Gesellschaft für Informatik e.V. (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in cooperation with GI and to publish the annual GI Award dissertation.

Broken down into
- seminars
- proceedings
- dissertations
- thematics

current topics are dealt with from the vantage point of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure high quality contributions.

The volumes are published in German or English.

Information: http://www.gi.de/service/publikationen/lni/

ISBN 978-3-88579-418-9

This book presents outstanding dissertations in informatics of the year 2013. Every year, the German Gesellschaft für Informatik (GI), the Swiss Informatics Society (SI), the Austrian Computer Society (OCG) and the German Chapter of the ACM (GchACM) jointly give an award for an excellent dissertation that represents an important advance in informatics. The award winner is chosen in a careful selection procedure from candidates proposed by Austrian, Swiss and German universities, where each university may suggest at most one dissertation per year. The series "Ausgezeichnete Informatikdissertationen" presents these outstanding dissertations to the informatics community, as well as to the public in order to support knowledge transfer from universities to industry and society. This volume is in German.

GI-Dissertationspreis 2013

14

GI-Edition

Lecture Notes in Informatics

Steffen Hölldobler et al. (Hrsg.)

Ausgezeichnete Informatikdissertationen 2013

Dissertations

Steffen Hölldobler et al. (Hrsg.)

Ausgezeichnete Informatikdissertationen 2013

Im Auftrag der GI herausgegeben durch die Mitglieder des Nominierungsausschusses

Abraham Bernstein, Universität Zürich
Wolfgang Effelsberg, Universität Mannheim
Steffen Hölldobler (Vorsitzender), Technische Universität Dresden
Hans-Peter Lenhof, Universität des Saarlandes
Klaus-Peter Löhr, Freie Universität Berlin
Paul Molitor, Martin-Luther-Universität Halle-Wittenberg
Gustaf Neumann, Wirtschaftsuniversität Wien
Rüdiger Reischuk, Universität zu Lübeck
Nicole Schweikardt, Goethe-Universität Frankfurt am Main
Myra Spiliopoulou, Otto-von Guericke-Universität Magdeburg
Harald Störrle, Technical University of Denmark
Sabine Süsstrunk, École Polytechnique Fédérale de Lausanne

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Dissertations
Series of the Gesellschaft für Informatik (GI), Volume D-14
ISBN 978-3-88579-418-9

printed by Köllen Druck+Verlag GmbH, Bonn

Vorwort

Die Gesellschaft für Informatik e.V. (GI) vergibt gemeinsam mit der Schweizer Informatik Gesellschaft (SI), der Österreichischen Computergesellschaft (OCG) und dem German Chapter of the ACM (GChACM) jährlich einen Preis für eine hervorragende Dissertation im Bereich der Informatik. Hierzu zählen nicht nur Arbeiten, die einen Fortschritt in der Informatik bedeuten, sondern auch Arbeiten aus dem Bereich der Anwendungen in anderen Disziplinen und Arbeiten, die die Wechselwirkungen zwischen Informatik und Gesellschaft untersuchen. Die Auswahl dieser Dissertationen stützt sich auf die von den Universitäten und Hochschulen für diesen Preis vorgeschlagenen Dissertationen. Jede dieser Hochschulen kann jedes Jahr nur eine Dissertation vorschlagen. Somit sind die im Auswahlverfahren vorgeschlagenen Kandidatinnen und Kandidaten bereits „Preisträger“ ihrer Hochschule.

Die 29 Einreichungen zum Dissertationspreis 2013 belegen die zunehmende Bedeutung und auch die Bekanntheit des Dissertationspreises. Wie jedes Jahr wurden die vorgeschlagenen Arbeiten im Rahmen eines Kolloquiums im Leibniz-Zentrum für Informatik Schloss Dagstuhl von den Nominierten vorgestellt. Für die Mitglieder des Nominierungsausschusses war das persönliche Zusammentreffen mit den Nominierten der Höhepunkt der Auswahlarbeit, und für die Nominierten hat das Kolloquium sicher eine Reihe neuer Erfahrungen und wissenschaftlicher Kontakte geboten. Das wissenschaftlich sehr hohe Niveau der Vorträge, die regen Diskussionen und die angenehme Atmosphäre in Schloss Dagstuhl wurde von allen Teilnehmerinnen und Teilnehmern des Kolloquiums sehr begrüßt.

Wie in jedem Jahr fiel es dem Nominierungsausschuss sehr schwer, eine einzige Dissertation auszuwählen, die durch den Preis besonders gewürdigt wird. Mit der Präsentation aller vorgeschlagenen Dissertationen in diesem Band wird die Ungerechtigkeit, eine aus mehreren ebenbürtigen Dissertationen hervorzuheben, etwas ausgeglichen. Dieser Band soll zudem einen Beitrag zum Wissenstransfer innerhalb der Informatik und von den Universitäten und Hochschulen in die Bereiche Technik, Wirtschaft und Gesellschaft leisten.

Die beteiligten Gesellschaften zeichnen Herrn Dr. Markus Anton Steinberger für seine hervorragende Dissertation „Dynamic Resource Scheduling on Graphics Processors“ mit dem Dissertationspreis 2013 aus.

Neuartige, parallele Hardware erfordert eine vollständige Anpasung der darauf ablaufenden Algorithmen. Herr Steinberger entwickelt ein neues Programmiermodell, das sich sehr gut für die Beschreibung paralleler Probleme eignet und gleichzeitig die Nutzung von gemischten parallelen Architekturen ermöglicht. Das Modell integriert die Speicherverwaltung mit einer dynamisches Arbeitsverteilung und führt in mehreren Anwendungsbereichen zu Lösungsstrategien, die um Größenordnungen effizienter sind als der bisherige Stand der Forschung.

Mit dieser Preisverleihung würdigen die beteiligten Gesellschaften – die Gesellschaft für Informatik e.V. (GI), die Schweizer Informatik Gesellschaft (SI), die Österreichische Computergesellschaft (OCG) und das German Chapter of the ACM (GChACM) – eine herausragende wissenschaftliche Arbeit, in der zwei, seit vielen Jahren offene Probleme mit neuen, bahnbrechenden Ideen elegant gelöst werden.

Ein besonderer Dank gilt dem Nominierungsausschuss, der sehr effizient und konstruktiv zusammengearbeitet hat. Bei Frau Julia Koppenhagen und Herrn Christoph Wernhard möchte ich mich für die Unterstützung bei der Entgegennahme der vorgeschlagenen Dissertationen, für die Organisation des Kolloquiums sowie für die Zusammenstellung und Anpassung der Beiträge an das Format der GI-Edition Lecture Notes in Informatik (LNI) bedanken. Für die finanzielle Unterstützung des Nominierungskolloquiums sei den beteiligten Gesellschaften gedankt. Die Gastfreundlichkeit und die hervorragende Bewirtung in Dagstuhl trugen zum Erfolg des Kolloquiums bei, wofür ich mich an dieser Stelle ebenfalls herzlich bedanke.

Steffen Hölldobler
Dresden im September 2014

Kandidaten für den GI-Dissertationspreis 2013

Dr. Ehlers, Rüdiger	Universität des Saarlandes
Dr.-Ing. Fischer, Ulrike	Technische Universität Dresden
Dr. Forkert, Nils Daniel	Universität Hamburg
Dr.-Ing. Gemünde, Mike	TU Kaiserslautern
Dr. Hofbauer, Heinz	Paris-Lodron-Universität Salzburg
Dr. Isele, Robert	Universität Mannheim
Dr. Kaufmann, Paul	Universität Paderborn
Dr.-Ing. Klement, Sascha	Universität zu Lübeck
Dr. Kuntsche, Stefan	Technische Universität Berlin
Dr. Langer, Alexander Johannes	RWTH Aachen
Dr. Lausser, Ludwig	Universität Ulm
Dr. Martin, Marcel	Technische Universität Dortmund
Dr.-Ing. Martinez Esturo, Janick	Otto-von-Guericke-Universität Magdeburg
Dr. Mitzlaff, Folke	Universität Kassel
Dr. Niessner, Matthias	Friedrich-Alexander-Universität Erlangen-Nürnberg
Dr. Otten, Jens	Universität Potsdam
Dr. Pham, Vinh	Universität Regensburg
Dr. Reinbacher, Thomas	Technische Universität Wien
Dr. Renner, Bernd-Christian	Technische Universität Hamburg-Harburg
Dr.-Ing. Rizk, Amr	Leibniz Universität Hannover
Dr. Salinger, Stephan	Freie Universität Berlin
Dr. Schatz, Roland	Johannes Kepler Universität Linz
Dr. Shokri, Reza	École polytechnique fédérale de Lausanne
Dr. Soeken, Mathias	Universität Bremen
Dr. Spehr, Jens	Technische Universität Braunschweig
Dr. Steinberger, Markus Anton	Technische Universität Graz
Dr. Tatu, Andrada	Universität Konstanz
Dr. Tuengerthal, Max	Universität Trier
Dr.-Ing. Wachsmann, Christian	Technische Universität Darmstadt

Mitglieder des Nominierungsausschusses für den GI-Dissertationspreis 2013

Von links nach rechts:

Prof. Dr.-Ing. Wolfgang Effelsberg	Universität Mannheim
Prof. Dr.-Ing. Klaus-Peter Löhr	Freie Universität Berlin
Prof. Dr. Harald Störrle	Technical University of Denmark
Prof. Dr. Gustaf Neumann	Wirtschaftsuniversität Wien
Prof. Dr. Paul Molitor	Martin-Luther-Universität Halle-Wittenberg
Prof. Dr. Hans-Peter Lenhof	Universität des Saarlandes
Prof. Dr. Steffen Hölldobler (Vorsitzender)	Technische Universität Dresden
Prof. Dr. Rüdiger Reischuk	Universität zu Lübeck
Prof. Dr. Nicole Schweikardt	Goethe-Universität Frankfurt am Main
Prof. Dr. Abraham Bernstein	Universität Zürich

Nicht im Bild:

Prof. Dr. Myra Spiliopoulou	Otto-von-Guericke-Universität Magdeburg
Prof. Dr. Sabine Süsstrunk	École Polytechnique Fédérale de Lausanne

Inhaltsverzeichnis

Symmetrische und effiziente Synthese*

Rüdiger Ehlers
ruediger.ehlers@uni-bremen.de
Universität Bremen

Abstract: Bei der reaktiven Synthese wird der Prozess der Konstruktion eines reaktiven Systems vereinfacht, indem dieses anhand einer formalen Spezifikation automatisch erzeugt wird. Trotz der offensichtlichen Vorteile dieses Konzeptes gegenüber der manuellen Konstruktion eines reaktiven Systems erzeugen aktuelle Syntheseansätze noch zu unstrukturierte Lösungen nach zu langer Rechenzeit, so dass reaktive Synthese noch nicht als Teil aktueller Systementwurfsvorgehensmodelle etabliert ist.

Die Dissertation „Symmetrische und effiziente Synthese“ behandelt sowohl das Problem der mangelnden Struktur als auch die zu verbessernde Effizienz der reaktiven Synthese. Zuerst wird die Synthese symmetrisch strukturierter Systeme betrachtet, welche dadurch, dass sie aus mehreren Instanzen desselben Prozesses bestehen, bessere Eigenschaften haben. Es werden Systemarchitekturen mit entscheidbaren und unentscheidbaren symmetrischen Syntheseproblemen charakterisiert. Für eine große Klasse von Architekturen wird ein komplexitätstheoretisch optimaler Synthesealgorithmus vorgestellt.

Zur Erhöhung der Effizienz der reaktiven Synthese wird ein aktueller Synthesealgorithmus, der seine praktische Anwendbarkeit bereits unter Beweis gestellt hat, um eine Klasse von unterstützten Systemeigenschaften erweitert, ohne dass er seine Effizienz verliert. Diese Erweiterung erlaubt auch die Synthese robuster Systeme, die sogar in unerwarteten Umgebungen geeignetes Verhalten zeigen. Außerdem wird der ACTL $\cap$ LTL Syntheseansatz vorgestellt, der die Effizienz des bereits erprobten Synthesealgorithmus in eine weit expressivere Spezifikationslogik überträgt.

1 Einleitung

Da Fehler in sicherheitskritischen reaktiven Systemen gravierende Auswirkungen haben können, wird formalen Methoden, die die Korrektheit eingebetteter Systeme sicherstellen können, eine große Bedeutung beigemessen. Ein reaktives System ist eine Datenverarbeitungseinheit, die in jedem Schritt ihrer Berechnung ein Eingabedatum einliest und ein passendes Ausgabedatum erzeugt. Die Berechnung terminiert nicht, d. h. das System hat keinen Zeitpunkt, zu dem die Berechnung abgeschlossen ist. Beispiele für solche Systeme sind Ampelsteuerungen, Steuerungseinheiten in Luft- und Landfahrzeugen sowie Maschinensteuerungen. Da das System nicht-terminierend ist, befasst sich die Spezifikation desselben mit dessen Verhalten während der Laufzeit, anstatt Bedingungen über die Ausgabe bei Berechnungsabschluss anzugeben.

Der meistverbreitete Ansatz zum Sicherstellen der Korrektheit reaktiver Systeme ist *Verifikation*, bei dem der Korrektheitsbeweis eines Systems erbracht wird, nachdem dieses

*Englischer Titel der Dissertation: “Symmetric and Efficient Synthesis”

konstruiert wurde. Verifikation beseitigt jedoch nicht die inhärente Schwierigkeit, korrekte Systeme zu konstruieren und bietet deshalb nur eine unbefriedigende Lösung. Zur Lösung dieses Problems wurde der Ansatz der *reaktiven Synthese* vorgeschlagen, welcher es dem Ingenieur/der Ingenieurin eines reaktiven Systems erlaubt, sich auf die Beschreibung der Spezifikation zu konzentrieren, von welcher die Systemimplementierung dann automatisch erzeugt wird. Ausgehend von der ersten Formulierung des Syntheseproblems für reaktive Systeme durch Church [Chu62] wurde in den letzten 40 Jahren eine Vielzahl von Spezifikationsformalismen und passenden Synthesemethoden entwickelt.

Trotz des Fortschritts der Forschung auf dem Gebiet der Synthese und der Attraktivität des Konzeptes ist dessen Einfluss auf die Praxis des Systementwurfs momentan eher als gering einzustufen. Für diese Diskrepanz gibt es eine Reihe von Gründen. Einer davon ist, dass die Qualität automatisch erzeugter Implementierungen im Allgemeinen vergleichbar mit der manuell erzeugter sein soll, und somit eine ähnliche Verständlichkeit, Wartbarkeit und technische Umsetzbarkeit haben soll. Aktuelle Synthesealgorithmen garantieren jedoch nicht, Implementierungen mit diesen Eigenschaften zu berechnen, und es zeigte sich, dass die Algorithmen dies auch in der Praxis nicht tun. Weil jedoch diese Qualitätseigenschaften schwer zu formalisieren sind, ist es am vielversprechendsten anzusehen, stattdessen Anforderungen an die Struktur der zu synthetisierenden Systeme in den Syntheseprozess aufzunehmen, die diese Eigenschaften implizieren.

Orthogonal zum Aspekt der Struktur synthetisierter Systeme ist die Frage der Skalierbarkeit des Syntheseprozesses. Aufgrund der hohen Komplexität des Syntheseproblems wurde von vielen Wissenschaftlern die Idee der Synthese als inpraktikabel verworfen. Beispielsweise wurde für Linearzeitlogik (LTL, [Pnu77]) als Spezifikationssprache gezeigt, dass das zugehörige Syntheseverfahren eine doppelt-exponentielle Zeitkomplexität hat. Dennoch zeigen die vergangenen Jahre ein Wiederaufleben des Interesses an reaktiver Synthese. Der Hauptgrund hierfür ist die Beobachtung, dass viele Spezifikationen in der Praxis entweder eine simple Struktur oder eine simple Lösung haben. Dies widerspricht der Komplexitätsanalyse nicht: Die doppelt-exponentielle Zeitkomplexität ist tatsächlich nur ein Zeichen dafür, dass der Spezifikationsformalismus es erlaubt, kompakte Spezifikationen zu schreiben, dessen kleinste Implementierungen doppelt-exponentiell in der Länge der Spezifikation groß sind. Wenn wir annehmen, dass solche Spezifikationen nicht als Eingabe für ein Syntheseverfahren verwendet werden, können Lösungstechniken angewandt werden, die auf die praktisch relevanten Spezifikationsfälle optimiert sind.

Die Syntheseansätze, die dieser Idee folgen, können grob in zwei Forschungsrichtungen unterteilt werden. Die erste dieser Richtungen befasst sich mit der Effizienzverbesserung der Synthese für vollständiges LTL. Dabei betrachtet sie eine Vielzahl von Techniken, wie z. B. solchen, die auf dem Lösen von Paritätsspielen basieren [SSR08], bishin zu Ansätzen wie *Bounded Synthesis* [FS07]. In der zweiten Forschungsrichtung wird die volle Expressivität von LTL gegen die Möglichkeit der Optimierung auf ein wohlgewähltes Subfragment von LTL eingetauscht, um die Skalierbarkeit der Synthese substanziell zu verbessern. Dieser Ansatz schränkt die Menge der Spezifikationen, die betrachtet werden können, ein. Trotzdem gibt es eine große Zahl praktischer Anwendungen, in denen Implementierungen mit ihm erfolgreich synthetisiert wurden (z.B. [BGJ$^+$07, OTMW11]). Dies zeigt, dass eine gute Auswahl der Spezifikationslogik es erlaubt, ein passendes Syntheseverfahren zu

entwickeln, welches die Effizienz und Expressivität in der Spezifikation, die in praktischen Anwendungen benötigt wird, kombiniert.

Die Doktorarbeit „Symmetrische und effiziente Synthese“ beschreibt neue Resultate und Techniken, die sowohl das Problem der ungenügenden Struktur synthetisierter reaktiver Systeme angehen als auch die Skalierbarkeit aktueller Syntheseverfahren für praktische Spezifikationen verbessern. Darüber hinaus enthält der Einführungsteil der Arbeit eine problemorientierte Einführung in das Gebiet der reaktiven Synthese und eine Beschreibung eines modernen Syntheseansatzes, einschließlich der vollständigen Herleitung und Motivation für dessen Komponenten. Dieser Einführungsteil ist in sich abgeschlossen und damit auch für reine Lehrzwecke geeignet.

2 Symmetische Synthese

Struktur in reaktiven Systemen kann viele Formen haben. So kann ein System kompakt durch ein kurzes Programm oder durch einen Schaltkreis geringer Tiefe dargestellt werden. In der bestehenden Literatur finden sich Syntheseverfahren zum Erzeugen explizit repräsentierter Systeme mit wenigen Zuständen [FS07], kleiner Schaltkreise [EKH12] und auch verteilter Systeme [PR90], bei denen ein reaktives System aus mehreren Subsystemen besteht, die durch eine vordefinierte Kommunikationsstruktur die Spezifikationen in Kooperation erfüllen.

In all diesen Arbeiten wurden jedoch keine Eigenschaften über die interne Struktur der Lösungsteile betrachtet. So präferieren Ingenieure z. B. *symmetrische* Systeme, d. h. Implementierungen, die aus multiplen Instanzen derselben Subimplementierung bestehen.

Die symmetrische Implementierbarkeit reaktiver Systeme ist eine klassische Forschungsfrage der Informatik. Das Philosophenproblem, bei dem eine Menge von n Philosophen sich bei der Benutzung ebensovieler Besteckteile koordinieren müssen, ist das wohl bekannteste Szenario in welchem man sich für die symmetrische Implementierbarkeit einer Spezifikation interessiert. Für ein reaktives System, welches einen kontinuierlichen Eingabestrom liest und gleichzeitig einen Ausgabestrom erzeugt, bedeutet eine symmetrische Implementierung, dass alle Subkomponenten parallel und zur selben Zeit laufen. Symmetrische Systeme sind oft einfacher zu handhaben als monolithische Systeme, da die Symmetrieanforderung garantiert, dass das System ohne zentrale Koordinationskomponente auskommt. Außerdem sind symmetrische Systeme typischerweise einfacher zu warten als andere verteilte Systeme, da sie weniger Komponenten haben und oft auch einfacher zu verstehen, da die Komponenten in einer Art und Weise agieren müssen, die eine zentrale Koordination unnötig machen. Zuguterletzt sei angemerkt, dass es spezielle Verifikationsalgorithmen gibt, die Symmetrie in zu verifizierenden Systemen ausnutzen können, so dass Symmetrie hilfreich für die Zertifizierung von Systemen ist.

Bei dem bereits genannten Philosophenproblem ist das Interesse an symmetrischer Implementierbarkeit von eher theoretischer Natur. Es ist jedoch allgemein bekannt, dass es keine synchrone symmetrische Implementierung für die Philosophen gibt, bei der diese nicht verhungern, was die inhärenten Beschränkungen symmetrischer Implementierungen

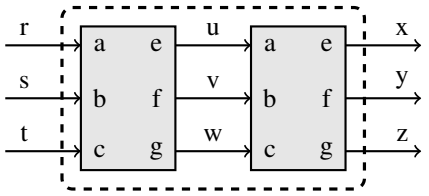


Abbildung 1: Grafische Darstellung der symmetrischen S2 Architektur

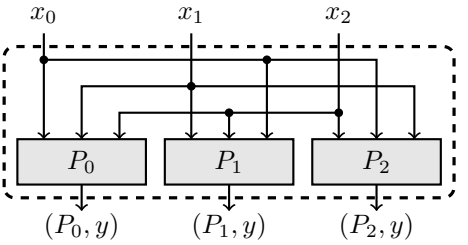


Abbildung 2: Eine einfache rotationssymmetrische Architektur.

zeigt. Um diese zu vermeiden, wurde eine breite Palette von Techniken zum *Brechen der Symmetrie* entwickelt. Typischerweise wird dies durch die Ausführung eines verteilten Algorithmus erreicht, nach dessen Terminierung ein Prozess als Koordinationsprozess bestimmt worden ist (auch „*leader election*“ genannt).

Die Forschung auf dem Gebiet der Symmetriebrechung ist gewissermaßen sehr pessimistisch, da sie dadurch motiviert ist, dass einige Anwendungen Symmetriebrechung benötigen. Dies trifft jedoch nicht auf alle Anwendungen zu. Das Einsparen des Symmetriebrechungsschrittes hat viele Vorteile. Beispielsweise benötigen Algorithmen zur Symmetriebrechung Kommunikationsverbindungen zwischen den Prozessen, die bei symmetrischen Implementierungen oft nicht nötig sind. Darüber hinaus verzögert die Wahl eines Koordinationsprozesses die Verfügbarkeit der Prozesse für deren Hauptaufgabe und vergrößert deren Implementierung. Daher sind Systemimplementierungen, die ohne Symmetriebrechnung auskommen, im Allgemeinen vorzuziehen. Dies wirft die Frage auf, mit welchen Methoden man für eine Anwendung eines verteilten Systems erkennen kann, ob Symmetriebrechnung überhaupt nötig ist.

2.1 Ergebnisse

Die Doktorarbeit „Symmetrische und effiziente Synthese“ beinhaltet die erste gründliche Analyse des Problems der Synthese symmetrischer Systeme. Die erste Kernfrage in diesem Kontext ist, welche *symmetrische Architekturen* entscheidbare Syntheseprobleme aufweisen. Eine Architektur definiert, mit welchen Signalen die Teilkomponenten, welche jeweils z. B. als Mealy- oder Moore-Maschinen dargestellt werden können und auch *Prozesse* genannt werden, interagieren können. Außerdem legt sie fest, wie die Ein- und Ausgabe des Gesamtsystems auf die Prozesse verteilt ist.

Abbildung 1 zeigt eine einfache Beispielarchitektur mit zwei Prozessen. Das symmetrische Syntheseproblem stellt sich in diesem Fall wie folgt dar: Gegeben ist eine Spezifikation über $r, s, \ldots, y, z$, die das gewünschte Verhalten des Gesamtsystems repräsentiert. Gesucht ist eine Prozessimplementierung, die die Eingabesignale a, b und c liest, auf die Ausgabesignale e, f und g schreibt und sicherstellt, dass das Gesamtsystem die Spezifikation erfüllt, wenn die Implementierung für beide Prozesse der Architektur verwendet wird.

Die Architektur wird als fest angenommen und ist nicht Teil der Eingabe des Syntheseproblems. Dies erlaubt es uns, zwischen Architekturen mit entscheidbaren und unentscheidbaren symmetrischen Syntheseproblemen zu unterscheiden.

Leider ist schon die sehr einfache *S2 Architektur* in Abbildung 1 für alle Spezifikationsformalismen von praktischer Relevanz unentscheidbar. Dies kann gezeigt werden, indem die Synthese für eine andere Architektur ohne Symmetriebedingung, in der die Prozesse somit unterschiedliche Implementierungen haben können, und für die von Pnueli und Rosner die Unentscheidbarkeit des Syntheseproblems gezeigt wurde [PR90], auf die symmetrische Synthese für die S2 Architektur reduziert wird. Die Archiektur von Pnueli und Rosner bezieht ihre Unentscheidbarkeit von der Tatsache, dass die beiden Prozesse die Eingabe für den jeweiligen anderen Prozess nicht lesen können. Durch geschickte Kodierung in der Spezifikation kann dies im symmetrischen Fall dadurch simuliert werden, dass die Prozesse nicht wissen, ob sie der erste oder der zweite Prozess der Kette sind.

Das Unentscheidbarkeitsergebnis kann dann für den Fall, dass die Prozesse jeweils nur ein Eingabe- und Ausgabesignal haben, generalisiert werden, indem die Werte der Signale serialisiert werden. Die Dissertation zeigt auch, wie dies mit *nicht-zählenden Spezifikationslogiken* wie z. B. LTL durchgeführt werden kann. Dadurch haben im Endeffekt alle Architekturen, die interne Kommunikation zwischen den Prozessen haben, ein unentscheidbares symmetrisches Syntheseproblem. Da jedoch eine der Motivationen für symmetrische Synthese das Einsparen ebensolcher Prozesskommunikation ist, stellt sich die Frage nach den verbleibenden Fällen.

Tatsächlich hat eine große Klasse von symmetrischen Architekturen ohne interne Prozesskommunikation ein entscheidbares Syntheseproblem für alle praktisch relevanten Spezifikationslogiken. In diesen so genannten *rotationssymmetrischen* Architekturen erhalten alle Prozesse die komplette Eingabe an das Gesamtsystem, jedoch in individuell rotierter Form. Abbildung 2 zeigt ein einfaches Beispiel. Architekturen von dieser Form erlauben die präzise Charakterisierung der Attribute derjenigen *Berechnungsbäume*, die das Verhalten eines rotationssymmetrischen Gesamtsystems darstellen, als sogenannte *Symmetrieeigenschaft*. Ein Berechnungsbaum beschreibt die Menge der möglichen Berechnungen eines reaktiven Systems. Der Baum verzweigt bezüglich der Eingabe an das System, von der wiederum die Ausgabe des Systems abhängen kann, mit dem die Knoten des Baumes markiert sind. Das Kernresultat, das die Synthese für rotationssymmetrische Gesamtsysteme erlaubt, ist die Dekomposition der Symmetrieeigenschaft in zwei Teile:

1. Die Bedingung, dass das zu synthetisierende System nicht spontan die Symmetrie durch seine Ausgabe bricht
2. Die Evaluation der Systemspezifikation und ihrer rotierten Versionen nur entlang *normalisierter* Eingabeworte an das System

Beide Bedingungen sind notwendig und für reaktive Systeme mit endlichem Zustandsraum, die wir bei der Synthese anstreben, auch hinreichend. Die erste Bedingung kann relativ leicht in einen reaktiven Syntheseprozess eingebettet werden, da sie z. B. in LTL beschrieben werden kann. Die zweite Bedingung wird durch Modifikation der *Baumautomaten*, die in Syntheseprozessen typischerweise verwendet werden, implementiert. Im Fall einer positiven Realisierbarkeitsantwort durch den Synthesealgorithmus ist dann nur

noch ein Nachbearbeitungsschritt der Implementierung erforderlich. In diesem wird das Systemverhalten für nicht-normalisierte Eingabeströme durch Rotationen des Systemverhaltens für normalisierte Eingabeströme ersetzt. Die Zeitkomplexität des Syntheseverfahrens für rotationssymmetrische Systeme ist exponentiell in der Anzahl der Prozesse für alle verbreiteten Spezifikationsformalismen. Eine Komplexitätsanalyse, die zeigt, dass der Algorithmus optimal ist, komplettiert die Betrachtung der symmetrischen Synthese.

3 Effiziente Synthese

Schon die ersten Lösungen zum Problem der reaktiven Synthese erlauben es, sehr expressive Spezifikationsformalismen zu verwenden [BL69]. Die hohe Komplexität des Syntheseproblems für eben diese Formalismen motiviert jedoch die Betrachtung schwächerer Spezifikationslogiken, die sich besser für effiziente Synthesealgorithmen eignen.

Unter denjenigen Ansätzen, die in diese Klasse fallen, ist *Generalised Reactivity(1) Synthese* [PPS06], welche typischerweise als *GR(1) Synthese* abgekürzt wird, die wohl meistverbreitete. In diesem Kontext besteht die Spezifikation aus *Annahmen* und *Garantien*. Erstere beschreiben die Eigenschaften der Umgebung des zu synthetisierenden reaktiven Systems, welche als gegeben angenommen werden können. Letztere beschreiben die gewünschten Systemeigenschaften. Um eine niedrige Komplexität des Syntheseprozesses zu erreichen, sind die erlaubten Formen der Annahmen und Garantien, verglichen mit der Temporallogik LTL, in welcher diese klassischerweise formuliert werden, syntaktisch eingeschränkt. Eine Vielzahl von Arbeiten zeigt jedoch, dass es möglich ist, unter diesen Einschränkungen erfolgreich Systeme mit praktischer Relevanz zu synthetisieren (siehe z.B. [BGJ+07, WTM10, OTMW11]). Leider hat der GR(1) Syntheseansatz dennoch zwei substanzielle Einschränkungen: *ungenügende Expressivität* und die Notwendigkeit zur *Präsynthese*.

3.1 Ergebnisse

Obwohl viele Eigenschaften in Spezifikationen praktischer reaktiver Systeme als Kombination von Eigenschaftstypen, die bei der GR(1) Synthese verwendet werden können, darstellbar sind, trifft dies auf eine wichtige Klasse von Spezifikationsteilen nicht zu: *Stabilitätseigenschaften*. Diese beschreiben, dass ab einem beliebigen Zeitpunkt der Ausführung eines reaktiven Systems eine gegebene einfache Subeigenschaft immer erfüllt sein muss. Ein Beispiel hierfür ist die Bedingung, dass ein reaktives System nur am Anfang seiner Ausführung eine endliche Anzahl von Reaktionsfristen auf bestimmte Stimuli verpassen darf, d. h., ab einem beliebigen Zeitpunkt ein Normalbetriebszustand erreicht ist.

Im Rahmen der Doktorarbeit „Symmetric and Efficient Synthesis“ wurde eine Erweiterung des GR(1) Syntheseverfahrens entwickelt, welche die Menge der erlaubten Eigenschaften in den Annahmen und Garantien um Stabilitätseigenschaften ergänzt. Die Erweiterung

behält diejenigen Attribute der GR(1) Synthese bei, die dessen effiziente Implementierung erlauben. Dies ist zum einen die Reduktion des Syntheseproblems auf das Lösen von Paritätsspielen mit einer konstanten Anzahl an Farben und einer Spielgröße, die nur exponentiell in der Anzahl der Propositionen in der Spezifikation ist. Zum anderen ist dies die Möglichkeit einer *symbolischen* Lösung dieses Spiels, z. B. mit binären Entscheidungsdiagrammen (BDDs). Darüber hinaus wird gezeigt, dass jede weitere substanzielle Erweiterung der unterstützten Eigenschaftsklassen für die Annahmen und Garantien dazu führen würde, dass diese Vorteile der GR(1) Synthese verloren gingen (sofern P$\neq$NP gilt).

Stabilitätseigenschaften komplettieren den GR(1) Syntheseansatz, dessen Erweiterung den Namen *generalisierte Rabin(1) Synthese* erhielt, auf zwei Arten. So charakterisieren Wongpiromsarn et al. [WTM10] die Eigenschaftstypen, die in Robotikanwendungen für die Annahmen und Garantien benötigt werden. Stabilitätseigenschaften waren die letzte Klasse, die bei der GR(1) Synthese noch fehlte. Darüber hinaus ist die Spezifikationsklasse mit der Erweiterung um Stabilitätseigenschaften nun unter *Härtung* abgeschlossen. Hierbei wird eine Systemspezifikation in eine neue Spezifikation übersetzt, die dafür sorgt, dass sich das zu synthetisierende System im Falle der Verletzung der gemachten Annahmen in einer robusten Art und Weise verhalten muss, d. h., nach kurzer Zeit wieder in den Normalbetriebszustand zurückkehren muss. Dies ist in unvorhergesehenen Betriebsbedingungen von Relevanz, da normalerweise keine Bedingungen an das Verhalten des reaktiven Systems gestellt werden, die nach Verletzung der Annahmen gelten müssen.

Neben der Erweiterung des GR(1) Syntheseansatzes um Stabilitätseigenschaften wurde im Rahmen der Doktorarbeit auch die Frage behandelt, wie dem Problem der *Präsynthese* begegnet werden kann. Dieser Begriff beschreibt die Aktivität, Spezifikationsteile in die von der GR(1) Synthese benötigte spezielle syntaktische Form zu überführen. Es hat sich gezeigt, dass es für die Effizienz des Syntheseverfahrens von großer Relevanz ist, wie genau eine Spezifikation in diese Form überführt wird. Als Konsequenz hieraus wird dieser Schritt typischerweise per Hand ausgeführt.

Dieser manuelle Schritt steht im starken Gegensatz zum Grundanspruch der Synthese, die Konstruktion von reaktiven Systemen vollständig zu automatisieren. Zur Behebung dieses Defizits wurde im Rahmen der Doktorarbeit der *ACTL $\cap$ LTL Syntheseansatz* entwickelt. Kernidee ist es, als Annahmen und Garantien alle Eigenschaften zuzulassen, deren Erfüllung in einem Berechnungsbaum sowohl in der Temporallogik ACTL (*Computation Tree Logic* ohne existenzielle Pfadquantoren) als auch in LTL darstellbar sind. Wie von Maidl [Mai00] gezeigt wurde, sind dies genau diejenigen Eigenschaften, die als *universelle sehr schwache Büchiautomaten* repräsentiert werden können.

Der Vorteil der Darstellung von Spezifikationen als universelle sehr schwache Automaten ist, dass diese effizient in symbolisch repräsentierte Synthesespiele übersetzt werden können. Dadurch sind sie gut als Zwischenebene für Spezifikationen in einer geeigneten Logik und Synthesespielen geeignet. Um die Vorteile dieses Automatentyps in praktischen Syntheseansätzen zu nutzen, bedarf es jedoch eines Verfahrens, um aus einer geeigneten Spezifikationssprache in Automaten des Typs zu übersetzen. Maidl [Mai00] zeigte für ACTL, wie dies funktioniert. Für praktische Anwendungen ist eine Linearzeitlogik jedoch wesentlich besser geeignet. Ein entsprechender Algorithmus hierzu war bisher noch unbekannt und wurde im Rahmen der Dissertation entwickelt.

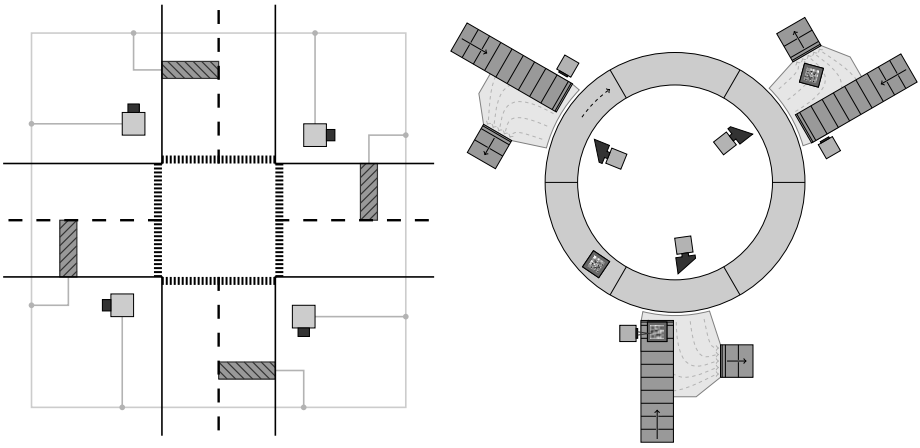


Abbildung 3: Grafische Darstellung zweier Anwendungen, für die reaktive Steuerungseinheiten symmetrisch implementiert werden können.

Die resultierende Konstruktion stellt auch den ersten Algorithmus dar, um eine Formel in LTL nach ACTL zu übersetzen, wann immer dies möglich ist. Eine Konstruktion für die Rückrichtung [CD88] wird mittlerweile in vielen Vorlesungen über formale Methoden gelehrt – nur die Hinrichtung war noch ungelöst. Mit den Resultaten zur ACTL $\cap$ LTL Synthese wurde diese Lücke nun geschlossen.

4 Zusammenführung und experimentelle Evaluation

Die Resultate zur symmetrischen Synthese, zur generalisierten Rabin(1) Synthese und zur ACTL $\cap$ LTL Synthese, die im Rahmen der Doktorarbeit erzielt wurden, können einzeln verwendet werden, aber insbesondere auch in Kombination. Dies erlaubt zum Beispiel die effiziente Synthese robuster Systeme mit Annahmen und Garantien in ACTL $\cap$ LTL.

Zur Evaluation der praktischen Nutzbarkeit der in der Dissertation vorgestellten Verfahren wurden diese in Kombination anhand von zwei Fallstudien bewertet. Abbildung 3 stellt diese grafisch dar. Es handelt sich im ersten Fall um ein Ampelsystem, das Sensoren für wartende Autos besitzt, sowie um einen Rotationssortierer, der Pakete mit Hilfe eines permanent rotierenden Verteiltisches in die jeweils richtigen Ausgangsrutschen schieben muss. Beide Fallstudien sind natürliche Anwendungsgebiete für symmetrische Systeme.

Die Experimente zeigen, dass die Anwendung symmetrischer Synthese auch Einblick in die inhärenten Eigenschaften eines verteilten Systems bietet. Beispielsweise zeigt sich beim Ampelsystem, dass es ausreichend ist, dass Symmetrie *partiell* gebrochen wird, damit das System dessen Funktion erfüllen kann. Alle durchgeführten Experimente benötigen höchstens ein paar Minuten auf aktuellen Rechnern und zeigen somit die praktische Anwendbarkeit der vorgestellten Techniken.

5 Zusammenfassung und Ausblick

Die Dissertation „Symmetrische und effiziente Synthese“ stellt Lösungen für die schnelle und automatische Synthese strukturierter reaktiver Systeme vor. Zur Verbesserung der Struktur synthetisierter Systeme wird das Problem der symmetrischen Synthese betrachtet und für eine große Klasse von Architekturen gelöst. Der Synthesealgorithmus besitzt eine optimale Zeitkomplexität. Für Architekturen, die Kommunikation zwischen den Prozessen enthalten, wird hingegen bewiesen, dass diese ein unentscheidbares Syntheseproblem aufweisen.

Zur Verbesserung der Effizienz der reaktiven Synthese wird ein effizienter Syntheseansatz, der seine Anwendbarkeit schon mehrfach unter Beweis gestellt hat, der aber nicht alle in praktischen Anwendungen nützlichen Spezifikationsteilarten unterstützt, um Stabilitätseigenschaften erweitert. Dies erlaubt insbesondere die Synthese *robuster* Systeme, die sich auch dann noch sinnvoll verhalten, wenn die in der Spezifikation gegebenen Annahmen über die Umgebung des Systems verletzt werden. Die Expressivitätserweiterung um Stabilitätseigenschaften ist die nachweisbar substanziell letztmögliche, ohne dass die guten Eigenschaften des Algorithmus verloren gehen (sofern P$\neq$NP gilt).

Darüber hinaus wird im Rahmen der Dissertation gezeigt, wie der Kodierungsprozess einer Spezifikation in eine Form, die von effizienten und vergleichsweise einfachen Synthesealgorithmen verwendbar ist, automatisiert werden kann. Dabei werden alle Eigenschaften in Spezifikationen unterstützt, die sowohl in der Temporallogik LTL, als auch in ACTL geschrieben werden können. Die Anwendung der resultierenden universellen sehr schwachen Automaten in Synthesespielen ist einfach und erlaubt die Benutzung symbolischer Berechnungstechniken.

Die Resultate über die symmetrische und effiziente Synthese reaktiver Systeme werden durch zwei Fallstudien komplementiert, in welchen die neu entwickelten Techniken in Kombination angewandt werden. Darüber hinaus enthält die Dissertation eine abgeschlossene leicht zugängliche Einführung in das Gebiet der reaktiven Synthese, welche auch ohne Betrachtung der erzielten Forschungsergebnisse von Interesse ist.

Neben der Erweiterung des Wissens über Methoden und Konzepte der reaktiven Synthese gibt die Dissertation auch einen Ausblick auf zukünftige Forschung auf dem Gebiet: Sie zeigt, dass nichtreguläre Eigenschaften zu synthetisierender Systeme, wie hier die Symmetrieeigenschaft, durch *Dekomposition* dieser Eigenschaften in den Syntheseprozess integriert werden können. Weiterhin zeigt sie, dass durch geschickte Wahl eines Automatenmodells als Zwischenrepräsentation für die Spezifikation in der Synthese Effizienz und Expressivität kombiniert werden können. Das in der Dissertation verwendete Automatenmodell ist z.B. sehr gut geeignet, um neuartige Repräsentationen für Spielpositionsmengen während der Synthesespiellösung zu untersuchen und somit die Syntheseeffizienz in der Zukunft weiter zu erhöhen. Schließlich gibt die Dissertation einen Ausblick auf zukünftige Lösungen zur Synthese von Systemen *hoher Qualität*, die sowohl auf algorithmischer Modifikation der Spezifikation basieren, als auch auf der Anwendung von Techniken, die eine fundamentalere Änderung des Synthesealgorithmus erfordern.

Literatur

[BGJ+07] Roderick Bloem, Stefan Galler, Barbara Jobstmann, Nir Piterman, Amir Pnueli und Martin Weiglhofer. Specify, Compile, Run: Hardware from PSL. *Electr. Notes Theor. Comput. Sci.*, 190(4):3–16, 2007.

[BL69] J. Richard Büchi und Lawrence H. Landweber. Definability in the Monadic Second-Order Theory of Successor. *J. Symb. Log.*, 34(2):166–170, 1969.

[CD88] Edmund M. Clarke und I. A. Draghicescu. Expressibility results for linear-time and branching-time logics. In *REX Workshop*, Seiten 428–437, 1988.

[Chu62] Alonzo Church. Logic, Arithmetic and Automata. In *Intl. Congr. Math.*, Seiten 23–25, 1962.

[Ehl13] Rüdiger Ehlers. *Symmetric and Efficient Synthesis*. Dissertation, Universität des Saarlandes, 2013.

[EKH12] Rüdiger Ehlers, Robert Könighofer und Georg Hofferek. Symbolically Synthesizing Small Circuits. In *FMCAD*, Seiten 91 – 100, 2012.

[FS07] Bernd Finkbeiner und Sven Schewe. SMT-Based Synthesis of Distributed Systems. In *Automated Formal Methods (AFM)*, 2007.

[Mai00] Monika Maidl. The Common Fragment of CTL and LTL. In *FOCS*, 2000.

[OTMW11] Necmiye Ozay, Ufuk Topcu, Richard M. Murray und Tichakorn Wongpiromsarn. Distributed Synthesis of Control Protocols for Smart Camera Networks. In *ICCPS*, 2011.

[Pnu77] Amir Pnueli. The Temporal Logic of Programs. In *FOCS*, Seiten 46–57. IEEE, 1977.

[PPS06] Nir Piterman, Amir Pnueli und Yaniv Sa'ar. Synthesis of Reactive(1) Designs. In *VMCAI*, Seiten 364–380, 2006.

[PR90] Amir Pnueli und Roni Rosner. Distributed Reactive Systems Are Hard to Synthesize. In *FOCS*, Seiten 746–757. IEEE Computer Society, 1990.

[SSR08] Saqib Sohail, Fabio Somenzi und Kavita Ravi. A Hybrid Algorithm for LTL Games. In *VMCAI*, Seiten 309–323, 2008.

[WTM10] Tichakorn Wongpiromsarn, Ufuk Topcu und Richard M. Murray. Receding horizon control for temporal logic specifications. In *HSCC*, Seiten 101–110, 2010.

Rüdiger Ehlers, geboren 1982, studierte von 2002 bis 2006 Informatik an der Technischen Universität Dortmund. Nach dem Abschluss als Diplom-Informatiker und einem zusätzlichen Master-Studium über die Schnittstelle zwischen Mathematik und Informatik an der University of Oxford in England begann er ein Promotionsstudium an der Universität des Saarlandes, das er 2013 abschloss. Von September 2012 bis August 2013 war er kollaborativer Wissenschaftler der University of California at Berkeley und der Cornell University in den Vereinigten Staaten von Amerika. Nach einer anschließenden Beschäftigung als postdoktoraler wissenschaftlicher Mitarbeiter an der Universität Kassel ist er seit Februar 2014 Nachwuchsgruppenleiter am Fachbereich Informatik der Universität Bremen.

Zeitreihenprognose in relationalen Datenbanksystemen*

Ulrike Fischer

Professur für Datenbanken
Fakultät Informatik
Technischen Universität Dresden
ulrike.fischer@tu-dresden.de

Abstract: Die Zeitreihenprognose ist eine wichtige Analysetechnik für eine Vielzahl von Anwendungsgebieten. Prognosen werden dabei oft von fachfremden Nutzern auf großen, multidimensionalen Datensätzen angefragt, die eine geringe Antwortzeit erwarten. Aktuelle Datenbanksysteme unterstützen jedoch gar nicht oder nur sehr eingeschränkt Prognoseverfahren, sodass Prognosewerte manuell von Experten in dedizierten Softwareumgebungen berechnet werden müssen. Im Rahmen dieser Dissertation stellen wir einen neuartigen Ansatz vor, der die Integration der Zeitreihenprognose in ein relationales Datenbanksystem verfolgt. Mittels SQL können Nutzer ohne statistische Vorkenntnisse deklarative Prognoseanfragen stellen, die dann transparent und automatisch vom Datenbanksystem verarbeitet werden. Dabei diskutieren wir Optimierungstechniken für drei verschiedene Arten von Prognoseanfragen: Ad-hoc Anfragen, wiederkehrende Anfragen und kontinuierliche Anfragen. Alle Optimierungsansätze reduzieren die Ausführungszeit von Prognoseanfragen und gewährleisten eine hohe Prognosegenauigkeit.

1 Einleitung

Die Zeitreihenprognose ist seit langer Zeit ein wichtiger Bestandteil der Forschungslandschaft und bietet immer noch viele offenen Forschungsfragen [GH06]. Die Prognose zukünftiger Daten ist in einer Vielzahl von Anwendungsgebieten unverzichtbar, zum Beispiel bei der Produktionsplanung, der Regulierung intelligenter Stromnetze oder im Online Marketing Bereich. Die Nutzung von Prognosetechniken ermöglicht es, vorausschauend zu agieren, anstatt nur auf vergangene Ereignisse zu reagieren.

In der Vergangenheit wurde die Zeitreihenprognose von hoch qualifizierten Experten mit langer Erfahrung im Unternehmen oft manuell in dedizierten, statistischen Programmen durchgeführt. In den letzten Jahren können wir einen Paradigmenwechsel beobachten, der nicht nur die Zeitreihenprognose sondern generell komplexe statistische Verfahren betrifft. Traditionelle Datenbanksysteme müssen immer größer werdende Datenmengen verarbeiten, die kontinuierlich aus einer Vielzahl von Datenquellen generiert werden. Diese Entwicklung bietet nicht nur große Herausforderungen sondern auch vielfältige Möglichkeiten

*Englischer Titel der Dissertation: “Forecasting in Database Systems”

hinsichtlich der Analyse komplexer Datenbestände in Echtzeit. Folglich beinhaltet die Datenanalyse in modernen Datenbankmanagementsystemen mehr und mehr komplexe statistische Methoden, die weit über klassische Roll-Up und Drill-Down Operationen hinausgehen. Dabei werden Prognosewerte oft von Nicht-Mathematikern und Nicht-Statistikern abgefragt, die sich nicht mit Aufgaben wie der Auswahl oder der Schätzung des Prognosemodells beschäftigen möchten. Auf der anderen Seite müssen Prognosewerte aber auch automatisiert und gänzlich ohne menschliches Eingreifen berechnet werden können, beispielsweise bei der Regulierung intelligenter Stromnetze.

Diverse Forschungsgruppen und kommerzielle Datenbankanbieter beschäftigen sich mit der Integration von statistischen Verfahren wie der Zeitreihenprognose in klassische Datenbankmanagementsysteme. Jedoch verfolgen die meisten existierenden Verfahren einen Black-Box Ansatz und versuchen, die Änderungen am Datenbanksystem so minimal wie möglich zu gestalten, beispielsweise durch die Verwendung benutzerdefinierter Funktionen (UDFs) [CDD+09] oder die Verknüpfung des Datenbanksystems mit externen, statistischen Programmen [GLW+11]. Zwar sind derartige Ansätze einfacher zu realisieren, jedoch entgehen ihnen vielfältige Optimierungsmöglichkeiten. Weiterhin erfordern die meisten existierenden Ansätze immer noch die explizite Erstellung und Nutzung von Modellen, und ignorieren damit das Konzept deklarativer Datenbankanfragen.

Im Rahmen der Dissertation [Fis14] stellen wir einen Ansatz vor, der eine vollständige Integration der Zeitreihenprognose in ein traditionelles, relationales Datenbanksystem verfolgt. Eine enge Kopplung zwischen dem Datenbanksystem und den statistischen Prognosemethoden sorgt für Konsistenz zwischen Modellen und Daten, und erhöht sowohl die Nutzbarkeit als auch die Effizienz des Prognoseprozesses. Im Gegensatz zu sogenannten Flash-Back Anfragen, die eine vergangene Sicht auf die Daten ermöglichen, unterstützen wir *Flash-Forward* Anfragen. Mittels SQL können Nutzer ohne statistische Vorkenntnisse deklarative Prognoseanfragen stellen, die dann transparent und automatisch vom Datenbanksystem verarbeitet werden.

Im folgenden Kurzbeitrag zur Dissertation gibt Abschnitt 2 zunächst einen Überblick über die Grundlagen der Zeitreihenprognose. Abschnitt 3 stellt dann die konzeptionelle Architektur unseres Gesamtsytems, des *Flash-Forward Datenbanksystems* (F^2DB), genauer vor. Die Abschnitte 4 bis 6 diskutieren unterschiedliche Komponenten und Optimierungstechniken von F^2DB. Abschließend fassen wir das Erreichte in Abschnitt 7 zusammen.

2 Grundlagen der Zeitreihenprognose

Die Prognose von Zeitreihen basiert auf einem generalisierten modellbasierten Prozess, aus diesem wir Anforderungen an und architektonische Umsetzungsmöglichkeiten für ein Prognosesystem ableiten können [FL14].

Modellbasierter Prognoseprozess Eine *Zeitreihe* bezeichnet eine zeitliche Folge von Beobachtungen in äquidistanten Zeitabständen. Die *Zeitreihenprognose* beschäftigt sich mit der Schätzung neuer, bisher ungesehener Beobachtungen, sogenannter *Prognosewerte*, für zukünftige Zeitpunkte. Der *Prognosehorizont* beinhaltet dabei das gesamte Intervall

vom aktuellen Zeitpunkt bis zu einem gegebenen zukünftigen Zeitpunkt. Zur Berechnung der Prognosewerte werden im Allgemeinen modellbasierte *Prognosemethoden* eingesetzt, die die historische Entwicklung der Zeitreihe in einem *Prognosemodell* abbilden. Weit verbreitete und oft eingesetzte Vertreter von Prognosemethoden sind beispielsweise das exponentielle Glätten und die ARIMA-Modelle [BJR08]. Die modellbasierte Zeitreihenprognose besteht aus drei wesentlichen Grundschritten. Der erste Schritt, die *Modellerstellung*, beinhaltet die Definition der relevanten Variablen, der Prognosemethode sowie die Schätzung der *Modellparameter*. Für die Parameterschätzung sind dabei oft aufwendige Optimierungsverfahren erforderlich, sodass dieser Schritt die zeitaufwändigste Komponente bei der modellbasierten Prognose darstellt. Ein einmal erstelltes Modell kann nun wiederholt und kostengünstig für die Berechnung der Prognosewerte verwendet werden (*Modellnutzung*). Treffen neue Zeitreihenwerte im System ein, ist gegebenfalls eine *Modellwartung* erforderlich. Dies beinhaltet zum einen eine Evaluierung des vorhandenen Modells und zum anderen eine Anpassung des Modells an die neuen Zeitreihencharakteristiken, die oft eine komplette Neuschätzung der Modellparameter nach sich zieht.

Systemanforderungen Die Zeitreihenprognose ist eine wichtige Analysetechnik für eine Vielzahl von Anwendungsgebieten. Damit sollte ein System, das sowohl den vorgestellten Prognoseprozess als auch typische Applikationsanforderungen unterstützt, gewisse Voraussetzungen erfüllen. Zunächst müssen weit verbreitete Zeitreihenprognosemethoden für die Modellerstellung verfügbar sein. Weiterhin ist eine generische Schnittstelle wünschenswert, die es ermöglicht, beliebige, domänenspezifische Prognosemethoden zum System hinzuzufügen. Eine einfache und deklarative Anfragesprache erlaubt es jedem Nutzer, auch ohne statistisches Hintergrundwissen Prognosen zu berechnen. Echtzeitanforderungen (z.B. weniger als eine Sekunde im Online Werbebereich) auf großen multidimensionalen Daten erfordern eine effiziente Erstellung von Prognosemodellen. Da aber gerade dies sehr zeitaufwändig ist, sollten Mechanismen zur Speicherung und Wiederverwendung von Prognosemodellen zur Verfügung gestellt werden, möglichst auch für andere Anfragen und Nutzer. Schlussendlich ist eine automatisierte Wartungsumgebung erforderlich, die Prognosemodelle transparent an neue Zeitreihencharakteristiken anpasst und die teure Modellwartung nur bei Bedarf ausführt.

Architektonische Umsetzung Existierende Ansätze zur Umsetzung statistischer Verfahren in Datenbankmanagementsystemen können in (1) externe Verfahren, (2) partielle Integrationsverfahren und (3) vollständige Integrationsverfahren klassifiziert werden. Externe Verfahren nutzen dedizierte statistische Programme (Matlab, R) und unterstützen dabei eine Vielzahl komplexer Prognosemethoden. Allerdings ist ein Export der Daten aus der Datenbank notwendig, es gibt keine Möglichkeit, deklarative Prognoseanfragen zu stellen und auch keine Mechanismen zur Speicherung, Wiederverwendung und automatisierten Wartung von Prognosemodellen. Partielle Integrationsansätze [CDD+09, GLW+11] gehen einen Schritt weiter und bieten eine eingeschränkte Integration der Zeitreihenprognose an. Jedoch wird hier der Prognoseprozess oft als Black-Box Ansatz innerhalb einer benutzerdefinierten Funktion oder eines R-Skriptes betrachtet, womit keine effiziente Optimierung, Wiederverwendung und Wartung von Modellen möglich ist. Vollständige Integrationsansätze, wie beispielsweise MauveDB [DM06], schlagen bereits eine vollständige Integration von statistischen Modellen als native Datenbankobjekte vor und bieten bei-

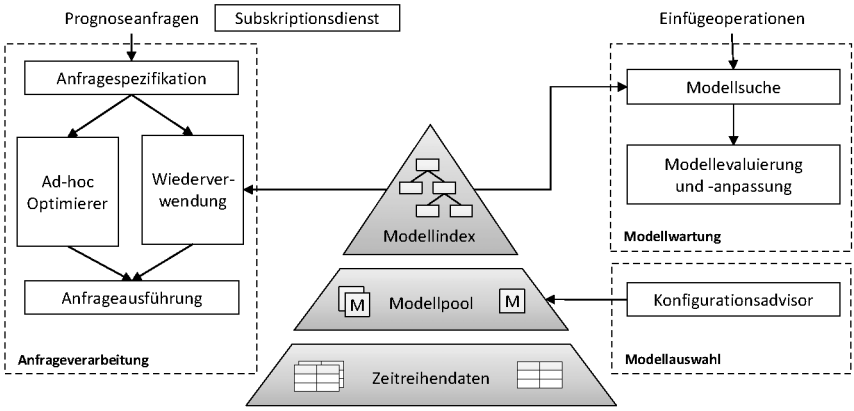


Abbildung 1: Flash-Forward Datenbanksystem F^{2}DB

spielsweise auch Wartungsmechansimen an. Existierende Ansätze adressieren die Zeitreihenprognose dabei jedoch nicht oder nur sehr eingeschränkt und erfordern weiterhin die explizite Spezifikation eines Modells in einer Anfrage.

3 Konzeptionelle Systemarchitektur

Basierend auf den vorgestellten Systemanforderungen und den Limitierungen existierender Ansätze erfolgte im Rahmen der Dissertation die Konzeption und prototypische Umsetzung eines Gesamtsystems mit dem Namen F^{2}DB (*Flash-Forward Datenbanksystem*) [FRL12a, FDS+12], welches eine vollständige Integration der modellbasierten Zeitreihenprognose in ein relationales Datenbanksystem realisiert. Der Prototyp wurde dabei auf Basis des frei verfügbaren Datenbanksystems PostgreSQL entwickelt. Die wesentlichen konzeptionellen Komponenten von F^{2}DB sind in Abbildung 1 dargestellt.

Zeitreihen stellen eine spezielle Sicht auf die Daten dar und können mittels beliebiger Anfragen auf den klassischen Datenbanktabellen erstellt werden, ähnlich wie bei klassischen Sichten. Prognosemodelle werden als native Datenbankobjekte in einem gemeinsamen *Modellpool* abgelegt. Ein *Modellindex* sichert das schnelle Auffinden existierender Modelle für eintreffende Prognoseanfragen und Einfügeoperationen.

Die Verarbeitung deklarativer Prognoseanfragen erfordert zunächst die Erweiterung der Anfrageverarbeitung eines traditionellen Datenbanksystems. Die *Anfragespezifikation* erkennt neue prognosespezifische Schlüsselwörter und überführt die Anfrage in einen internen, logischen Anfragebaum. Ausgehend von der initialen Analyse der Prognoseanfrage ergeben sich zwei grundsätzliche Pfade zur Anfrageverarbeitung. Auf der einen Seite können neue Prognosemodelle zur Laufzeit erstellt werden, wobei die Modellerstellung im Rahmen der *Anfrageausführung* mittels neuer physischer Prognoseoperatoren umgesetzt ist. Der *Ad-hoc Optimierer* hat dabei die Aufgabe, die Laufzeit und den Prognosefehler

derartiger Ad-hoc Prognoseanfragen zu minimieren. Auf der anderen Seite können existierende Modelle aus dem Modellindex geladen und für die Berechnung der Prognose *wiederverwendet* werden. Neben der Anfrageverarbeitung ergeben sich Änderungen bei der Verarbeitung von Einfügeoperationen, da nun eine Wartung der materialisierten Prognosemodelle notwendig ist. Dazu müssen die von einer Einfügeoperation betroffenen Modelle im Modellindex *gesucht*, *evaluiert* und bei Bedarf *angepasst* werden. Daraus ergibt sich die Frage, welche Modelle im Modellindex abgelegt werden sollten, um zum einen die Wartungskosten zu minimieren, zum anderen aber eine möglichst effiziente und genaue Wiederverwendung zu ermöglichen. Basierend auf einer gegebenen Anfragelast schlägt der *Konfigurationsadvisor* eine möglichst optimale Konfiugration von Modellen für den Modellpool vor. Eine letzte Komponente, der *Subskriptionsdienst*, ermöglicht die einmalige Registrierung kontinuierlicher Prognoseanfragen. Neu eintreffende Zeitreihenwerte führen dann zu automatischen Notifikationen an den Subskribenten.

4 Verarbeitung und Optimierung von Ad-hoc Prognoseanfragen

In diesem Abschnitt diskutieren wir zunächst die Grundlagen der Verarbeitung und Optimierung von Ad-hoc Prognoseanfragen.

Spezifikation und Ausführung von Prognoseanfragen Prognoseanfragen werden mittels eines einfachen Schlüsselwortes `AS OF` spezifiziert, welches den Prognosehorizont bestimmt [FRL12a]. Eine einfache Prognoseanfrage könnte beispielsweise Verkaufsprognosen für Telefone im nächsten Monat abfragen:

```
SELECT    orderdate, quantity
FROM      facts
WHERE     pgroup = 'phones'
GROUP BY  orderdate
AS OF     now() + interval '3 months'
```

Das SQL Konstrukt `AS OF` ist dabei bereits Teil des SQL:2011 Standards und wurde bis jetzt verwendet, um eine Sicht auf die Daten in der Vergangenheit zu ermöglichen (Flash-Back Anfragen). In unserem System ist es nun möglich, mit dem selben Sprachkonstrukt eine Sicht auf die zukünftigen Daten abzufragen (Flash-Forward Anfragen). Optionale zusätzliche Schlüsselworte ermöglichen die Spezifikation von abhängigen und unabhängigen Variablen oder der Prognosemethode.

Analog zur klassischen Anfrageverarbeitung wird die Prognoseanfrage in einen internen, logischen Anfragebaum transformiert, wofür wir einen neuen, logischen *Prognoseoperator* eingeführt haben. Auf physischer Ebene wird der logische Prognoseoperator durch zwei wesentliche, physische Operatoren repräsentiert - den Operator zur Modellerstellung und den Operator zur Modellnutzung. Die entwickelten Operatoren sind dabei unabhängig vom eigentlichen Prognoseverfahren und nutzen vordefinierte Funktionen einer generischen Schnittstelle. Diese erlaubt es Datenbankadministratoren, beliebige domänenspezifische Prognosemethoden in F^2DB zu integrieren.

Im Gegensatz zur klassischen Anfrageverarbeitung sind Prognosewerte allerdings per se

ungenau. Dies führt zu einer neuen Semantik bei der Restrukturierung und Optimierung von Anfrageplänen. Unterschiedliche Anfragepläne beeinflussen nicht nur die Laufzeit sondern auch die Genauigkeit des Anfrageergebnisses. Im Rahmen der Dissertation haben wir deshalb sowohl exakte als auch approximative Restrukturierungsregeln untersucht, neue Kostenmodelle für die physischen Operatoren eingeführt und spezielle stichprobenbasierte Optimierungstechniken für zwei Gruppen von Prognoseanfragen entwickelt.

Stichprobenbasierte Anfrageoptimierung Stichprobenverfahren führen zu einer Reduktion der zu verarbeitenden Datenmenge und wurden in einer Vielzahl von Bereichen erfolgreich angewandt. Im Falle von Prognoseanfragen nutzen wir Stichprobenverfahren auf zwei verschiedene Arten: *vertikales Sampling* und *horizontales Sampling*. Beide Verfahren haben zum Ziel, die Datenmenge und damit die Anfragelaufzeit mit keinem oder nur sehr geringem Genauigkeitsverlust zu reduzieren. Vertikales Sampling adressiert Aggregationsanfragen, die eine aggregierte Prognose über eine Vielzahl von Einzelzeitreihen berechnen [FRL12b]. Die Verwendung von Einzelmodellen ist dabei oft sehr teuer, ein Modell über die aggregierte historische Zeitreihe liefert eine geringere Genauigkeit. Vertikales Sampling verwendet stattdessen nur eine Stichprobe der Einzelmodelle und gewinnt die Prognose durch eine Hochrechnung der Einzelprognosen. Spezielle Stichprobenschätzer, die auf den historischen Entwicklungen der Zeitreihen basieren, stellen eine hohe Genauigkeit selbst bei kleinen Stichprobengrößen sicher. Horizontales Sampling reduziert die Länge historischer Zeitreihen in Gruppierungsanfragen mit dem Ziel der Beschleunigung der Modellerstellung. Dazu erfolgt eine Analyse der optimalen Länge für eine kleine, ausgewählte Anzahl von Zeitreihen. Anschließend wird der aktuelle Plan neu parametriert, sodass für spätere Zeitreihen im Plan eine kürzere Historie verwendet werden kann.

Evaluierung Im Rahmen der Evaluierung haben wir sowohl den Laufzeitgewinn der vollständigen Datenbankintegration der Zeitreihenprognose gegenüber alternativen Integrationsansätzen als auch die Vorteile der vorgeschlagenen Optimierungsverfahren untersucht. Um möglichst unterschiedliche Anfragecharakteristiken abzubilden, haben wir dafür den klassischen TPC-H Benchmark angepasst. Dies erforderte zum einen Anpassungen bei der Datengenerierung, sodass Zeitreihendaten anstatt zufälliger Werte erstellt werden, und zum anderen Anpassungen bei der Anfragegenerierung, sodass Prognoseanfragen anstatt historischer Anfragen an das Datenbanksystem gesendet werden. Die Evaluierung zeigte eine signifikante Laufzeitreduzierung der vollständigen Integration gegenüber der externen Ausführung von Prognoseanfragen beziehungsweise partiellen Integrationsansätzen. Eine weitere Verbesserung der Laufzeit wurde durch die vorgestellten stichprobenbasierten Optimierungstechniken erreicht.

5 Materialisierung von Prognosemodellen

Die Vorauswahl und Materialisierung von Prognosemodellen ermöglicht es, sowohl die Laufzeit als auch die Genauigkeit von Prognoseanfragen weiter zu reduzieren. Dabei ist unser Ziel, die deklarative Verarbeitung von Prognoseanfragen beizubehalten, wofür wir zunächst einen Mechanismus für die transparente Wiederverwendung und Wartung von Prognosemodellen benötigen.

Wiederverwendung und Wartung von Prognosemodellen Die Materialisierung von Modellen wird durch zwei neue Datenstrukturen unterstützt. Ähnlich zu klassischen Datenbankindexen werden physische Prognosemodelle (Modellparameter und -zustand) in einem *Modellpool* gespeichert. Zur schnellen Identifikation passender Modelle haben wir einen *Modellindex* entwickelt, der komplett im Hauptspeicher gehalten wird [FRBL10]. Ein Prädikatenindex indexiert dabei atomare Prädikate, welche dann zu beliebigen Anfrageausdrücken verknüpft werden. Einmal erstellte und indizierte Modelle werden automatisch und transparent für Prognoseanfragen verwendet. Ein Suchalgorithmus findet hierbei während der Anfrageoptimierung passende Modelle für eine gegebene Anfrage. Wird kein passendes Modell gefunden, muss zur Laufzeit ein neues Modell erstellt werden, welches dann im Modellindex abgelegt wird und damit zukünftigen Anfragen zur Verfügung steht. Neu eintreffende Daten erfordern die Wartung der Prognosemodelle. Hierbei ist erneut eine Suche im Modellindex erforderlich, um die betroffenen Modelle zu identifizieren. Nach der Aktualisierung des Modellzustandes (interne Historie) erfolgt eine Evaluierung des Modells, um die Notwendigkeit einer Modellwartung festzustellen. Dafür haben wir unterschiedliche Evaluierungsstrategien in F^2DB integriert. Zeigt die Evaluierung, dass eine Neuschätzung der Modellparameter erforderlich ist, dann wird das Modell markiert und bei der nächsten Referenzierung durch eine Anfrage gewartet.

Optimierung von Modellkonfigurationen Zeitreihen in typischen Datenbanksystemen sind entlang einer Vielzahl von Dimensionen und Aggregationsebenen organisiert, auf denen traditionelle Anfragen mit Drill-Down und Roll-Up Operationen arbeiten. Die Erstellung und Wartung eines Modells für jede einzelne Zeitreihe ist extrem aufwändig und kaum praktikabel. Analog zu materialisierten Sichten können Ableitungsschemata genutzt werden, um Modelle einzusparen, beispielsweise durch Aggregation von Prognosewerten. Allerdings und damit im Gegensatz zu materialisierten Sichten haben derartige Ableitungsschemata Einfluss auf die Genauigkeit der Prognosen. Interessanterweise kann die Genauigkeit sogar durch Ableitungen erhöht werden. Beispielsweise haben empirische Studien gezeigt, dass bei verrauschten Zeitreihen, Modelle auf höherer Aggregationsebene zu bevorzugen sind, während niedrige Aggregationsebenen zu einer höheren Genauigkeit bei sehr regelmäßigen Zeitreihen führen können. In der Dissertation haben wir dazu ein verallgemeinertes Ableitungsmodell konzeptioniert, das die Ableitung einer Zielzeitreihe aus einer beliebigen Anzahl von Quellzeitreihen erlaubt.

Konfigurationsadvisor Basierend auf diesem Ableitungsmodell bestimmt der Konfigurationsadvisor für eine gegebene Anfragelast eine möglichst optimale Konfiguration von Modellen [FBL11, FSHL13]. Eine Modellkonfiguration besteht dabei aus einer Menge von Modellen und einer Menge von Ableitungsschemata, die beschreiben wie die Modelle zur Anfrageverarbeitung zu nutzen sind. Ziel des Konfigurationsadvisors ist nun zum einen die Maximierung der Prognosegenauigkeit für die gegebene Anfragelast und zum anderen die Minimierung der dafür notwendigen Modelle, d.h. der Modellwartungskosten. Während die Modellkosten einfach zu berechnen sind, lässt sich die Genauigkeit nicht mathematisch bestimmen. Hier müssen Modelle zunächst erstellt und empirisch evaluiert werden. Gerade bei großen Datensätzen ist die Erstellung aller Modelle und die Überprüfung aller möglichen Ableitungsschemata allerdings kaum praktikabel.

Der Konfigurationsadvisor arbeitet deshalb nach einem iterativen Prinzip (Abbildung 2).

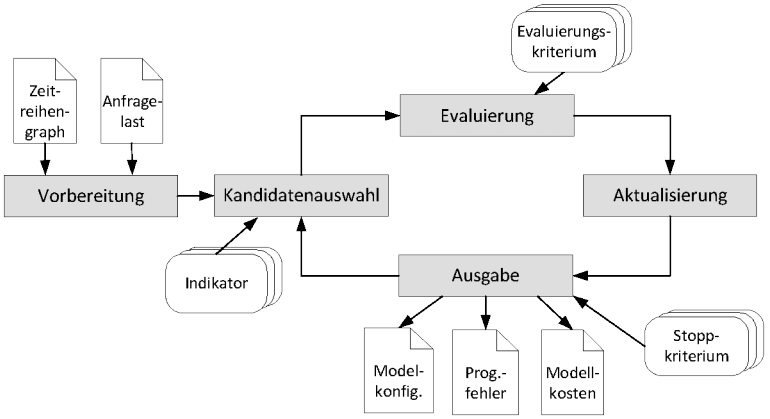


Abbildung 2: Iterative Ausführung des Konfigurationsadvisors

Er erhält als Eingabe den aktuellen Datensatz, repräsentiert als Zeitreihenhypergraph, und die Anfragelast. Nach der Erstellung einer Initialkonfiguration (*Vorbereitung*) wird eine iterative Prozesskette ausgeführt. Basierend auf Heuristiken, sogenannten Indikatoren, wird in jeder Iteration eine Menge von Modellen ausgewählt (*Kandidatenauswahl*) und für diese empirisch die Genauigkeit für die gegebenen Anfragen bestimmt (*Evaluierung*). Am Ende jeder Iteration werden die aktuellen Genauigkeiten und Wartungskosten der Modellkonfiguration ausgeben (*Ausgabe*), sodass ein automatischer oder manueller Abbruch des Advisors erfolgen könnte. Diverse Parameter (zum Beispiel die Reduktion der Kandidatenanzahl oder Ableitungsmöglichkeiten) erlauben die Skalierung des Advisors auch für größere Datensätze (*Aktualisierung*). Die Evaluierung auf realen und synthetischen Daten zeigte, dass der Advisor effizient und auch auf großen Datensätzen eine Modellkonfiguration mit hoher Genauigkeit und geringen Wartungskosten berechnet.

6 Subskriptionsbasierte Prognoseanfragen

Abschließend erweitern wir Prognoseanfragen um kontinuierliche Aspekte, die es einer Applikation ermöglichen, Anfragen einmalig im F^{2}DB System zu registrieren [FKK+12, FBLP12]. Dies erlaubt es Applikationen kontinuierliche Prognoseergebnisse zu erhalten und ermöglicht es, diese weiter zu verarbeiten. Ein wichtiger Anwendungsbereich sind beispielsweise intelligente Stromnetze, so genannte Smart Grids. Hier ist eine kontinuierliche Balancierung von Energienachfrage und -angebot notwendig, wofür kontinuierlich Notifikationen mit neuen Prognoseergebnissen an die Applikation verschickt werden müssen.

Eine subskriptionsbasierte Prognoseanfrage registriert sich am Datenbanksystem mit einem Prognosehorizont und einer Fehlergrenze. Das Datenbanksystem schickt nur dann neue Notifikationen, wenn der Prognosehorizont ausgeschöpft ist oder eine signifikante Abweichung von der Fehlergrenze festgestellt wurde. Aufgabe des *Subskriptionsdienstes* ist dabei die Optimierung derartiger Notifikationen zur Reduktion der Kommunikations-

und Applikationskosten. Dazu besteht die Möglichkeit, auch mehr Prognosewerte als gefordert, das heißt einen größeren Prognosehorizont, zu versenden. Dies hat den Vorteil, dass der Prognosehorizont länger gültig bleibt und damit weniger Notifikationen versendet werden müssen. Auf der anderen Seite zeigen sehr lange Prognosehorizonte einen höheren Prognosefehler, womit eine schnellere Überschreitung der Fehlergrenze zu befürchten ist.

Die Grundlage zur Bestimmung des optimalen Prognosehorizontes bildet ein Kostenmodell, welches die Verarbeitungskosten der Applikation abbildet. Weiterhin unterscheiden wir drei verschiedene Optimierungsansätze, die basierend auf dem entwickelten Kostenmodell und der historischen Zeitreihe einen festen Prognosehorizont, variable Prognosehorizonte für verschiedene Zeitscheiben sowie eine dynamische Anpassung des Prognosehorizontes berechnen. Die abschließende Evaluierung zeigte einen geringen Aufwand der Optimierungsansätze, eine signifikante Reduktion der Applikationskosten sowie die Gültigkeit des Kostenmodells auf realen Datensätzen.

7 Zusammenfassung

Wir haben einen Ansatz zur vollständigen Integration der Zeitreihenprognose in ein Datenbankmanagementsystem vorgestellt. Deklarative Prognoseanfragen ermöglichen dabei die einfache Abfrage von Prognoseergebnissen und erfordern keine statistischen Kenntnisse des Nutzers. Prognoseergebnisse werden direkt in der Datenbank berechnet und verarbeitet, und können mit anderen Quelldaten verknüpft werden (z.B. mittels Join-Operatoren). Analog zu klassischen Indexstrukturen werden Prognosemodelle als native Datenbankobjekte unterstützt. Dies erlaubt die transparente Speicherung, Wiederverwendung und Wartung von Prognosemodellen. Zusammenfassend haben wir im Rahmen dieser Dissertation ein neuartiges Flash-Forward Datenbanksystem entwickelt und viele weitere interessante Forschungsmöglichkeiten eröffnet. Zukünftige Forschungsarbeiten könnten beispielsweise neue Anfrageoptimierungstechniken integrieren, die Parallelisierung von Prognoseoperatoren betrachten oder die Online Wartung von Modellkonfigurationen untersuchen.

Literatur

[BJR08] George E. P. Box, Gwilym M. Jenkins und Gregory C. Reinsel. *Time Series Analysis: Forecasting and Control, 4th ed.* Wiley, 2008.

[CDD+09] Jeffrey Cohen, Brian Dolan, Mark Dunlap, Joseph M. Hellerstein und Caleb Welton. MAD Skills: New Analysis Practices for Big Data. *PVLDB*, 2:1481–1492, 2009.

[DM06] Amol Deshpande und Samuel Madden. MauveDB: Supporting Model-based User Views in Database Systems. In *SIGMOD*, Seiten 73–84, 2006.

[FBL11] Ulrike Fischer, Matthias Böhm und Wolfgang Lehner. Offline Design Tuning for Hierarchies of Forecast Models. In *BTW*, Seiten 167–186, 2011.

[FBLP12] Ulrike Fischer, Matthias Boehm, Wolfgang Lehner und Torben Bach Pedersen. Optimizing Notifications of Subscription-Based Forecast Queries. In *SSDBM*, Seiten 449–466, 2012.

[FDS+12] Ulrike Fischer, Lars Dannecker, Laurynas Siksnys, Frank Rosenthal, Matthias Boehm und Wolfgang Lehner. Towards Integrated Data Analytics: Time Series Forecasting in DBMS. *Datenbank-Spektrum*, Seiten 1–9, 2012.

[Fis14] Ulrike Fischer. *Forecasting in Database Systems*. Dissertation, Technische Universität Dresden, 2014.

[FKK+12] Ulrike Fischer, Dalia Kaulakiene, Mohamed E. Khalefa, Wolfgang Lehner, Laurynas Siksnys Torben Bach Pedersen und Christian Thomsen. Real-time Business Intelligence in the MIRABEL Smart Grid System. In *BIRTE*, Seiten 1–22, 2012.

[FL14] Ulrike Fischer und Wolfgang Lehner. Transparent Forecasting Strategies in Database Management Systems. In *Business Intelligence*. Springer Berlin Heidelberg, to appear 2014.

[FRBL10] Ulrike Fischer, Frank Rosenthal, Matthias Böhm und Wolfgang Lehner. Indexing Forecast Models for Matching and Maintenance. In *IDEAS*, Seiten 26–31, 2010.

[FRL12a] Ulrike Fischer, Frank Rosenthal und Wolfgang Lehner. F2DB: The Flash-Forward Database System. In *ICDE*, Seiten 1245–1248, 2012.

[FRL12b] Ulrike Fischer, Frank Rosenthal und Wolfgang Lehner. Sample-Based Forecasting Exploiting Hierarchical Time Series. In *IDEAS*, Seiten 120–129, 2012.

[FSHL13] Ulrike Fischer, Christopher Schildt, Claudio Hartmann und Wolfgang Lehner. Forecasting the Data Cube: A Model Configuration Advisor for Multi-Dimensional Data Sets. In *ICDE*, 2013.

[GH06] Jan G. De Gooijera und Rob J. Hyndman. 25 Years of Time Series Forecasting. *International Journal of Forecasting*, 22:443–473, 2006.

[GLW+11] Philipp Große, Wolfgang Lehner, Thomas Weichert, Franz Färber und Wen-Syan Li. Bridging Two Worlds with RICE Integrating R into the SAP In-Memory Computing Engine. *PVLDB*, 4:1307–1317, 2011.

Ulrike Fischer hat von 2003 bis 2009 Informatik an der Technischen Universität Dresden studiert. Seit 2009 ist sie dort als wissenschaftliche Mitarbeiterin am Lehrstuhl für Datenbanken tätig. Zu ihren Forschungsschwerpunkten zählen die Zeitreihenprognose, insbesondere im Energiebereich, und die Integration von analytischen Verfahren in Datenbankmanagementsysteme. Im Jahr 2013 schloss sie ihre Dissertation mit dem Titel "Forecasting in Database Systems" mit Auszeichnung ab. Sie hat weiterhin zu einer Vielzahl von Forschungsprojekten beigetragen, unter anderem dem DFG-geförderten Grundlagenprojekt FFQ (Flash-Forward Query Framework) und dem EU-geförderten MIRABEL Verbundprojekt (Micro-Request-Based Aggregation, Forecasting and Scheduling of Energy Demand, Supply and Distribution). Letzteres beschäftigt sich mit der automatisierten Balancierung moderner Energienetze, wofür effiziente und genau Energieprognosen eine wichtige Voraussetzung bilden.

Modellbasierte Analyse zerebrovaskulärer Erkrankungen durch Kombination von 3D und 4D MRA Datensätzen*

Nils Daniel Forkert

Department Informatik, Universität Hamburg
nforkert@uke.uni-hamburg.de

Abstract: Der Schlaganfall ist eine der häufigsten Todesursachen und Hauptgrund für schwere Behinderungen in Deutschland. Moderne Bildgebungstechniken ermöglichen eine präzise 3D-Abbildung der Gefäßanatomie und der zeitlichen Blutflussdynamik bzw. der Perfusion, was die Basis für Diagnose, Therapie sowie für die bildgestützte Forschung darstellt. Die Begutachtung dieser immensen Bilddaten kann allerdings zeitintensiv und auch sehr komplex sein. In der hier zusammengefassten Dissertation wurden Algorithmen und Methoden entwickelt, die es erlauben mehre 3-dimensionale und zeitaufgelöste Bildsequenzen eines Patienten computergestützt und kombiniert zu analysieren und visualisieren. Dieses ermöglicht nicht nur die patientenspezifische Struktur der Blutgefäße zu segmentieren und quantitativ auszuwerten, sondern auch den entsprechenden Blutfluss im Gefäß und Gehirngewebe. Neben der Möglichkeit einer direkten Verwendung in der Diagnostik, wurden die entwickelten Methoden auch bereits in mehreren klinischen Studien eingesetzt.

1 Einführung

Der zerebrale Schlaganfall ist weltweit eine der Haupttodesursachen und ein Hauptgrund für neurologische Defizite [DFMD08]. Es wird geschätzt, dass jeder zwanzigste Erwachsene in seinem Leben einen Schlaganfall erleiden wird [FLBCP09]. Die alternde Bevölkerung wird voraussichtlich zu einem weiteren Anstieg der Inzidenz zerebraler Schlaganfälle führen. Daher wird der optimalen Behandlung von Schlaganfallpatienten zukünftig eine noch größere Bedeutung zukommen, als das schon zurzeit der Fall ist.
Ungefähr 72% aller zerebralen Schlaganfälle zählen zu den ischämischen Schlaganfällen [TDM+01]. Hierbei kommt es zu dem Verschluss einer Arterie, was zu einer partiellen bis hin zu einer vollständigen Einschränkung der zerebralen Perfusion in den versorgten Gehirnarealen führt. Im Gegensatz dazu werden ca. 20% aller Schlaganfälle durch eine akute Blutung aus einem Blutgefäß aufgrund einer zerebralen Gefäßerkrankung, wie zum Beispiel Aneurysmen und arteriovenöse Malformationen, hervorgerufen [TDM+01]. In beiden Fällen ist eine genaue Analyse der zerebralen Gefäßanatomie notwendig, um eine bestmögliche Diagnose, Therapieentscheidung, Intervention und Nachuntersuchung zu

*Englischer Titel der Dissertation: "Model-Based Analysis of Cerebrovascular Diseases Combining 3D and 4D MRA Datasets"

ermöglichen. Darüber hinaus bietet ein detailliertes Verständnis des zerebralen Blutflusses auf makrovaskulärer Ebene und/oder der mikrovaskulären Perfusion weitere Vorteile für viele Fragestellungen in der klinischen Praxis.
Ein fundiertes Wissen über diese Aspekte kann zum Beispiel aus Magnetresonanztomographie (MRT) Bildsequenzen extrahiert werden. Ein typischer MRT Datensatz, der sowohl hochaufgelöste 3D als auch zeitaufgelöste (4D) Bildsequenzen enthält, führt jedoch auch zu einer immensen Masse an Daten, die in der klinischen Praxis durch Ärzte visuell analysiert werden muss. Eine genaue Diagnose ist daher nicht nur sehr zeitaufwändig, sondern auch fehleranfällig. Computergestützte Analysemethoden eines solchen MRT Datensatzes könnten nicht nur die Diagnose selbst verbessern, sondern auch die Zeit, die hierfür benötigt wird, verringern.
Aus methodischer Sicht sind hierbei drei Probleme von besonderer Bedeutung, die im Fokus dieser Arbeit stehen: Die automatische Segmentierung der zerebralen Blutgefäße, die Analyse der Hämodynamik auf Basis von zeitaufgelösten MRT Datensätzen sowie die kombinierte Analyse und Visualisierung von zerebralen Gefäßen und Hämodynamik.

2 Methoden

2.1 Gefäßsegmentierung

In der Vergangenheit wurde eine Vielzahl automatischer Segmentierungsmethoden zur Extraktion von zerebralen Gefäßen aus räumlich hochaufgelösten 3D Magnetresonanzangiographie (MRA) Datensätzen vorgestellt. Für eine detaillierte Übersicht zu Gefäßsegmentierungsansätzen sei an dieser Stelle auf die Arbeit von Lesage et al. [LABFL09] verwiesen. Die meisten dieser Methoden führen jedoch dazu, dass die Ergebnisse einen Kompromiss zwischen ausreichender Präzision bei der Detektion von kleinen oder von pathologischen Gefäßen darstellen.
Aus diesem Grund wurde im Rahmen der Promotionsarbeit ein mehrstufiges Segmentierungsverfahren entwickelt, das in der Lage ist, sowohl kleine als auch pathologische Blutgefäße mit hoher Präzision zu segmentieren. In einem ersten Schritt wird hierbei das Gehirngewebe mittels eines graphenbasierten Ansatzes aus einem 3D MRA Datensatz extrahiert [FSF+09], um in den folgenden Schritten eine Übersegmentierung von nicht-zerebralen Strukturen zu verhindern. Im zweiten Schritt werden dann Intensitäts- und Vesselness-Werte mittels Fuzzy Logik kombiniert [FSRF+11], wobei der Vesselness-Parameter [SNS+98] die lokale Ähnlichkeit zu einer tubulären Struktur quantifiziert, was insbesondere eine verbesserte Darstellung kleiner Gefäße ermöglicht. Die nicht-lineare Kombination von Intensitäts- und Vesselnesswerten mittels Fuzzy-Logik hat den Vorteil, dass in den hieraus resultierenden Parameterbildern sowohl pathologische als auch kleine Gefäße im Vergleich zum Hintergrund hervorgehoben werden. Im dritten Schritt des mehrstufigen Segmentierungsverfahrens wird das Gefäßsystem aus diesem Fuzzy-Datensatz mittels einer Level-Set Methode extrahiert, welche unter Zuhilfenahme der Richtungsinformation aus dem Vesselnessfilter eine verbesserte Gefäßsegmentierung erlaubt [FSRF+12a]. Hierdurch wird insbesondere ermöglicht, dass die interne Energie nicht

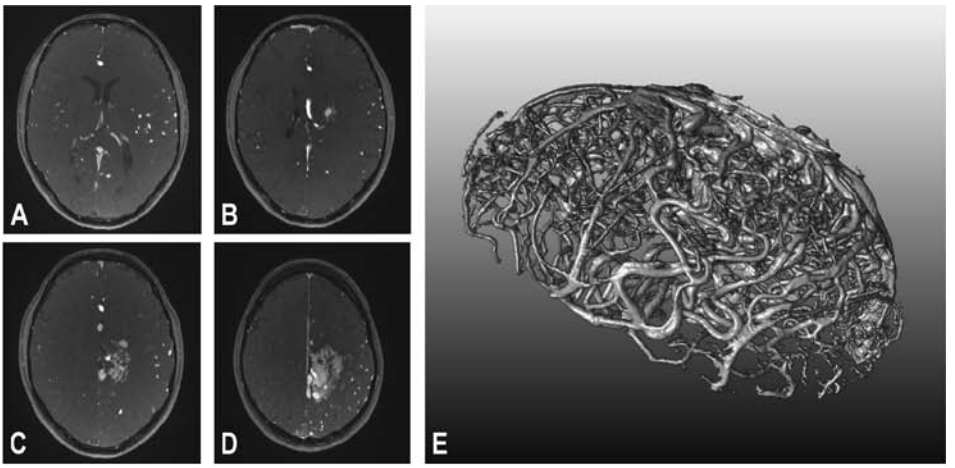


Abbildung 1: Vier ausgewählte Schichten aus einem 3D MRA Datensatz von einem Patienten mit einer arteriovenösen Malformation (A-D) und das korrespondierende 3D Oberflächenmodell des segmentierten Gefäßsystems (E).

die Level-Set Evolution in kleine Gefäße verhindert. Trotz dieser Adaption weisen die resultieren Segmentierungen immer noch Unterbrechungen von Gefäßen, insbesondere von sehr dünnen Blutgefäßen, auf. Diese werden im letzten Schritt des vorgestellten Segmentierungsansatzes korrigiert [FSRF+12b]. Hierfür werden zunächst alle Gefäßenden in der vorhandenen Gefäßsegmentierung bestimmt, die dann unter Verwendung des Dijkstra-Algorithmus im dreidimensionalen Raum zu anderen bereits segmentierten Gefäßen verbunden werden. Die extrahierten Pfade werden mittels eines Level-Set Ansatzes an die Gefäßgrenzen expandiert und diese dann mit der ursprünglichen Gefäßsegmentierung kombiniert.

2.2 Analyse der Hämodynamik und Perfusion

Die automatische Segmentierung der zerebralen Gefäße kann zum Beispiel für eine oberflächenbasierte 3D-Visualisierung verwendet werden, was eine intuitive 3D-Darstellung des Gefäßsystems erlaubt (siehe Abb. 1). Eine solche Darstellung beinhaltet jedoch noch keine Information über die Hämodynamik, welche aber häufig von Interesse für die Diagnostik ist.

Für diesen Zweck müssen zusätzlich zu den räumlich hochaufgelösten MRA Datensätzen noch zeitaufgelöste 3D-Magnetresonanzangiographien analysiert werden, welche die zeitliche Dynamik des Kontrastmittels darstellen. In der Vergangenheit wurde eine Vielzahl an Parametern beschrieben, die sich auf Basis von zeitlichen Kontrastmittelkurven bestimmen lassen. Ein Parameter, der hierbei von generellem Interesse für die Analyse des Blutflusses sowohl auf makrovaskulärer Ebene als auch der Perfusion ist, ist der Time-to-Peak (TTP) Parameter. Typische Kontrastmittelkurven weisen jedoch nicht nur Rauschartefakte und

Rezirkulationen des Kontrastmittels auf, sondern bestehen auch nur aus diskreten Messpunkten, weshalb die direkte Bestimmung des TTP-Parameters häufig zu ungenau ist. Um dieses Problem zu vermeiden, wurden in der Vergangenheit mehrere parametrische Modelle, wie zum Beispiel das Gamma Variate Model [TSWM64] oder das Local Density Random Walk Model [BJvR$^+$86], vorgestellt, die vor der Parameterbestimmung an die diskreten Kurven angepasst werden (siehe Abb. 2). Diese Modelle machen jedoch sehr starke Annahmen über die Form von typischen Kontrastmittelkurven, die im Falle von zerebralen Gefäßkrankheiten häufig nicht mehr zutreffend sind. Deshalb wurde im Rahmen dieser Arbeit das Model der referenzbasierten Kurvenanpassung zur Analyse der Hämodynamik entwickelt [FFR$^+$11]. Hierbei wird eine patientenindividuelle Referenzkurve direkt aus dem zeitaufgelösten MRA Datensatz extrahiert und dann für die Kurvenanpassung verwendet. Genauer gesagt werden hierfür in einem ersten Schritt n Kontrastmittelkurven zufällig aus dem zeitaufgelösten MRA Datensatz extrahiert und hieraus mittels iterativer Kurvenanpassung eine Referenzkurve generiert. Nach einer B-Spline Approximation der Referenzkurve, um eine Abtastung unterhalb der zeitlichen Auflösung der Bildakquisition zu ermöglichen, wird der Referenz-TTP bestimmt. In einem folgenden Schritt wird die generierte Referenzkurve unter Berücksichtigung von Zeitverzögerung und Dispersion des Kontrastmittels, an jede Kontrastmittelkurve in dem zeitaufgelösten MRA Datensatz angepasst. Der TTP Wert für jede einzelne Kontrastmittelkurve kann durch Transformation des Referenz-TTP entsprechend der Optimierungsparameter aus der Kurvenanpassung bestimmt werden.

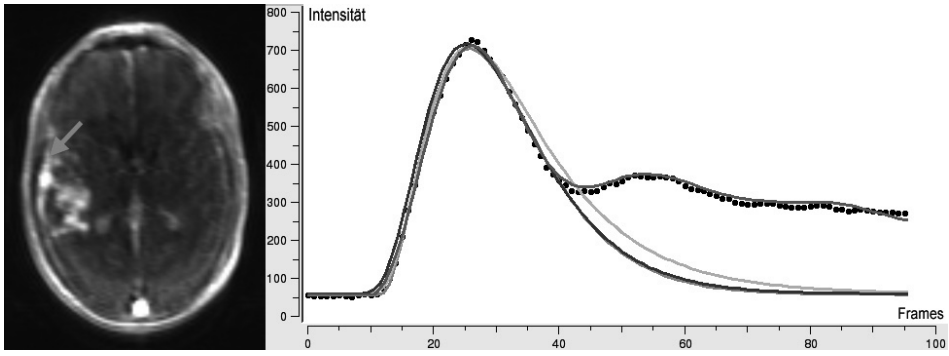


Abbildung 2: Beispiele für parametrische hämodynamische Modelle, die an eine diskrete zeitliche Kontrastmittelkurve (schwarz gepunktet) aus einem zeitaufglösten MRA Datensatz angepasst wurden: Gamma Variate Model (rot), Local Density Random Walk Model (blau), Log-Normal Model (orange) und die referenzbasierte lineare Kurvenanpassung (lila).

2.3 4D Blutflussvisualisierung

Die hämodynamische Analyse reduziert den zeitaufgelösten MRA Datensatz auf eine dreidimensionale TTP-Parameterkarte. Hierdurch wird zwar eine quantitative Auswertung der Hämodynamik möglich, jedoch noch keine direkte kombinierte Analyse und Visualisie-

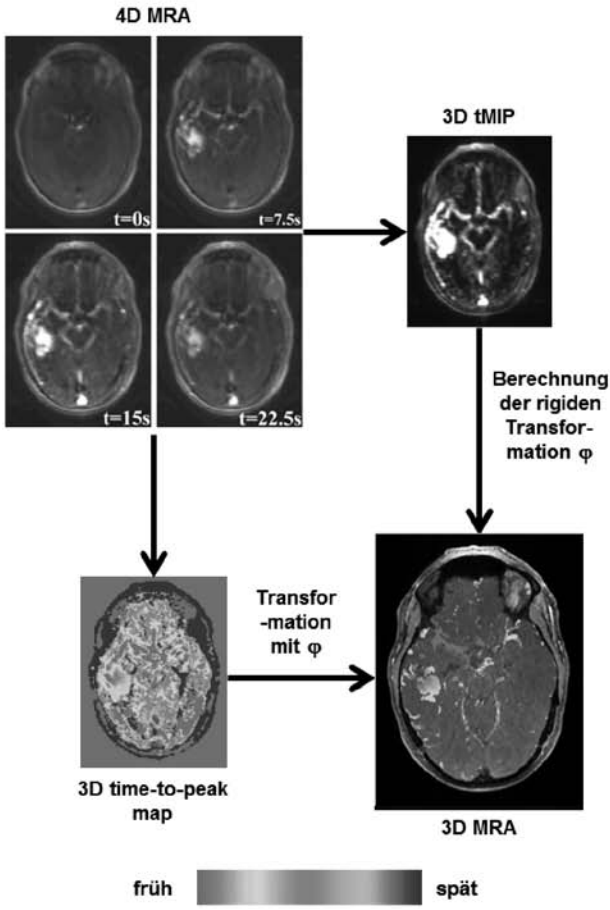


Abbildung 3: Illustration des Registrierungsverfahrens.

rung des zerebralen Gefäßsystems und dessen Hämodynamik. Aus diesem Grund wurde eine neue Methode zur 4D-Blutflussvisualisierung [FFI+12] entwickelt, die dieses Problem behebt. Hierbei wird der zeitaufgelöste MRA Datensatz eines Patienten zunächst unter Verwendung einer zeitlichen Maximumintensitätsprojektion (tMIP) auf einen 3D Datensatz reduziert. In dem entstehenden tMIP Datensatz sind alle Gefäße unabhängig von der Ankunftszeit des Kontrastmittels dargestellt, so dass dieser eine verbesserte Registrierung auf den entsprechend räumlich hochaufgelösten MRA Datensatz unter Verwendung einer rigiden Transformation und Maximierung der Mutual Information Metrik [WVA+96] ermöglicht. Die berechnete optimale rigide Transformation wird dann dazu verwendet, die TTP-Werte auf den hochaufgelösten MRA Datensatz zu übertragen und dort farbkodiert anzuzeigen (siehe Abb. 3). Darüber hinaus kann die hämodynamische Information farbkodiert und auch dynamisch auf dem Oberflächenmodell des Gefäßsystems visualisiert werden (siehe Abb. 4).

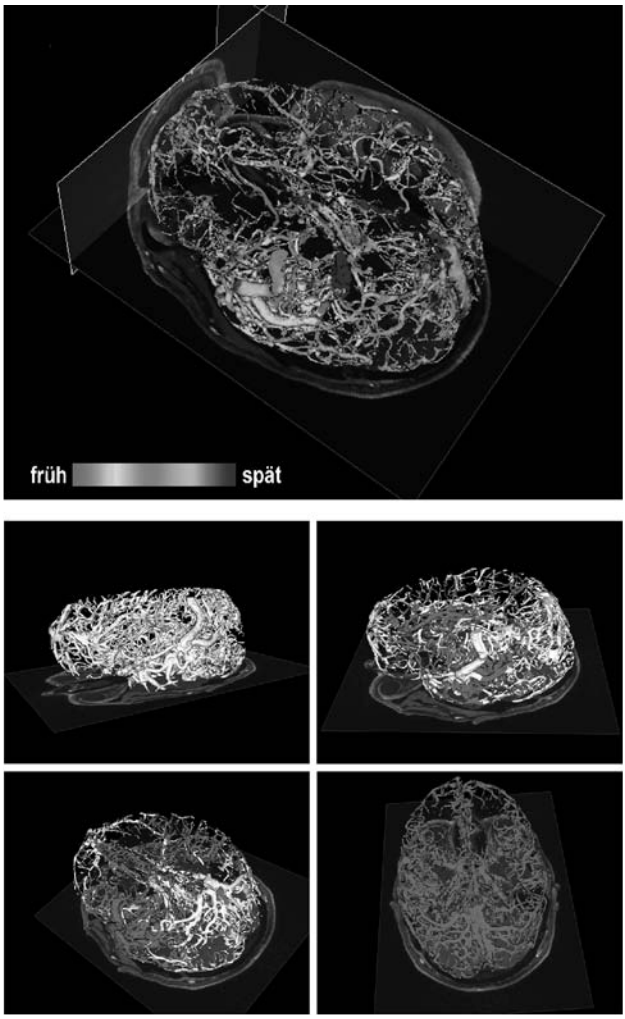


Abbildung 4: Farbkodierte statische Visualisierung des zerebralen Blutflusses auf dem 3D Oberflächenmodell sowie die korrespondierende 4D Blutflussvisualisierung aus verschiedenen Perspektiven für vier ausgewählte Zeitpunkte.

3 Ergebnisse

Das mehrstufige Segmentierungsverfahren zur Extraktion von zerebralen Blutgefäßen wurde auf Basis von 10 Time-of-Flight (TOF) MRA Datensätzen von Patienten mit einer arteriovenösen Malformation evaluiert. Hierbei standen zu jedem TOF-MRA Datensatz manuelle Gefäßsegmentierungen von zwei erfahrenen medizinischen Experten zur Verfügung, welche als Goldstandard gedient haben. Neben dem in dieser Arbeit beschriebenen mehrstufigen Segmentierungsansatz wurden noch vier weitere etablierte Methoden verwendet, um die Gefäße aus den MRA Datensätzen zu extrahieren. Für die quantitative Evaluation der Segmentierungsergebnisse wurde unter anderem der Dice-Koeffizient verwendet: $D(A,B) = \frac{2*|A \cap B|}{|A|+|B|}$, wobei Werte nahe 1 auf eine gute Übereinstimmung zwischen zwei Segmentierungen hindeuten. Insgesamt konnte gezeigt werden, dass das beschriebene mehrstufige Segmentierungsverfahren in der Lage ist, die Blutgefäße mit einer mittleren Genauigkeit von D = 0,806 zu segmentieren, was sogar leicht besser als die Inter-Observer Übereinstimmung (D = 0,791) ist. Darüber hinaus konnte nachgewiesen werden, dass der vorgestellte mehrstufige Segmentierungsansatz zu signifikant besseren Ergebnissen führt als andere etablierte automatische Segmentierungsverfahren. Insbesondere zeigte sich, dass die entwickelte Methode zu einem größeren Anteil an segmentierten kleinen Gefäßen führt.

Die referenzbasierte Kurvenanpassung zur Analyse der Hämodynamik wurde unter Verwendung von Monte-Carlo Simulationen und klinischen Datensätzen evaluiert. Im Rahmen der Monte-Carlo Evaluation konnte nachgewiesen werden, dass der Time-to-Peak Parameter im Vergleich zu anderen Kriterien am besten geeignet ist, um die Ankunftszeit des Kontrastmittelbolus zu bestimmen. Es konnte weiterhin gezeigt werden, dass bei Verwendung dieses Parameters die referenzbasierte Kurvenanpassung die Genauigkeit der Time-to-Peak Abschätzung in zeitlichen Kontrastmittelkurven mit niedrigen Signal-zu-Rausch Verhältnissen im Vergleich zu den anderen etablierten Modellen um bis zu 59% verbessern kann. Im Rahmen einer weitergehenden klinischen Analyse konnte darüber hinaus nachgewiesen werden, dass eine zeitliche Auflösung von 1.5 Sekunden ausreichend ist, um die zerebrale Hämodynamik zu analysieren [FIM$^+$12]. Hierdurch ist es möglich, die örtliche Auflösung der zeitaufgelösten MRA Datensätze im Vergleich zu den bisher verwendeten 4D MRA Datensätzen, die mit einer zetilichen Auflösung von 0.5 Sekunden aufgenommen wurden, deutlich zu steigern.

Die 4D Blutflussvisualisierung wurde auf Basis von 31 Datensätzen von Patienten mit einer arteriovenösen Malformation klinisch evaluiert [IFR$^+$13]. Hierbei konnte demonstriert werden, dass die 4D Blutflussvisualisierung zu einer guten Inter-Rater Übereinstimmung führt. Gleichzeitig wurde auch eine hohe Übereinstimmung der klinischen Bewertungen auf Basis von den 4D Visualisierungen und denen auf Basis von 2D digitalen Subtraktionsangiographien, welche den Goldstandard in der Klinik darstellen, festgestellt. Insgesamt zeigte sich, dass die 4D Blutflussvisualisierung eine zuverlässige, intuitive und schnelle klinische Begutachtung der Anatomie und Hämodynamik erlaubt.

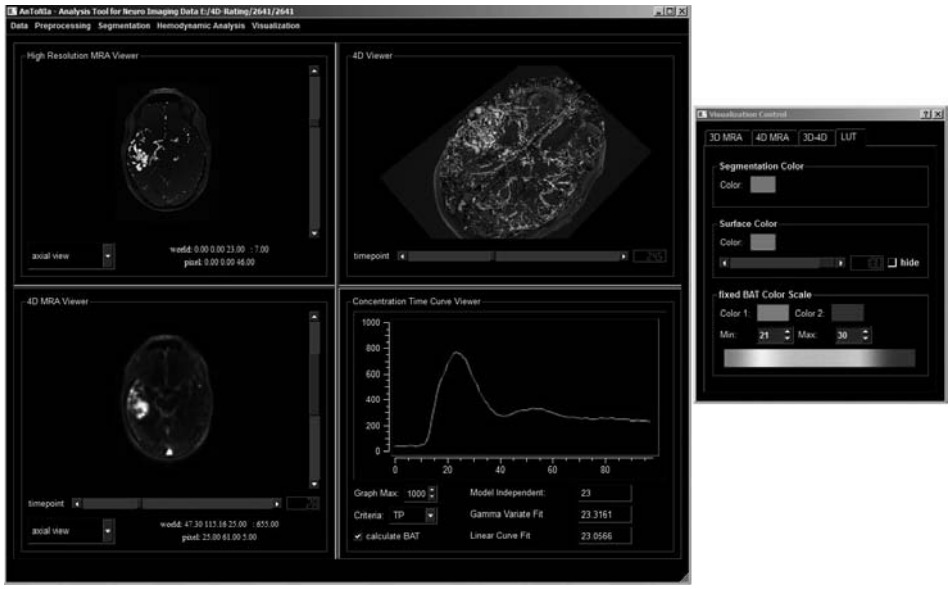


Abbildung 5: Abbildung der Benutzungsoberfläche von dem entwickelten Softwareprogramm AnToNIa

4 Diskussion

Die entwickelten Bildverarbeitungsmethoden wurden in das Softwareprogramm AnToNIa (Analysis Tool for Neuroimaging Data) implementiert, das ebenfalls im Rahmen dieser Arbeit entstandenen ist. Dieses Softwaretool wurde in einem stark interdisziplinären Forschungsumfeld in direkter Kooperation mit den lokalen Kliniken für Neuroradiologie und Neurologie entwickelt und verfügt über eine intuitiv verwendbare Benutzungsoberfläche (siehe Abb. 5). Neben der 4D-Blutflussvisualisierung wurde die Software bereits für eine Vielzahl von klinischen Studien eingesetzt. So wurde das Programm zum Beispiel für die Analyse der Auswirkung unterschiedlicher anatomischer Risikofaktoren für eine Ruptur einer arteriovenösen Malformation auf die zerebrale Hämodynamik [IFS+12] verwendet sowie für eine computergestützte Segmentierung und angiographische Charakterisierung von arteriovenösen Malformationen eingesetzt [FIG+13]. Im Bereich von Aneurysmen hat sich insbesondere die Gefäßsegmentierung als nützlich erwiesen und wurde zum Beispiel als Grundlage für eine verbesserte 3D-Vermessung von Aneurysmen [GFB+13] verwendet. Weiterhin wurde unter Verwendung der entwickelten Methoden ebenfalls der Einfluss hämodynamischer Modelle auf die Tissue-at-Risk Quantifizierung bei ischämischen Schlaganfällen [FKT+13] untersucht und ein neues Konzept zur Identifikation von Patienten mit ischämischem Schlaganfall, die sich innerhalb eines 4,5 Stunden-Fensters nach Symptomauftritt befinden, vorgestellt und in einer großen multi-zentrischen Studie evaluiert [CBF+13].

Zusammenfassend hat sich gezeigt, dass das mehrstufige Segmentierungsverfahren zur automatischen Extraktion von Gefäßstrukturen und die hämodynamische Analyse wertvoll für die klinische Anwendung sind und das Fundament für viele weitere Forschungsprojekte legen, die letztendlich zu einem tieferen Verständnis und einer verbesserten Behandlung von zerebralen Gefäßerkrankungen führen können.

Literatur

[BJvR+86] J.M. Bogaard, J.R. Jansen, E.A. von Reth et al. Random walk type models for indicator-dilution studies. *Cardiovasc Res*, 20(1):789–96, 1986.

[CBF+13] B. Cheng, M. Brinkmann, N.D. Forkert et al. Quantitative measurements of relative fluid-attenuated inversion recovery (FLAIR) signal intensities in acute stroke for the prediction of time from symptom onset. *J Cereb Blood Flow Metab*, 33(1):76–84, 2013.

[DFMD08] G.A. Donnan, M. Fisher, M. Macleod und S.M. Davis. Stroke. *Lancet*, 371:1612–1623, 2008.

[FFI+12] N.D. Forkert, J. Fiehler, T. Illies et al. 4D blood flow visualization fusing 3D and 4D MRA image sequences. *J Magn Reson Imaging*, 36(2):443–553, 2012.

[FFR+11] N.D. Forkert, J. Fiehler, T. Ries et al. Reference-based linear curve fitting for bolus arrival time estimation in 4D MRA and MR perfusion-weighted image sequences. *Magn Reson Med*, 65(1):289–94, 2011.

[FIG+13] N.D. Forkert, T. Illies, E. Goebell et al. Computer-aided nidus segmentation and angiographic characterization of arteriovenous malformations. *Int J Comput Assist Radiol Surg*, 8(5):775–86, 2013.

[FIM+12] N.D. Forkert, T. Illies, D. Möller et al. Analysis of the Influence of 4D MR Angiography Temporal Resolution on Time-to-Peak Estimation Error for Different Cerebral Vessel Structures. *Am J Neuroradiol*, 33(11):2103–9, 2012.

[FKT+13] N.D. Forkert, P. Kaesemann, A. Treszl et al. Comparison of Ten TTP and Tmax Estimation Techniques for MR Perfusion-diffusion Mismatch Quantification. *Am J Neuroradiol*, 34(9):1697–703, 2013.

[FLBCP09] V.L. Feigin, C.M. Lawes, D.A. Bennett S.L. Barker-Collo und V. Parag. Worldwide stroke incidence and early case fatality reported in 56 population-based studies: A systematic review. *Lancet Neurol*, 8(4):355–69, 2009.

[FSF+09] N.D. Forkert, D. Säring, J. Fiehler et al. Automatic brain segmentation in Time-of-Flight MRA images. *Methods Inf Med*, 48(5):399–407, 2009.

[FSRF+11] N.D. Forkert, A. Schmidt-Richberg, J. Fiehler et al. Fuzzy-based vascular structure enhancement in Time-of-Flight MRA images for improved segmentation. *Methods Inf Med*, 50(1):74–83, 2011.

[FSRF+12a] N.D. Forkert, A. Schmidt-Richberg, J. Fiehler et al. 3D cerebrovascular segmentation combining fuzzy vessel enhancement and level-sets with anisotropic energy weights. *Magn Reson Imaging*, 31(2):262–271, 2012.

[FSRF+12b] N.D. Forkert, A. Schmidt-Richberg, J. Fiehler et al. Automatic correction of gaps in cerebrovascular segmentations extracted from 3D Time-of-Flight MRA datasets. *Methods Inf Med*, 51(5):415–22, 2012.

[GFB+13] M. Groth, N.D. Forkert, J.-H. Buhk et al. Comparison of 3D computer-aided with manual cerebral aneurysm measurements in different imaging modalities. *Neuroradiology*, 55(2):171–178, 2013.

[IFR+13] T. Illies, N.D. Forkert, T. Ries et al. Classification of cerebral arteriovenous malformations and intranidal flow Patterns by color-encoded 4D-Hybrid-MRA. *Am J Neuroradiol*, 34(1):46–53, 2013.

[IFS+12] T. Illies, N.D. Forkert, D. Säring et al. Persistent hemodynamic changes in ruptured brain arteriovenous malformations. *Stroke*, 43(11):2910–5, 2012.

[LABFL09] D. Lesage, E.D. Angelini, I. Bloch und G. Funka-Lea. A review of 3D vessel lumen segmentation techniques. *Med Image Anal*, 13(6):819–45, 2009.

[SNS+98] Y. Sato, S. Nakajima, N. Shiraga et al. Three-dimensional multi-scale line filter for segmentation and visualization of curvelinear structures in medical images. *Med Image Anal*, 2(2):143–68, 1998.

[TDM+01] A.G. Thrift, H.M. Dewey, R.A. Macdonell et al. Incidence of the major stroke subtypes - Initial findings from the north east Melbourne stroke incidence study (NEMESIS). *Stroke*, 32(8):1732–38, 2001.

[TSWM64] H.K. Thompson, C.F. Starmer, R.E. Whalen und H. McIntosh. Indicator transit time considered as a gamma variate. *Circ Res*, 14:502–15, 1964.

[WVA+96] W.M. Wells, P. Viola, H. Atsumi et al. Multi-modal volume registration by maximization of mutual information. *Med Image Anal*, 1(1):35–51, 1996.

Nils Daniel Forkert schloss 2009 sein Diplomstudium der Informatik (Schwerpunkt Bildverarbeitung und Ergänzungsfach Medizin) an der Universität Hamburg ab. Im Anschluss daran arbeitete er als Doktorand am Institut für Medizinische Informatik und dem hieraus hervorgegangenen Institut für Computational Neuroscience am Universitätsklinikum Hamburg-Eppendorf. Gleichzeitig hat er das postgraduale Fernstudium Medizinische Physik (Schwerpunkt Strahlenphysik) an der Technischen Universität Kaiserslautern begonnen und 2012 mit dem akademischen Grad Master of Science abgeschlossen. Seine Dissertation hat er im selben Jahr (2012) am Department Informatik der Universität Hamburg eingereicht und das Promotionsverfahren mit der Disputation im April 2013 mit Auszeichnung abgeschlossen. Seit August 2013 forscht Nils Daniel Forkert als Postdoctoral Research Fellow am Radiological Sciences Laboratory an der Universität Stanford.

Takthierarchien für imperative synchrone Sprachen*

Mike Gemünde

Fachbereich Informatik
Technische Universität Kaiserslautern
gemuende@cs.uni-kl.de

Abstract: Synchrone Programmiersprachen eignen sich besonders zur Beschreibung von reaktiven Systemen, also von Systemen die in ständiger Interaktion mit ihrer Umgebung stehen. Dabei abstrahieren sie in einer Reaktion von der Berechnungszeit und nehmen an, dass Ausgaben unmittelbar nach den Eingaben verfügbar sind. Auch wenn diese Annahme Herausforderungen an die spätere Implementierung stellt, bietet diese Beschreibung auch einige Vorteile. Die Abstraktion ermöglicht beispielsweise eine wohldefinierte parallele Komposition, und durch die formal definierte Semantik sind diese Sprachen sehr gut handhabbar für formale Verifikationstechniken.

Die existierenden Sprachen haben dabei den Nachteil, dass die parallele Komposition nur strukturell und nicht zeitlich möglich ist. Jede Reaktion eines Systems oder Moduls synchronisiert sich jeweils mit den Reaktionen der anderen Module, auch wenn keine Interaktion stattfindet. Diese Arbeit betrachtet die Erweiterung imperativer synchroner Sprachen durch Takthierarchien, welche es erlauben, zusätzlich zu den Reaktionen weitere Abstraktionsebenen einzuführen. Eine Reaktion kann dabei in kleinere logische Schritte unterteilt werden, die wiederum Reaktionen eines anderen Moduls sind. Dies erlaubt die Komposition von Modulen auf zeitlicher Ebene und führt weitere Beschreibungsmöglichkeiten ein.

1 Einleitung

Man braucht heute nicht mehr viele Worte darüber zu verlieren, dass *eingebettete Systeme* eine immer größer werdende Rolle spielen und für immer mehr Aufgaben eingesetzt werden. Spezielle Anforderungen wie Funktionalität, Korrektheit, aber auch die Einhaltung der Entwicklungs- und Produktionskosten werden an sie gestellt. Auch der Begriff der *reaktiven Systeme*, welche in ständiger Interaktion mit ihrer Umgebung stehen, hat sich etabliert und die genannten Anforderungen treffen oft auch auf sie zu. Ständige Interaktion bedeutet dabei, dass die Reaktion des Systems *schnell genug* sein muss, um den Anforderungen der Anwendung zu genügen.

Durch die Verteilung der Systeme als auch durch die speziellen Anforderungen wurden neue Beschreibungsformalismen, sogenannte Berechnungsmodelle *(Models of Computation (MoC))* [Jan04], entwickelt um diesen Gegebenheiten Rechnung zu tragen. Dabei fokussieren sich Berechnungsmodelle auf spezielle Eigenschaften des Systems. Generell un-

*Englischer Titel der Dissertation: „Clock Refinement in Imperative Synchronous Languages“

terscheiden sie sich im Verständnis wie und wann Berechnungen angestoßen werden und wie Kommunikation, speziell zwischen parallelen Teilen, stattfindet. Entwickler können sich dabei auf diese Eigenschaften des System konzentrieren.

Eines dieser Berechnungsmodelle ist das *synchrone Berechnungsmodell* [Hal93, BCE+03], welches Systeme idealisiert betrachtet und annimmt, dass Ausgaben *zeitgleich* mit der Verfügbarkeit der Eingaben produziert werden. Diese Eigenschaft, oft auch als *perfekte Synchronität* bezeichnet, führt zur wohldefinierten, deterministischen Komposition von Systemen, da die Kommunikationszeiten ignoriert und sämtliche für die Berechnungen relevanten Teile (in der Annahme) zeitgleich ausgeführt werden. Diese Sichtweise wird im Übrigen auch bei der Entwicklung synchroner Schaltkreise angewendet und konzentriert sich hier auf die eigentliche Funktion, ignoriert aber die Gatterlaufzeiten. Zusätzlich muss sichergestellt werden, dass in jeder Reaktion (jedem Moment) des Systems, die Berechnung der Ausgaben schnell genug erfolgt, um der Anwendung Rechnung zu tragen.

Es existieren die unterschiedlichsten Sprachen, welche das synchrone Berechnungsmodell implementieren. Auf der einen Seite sind hier die datenflussorientierten Sprachen wie LUSTRE [HCRP91] und SIGNAL [GLB87] zu nennen, auf der anderen Seite steht bei den Sprachen ESTEREL [BC85] und QUARTZ [Sch09] der Kontrollfluss im Vordergrund. Darüber hinaus sollten die graphischen Varianten wie *Statecharts* [HP85] für den Kontrollfluss und *Scade* [BCE+03] für den Datenfluss nicht unerwähnt bleiben.

Der Inhalt der hier vorgestellten Dissertation [Gem13] beschreibt die Erweiterung der kontrollflussorientieren Sprache QUARTZ mit Takthierarchien. Diese Takthierarchien erlauben es, zusätzlich zu den bereits existierenden Modulhierarchien, zeitliche Abstraktion direkt in der Sprache zu beschreiben. Damit wird auf der einen Seite die Wiederverwendbarkeit gefördert, auf der anderen Seite aber auch ein neues Programmiermodell eingeführt, zu dem es bisher noch keine Sprache gab.

Diese Zusammenfassung ist wie folgt strukturiert: Kapitel 2 führt in die synchrone Sprache QUARTZ ein, bevor die eigentliche Spracherweiterung in Kapitel 3 motiviert und informell vorgestellt wird. In Kapitel 4 werden die Herausforderungen dieser Arbeit dargestellt und in Kapitel 5 anschließend die Nützlichkeit anhand eines Beispiels demonstriert. Kapitel 6 schließt die Darstellung ab.

2 Synchrone Sprache QUARTZ

Die imperative synchrone Sprache QUARTZ [Sch09] basiert auf dem synchronen Berechnungsmodell und stellt die Grundlage für die in der Dissertation entwickelte Erweiterung dar. Die Sprache selbst wird hier anhand des Beispielprogramms in Abbildung 1 (a) eingeführt. Das Beispiel zeigt ein Modul mit zwei Eingaben (`a`, `b`), zwei Ausgaben (`x`, `y`) und der lokalen Variablen `z`. Durch die `pause`-Anweisung wird ein Schritt bzw. eine Reaktion des Moduls beendet. Dabei werden in jedem Schritt neue Eingaben durch die Umgebung an das Modul übergeben und neue Ausgaben berechnet. Die Werte bleiben für den kompletten Schritt konstant; Variablen haben also genau einen Wert pro Schritt. Diese Annahme ist die grundlegende Eigenschaft synchroner Sprachen und erlaubt u.a. die

```
module M(nat ?a,?b,x,y)
{
  nat z;
  loop {
    x = a;
    next(z) = b;
    l1: pause;
    x = y + z;
    if(a > 4) {
      y = b;
      l2: pause;
    }
    l3: pause;
    z = 3;
  }
}
```

(a) Quelltext von Modul `M`

		1	2	3	4	5	6
Eingaben	a	2	5	1	1	2	2
	b	5	2	1	3	4	1
Markierungen	st	T	F	F	F	F	F
	l1	F	T	F	F	T	F
	l2	F	F	T	F	F	F
	l3	F	F	F	T	F	T
lokale Variable	z	0	5	5	3	3	3
Ausgaben	x	2	7	7	1	5	2
	y	0	2	2	2	2	2

(b) Beispielhafte Ausführung

Abbildung 1: Beispielmodul `M`

deterministische Komposition von Modulen, aber auch deterministisches Abbrechen oder Anhalten von Modulen.

Die Variablenwerte einer beispielhaften Ausführung des Moduls `M` sind in Abbildung 1 (b) dargestellt. Der `pause`-Anweisung wird, zumindest in der Literatur, oft eine Markierung vorangestellt, da sie eine wichtige Rolle bei der Übersetzung spielt. Die Markierung stellt dabei einen Zustand des Kontrollflusses dar. Im ersten Schritt werden alle Anweisungen bis zum Erreichen der ersten `pause`-Anweisung mit Markierung `l1` zeitgleich ausgeführt. Dabei wird der Variablen `x` der Wert 2 zugewiesen und die verzögerte Zuweisung `next(z) = b` weist der Variablen `z` den aktuellen Wert von `b` im nächsten Schritt zu. Der zweite Schritt der Ausführung beginnt bei der `pause`-Anweisung, an der der erste endete, und verläuft bis zum Erreichen der nächsten `pause`-Anweisung mit Markierung `l2` (da der `if`-Zweig betreten wird). In diesem zweiten Schritt wird `y` der Wert von `b` zugewiesen, aber auch der Wert von `y` zur Berechnung des Wertes von `x` verwendet. Nachdem jede Variable genau einen Wert in einem Schritt hat, kann das nur der gleiche Wert sein, der für beide Zuweisungen verwendet wird, auch wenn, der Wert *vor* der eigentlichen Zuweisung im Quelltext erscheint. Dies ist eine logische Konsequenz der synchronen Annahme und auch wenn sie in diesem Fall künstlich wirkt, kann sie bei der parallelen Ausführung von Modulen sehr nützlich sein.

Neben den im Beispiel gezeigten Möglichkeiten, bietet die Sprache QUARTZ weitere Anweisungen um den Programmablauf zu steuern. Eine Übersicht der grundlegenden Anweisungen ist in Abbildung 2 gegeben. Die parallele Ausführung erlaubt es, Programmteile oder ganze Module nebenläufig auszuführen. Dabei erfolgt die Ausführung immer schrittweise: jeweils ein Schritt jedes parallelen Fadens wird synchron ausgeführt. Durch die eindeutige Zuordnung von Variablenwerten zu Schritten ist die Ausführung determinis-

Beenden eines Schrittes	`l: pause`
direkte Zuweisung	`x =` τ
verzögerte Zuweisung	`next(x) =` τ
bedingte Ausführung	`if(`σ`) ... else ...`
lokale Deklaration	{α `x; ...`}
Schleife	`do ... while(`σ`)`
parallele Ausführung	`{ ... } || { ... }`
Abbruch der Ausführung	`[weak] [immediate] abort ... when(`σ`)`
Anhalten der Ausführung	`[l: weak] [immediate] suspend ... when(`σ`)`

Abbildung 2: QUARTZ-Anweisungen

tisch. Die Anweisung zum Abbruch kann die Ausführung jeweils zum Anfang oder zum Ende (Schlüsselwort `weak`) eines Schrittes beenden. Das weitere optionale Schlüsselwort `immediate` gibt an ob der Abbruch bereits im initialen Schritt geschehen kann.

Die Semantik der Sprache QUARTZ ist formal mittels struktureller operationaler Semantik (SOS) beschrieben. Die Korrektheit der existierenden Übersetzung ist bezüglich dieser Semantik verifiziert worden und diese übersetzt Programme in ein einfaches Zwischenformat welches keine komplexen Kontrollflussanweisungen mehr enthält. Ausgehend von diesem Zwischenformat werden Codegeneratoren eingesetzt, um Programme in Software (z.B. die Sprache C) oder in Hardware (z.B. die Sprache VERILOG) zu übersetzen. Die Übersetzung und die Codegeneratoren werden allerdings auch vor einige Herausforderungen gestellt, auf die später eingegangen wird.

3 Erweiterung der Sprache QUARTZ

Die Erweiterung der Sprache QUARTZ wird anhand eines einfachen Beispiels in Abbildung 3 illustriert. Zu sehen ist eine Implementierung des euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers (ggT) zweier Zahlen. Das Modul auf der linken Seite ist in QUARTZ ohne Verwendung der Erweiterung implementiert. Im ersten Berechnungsschritt werden die gegebenen Eingaben den beiden lokalen Variablen zugewiesen, mit deren Hilfe in den folgenden Schritten der ggT berechnet und schließlich als Ausgabe wieder zur Verfügung gestellt wird. Dabei sind die entsprechenden Eigenschaften der Sprache zu beachten, nämlich dass pro Schritt neue Eingaben vorhanden und auch neue Ausgaben nach außen sichtbar sind. Dies bedeutet auch, dass die einzelnen Berechnungsschritte nach außen hin sichtbar sind. Die beispielhafte Berechnung in Abbildung 4 (a) illustriert dies: Das Ergebnis ist erst im sechsten Schritt für die Umgebung sichtbar. Diese Tatsache macht die Wiederverwendung des gezeigten Moduls schwierig, da sich die Umgebung bzw. das aufrufende Modul an das zeitliche Verhalten anpassen muss. Eingangs wurde allerdings auch die zeitliche Abstraktion gerade bei synchronen Sprachen hervorgehoben. Diese Eigenschaft ist hier also nicht konsequent umgesetzt.

```
module GCD1(nat ?a,?b,!gcd)
{

    nat x, y;
    x = a;
    y = b;
    while(x > 0) {
      if(x >= y)
        next(x) = x-y;
      else
        next(y) = y-x;
      l: pause;
    }
    gcd = y;

}
```

(a) Original

```
module GCD2(nat ?a,?b,!gcd)
{
  clock(C1) {
    nat x, y;
    x = a;
    y = b;
    while(x > 0) {
      if(x >= y)
        next(x) = x-y;
      else
        next(y) = y-x;
      l: pause(C1);
    }
    gcd = y;
  }
}
```

(b) Erweiterung

Abbildung 3: ggT-Implementierungen

Das Beispiel in Abbildung 3 (b) zeigt den gleichen Algorithmus, nur verwendet dieses Modul die QUARTZ-Erweiterung. Das neue Schlüsselwort `clock` erlaubt es, lokal einen neuen Takt zu definieren, auf dem die Schritte zur Berechnung des ggT ausgeführt werden. Diese Unterschritte sind also nicht nach außen hin sichtbar, sondern nur der gesamte Berechnungsschritt, der erst am Ende des Moduls endet. Mit Hilfe des lokalen Taktes können also Schritte in Unterschritte unterteilt werden. Das Berechnungsbeispiel in Abbildung 4 (b) illustriert das Verhalten. Während die Eingaben für den gesamten (äußeren) Berechnungsschritt ihre Gültigkeit behalten, verändern sich die lokalen Variablen mit jedem Schritt des lokalen Taktes. Die lokalen Variablen sind auf einem schnelleren Takt definiert als die Variablen an der Schnittstelle des Moduls. Das Ergebnis der Berechnung hat auch für den gesamten Schritt nur einen gültigen Wert und die Umgebung nimmt die gesamte Berechnung als einen einzigen Schritt war.

Das sehr einfach gehaltene Beispiel zeigt die grundlegende Idee der Erweiterung. Jedoch können die Taktdeklarationen generell beliebig geschachtelt werden. Dabei bilden sie eine Baumstruktur, da es jeweils nur einen nächst höheren Takt geben kann. Ein Designziel der Erweiterung war es, das synchrone Modell auf jeder dieser Ebenen zu erhalten. Dies bedeutet, dass Variablen, die auf einer Taktebene definiert sind, auch für jeweils diesen Takt konstant bleiben, auch wenn sich Variablen, die auf niedrigeren Takten definiert sind, während eines solchen Schrittes öfter ändern können. Dies wurde auch durch das ggT Beispiel gezeigt. Wenn ein Schritt eines Taktes beendet wird, d.h. der Kontrollfluss des Programms eine `pause`-Anweisung des Taktes erreicht, werden auch alle darunterliegenden Taktschritte beendet. Dies ist auch eine logische Konsequenz, um die Abstraktion auf jeder Schicht zu erhalten.

	1	2	3	4	5	6
a	7	7	7	7	7	7
b	3	3	3	3	3	3
st	T	F	F	F	F	F
l	F	T	T	T	T	T
x	7	4	1	1	1	0
y	3	3	3	2	1	1
gcd	0	0	0	0	0	1

(a) Original

	1	2	3	4	5	6
a	7					
b	3					
st	T	F	F	F	F	F
l	F	T	T	T	T	T
x	7	4	1	1	1	0
y	3	3	3	2	1	1
gcd						1

(b) Erweiterung

Abbildung 4: Beispielhafte Ausführungen der ggT-Implementierungen

Neben der Erweiterung des Verhaltens aller vorhandenen QUARTZ-Anweisungen wie beispielsweise der `abort`-Anweisung, ist speziell die parallele Ausführung hervorzuheben. Es ist nun möglich, neue *unabhängige* Takte in den parallelen Fäden zu definieren, die dann unabhängig voneinander ausgeführt werden können:

```
clock(c1) {...} || clock(c2) {...}
```

Eine *Synchronisation* findet nur dann statt, wenn beide Seiten eine `pause`-Anweisung erreichen, die einem gemeinsamen Takt angehört. Die lokal definierten Unterschritte sind dagegen für die jeweils andere Seite nicht sichtbar. Speziell dieses Konstrukt erlaubt es, die Ausführungszeit, d.h. die Anzahl der eigentlich benötigten Schritte, eines Modules zu verstecken. Es erlaubt somit die zeitliche Komposition und fördert die Wiederverwendbarkeit, erhält dabei aber auch das synchrone Berechnungsmodell mit all seinen Vorteilen. Ähnliche Konstrukte sind bereits in der synchronen Datenflusssprache SIGNAL möglich, allerdings betrachtet diese Arbeit dieses Konzept auf Basis einer kontrollflussorientierten synchronen Sprache.

4 Herausforderungen

Im Kapitel 2 wurden bereits die Herausforderungen der Sprache z.B. für die Übersetzung erwähnt, auf welche hier im Kontext der Erweiterung eingegangen werden soll. Speziell diese Themen mussten erneut für die Erweiterung gelöst werden und beinhalten die ursprünglichen Lösungen als Spezialfall.

Die Tatsache, dass jede Variable genau einen Wert in einem Schritt besitzt, führt dazu, dass die Abhängigkeiten, welche Variablen für die Berechnung einer anderen benötigt werden, von Bedeutung sind. Diese Abhängigkeiten werden in einer kontrollflussbasierten Sprache wie QUARTZ durch die Bedingungen des Programmablaufs ergänzt. So kann beispielsweise die Ausführung einer Anweisung durch eine `if`-Bedingung eingeschränkt sein und die Variablen, die zur Auswertung der Bedingung benötigt werden, in einem

parallelen Faden berechnet werden. Solche Abhängigkeiten entstehen oft durch die parallele Komposition und sind bei normalen Programmen in der Regel einfach zu lösen, müssen aber generell von Compiler und Codegerneratoren beachtet werden. Dieser Aspekt wird als *Kausalität* bezeichnet: die Abhängigkeiten müssen sich, entweder statisch zum Übersetzungszeitpunkt oder dynamisch zur Laufzeit, auflösen lassen.

Ein zweiter wichtiger Aspekt imperativer synchroner Sprachen sind lokale Deklarationen, also Variablen, die einen eingeschränkten Sichtbarkeitsbereich haben und bei Betreten dessen jeweils neu initialisiert werden. Durch Schleifen kann es allerdings passieren, dass dieser Sichtbarkeitsbereich in einem Schritt verlassen und in demselben auch wieder betreten wird. Ein Teil der Anweisungen bezieht sich also auf die *alte* Variable, während ein anderer Teil mit dem Wert der *neuen* Variablen ausgeführt wird. Für die Analyse der Abhängigkeiten müssen also mehrere *Inkarnationen* derselben Variablen betrachtet werden. Die Anzahl ist hier aber durch die Schachtelungstiefe der Schleifen beschränkt. Dieses Problem löst der Compiler bei der Übersetzung in das Zwischenformat und muss für die Codegeneratoren nicht erneut betrachtet werden. Betroffene Anweisungen und Variablen werden als *schizophren* bezeichnet.

Für die Erweiterung war es im ersten Schritt nötig, die formale Semantik, die für die Sprache QUARTZ definiert ist, auf die neuen Taktdefinitionen zu erweitern. Da hierbei das komplette Ausführungsmodell verändert wird, reichte es dazu nicht aus, SOS-Regeln für die neuen Anweisungen zu ergänzen. Vielmehr muss jede Anweisung die Untertakte beachten und aus diesem Grund wurden die SOS-Regeln für die Erweiterung weitestgehend neu geschrieben. Viele Regeln, sowie die generelle Struktur der Regeln sind allerdings ähnlich geblieben. So hat es sich bewährt, die SOS-Regeln in zwei unterschiedliche Mengen einzuteilen, die Reaktionsregeln und die Transitionsregeln. Erstere lösen durch sukzessives Anwenden die Abhängigkeiten zwischen den Variablen auf und bestimmen deren Werte für den aktuellen Schritt. Dabei führen sie mit nur teilweise vorhandenem Wissen über die Variablenwerte das Programm aus und versuchen neue Informationen über andere Variablen zu erhalten. Als Beispiel sei hier wieder die `if`-Anweisung zu nennen, bei der, solange die Variablen zur Auswertung der Bedingung noch unbekannt sind, keine Aussage darüber getroffen werden kann, ob der eine oder andere Zweig ausgeführt werden soll. Auf der anderen Seite führen die Transitionsregeln anschließend einen Schritt des Programms mit Hilfe der vorher bestimmten Variablenwerten aus. Die Anwendung der Regeln wurde, wie auch in der bisherigen Sprache, in einem Simulationsalgorithmus zusammengefasst, um die Ausführung des Programms zu beschreiben. Neu hinzugekommen ist neben der Entscheidung, wann Variablen ihren Wert behalten und wann sie einen neuen bekommen können, die Frage welcher Unterschritt als nächstes ausgeführt werden soll. Durch die parallelen unabhängigen Untertakte ist diese Entscheidung nicht eindeutig. Allerdings ist eine Erkenntnis an dieser Stelle, dass die Reihenfolge der Ausführung keinen Einfluss auf das übergeordnete Verhalten hat: Das Gesamtverhalten bleibt deterministisch.

Der zweite Schritt für die Erweiterung war die Übersetzung der neuen Sprachkonstrukte in das Zwischenformat. Wie schon bei der Semantik konnte sich diese Änderung nicht nur auf die neuen Anweisungen beziehen, sondern erstreckt sich durch den kompletten Algorithmus und führte auch zur Erweiterung des Zwischenformats, welches nun mit verschiedenen Takten und deren Abhängigkeiten umgehen muss.

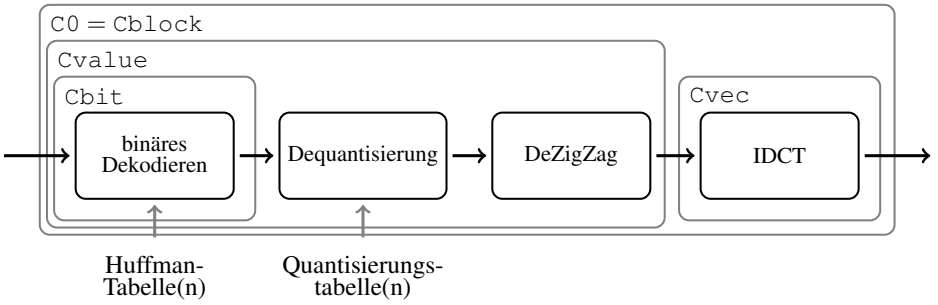


Abbildung 5: Taktgrenzen des JPEG-Dekoders

Der letzte Schritt in der Kette bildet die Übersetzung aus dem Zwischenformat in eine Zielsprache. In der Arbeit wurde dazu die Übersetzung in eine Hardware-Beschreibungssprache betrachtet. Die Übersetzung wird in zwei unterschiedlichen Teilen beschrieben. Der eine bildet den funktionalen Kern ab, also das Verhalten, das durch das Programm vorgegeben ist. Wie aber schon im Absatz über die Semantik erwähnt wurde, kann es Freiheiten in der Ausführung der Untertakte geben. Dieser Teil wird in einen Scheduler ausgelagert und kann auf mehrere Weisen implementiert werden, so lange er sich an bestimmte, durch das Programm gegebene Abhängigkeiten, hält. Damit ist, wie bei der Semantik, der Determinismus des Gesamtsystems sichergestellt.

5 Evaluierung

In der Dissertation wurde die Übersetzung aus dem Zwischenformat in eine Hardwarebeschreibungssprache erläutert. Diese wurde anschließend benutzt, um die Spracherweiterung anhand von Beispielen zu evaluieren. Eines dieser Beispiele ist ein JPEG-Dekoder, der ohne und mit der neuen Spracherweiterung implementiert wurde, um die Ergebnisse zu vergleichen.

Das Datenflussmodell des Dekoders ist in Abbildung 5 illustriert. Entscheidend ist bei der Darstellung, dass der Datenaustausch zwischen den Modulen mit unterschiedlicher Frequenz geschieht und auch von den Eingaben selbst abhängig ist. Beispielsweise ist nicht festgelegt, wie viele Bit mittels Huffman-Tabelle dekodiert werden und auch nicht wie viele einzelne Werte daraus resultieren. Die einzelnen Werte werden anschließend zu einer Tabelle der Größe 8×8 zusammengesetzt und mittels IDCT weiterverarbeitet. Die umschließenden Kästen illustrieren die Takte, die in den entsprechenden Modulen sichtbar sind. So wird beispielsweise in jedem Schritt des Taktes `Cblock` ein Datenwert zwischen dem DeZigZag-Modul und dem IDCT-Modul ausgetauscht. Lokal wird der Takt `Cvec` verwendet, um die interne Berechnung der IDCT nach außen hin unsichtbar zu machen. Somit zeigt das Beispiel zwei Dinge: Zum einen werden Takte zur Synchronisation und zum Datenaustausch verwendet, zum anderen wird von der lokalen Berechnung durch die

Takte abstrahiert. Dieses Verhalten wäre in der ursprünglichen Sprache ohne explizites Programmieren dieser Eigenschaften nicht möglich gewesen.

6 Zusammenfassung

Die hier vorgestellte Dissertation erweitert die synchrone Sprache QUARTZ um Takthierarchieen, möchte allerdings nicht beliebige Komplexität zulassen, sondern die Merkmale der synchronen Sprache selbst auf jeder dieser Ebenen erhalten. Das vorgestellte Konzept zieht sich durch den kompletten Entwicklungsablauf der Sprache QUARTZ und behandelt die formale Definition der Semantik, das Übersetzen in ein Zwischenformat und die Codegenerierung. Abschließend wurden Beispiele gegeben, die zeigen, dass sich die neuen Konzepte durchaus praktisch nutzen lassen und eine sinnvolle Erweiterung der Sprache QUARTZ darstellen.

Literatur

[BC85] G. Berry und L. Cosserat. The Esterel Synchronous Programming Language and its Mathematical Semantics. In S.D. Brookes, A.W. Roscoe und G. Winskel, Hrsg., *Seminar on Concurrency (CONCUR)*, vol. 197 of *LNCS*, Seiten 389–448, Pittsburgh, Pennsylvania, USA, 1985. Springer.

[BCE+03] A. Benveniste, P. Caspi, S. Edwards, N. Halbwachs, P. Le Guernic und R. de Simone. The Synchronous Languages Twelve Years Later. *Proceedings of the IEEE*, 91(1):64–83, 2003.

[Gem13] M. Gemünde. *Clock Refinement in Imperative Synchronous Programs*. Dissertation, Department of Computer Science, University of Kaiserslautern, Germany, Kaiserslautern, Germany, October 2013. PhD.

[GLB87] T. Gautier, P. Le Guernic und L. Besnard. SIGNAL, a declarative language for synchronous programming of real-time systems. In G. Kahn, Hrsg., *Functional Programming Languages and Computer Architecture*, vol. 274 of *LNCS*, Seiten 257–277, Portland, Oregon, USA, 1987. Springer.

[Hal93] N. Halbwachs. *Synchronous programming of reactive systems*. Kluwer, 1993.

[HCRP91] N. Halbwachs, P. Caspi, P. Raymond und D. Pilaud. The Synchronous Dataflow Programming Language LUSTRE. *Proceedings of the IEEE*, 79(9):1305–1320, September 1991.

[HP85] D. Harel und A. Pnueli. On the development of reactive systems. In K.R. Apt, Hrsg., *Logic and Models of Concurrent Systems*, Seiten 477–498. Springer, 1985.

[Jan04] A. Jantsch. *Modeling Embedded Systems and SoCs*. Morgan Kaufmann, 2004.

[Sch09] K. Schneider. The Synchronous Programming Language Quartz. Internal Report 375, Department of Computer Science, University of Kaiserslautern, Kaiserslautern, Germany, December 2009.

Mike Gemünde wurde am 26. April 1983 in Bad Kreuznach geboren. Nach Abitur und Wehrdienst studierte er von 2003 bis 2008 Informatik an der Technischen Universität Kaiserslautern. Nach dem Abschluss des Studiums als Diplom-Informatiker arbeitete er als wissenschaftlicher Mitarbeiter in der Arbeitsgruppe Eingebettete Systeme des Fachbereichs Informatik der Technischen Universität Kaiserslautern. Die Stelle wurde in dem Projekt „Modellbasierter Entwurf von Hardware-Softwaresystemen mit synchronen Sprachen“ durch die deutsche Forschungsmeinschaft (DFG) gefördert und mit der Promotion 2013 abgeschlossen. Seit September 2013 arbeitet er in der Entwicklung elektronischer Bremssysteme bei der Continental AG in Frankfurt a.M.

Visuelle Evaluierung, Skalierung und Transport von sicheren Videos*

Heinz Hofbauer

Fachbereich Computerwissenschaften
Universität Salzburg
hhofbaue@cosy.sbg.ac.at

Abstract: Diese Kurzfassung der Dissertation gibt eine Stand-der-Technik-Analyse eines effizienten Video-on-Demand Systems. Unter effizient verstehen wir in diesem Zusammenhang einerseits die Optimierung des Speicherverbrauchs am Server als auch die Minimierung von Verzögerungen beim Ausliefern an Verbraucher, die durch Skalierung und Verschlüsselung des Videomaterials zur sicheren Übertragung entstehen. Zusätzlich stellen wir theoretische und praktische Überlegungen an, wie man ein effizientes und sicheres System mittels Wavelet-basierter Videocodecs verwirklichen kann.

1 Einleitung

Durch die weite Verbreitung von Breitbandinternet im Heim und im mobilen Bereich werden *Video-on-Demand* (VoD) und Streamingsysteme immer beliebter. Konsumenten von VoD wollen diese Systeme überall nutzen, Stichwort "ubiquitous computing", von Smartphones mittels 3G-Verbindung bis zu Heimkinosystemen über Breitbandinternet. Dies verlangt von VoD-Anbietern dass Videos in diverser Form gespeichert werden, also in verschiedener Auflösung und Qualität, angepasst an das jeweilige Endgerät und die mögliche Verbindungsgeschwindigkeit. In weiterer Folge bedeutet das einen gesteigerten Speicherverbrauch und damit höhere Kosten für Anbieter.
Für dieses Problem gibt es eine Lösung, den *Universal Multimedia Access* (UMA), [VCE03]. Gemeint ist damit, dass ein Video auf eine Art gespeichert wird, die eine Adaption an die eventuellen Anforderungen eines Benutzers aus dem gleichen Quellmaterial erlaubt. Damit wird der benötigte Speicherbedarf gesenkt und die Möglichkeit gewährleistet, auf neue Anforderungen einzugehen die im Ursprungssystem nicht vorgesehen waren.
Der Nachteil dieser Systeme ist die benötigte Rechenleistung zur Adaption der Videos. Allerdings führt dies nicht zu einem Flaschenhals beim Anbieter, da moderne Netzwerktechnologien erlauben, diese Adaption *Just-in-Time* (JIT), also erst an der letztmöglichen Stelle im Netzwerk, zu erledigen. Durch diese *multimedia aware network elements* (MANE), [KKRH08], ist theoretisch sogar eine Einsparung der Bandbreite beim Anbieter möglich, da verschiedene Endnutzer mit einem einzigen Strom seitens des Anbieters bedient werden

*Englischer Titel der Dissertation "Visual Evaluation, Scaling and Transport of Secure Videos" [Hof13]

können, indem die Adaption JIT im Netzwerk ausgeführt wird.

Waveletbasierte Video Codecs sind durch ihre inhärente Skalierbarkeit gut für diese Anwendung geeignet und bieten die gleiche Qualität wie herkömmliche Video-Codecs. Der *Motion Compensated Embedded Zero Bit Codec* (MC-EZBC), [CW04, WGW04], ist ein moderner Video-Codec der auf Wavelets basiert und damit den Anforderungen von UMA grundsätzlich genügt.
Die Lösung für das UMA-Problem einfach einen Waveletbasierten Codec zu verwenden klingt offensichtlich, wird aber vorerst dadurch verhindert, dass MANEs und die dazugehörigen Netzwerkprotokolle nur standardisierte Formate, nämlich H.264, verstehen. Zusätzlich wollen Anbieter von VoD Systemen auf eine sichere Art mit ihren Endnutzern kommunizieren, um Piraterie zu vermeiden, was eine JIT-Skalierung im Netzwerk ausschließt, da die MANEs vollen Zugriff auf das übertragene Video zur Adaption benötigen.

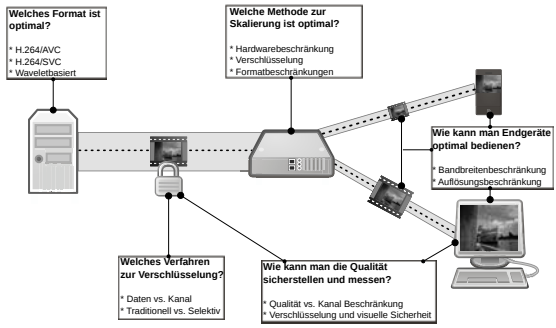


Abbildung 1: Übersicht über ein VoD-Setup mit Darstellung der Fragestellungen (und Probleme), die in einem solchen System auftreten.

In der kumulativen Dissertation geht es also darum, folgende Probleme (siehe auch Abb. 1) zu lösen: Es muss gewährleistet werden, dass MC-EZBC Videoströme über die existierenden Transporttechnologien übertragen werden können; Es müssen Methoden zur sicheren Ende-zu-Ende-Verbindung entwickelt werden, die eine JIT Skalierung erlauben; und es müssen Methoden zur Evaluierung der Sicherheit des Visuellen Inhalts von Videos entwickelt und evaluiert werden.
Da eine detaillierte Beschreibung der Methodologien den Rahmen des Artikels bei weitem übersteigen würde, beschränken wir uns darauf, zu jedem der obig gestellten Fragen einen Überblick über die aktuellen Verfahren zu geben und unseren Beitrag zu umreißen. Damit sich der geneigte Leser zu jedem Gebiet vertiefen kann werden wir ausführlich auf die Literatur verweisen.

2 Wavelet-basierte Videocodecs und existierende Transporttechnologien

2.1 Videokodierung und UMA

Herkömmliche Videokodierung zielt auf eine bestimmte Auflösung und Bitrate ab. Die Kodierung erfolgt hauptsächlich durch Ausnutzung von Redundanzen im Bildbereich und Ähnlichkeiten von zeitlich nahe beisammen liegenden Bildern. Wenn ein VoD-Anbieter nun mehr als eine Zielplattform bedienen will, z.B., Heimkino mit 1080p Auflösung, re-

guläre Fernseher mit PAL-Auflösung und Handys mit 720p, dann muss für jede Zielplattform ein Video, das explizit dafür kodiert ist, gespeichert werden. Serverseitig führt das zu einem nicht unerheblichen Mehraufwand an Speicherplatz. Eine Lösung für dieses Problem wäre das Prinzip des UMA, welches nur ein skalierbares Ursprungsvideo speichert und aus diesem für jede Zielanwendung ein angepasstes Video extrahieren kann.

Der derzeitige Standard für Videokodierung, H.264/AVC [ISO05, ITU07], ist nicht skalierbar und daher für UMA nicht geeignet. Es gibt jedoch eine Erweiterung von H.264/AVC die skalierbar ist, nämlich H.264/SVC [ITU12, SMW07]. Es gibt jedoch einen Nachteil bei der Nutzung von H.264/SVC als Ursprungsvideo. Es erlaubt nämlich nur Skalierung auf Ziele, die beim Erstellen des Ursprungsvideos explizit eingestellt wurden. Damit ist es insofern nicht zukunftssicher, als dass neue Zielanwendungen entweder nicht optimal bedient werden können, oder eine neu Kodierung aller Ursprungsvideos nach sich zieht. Zusätzlich entstehen durch weitere mögliche Zielanwendungen zusätzliche Speicherkosten. In Folge können auch Leitungskapazitäten nicht optimal genutzt werden, da ein Teil der Leitung für unnötige Skalierung verschwendet wird.

Eine Alternative zu H.264/SVC sind Waveletbasierte Videokodierungsverfahren. Diese nutzen ebenso wie H.264 die räumlichen und temporalen Redundanzen in einem Video aus. Durch die Kodierung mittels Wavelets sind sie allerdings automatisch skalierbar. Zwar gibt es bei der räumlichen und zeitlichen Skalierung eine Einschränkung auf dyadische Faktoren, dafür ist die Qualitätsanpassung stufenlos möglich. In unseren Arbeiten haben wir den MC-EZBC von Woods et al. [WGW04] verwendet. Durch die Art der Kodierung ist also sichergestellt, dass Ursprungsvideos zukunftssicher gespeichert werden können. Zudem kann mit der stufenlosen Qualitätsanpassung sichergestellt werden, dass die verfügbare Kapazität eines Übertragungskanals optimal genutzt werden kann.
Es bleibt noch die Frage zu klären, ob der MC-EZBC grundsätzlich eine vergleichbare Qualität wie H.264 liefert. Lima et al. [LMA+07] haben Waveletbasierte Codecs mit H.264/AVC verglichen und kamen zum Schluss, dass zwischen beiden Codecs keine großen Qualitätsunterschiede merklich sind. Da wir die Waveletbasierten Codecs anstelle von H.264/SVC verwenden wollen, haben wir dies gesondert getestet [HSU09]. Wir haben festgestellt, dass es bei einer geringen Anzahl von Skalierungszielen für H.264/SVC keinen merklichen Qualitätsunterschied festzustellen gibt. Erhöht man allerdings die Anzahl von Skalierungszielen, verringert sich bei gleichbleibendem Speicherverbrauch die Qualität von H.264/SVC gegenüber dem MC-EZBC. Wir können also festhalten, dass für UMA die Waveletbasierten Codecs von Vorteil sind, da sie bei gleichem Speicher Verbrauch optimale Leitungsnutzung erlauben und zukunftssicher sind.

2.2 Transport von Waveletbasierten Videos

Die Nutzung von Wavelets als Codec für VoD-Systeme hat allerdings auch Nachteile, nämlich die Tatsache, dass die Hardware und Software für den Transport der Videos durch das Netzwerk und für die Skalierung auf dem Standard (H.264) aufbaut. Es gibt diesbezüglich auch einiges an Literatur, sowohl für den Transport [WHP+07, Apo06] als auch für die Skalierung [KKH10, TPP08, KKRH08].

Um das System optimal zu gestalten, geht es allerdings nicht nur um den Transport der Videos, sondern auch um die Skalierung im Netzwerk. Dadurch kann man die Kanalkapazität besser nutzen, indem man die Skalierung ins Netzwerk verschiebt und nur dann skaliert wenn es tatsächlich nötig ist. Beispielsweise kann die Kanalkapazität bei drahtlosen Anwendungen schwanken, z.B. durch die Anzahl der Benutzer in einer Funkzelle. In diesem Fall ist es möglich, ein Video bis zum Übergang auf den drahtlosen Kanal mit der maximal möglichen Qualität zu schicken. Die Anpassung an die aktuelle Kapazität kann dann auf dem Zugriffspunkt erfolgen, um dem Anwender immer die optimale Qualität zu liefern. Damit die Skalierung im Netzwerk passieren kann müssen die MANEs eine Information geliefert bekommen, die beschreibt, wie die gelieferten Daten aufgebaut sind.

Eine Art, die gelieferten Daten zu beschreiben ist, einen parallelen Datenstrom zu verwenden, der die Videodaten beschreibt. Die standardisierte Weise dies zu tun basiert auf MPEG-21 "Digital Item Adaptation" [ISO07] und wird passenderweise als "generic Bitstream Syntax Description" (gBSD) bezeichnet [PHH+03]. In [HU09a] haben wir ein Beschreibungsmodell entwickelt, mit dem sich ein auf Waveletbasiertes Video mittels gBSD gut beschreiben lässt. Zudem haben wir untersucht, wie gut dieses in der Praxis verwendbar ist. Das Problem mit gBSD ist, dass die Größe der Beschreibung linear mit der Anzahl der Skalierungspunkte steigt. In einem Anwendungsfall, in dem eine feingranulare Skalierung nötig ist, etwa das Beispiel zur drahtlosen Übertragung von oben, kann gBSD deshalb nicht effizient verwendet werden.

Eine andere Option ist die standardisierte Art zum Transport von Multimediadaten zu verwenden, nämlich das "Real-time Transport Protocol" (RTP) [SCFJ03] und das "Real Time Streaming Protocol" (RTSP) [SRL98], welches wiederum auf RTP basiert. Für H.264/AVC gibt es durch [WHS+05] eine Beschreibung wie RTP und RTSP zu nutzen sind. Kuschnig et al. [KKRH08] haben diese Beschreibung auf H.264/SVC erweitert. Grundsätzlich funktioniert dass, indem ein H.264/SVC Strom in "network abstraction layer units" (NALUs) aufgebrochen wird und diese als Basis für die Skalierung auf den MANEs verwendet werden. Wenn wir also eine Möglichkeit finden können, einen Waveletbasierten Strom in solche NALUs zu zerlegen, die bezüglich Skalierung die gleichen Eigenschaften aufweisen wie vergleichbare H.264/SVC-NALUs, so können wir die bestehende Hard- und Software für diese Art des Transports nutzen. In [HHK+11] haben wir eine solche Zerlegung auf Basis des MC-EZBC erzeugt. Allerdings ist sie so allgemein, dass sie mit kleineren Anpassungen für alle Waveletbasierten Ströme nutzbar ist. Zudem haben wir Untersuchungen zur Effizienz angestellt, die zeigen, dass der Aufwand für diese Art des Transports und der Skalierung geringer ist als die Nutzung von gBSD.

3 Sichere Methoden für Ende-zu-Ende Verbindungen mit Skalierung

3.1 Datenverschlüsselung

Die traditionelle Art der Verschlüsselung basierend auf der Arbeit von Shannon [Sha49] würde eine sichere Ende-zu-Ende Verbindung erlauben. Da für die Skalierung zumindest

die Beschreibungsdaten im Klartext erhalten werden müssen, kann damit im Netzwerk keine Skalierung vorgenommen werden. Eine Variante die in diesem Fall üblicherweise Anwendung findet, lautet selektive Verschlüsselung. Dabei wird nicht der gesamte Datenstrom verschlüsselt sondern nur ausgewählte Teile. Üblicherweise werden Beschreibungsdaten im Klartext belassen und nur die Bilddaten verschlüsselt. Allerdings haben sich in diesem Zusammenhang und im Kontext von "digital rights management" (DRM) noch andere Arten der selektiven Verschlüsselung etabliert. Man unterscheidet grob nach dem Anwendungsfall von DRM folgende Methoden:

Strenge Sicherheit wird oft auch als "message privacy" Sicherheit (MP security) bezeichnet. Formell heißt MP-Sicherheit, dass aus dem verschlüsselten Strom keine Rückschlüsse auf den Klartext gezogen werden können [BRRS09]. Im Falle von MC-EZBC-Daten konnten wir zeigen [HU10a], dass bereits die Beschreibungsdaten ausreichen, um Rückschlüsse zu ziehen. Es kann hier also nur mit traditioneller Verschlüsselung eine strenge Sicherheit gewährleistet werden.

Inhaltssicherheit ist eine Aufweichung der strengen Sicherheit. Es wird erlaubt, dass Beschreibungsdaten rekonstruiert werden dürfen. Allerdings dürfen bei der Inhaltssicherheit die Inhaltsdaten, also bei Video etwa das Bildmaterial, nicht in einer Art und Weise rekonstruierbar sein die eine Erkennung der Inhalte erlaubt [SU12].

Hinreichende Sicherheit bezeichnet eine Art der Verschlüsselung, die es durchaus erlaubt, Inhaltsdaten teilweise zu rekonstruieren. Allerdings wird gefordert, dass ein Missbrauch der Daten verhindert wird. Dies geschieht üblicherweise dadurch, dass die Inhaltsdaten nur auf eine Art rekonstruiert werden können welche die Qualität der Daten extrem verringert [SU10]. Im Falle von Videos reicht es üblicherweise, die Qualität soweit zu mindern dass ein Konsum der Daten schlecht bis gar nicht möglich ist.

Transparente Sicherheit ist eine Art von hinreichender Verschlüsselung bei der allerdings eine Mindestqualität gefordert wird. Anders als bei hinreichender Sicherheit, wo es nur darum geht, die Qualität hinreichen zu verringern, will man bei transparenter Sicherheit die Qualität nur in einem bestimmen Maß vermindern. Das Ziel der transparenten Sicherheit ist es üblicherweise eine Vorschau mit niederer Qualität zu geben, die mit dem korrekten Schlüssel auf eine hohe Qualität zurückgeführt werden kann [LC07].

Bezüglich der Sicherheit dieser selektiven Verschlüsselungsverfahren, speziell im Zusammenhang mit Shannons Arbeit, hat Lookabaugh et al. [LS04] gezeigt, dass die Sicherheit gewährleistet ist.

In [HU09b] haben wir für MC-EZBC-basierte Videos ein Verschlüsselungsverfahren entwickelt, welches die Bilddaten verschlüsselt und damit für hinreichende und transparente Sicherheit geeignet ist. In [HU10a] haben wir untersucht, ob es sich auch für Inhaltssicherheit eignet und feststellen müssen, dass temporale Information die Sicherheit in dieser Hinsicht kompromittiert. Wir haben eine Erweiterung für das Verschlüsselungsverfahren aus [HU09b] entwickelt, welches Inhaltssicherheit erlaubt. Zudem haben wir zeigen können dass keine Form der selektiven Verschlüsselung für MC-EZBC-basierte Videos eine strenge Sicherheit gewährleisten kann.

3.2 Kanalverschlüsselung

Eine andere Form der sicheren Ende-zu-Ende Verbindung kann erreicht werden, indem man unverschlüsselte Daten über einen sicheren Kanal versendet. Dafür gibt es einen auf RTP basierenden Standard namens "Secure Real-time Transport Protocol", der in [BMN+04] definiert ist. Dabei wird ein sicherer Kanal aufgebaut und die Daten werden darüber versendet. Das zieht nach sich, dass an jeder Stelle im Netzwerk, die auf die Daten zugreift, der Schlüssel bekannt sein muss. Das heißt nicht nur, dass das Schlüsselmanagement zu einem Problem wird, sondern auch, dass jede Stelle im Netzwerk, die auf den Datenstrom zugreift, ein potentielles Angriffsziel ist.
Im Gegensatz zur selektiven Verschlüsselung der Daten muss außerdem der Datenstrom komplett entschlüsselt und wieder verschlüsselt werden, wenn auf die Daten zugegriffen wird. Dadurch entstehen einerseits zeitliche Verzögerungen bei der Auslieferung der Daten und andererseits steigen die Hardwareanforderungen an die MANEs, die die Last der Ent- und Verschlüsselung tragen. In [HHK+11] haben wir Messungen angestellt, die sowohl Latenz als auch Hardwareanforderungen von Kanal- und Datenverschlüsselung vergleichen. Die Messungen haben klar ergeben, dass auf Basis der Effizienz die Kanalverschlüsselung der Datenverschlüsselung unterlegen ist.

4 Visuelle Evaluierung und Sicherheit

Bei der visuellen Evaluierung von Bildinhalten geht es darum, die Qualität zu bestimmen, in der ein Konsument das Bildmaterial wahr nimmt. Das ist einerseits bei der Kompression wichtig, spielt aber auch bei der selektiven Verschlüsselung, bei Inhalts-, hinreichender und transparenter Verschlüsselung eine Rolle. Optimalerweise wird die Qualität direkt durch Befragung von Menschen ermittelt. Es gibt dafür Standards, wie das zu geschehen hat. Die wichtigsten sind die Empfehlungen der ITU, speziell die "Methodology for the subjective assessment of the quality of television pictures" [ITU02] und "Audiovisual quality in multimedia services" [ITU96]. Dabei werden Laborbedingen spezifiziert, etwas Bildschirmauflösung, der Abstand zum Bild, die Dauer der Anzeige und Beleuchtung. Zusätzlich wird die Anzahl der Testsubjekte auf mindestens fünfzehn festgelegt. Der dadurch entstehende Zeit- und Kostenaufwand ist schon für kleine Testreihen recht hoch.
Üblicherweise wird anstelle der Testreihen mit menschlichen Betrachtern dazu übergegangen, Bildmetriken zu verwenden. Diese bilden das menschliche Sehvermögen ab und werden auch als objektive Metriken bezeichnet. Entwickelt werden diese Bildmetriken mit der Hilfe von Datenbanken, die eine Vielzahl von Bildern mit unterschiedlichen Veränderungen und Verzerrungen enthalten, zusammen mit Messwerten durch menschliche Betrachter laut ITU-Empfehlung. Die Veränderungen der Bilder unterscheiden sich von Datenbank zu Datenbank, enthalten aber üblicherweise typische Arten der Kompression, z.B. JPEG oder JPEG2000, und häufig auftretende Veränderungen wie Rauschen, Unschärfe oder Kontraständerungen. Typische Vertreter solcher Datenbanken sind die "LIVE Image Quality Assessment Database" [SWCB] und "Tampere Image Database" [PBE+08, PLZ+09].

Typischerweise werden Bildmetriken verwendet, die sowohl eine schnell Evaluierung er-

lauben als auch annähernd dem menschlichen Sehen entsprechen. Die am weitesten verwendet Bildmetrik ist immer noch der Signal-Rausch-Abstand [Lia09], da diese Metrik einfach zu implementierten und schnell anzuwenden ist. Allerdings ist von dieser Metrik auch bekannt, dass sie das menschliche Sehen nur schlecht abbildet [HTG08, HU10b]. Daher werden in letzter Zeit immer öfter Metriken verwendet, die besser dem menschlichen Sehen entsprechen, aber immer noch schnell anwendbar sind. Typische Vertreter dieser Kategorie sind SSIM [WBSS04] und NICE [DR09]. Obwohl es bessere Bildmetriken gibt, etwa VIF [SB06] oder CPA1 [CPA10], werden diese vergleichsweise selten verwendet, weil sie kompliziert und sehr langsam sind. Um die Evaluierung weiter zu optimieren, haben wir in [HU11] eine Bildmetrik entwickelt die besser dem menschlichen Sehen entspricht als SSIM und NICE. Die Leistung ist am ehesten mit der CPA1 vergleichbar. Dabei ist die Metrik allerdings schneller als SSIM und NICE.
Diese Bildmetriken sind generell gut geeignet, wenn es um hohe Qualität geht, die hauptsächlich bei Kompression oder Bildbearbeitung interessant ist. Allerdings eignen sie sich weniger, wenn die Qualität merklich nachlässt, wie es etwa bei hinreichender oder transparenter Verschlüsselung der Fall sein kann. Wir haben in [HU10b] gezeigt, dass die normale Leistung einer Metrik nicht für niederqualitatives Bildmaterial gilt. In [HU13] haben wir eine Methodologie für die Evaluierung von Sicherheitsmetriken vorgestellt und herkömmliche Metriken evaluiert. Dabei hat sich herausgestellt, dass nur die VIF annähernd die Leistung bringt, die man von einer Sicherheitsmetrik erwarten würde.

5 Zusammenfassung

Wir haben die Fragestellung behandelt in welchen Gebieten Probleme bei Video-on-Demand Systemen auftreten die auf Wavelet-basierte Videokompression aufbauen. Zu den gefundenen Themengebieten haben wir ein Übersicht über derzeitige Standards gegeben und Lösungen für die gefundenen Probleme präsentiert.

Literatur

[Apo06] J. Apostolopoulos. Architectural Principles for Secure Streaming & Secure Adaptation in the Developing Scalable Video Coding (SVC) Standard. In *Proceedings of the IEEE International Conference on Image Processing, ICIP '06*, Seiten 729–732, Oktober 2006.

[BMN+04] M. Baugher, D. McGrew, M. Naslund, E. Carrara und K. Norrman. The Secure Real-time Transport Protocol (SRTP). RFC 3711 (Proposed Standard), Marz 2004.

[BRRS09] M. Bellare, T. Ristenpart, P. Rogaway und T. Stegers. Format-preserving encryption. In *Proceedings of Selected Areas in Cryptography, SAC '09*, Jgg. 5867, Seiten 295–312, August 2009.

[CPA10] Maurizio Carosi, Vinod Pankajakshan und Florent Autrusseau. Towards a simplified perceptual quality metric for watermarking applications. In *Proceedings of SPIE, Multimedia on Mobile Devices*, Jgg. 7542, Januar 2010.

[CW04] Peisong Chen und John W. Woods. Bidirectional MC-EZBC With Lifting Implementation. *IEEE Transactions on Circ. and Systems for Video Technology*, 14(10):1183–1194, 2004.

[DR09] S. S. Hemami D. Rouse. Natural Image Utility Assessment Using Image Contours. In *IEEE International Conference on Image Processing (ICIP'09)*, Seiten 2217–2220, November 2009.

[HHK+11] Hermann Hellwagner, Heinz Hofbauer, Robert Kuschnig, Thomas Stütz und Andreas Uhl. Secure Transport and Adaptation of MC-EZBC Video Utilizing H.264-based Transport Protocols. *Elsevier Journal on Signal Processing: Image Communication*, 27(2):192–207, 2011.

[Hof13] Heinz Hofbauer. *Visual Evaluation, Scaling and Transport of Secure Videos*. Dissertation, University of Salzburg, Department of Computer Sciences, April 2013.

[HSU09] H. Hofbauer, T. Stütz und A. Uhl. Secure Scalable Video Compression for GVid. In *Proceedings of the 3rd Austrian Grid Symposium*, Jgg. 269 of *books@ocg.at*, Seiten 88–102, 2009.

[HTG08] Q. Huynh-Thu und M. Ghanbari. Scope of validity of PSNR in image/video quality assessment. *Electronics Letters*, 44(13):800–801, Juni 2008.

[HU09a] Heinz Hofbauer und Andreas Uhl. The Cost of In-Network Adaption of the MC-EZBC for Universal Multimedia Access. In *Proceedings of the 6th International Symposium on Image and Signal Processing and Analysis (ISPA '09)*, September 2009.

[HU09b] Heinz Hofbauer und Andreas Uhl. Selective Encryption of the MC EZBC Bitstream for DRM Scenarios. In *Proceedings of the 11th ACM Workshop on Multimedia and Security*, Seiten 161–170, September 2009.

[HU10a] Heinz Hofbauer und Andreas Uhl. Selective Encryption of the MC-EZBC Bitstream and Residual Information. In *18th European Signal Processing Conference, 2010 (EUSIPCO-2010)*, Seiten 2101–2105, August 2010.

[HU10b] Heinz Hofbauer und Andreas Uhl. Visual Quality Indices and Low Quality Images. In *IEEE 2nd European Workshop on Visual Information Processing*, Seiten 171–176, Juli 2010.

[HU11] Heinz Hofbauer und Andreas Uhl. An Effective and Efficient Visual Quality Index based on Local Edge Gradients. In *IEEE 3rd European Workshop on Visual Information Processing*, Seite 6pp., Juli 2011.

[HU13] Heinz Hofbauer und Andreas Uhl. An Evaluation of Visual Security Metrics. *IEEE Transactions on Multimedia*, Seite 15 pages, 2013. submitted.

[ISO05] ISO/IEC 14496-10. Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding, 2005.

[ISO07] ISO/IEC 21000-7:2007. Information technology – Multimedia framework (MPEG-21) – Part 7: Digital Item Adaptation, November 2007.

[ITU96] ITU-T REC P.910. SERIES P: TELEPHONE TRANSMISSION QUALITY Audiovisual quality in multimedia services, 1996.

[ITU02] ITU-R BT.500-11. Methodology for the subjective assesment of the quality of television pictures, 2002.

[ITU07] ITU-T H.264. Advanced video coding for generic audiovisual services, November 2007.

[ITU12] ITU-T REC H.264. SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS Infrastructure of audiovisual services Coding of moving video, Januar 2012.

[KKH10] R. Kuschnig, I. Kofler und H. Hellwagner. Improving Internet Video Streaming Performance by Parallel TCP-based Request-Response Streams. In *Proceedings of the 7th Annual IEEE Consumer Communications and Networking Conference (IEEE CCNC 2010)*, Januar 2010.

[KKRH08] R. Kuschnig, I. Kofler, M. Ransburg und H. Hellwagner. Design options and comparison of in-network H.264/SVC adaptation. *Journal of Visual Communication and Image Representation*, 19(8):529–542, September 2008.

[LC07] Q. Li und I. J. Cox. Using perceptual models to improve fidelity and provide resistance to valumetric scaling for quantization index modulation watermarking. *IEEE Transactions on Information Forensics and Security*, 2(2):127–139, Juni 2007.

[Lia09] Shiguo Lian. Efficient image or video encryption based on spatiotemporal chaos system. *Chaos, Solitons & Fractals*, 40(5):2509 – 2519, 2009.

[LMA+07] L. Lima, F. Manerba, N. Adami, A. Signoroni und R. Leonardi. Wavelet-Based Encoding for HD Applications. In *2007 IEEE International Conference on Multimedia and Expo*, Seiten 1351–1354, Juli 2007.

[LS04] T. D. Lookabaugh und D. C. Sicker. Selective Encryption for consumer applications. *IEEE Communications Magazine*, 42(5):124–129, 2004.

[PBE+08] N. Ponomarenko, F. Battisti, K. Egizarian, J. Astola und V. Lukin. Color Image Database for Evaluation of Image Quality Metrics. In *Proceedings of International Workshop on Multimedia Signal Processing*, Seiten 403–408, Oktober 2008.

[PHH+03] Gabriel Panis, Andreas Hutter, Jörg Heuer, Herman Hellwagner, Harald Kosch, Christian Timmerer, Sylvain Devillers und Myriam Amielh. Bitstream syntax description: a tool for multimedia resource adaptation within MPEG-21. In *Special Issue on Multimedia Adaptation*, Jgg. 18 of *Signal Processing: Image Communication*, Seiten 721–747, Sept. 2003.

[PLZ+09] N. Ponomarenko, V. Lukin, A. Zelensky, K. Egiazarian, M. Carli und F. Battisti. TID2008 - A Database for Evaluation of Full-Reference Visual Quality Assessment Metrics. In *Advances of Modern Radioelectronics*, Jgg. 10, Seiten 30–45, 2009.

[SB06] H. R. Sheikh und A. C. Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, Mai 2006.

[SCFJ03] H. Schulzrinne, S. Casner, R. Frederick und V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 3550, Juli 2003.

[Sha49] Claude E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28:656–715, Oktober 1949.

[SMW07] H. Schwarz, D. Marpe und T. Wiegand. Overview of the Scalable Video Coding Extension of the H.264/AVC Standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1103–1120, 2007.

[SRL98] H. Schulzrinne, A. Rao und R. Lanphier. Real Time Streaming Protocol (RTSP). RFC 2326, April 1998.

[SU10] Thomas Stütz und Andreas Uhl. Efficient Format-Compliant Encryption of Regular Languages: Block-Based Cycle-Walking. In *Proceedings of the 11th Joint IFIP TC6 and TC11 Conference on Communications and Multimedia Security, CMS '10*, Jgg. 6109, Seiten 81–92, Mai 2010.

[SU12] Thomas Stütz und Andreas Uhl. A Survey of H.264 AVC/SVC Encryption. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(3):325–339, 2012.

[SWCB] H. R. Sheikh, Z. Wang, L. Cormack und A. C. Bovik. LIVE Image Quality Assessment Database Release 2.

[TPP08] Nicolas Tizon und Beatrice Pesquet-Popescu. Scalable and Media Aware Adaptive Video Streaming over Wireless Networks. *EURASIP Journal on Advances in Signal Processing*, Volume 2008(Article ID 218046):11 pages, 2008.

[VCE03] A. Vetro, C. Christopoulos und T. Ebrahimi. From the guest editors - Universal multimedia access. *IEEE Signal Processing Magazine*, 20(2):16 – 16, 2003.

[WBSS04] Z. Wang, A.C. Bovik, H.R. Sheikh und E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004.

[WGW04] Y. Wu, A. Golwelkar und J. W. Woods. MC-EZBC video proposal from Rensselaer Polytechnic Institute. *ISO/IEC JTC1/SC29/WG11, MPEG2004/M10569/S15*, Marz 2004.

[WHP+07] Y. Wang, M. Hannuksela, S. Pateux, A. Eleftheriadis und S. Wenger. System and Transport Interface of SVC. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1149–1163, September 2007.

[WHS+05] S. Wenger, M.M. Hannuksela, T. Stockhammer, M. Westerlund und D. Singer. RTP Payload Format for H.264 Video. RFC 3984, Februar 2005.

Heinz Hofbauer, geboren am 15.12.1977, erlangte seine Hochschulreife an der Höheren Technische Bundeslehranstalt für Elektrotechnik in Salzburg. Anschließend absolvierte er sein Studium an der Universität Salzburg im Bereich Angewandte Informatik. Er schrieb seine Diplomarbeit über die Nutzung von Quasi-Monte-Carlo-Methoden für die parallele Integration von uneigentlichen Integralen und erlangte seinen Abschluss als Diplom-Ingenieur. Danach widmete er sich dem Studium der Bild- und Videoverschlüsselung und der Bestimmung der Qualität von Bild- und Videokompression. Zudem kam das Studium von Videokompression und Übertragung von skalierbarem Videomaterial. Das Studium und die Forschung kulminierte 2013 in der Dissertation mit dem Titel "Visual Evaluation, Scaling and Transport of Secure Videos" und erlangte den Abschluss als Doctor technicæ. Seine Forschungsschwerpunkte liegen derzeit in der visuellen Evaluierung selektiver Verschlüsselungsverfahren, Watermarking und Verschlüsselung von Bild und Video sowie der Biometrie.

Maschinelles Lernen expressiver Verknüpfungsregeln zur Duplikaterkennung mittels genetischer Programmierung*

Robert Isele
mail@robertisele.com

Abstract: Die vorliegende Arbeit führt neuartige Algorithmen für ein zentrales Problem der Datenintegration und Datenbereinigung ein, das Finden von Paaren von Entitäten in Datensets, welche das gleiche Realwelt-Objekt beschreiben. Sie deckt dabei den gesamten Entity-Matching-Prozess ab, von der automatischen Erzeugung von effektiven Verknüpfungsregeln mittels genetischen Algorithmen, bis zu deren effizienten Ausführung. Die eingeführten Algorithmen tragen dazu bei den Konfigurationsaufwand des Verknüpfungsprozesses zu reduzieren, die Genauigkeit der generierten Links zu erhöhen, sowie schließlich die Ausführung auf Einzelmaschinen oder Rechenclustern zu beschleuningen.

1 Einführung

Ein zentrales Problem der Datenintegration und Datenbereinigung ist das Finden von Paaren von Entitäten in Datensets, welche das gleiche Realwelt-Objekt beschreiben. Viele bestehende Methoden für das Identifizieren solcher Paare basieren auf domänenspezifischen Verknüpfungsregeln, die festlegen wie zwei Entitäten auf Äquivalenz verglichen werden. Allerdings ist das Schreiben solcher Verknüpfungsregeln von Hand in der Praxis aufwendig und erfordert eine detaillierte Kenntnis der verwendeten Datensets. Ein weiteres wichtiges Problem stellt das effiziente Ausführen der generierten Verknüpfungsregeln dar. Abbildung 1 zeigt die wichtigsten Schritte eines regelbasierten Prozesses zur Duplikaterkennung. Im Folgenden gehen wir für jeden Schritt näher ins Detail.

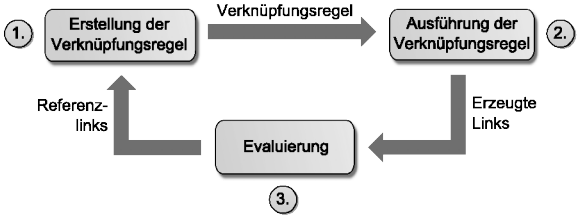


Abbildung 1: Duplikaterkennung mit Verknüpfungsregeln.

*Englischer Titel der Dissertation: "Learning Expressive Linkage Rules for Entity Matching using Genetic Programming"

1.1 Erstellung der Verküpfungsregel

Bevor zwei Datenquellen verlinkt werden können, muss eine Verknüpfungsregel definiert werden, welche die Voraussetzungen angibt, die gelten müssen damit zwei Entitäten verlinkt werden. Verknüpfungsregeln können entweder von einem Domänenexperten oder einem überwachten Lernalgorithmus erstellt werden. Das Erstellen einer effektiven Verknüpfungsregel von Hand ist ein nicht-triviales Problem und verlangt vom Autor detaillierte Kenntnisse über die Struktur der zu verlinkenden Datensätze. Wir veranschaulichen dies am einfachen Beispiel zweier Datensätze über Filme. Zunächst ist ein Vergleich der Filme alleinig basierend auf den Filmtiteln in der Regel nicht ausreichend, da Filme mit dem gleichen Titel tatsächlich unterschiedliche Filme bezeichnen können, die in verschiedenen Jahren veröffentlicht wurden. Daher muss die Verknüpfungsregel zumindest die Filmtitel sowie ihre Veröffentlichungstermine kombinieren und dabei eine geeignete Aggregationsfunktion wählen, welche die Ähnlichkeiten beider Eigenschaften vereint. In der Regel muss mit Unregelmässigkeiten in beiden Datensätzen gerechnet werden muss. So können die Veröffentlichungsdaten des gleichen Films in verschiedenen Datenquellen um mehrere Tage abweichen. Deshalb muss der Autor der Verknüpfungsregel auch geeigneten Distanzmaße zusammen mit adäquaten Schwellenwerten auswählen. Die Abdeckung aller in den Datenquellen vorhandener Heterogenitäten durch die Verknüpfungsregel ist oft schwierig und kann nur durch mehrere Iterationen erreicht werden.

1.2 Ausführung der Verknüpfungsregel

Die Ausführung einer Verknüpfungsregel hat das Ziel alle Paare von Entitäten zu identifizieren, für welche die gegebene Regel erfüllt ist. Das Ergebnis ist eine Menge von Links, worin jeder Link zwei Entitäten verbindet, welche laut Verknüpfungsregel ähnlich sind.

Abbildung 2 zeigt einen typischen Ausführungsprozess für zwei Datenquellen. Um zu

Abbildung 2: Ausführung einer Verknüpfungsregel.

Vermeiden, dass die Verknüpfungsregel für jedes mögliche Paar von Entitäten ausgewertet werden muss, kann zunächst eine Indizierung durchgeführt werden. Das Ziel der Indizierung ist es Paare von Entitäten herauszufiltern welche potentiell ähnlich sind um dabei andererseits Paare die definitiv verschieden sind früh verwerfen zu können. Basierend auf dem Index wird die Verknüpfungsregel nur für die potentiell ähnlichen Paare von Entitäten ausgewertet.

Verschiedene Indizierungsverfahren wurden entwickelt um die Effizienz des Prozesses zu

erhöhen [Chr12]. Allerdings können viele dieser Verfahren dazu führen, dass korrekte Links durch das Indizierungsverfahren fälschlicherweise verworfen werden und damit weniger Links generiert werden als durch die Verknüpfungsregel vorgegeben.

Im vorigen Beispiel zweier Filmdatenbanken, könnte eine einfache Indizierungsmethode so aussehen, dass jedem Film basierend auf seinem Erscheinungsjahr ein Index zugewiesen wird. In diesem Fall müssten anschließen lediglich Filme mit dem gleichen Erscheinungsjahr auf Ähnlichkeit mit der Verknüpfungsregel getestet werden. Der Nachteil dieser Methode wäre allerdings der Verlust von Links zwischen Filmen, für welche das falsche Erscheinungsjahr angegeben ist. Die Entwicklung von Indizierungsverfahren, welche die Anzahl der nötigen Vergleiche reduzieren und zugleich den Verlust korrekter Links vermeiden, ist ein wichtiges Forschungsgebiet.

1.3 Evaluierung

Der Zweck des Evaluierungsschritts ist es die Qualität der Verknüpfungsregel zu bestimmen und potentielle Fehler in den generierten Links zu finden. In der Regel erfolgt die Überprüfung basierend auf manuell erstellten Referenzlinks. Eine Menge von Referenzlinks besteht hierbei aus positiven Referenzlinks, welche Paare von Entitäten verknüpfen welche nachgeprüft das gleiche Real-Welt Objekt beschreiben, sowie negativen Referenzlinks, welche Paare von Entitäten verknüpfen, welche verschiedene Objekte beschreiben.

Nach der Ausführung der Verknüpfungsregel, können mit Hilfe der Referenzlinks zwei Arten von fehlerhaften Links identifiziert werden: Falsch positive Links, d.h. für einen negativen Referenzlink wurde basierend auf der aktuellen Verknüpfungsregel ein Link erstellt und falsch negative Links, d.h. positive Referenzlinks für die kein Link generiert wurde. Die gefundenen Fehler können genutzt werden um die Verknüpfungsregel zu verbessern. Auf diese Weise kann die Verknüpfungsregel iterativ verbessert werden.

1.4 Vorliegende Arbeit

Die vorliegende Arbeit führt neuartige Algorithmen ein, die den gesamten Prozess der Duplikaterkennung abdecken, von der automatischen Erzeugung von effektiven Verknüpfungsregeln, bis zu deren effizienten Ausführung. Die folgenden Abschnitte fassen die Hauptbeiträge zum aktuellen Stand der Forschung zusammen:

Abschnitt 2 stellt eine expressive Repräsentation für Verknüpfungsregeln vor, welche Regeln als Operatorbaum darstellt und expressiver als durch bestehende Lernverfahren unterstützte Repräsentationen ist. In *Abschnitt 3* wird zunächst ein genetischer Algorithmus vorgestellt, der Verknüpfungsregeln aus einem Goldstandard lernt, welcher aus einer Menge von Paaren von Entitäten besteht, worin jedes Paar als Duplikat oder Nicht-Duplikat gekennzeichnet ist.

Um auch Anwendungsfälle abzudecken, in denen kein Goldstandard verfügbar ist, wird

anschließend in *Abschnitt 4* ein komplementärer Algorithmus eingeführt, welcher Verknüpfungsregeln interaktiv lernt indem er dem Nutzer eine kleine Anzahl von Beispielpaaren präsentiert, welcher dieser als Duplikat oder Nicht-Duplikat kennzeichnet. Abschließend führt *Abschnitt 5* ein neuartiges Indizierungsverfahren ein, welches die Anzahl der notwendigen Vergleiche signifikant reduziert und die Ausführung der gelernten Verknüpfungsregeln auf verteilten Architekturen ermöglicht.

2 Expressive Verknüpfungsregeln

Die Aufgabe einer *Verknüpfungsregel* ist es, einem gegebenen Paar von Entitäten einen Ähnlichkeitswert zuzuweisen. Um den Ähnlichkeitswert zu bestimmen, verbindet eine Verknüpfungsregel einzelne Vergleiche. Im Laufe der Zeit wurden verschiedene Modelle vorgeschlagen, die das Ziel haben die einzelnen Distanzen zu einem Ähnlichkeitswert zu kombinieren. Die meisten Modelle basieren auf zwei Typen von Klassifikatoren:

- Ein *linearer Klassifikator* kombiniert mehrere Distanzen, indem die gewichtete Summe der einzelnen Distanzen berechnet wird.Die Größe der Gewichte reguliert hierbei den Einfluss eines bestimmten Vergleichs auf den Distanzwert.
- Ein *schwellwertbasierter Klassifikator* kombiniert verschiedene Ähnlichkeitstests mit logischen Operatoren.Ein Ähnlichkeitstest besteht dabei aus einem Distanzmaß und einem Schwellwert.

In der vorliegenden Arbeit wird ein Modell für die Repräsentation von Verknüpfungsregeln vorgeschlagen welches expressiver ist als bisher verwendete Modelle und Datentransformationen enthalten kann, welche Werte vor dem Vergleich normalisieren. Das vorgeschlagene Modell repräsentiert eine Verknüpfungsregel als einen Baum der aus vier Typen von Operatoren aufgebaut ist:

Ein **Eingabeoperator** gibt alle Werte eines bestimmten Attributs, beispielsweise die Adresse einer Person, zurück. Ein **Transformationsoperator** transformiert Werte basierend auf eine vorgegebenen Transformationsfunktion.Mehrere Transformationsoperatoren können verschachtelt werden um Ketten von Transformationen abzubilden. Ein **Vergleichsoperator** vergleicht zwei Werte mit Hilfe eines vorgegebenen Distanzmaßes und einem Distanzschwellwert und gibt einen Ähnlichkeitswert zurück. Ein **Aggregationsoperator** vereint mehrere Ähnlichkeitswerte zu einem Ähnlichkeitswert mit Hilfe einer Aggregationsfunktion, z.B. dem gewichteten Mittelwert.

Abbildung 3 zeigt eine Verknüpfungsregel um geographische Orte zu verlinken.

3 GenLink — Überwachtes Lernen von Verknüpfungsregeln

Im vorhergehenden Abschnitt haben wir bereits gezeigt wie Verknüpfungsregeln verwendet werden können um die genauen Bedingungen anzugeben die erfüllt sein müssen damit

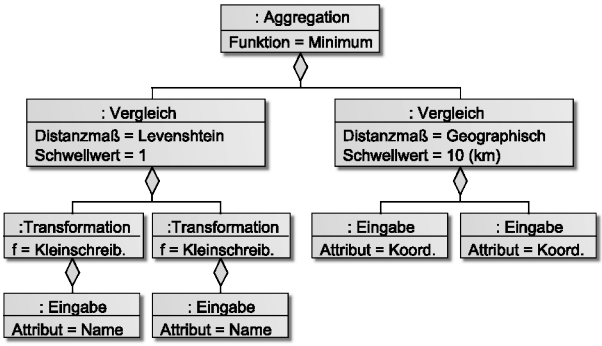


Abbildung 3: Beispiel einer Verknüpfungsregel.

zwei Entitäten verlinkt werden. Allerdings ist das Schreiben und die Feinoptimierung solcher Verknüpfungsregeln von Hand oft schwierig und zeitaufwendig.

Eine Möglichkeit diesen Aufwand zu reduzieren, stellen Lernverfahren dar, welche Verknüpfungsregeln aus bestehenden Referenzlinks generieren. Das Erstellen von Referenzlinks ist viel einfacher als das Schreiben von Verknüpfungsregeln, da es keine Vorkenntnisse über verschiedene Distanzmaße oder das durch das vorliegende System verwendete Modell für die Repräsentation von Verknüpfungsregeln erfordert. Referenzlinks können von Domain-Experten erstellt werden, indem sie die Gleichwertigkeit von einer Menge von Paaren von Entitäten aus den Datensätzen bestätigen oder ablehnen.

3.1 Der Algorithmus

Der GenLink-Algorithmus basiert auf genetischer Programmierung. Genetische Programmierung ist eine Erweiterung des genetischen Algorithmus, welche ursprünglich von Cramer vorgeschlagen wurde. Genetische Programmierung eignet sich für Probleme deren Lösungen als Bäume von Operatoren dargestellt werden können, so wie es bei der eingeführten Repräsentation für Verknüpfungsregeln der Fall ist.

Abbildung 4 veranschaulicht den allgemeinen Ablauf des GenLink-Algorithmus. Zu Be-

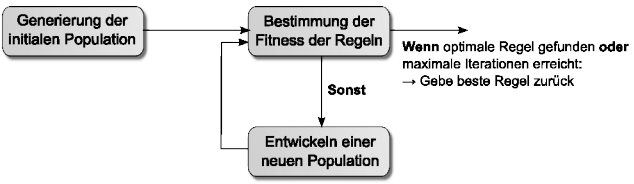


Abbildung 4: GenLink Iterationen.

ginn generiert GenLink eine Population von initialen Verknüpfungsregeln, welche nach einem Zufallsalgorithmus erstellt werden. In jeder Iteration überprüft GenLink die Fitness aller Verknüpfungsregeln in der aktuellen Population indem für jede Regel überprüft wird wie viele der aktuellen Referenzlinks korrekt bestimmt werden. Anschließen erzeugt GenLink neue Population indem basierend auf einer Selektionsstrategie, welche Regeln mit höherer Fitness bevorzugt, Regeln aus der Population ausgewählt werden und zu einer neuen Regel kombiniert werden. Dazu werden zwei Hauptoperationen eingesetzt: 1. Mutation modifiziert eine einzelne Regel zufällig. 2. Crossover rekombiniert zwei Regeln zu einer neuen Regel. Der Algorithmus ist beendet sobald entweder eine optimale Verknüpfungsregel gefunden wurde oder eine vorgegebene maximale Anzahl von Iterationen erreicht wurde.

GenLink erweitert die traditionelle genetische Programmierung in drei wichtigen Punkten:

1. Während bei genetischen Algorithmen die initiale Population in der Regel vollständig zufällig generiert wird, verwendet GenLink ein Verfahren, welches eine initiale Population erstellt, die nur Regeln enthält die Eigenschaften mit ähnlichen Werten vergleichen. Somit werden weniger Iterationen gebraucht um Regeln zu lernen.

2. Die meisten Algorithmen der genetischen Programmierung im Allgemeinen und das einzige GenLink vorrausgehende überwachte Lernverfahren von Carvalho et al. [dCLGdS12] verwenden einen generischen Crossover-Operator der Subtree-Crossover genannt wird. GenLink dagegen setzt eine Menge von spezialisierten Crossover-Operatoren ein, worin jeder Operator nur einen Aspekt der Verknüpfungsregeln kombiniert. Beispielweise ist ein Crossover-Operator enthalten welcher Transformationen beider Regeln kombiniert und damit Transformationsketten aufbauen kann.

3. Ein häufiges Problem von Algorithmen der genetischen Programmierung ist das Aufblähen der gelernten Lösungen, d.h. das Entwickeln von Lösungsbäumen, die zunehmend grösser werden, ohne dass deren Qualität im gleichem Maße zunimmt. GenLink verwendet ein neuartiges Verfahren um ein Aufblähen der Lösungen zu verhindern, ohne dass die Qualität der gelernten Verknüpfungsregeln leidet.

3.2 Ergebnisse

3.2.1 Evaluierung der Erweiterungen

Der vorgestellte GenLink-Algorithmus enthält Erweiterungen des verbreiteten Algorithmus für genetische Programmierung. Der Betrag jeder der eingeführten Erweiterungen wurde experimentell auf sechs Datensets aus drei verschiedenen Gebieten evaluiert:

(1) Die eingesetzte Strategie zur Generierung der initialen Population erhöhte die Qualität der Verknüpfungsregeln auf Datensets mit vielen Attributen. **(2)** Die spezialisierten Crossover-Operatoren erzielten auf allen sechs Datensets eine vergleichbare oder höhere Performance als traditionelles Subtree-Crossover. **(3)** Die Effektivität des eingesetzten Verfahren um ein Aufblähen der Regeln zu verhindern konnte ebenfalls gezeigt werden.

3.2.2 Expressivität der gelernten Verknüpfungsregeln

Die eingesetzte, expressive Repräsentation der Verknüpfungsregeln konnte die Performance der gelernten Verknüpfungsregeln auf allen sechs Datensets erhöhen. Dafür zeigte sich vorwiegend die Einbeziehung von Datentransformationen in die Regeln verantwortlich.

3.2.3 Vergleich mit bestehenden Lernverfahren

Zwei Datensets haben sich dadurch, dass sie bereits für die Evaluierung von mehreren existierenden Lernalgorithmen verwendet wurden, als Benchmarkdatensets etabliert: Das *Cora* Datenset [MNRS00] enthält Zitierungen für Publikationen aus der Cora Computer Science Datenbank für Forschungspublikationen. Das *Restaurant* Datenset [TKM01] enthält Einträge aus dem *Fodor's and Zagat's Restaurant Guide*.

Tabelle 1 vergleicht die Qualität der mit unterschiedlichen Verfahren gelernten Verknüpfungsregeln mit GenLink. GenLink konnte auf beiden Datensets eine höhere Performance erzie-

Ansatz	Cora	Restaurant
Lineare und schwellwertbasierte Klassifikatoren		
Active Atlas [TKM01]	(not available)	99.7% accuracy
MARLIN [BM03]	86.7% F-measure	92.2% F-measure
Genetische Programmierung		
[dCLGdS12]	91.0% F-measure	98.0% F-measure
GenLink	96.6% F-measure	99.3% F-measure
Kollektive Ansätze		
[Dom04]	87.0% F-measure	(not available)
[DHM05]	95.4% F-measure	(not available)
Unüberwachte Ansätze (Top 3)		
RiMOM [WZH+10]	(not available)	81% F-measure
LN2R [SNPR10]	(not available)	75% F-measure
ObjectCoref [HCCQ10]	(not available)	73% F-measure

Tabelle 1: Vergleich verschiedener Ansätze mit GenLink.

len als der bestehende genetische Algorithmus von Carvalho et al. [dCLGdS12]. Darüber hinaus konnte GenLink eine Reihe von bestehenden Lernalgorithmen die unterschiedliche Verfahren benutzen übertreffen.

4 ActiveGenLink — Interaktives Lernen von Verknüpfungsregeln

Der ActiveGenLink-Algorithmus kombiniert aktives Lernen (engl. active learning) mit genetischer Programmierung um Verknüpfungsregeln interaktiv zu lernen. Die für das Lernen der Verknüpfungsregel notwendigen Referenzlinks werden in Interaktion mit einem menschlichen Experten erstellt. Abbildung 5 illustriert die drei Hauptschritte, welche iterativ ausgeführt werden.

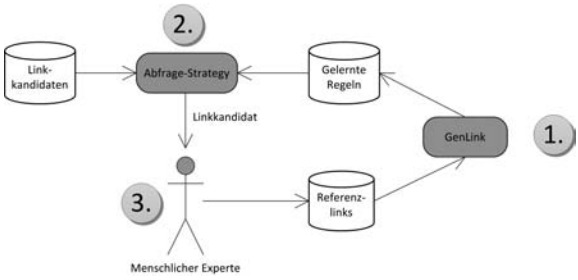


Abbildung 5: Ablauf des ActiveGenLink-Algorithmus.

(1) Die Abfrage-Strategie (engl. query strategy) wählt einen Linkkandidaten aus, für welchen die Verknüpfungsregeln in der gelernten Population unsicher sind. **(2)** Ein menschlicher Experte markiert den ausgewählten Linkkandidaten als korrekt oder inkorrekt woraus ein neuer Referenzlink generiert wird. **(3)** Der GenLink-Algorithmus lernt eine Population von Verknüpfungsregeln aus den aktuellen Referenzlinks.

In der Evaluierung konnten drei Eigenschaften von ActiveGenLink gezeigt werden:

- Auf allen sechs Datenset erreichte ActiveGenLink, nach dem der Nutzer maximal 50 Kandidaten manuell gekennzeichnet hat, eine vergleichbare Genauigkeit als der überwachte GenLink Algorithmus auf dem gesamten Goldstandard, reduzierte also die Anzahl der Linkkandidaten die manuell überprüft werden müssen erheblich.
- Auf einem großen geografischen Datenset konnte die Skalierbarkeit von ActiveGenLink gezeigt werden. Die zwei zu verlinkenden Datensets enthielten 323.257 und 560.123 Einträge und ergeben damit über 180 Milliarden potentielle Linkkandidaten, welche durch die Abfrage-Strategie selektiert werden können. Im Durschnitt dauerte jede Iteration etwa 1,5 Sekunden, wobei jede Iteration das Lernen einer neuen Verknüpfungsregel aus den aktuellen Referenzlinks, sowie das Selektieren eines neuen Linkkandidaten durch die Abfrage-Strategie umfasst.
- Die für ActiveGenLink entwickelte, neuartige Abfrage-Strategie benötigt weniger Referenzlinks als bestehende Strategien.

5 MultiBlock — Effizientes Ausführen von Verknüpfungsregeln

Die Grundidee des MultiBlock-Indizierungsalgorithmus ist das Generieren eines mehrdimensionalen Indexes in welchem die Distanzen der Entitäten zueinander beibehalten werden, d.h. laut Verknüpfungsregel ähnliche Entitäten im Index nahe beieinander liegen. In der Evaluierung konnten drei Haupteigenschaften von MultiBlock gezeigt werden:

Effizienz: Auf einem Datenset mit Personendaten verglichen wir die Leistung von Multiblock mit vier etablierten Indizierungsverfahren. Multiblock war die einzige Me-

thode, die eine Trefferquote von 100% erreichte. Unter den Verfahren mit einer Trefferquote von mindestens 50%, erreichte Multiblock die niedrigste Laufzeit. Auf einem großen geographischen Datenset verglichen wir die Leistung von Multiblock mit dem StringMap-Verfahren, welches ebenfalls einen mehrdimensionalen Index verwendet. Obwohl StringMap eine größere Reduzierung der Anzahl der notwendigen Vergleiche erreichen konnte, dauerte das Generieren des mehrdimensionalen Indexes damit etwa 10-mal länger als der gesamte Prozess mit MultiBlock veranschlagte. Multiblock reduziert die Anzahl der Vergleiche mit einem Faktor von 25.000 und war etwa 800-mal schneller als die vollständige Auswertung aller Paare.

Effektivität: Selbst mit komplexen Verknüpfungsregeln konnte MultiBlock die Anzahl der notwendigen Vergleiche stark reduzieren, ohne korrekte Links auszulassen.

Skalierbarkeit: Es wurde gezeigt, dass MultiBlock mit Hilfe von MapReduce auf einem Rechencluster ausgeführt werden kann. Dazu wurde MultiBlock auf einem Cluster mit 32 Maschinen für ein sehr großes Datenset ausgeführt, wodurch 1 Million Links in 15 Minuten generiert wurden.

6 Fazit

Die vorgestellte Arbeit deckt den vollständigen Prozess der Duplikaterkennung, von der Generierung der Verknüpfungsregel bis zu ihrer Ausführung, ab. Die vorgestellten Verfahren erbringen drei wichtige Beiträge zum aktuellen Stand der Forschung:

Genauigkeit: Der vorgestellte GenLink-Algorithmus verwendet genetische Programmierung zusammen mit einer expressiven Repräsentation für Verknüpfungsregeln um Datenquellen mit *höherer Genauigkeit* zu Verlinken als bisherige Verfahren.

Aufwand: Der vorgestellte ActiveGenLink-Algorithmus *reduziert den Aufwand* der für die Verlinkung zweier Datenquellen notwendig ist, indem er Verknüpfungsregeln interaktiv generiert, während ein menschlicher Experte eine Reihe von Linkkandidaten überprüft.

Effizienz: Das MultiBlock Indizierungsverfahren ermöglicht die *effiziente Ausführung* der gelernten Regeln und schließt somit den Duplikaterkennungsprozess ab. MultiBlock ist effizienter als verglichene Indizierungsverfahren und kann auf einem Rechencluster ausgeführt werden.

Die vorgestellten Algorithmen wurden im *Silk Link Discovery Framework*[1] implementiert, wodurch ein System zur Duplikaterkennung zur Verfügung gestellt wird, welches den vollständigen Duplikaterkennungsprozess unterstützt.

[1] `http://silk.wbsg.de/`

Literatur

[BM03] M. Bilenko und R.J. Mooney. Adaptive Duplicate Detection Using Learnable String Similarity Measures. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seiten 39–48, 2003.

[Chr12] Peter Christen. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer, 2012.

[dCLGdS12] Moisés G de Carvalho, Alberto HF Laender, Marcos André Gonçalves und Altigran S da Silva. A Genetic Programming Approach to Record Deduplication. *IEEE Transactions on Knowledge and Data Engineering*, 24(3):399–412, 2012.

[DHM05] Xin Dong, Alon Halevy und Jayant Madhavan. Reference reconciliation in complex information spaces. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Seiten 85–96, 2005.

[Dom04] Pedro Domingos. Multi-relational record linkage. In *In Proceedings of the KDD Workshop on Multi-Relational Data Mining*, Seiten 31–48, 2004.

[HCCQ10] Wei Hu, Jianfeng Chen, Gong Cheng und Yuzhong Qu. ObjectCoref & Falcon-AO: Results for OAEI 2010. In *Proceedings of the Fifth International Workshop on Ontology Matching (OM)*, Seiten 158–165, 2010.

[MNRS00] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie und Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000. First mention of the Cora dataset.

[SNPR10] Fatiha Saıs, Nobal Niraula, Nathalie Pernelle und Marie-Christine Rousset. LN2R–a knowledge based reference reconciliation system: OAEI 2010 Results. In *Proceedings of the Fifth International Workshop on Ontology Matching (OM)*, Seiten 172–180, 2010.

[TKM01] Sheila Tejada, Craig A. Knoblock und Steven Minton. Learning object identification rules for information integration. *Information Systems*, 26(8):607–633, 2001.

[WZH+10] Zhichun Wang, Xiao Zhang, Lei Hou, Yue Zhao, Juanzi Li, Yu Qi und Jie Tang. RiMOM results for OAEI 2010. In *Proceedings of the Fifth International Workshop on Ontology Matching (OM)*, Seiten 194–201, 2010.

Robert Isele, geboren am 27. Juni 1983 in Waldshut-Tiengen, wohnhaft in Berlin, schloss 2008 das Studium der Informatik an der Technischen Universität München mit dem Diplom mit der Gesamtnote 1,2 ab, worauf er zunächst als Softwareentwickler bei der REC GmbH angestellt war.

Zwischen Oktober 2009 und März 2013 arbeitete er als wissenschaftlicher Mitarbeiter, zunächst bei der Freien Universität Berlin und ab September 2012 an der Universität Mannheim, wo er 2013 seine Promotion mit dem Titel Dr. rer. nat. abschloss. Das Thema der Arbeit lautete “Learning Expressive Linkage Rules for Entity Matching using Genetic Programming” und wurde mit der Gesamtnote “summa cum laude” bewertet. Während seiner Promotion absolvierte er unter anderem einen Forschungsaufenthalt am Yahoo! Research Lab in Barcelona. Seit dem April 2013 ist Robert Isele bei der Brox IT-Solutions GmbH als Consultant angestellt.

Multikriterielle Evolution adaptiver eingebetteter Systeme*

Paul Kaufmann

Institut für Informatik
Universität Paderborn
paul.kaufmann@uni-paderborn.de

Abstract: Die Kombination rekonfigurierbarer elektronischer Bausteine mit den Methoden der künstlichen Intelligenz erschafft für eingebettete Systeme einen eleganten und uniformen Ansatz zur Adaptation an Veränderungen der Umwelt, Defekte der Hardware und Variationen in den Anforderungen an Systemressourcen. Dieses, unter dem Begriff Evolvable Hardware bekannte Prinzip, hat auf eindrucksvolle Weise das Entdecken neuer Entwurfsprinzipien und neuartiger sowie leistungsfähiger Lösungen für bestehende Ingenieursaufgaben aufgezeigt.

In dieser Arbeit präsentieren wir einen ganzheitlichen Ansatz für Evolvable Hardware und stellen unsere Ergebnisse auf dem Gebiet des effizienten Entwurfs Boolescher Schaltungen vor. Die entwickelten Methoden werden für die Evolution von hardwarebasierten Mustererkennungsarchitekturen sowie Prozessoroptimierung eingesetzt.

1 Das Evolvable Hardware Prinzip

Evolvable Hardware ist die immerwährende Optimierung eines rekonfigurierbaren Systems. Vorgestellt 1993 von Higuchi, de Garis u.a. in [HNT+93], liegt Evolvable Hardware folgende Funktionsweise zu Grunde:

1. Basierend auf dem iterativen Zyklus eines evolutionären Algorithmus, startet die lebenslange Optimierung mit einem oder mehreren Bauplänen des zu adaptierenden Systems. In der Terminologie der evolutionären Algorithmen wird ein Bauplan als Genotyp oder Individuum und die Gesamtheit der Individuen als Population bezeichnet. Das zu optimierende System ist bereits mit einer vorher evolvierten oder entwickelten Lösung im Einsatz. Der geschilderte Ablauf wird in der Abbildung 1(a) anhand der Adaptation eines Bildkomprimierers veranschaulicht.

2. In der nächsten Phase werden die Individuen bezüglich ihrer Leistungsfähigkeit (Fitness) bewertet. Dazu bedarf es aktueller Messwerte, wie etwa der neusten Kameraaufnahmen, wie in Abbildung 1(b) dargestellt. Mit diesen Eingabedaten kann für jeden Genotyp die Kompressionsrate berechnet werden, die z.B. vom aktuellen Ausleuchtungsgrad abhängt (Abbildung 1(c)).

*Englischer Titel der Dissertation: “Adapting Hardware Systems by Means of Multi-Objective Evolution” [Kau13]

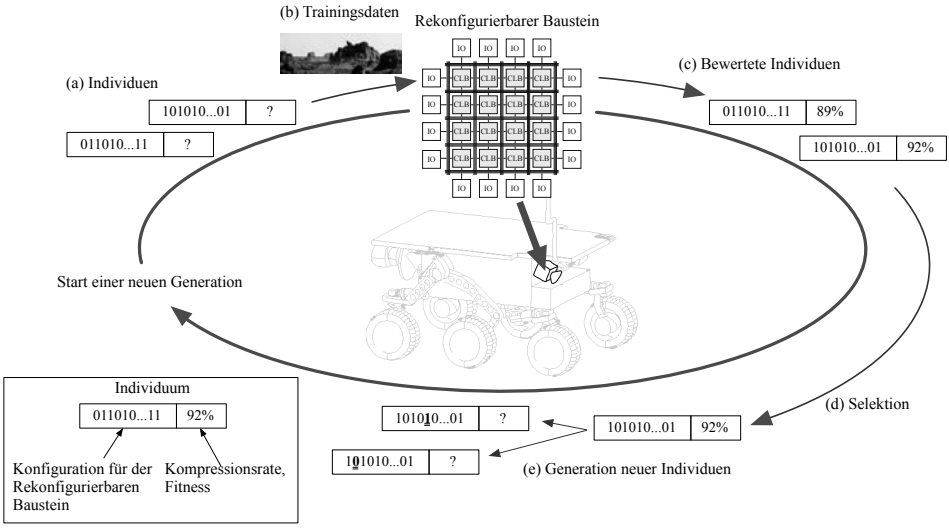


Abbildung 1: Adaptiver Bildkompressor mithilfe von Evolvable Hardware.

Bei der Auswertung der Fitness gibt es einen entscheidenden Punkt: Wird die Fitness auf der selben Hardware ausgewertet, die später auch das Individuum operativ ausführt, ist auch eine Kompensation von Hardwaredefekten möglich. Dies liegt daran, dass ein evolutionärer Algorithmus in seiner klassischen Ausprägung stets die Funktionalität einer Lösung zum Ziel hat. Die innere Struktur der Lösung unterliegt dabei dem Optimierungsprozess. Veränderungen und Fehler in den Rechenressourcen können somit implizit berücksichtigt und kompensiert werden. Die Auswertung der Fitness auf der Zielplattform erlaubt auch die Evolution von Lösungen für Rechenplattformen, die nur angenähert formalisiert werden können und die Schwankungen im Herstellungsprozess unterliegen. Solche Plattformen sind z.B. rekonfigurierbare Transistor- und Operationsverstärkerbausteine.

3. Nun kommen die Prinzipien der biologischen Evolution zum Zuge. Geleitet von der Fitness, werden zielgerichtet bestehende und gute Eigenschaften verschiedener Individuen kombiniert (das Prinzip der geschlechtlichen Fortpflanzung, Rekombination) sowie bestehende Lösungen durch Hinzufügen fundamental neuer Eigenschaften mittels zufälliger Veränderungen des Genotyps (Mutation) sukzessiv verbessert (Abbildung 1(d),(e)). Mit der Erzeugung einer neuen Population schließt sich der Optimierungszyklus eines evolutionären Algorithmus und im nächsten Schritt kann die Fitness der neuen Individuen bewertet werden.

Das Evolvable Hardware Prinzip eignet sich insbesondere für die Adaptation von Zielfunktionen, die eine graduelle Funktionsgüte haben. Das können Funktionen sein, deren interne Struktur von der Verteilung der Eingabedaten abhängt und Funktionen, die zwar eine perfekte Lösung besitzen, aufgrund der endlichen Ressourcen aber mit einer kleinen und suboptimalen Realisierung auskommen müssen. Beispiele dafür sind Mustererkennungsalgorithmen, Regel- und Einbahnfunktionen.

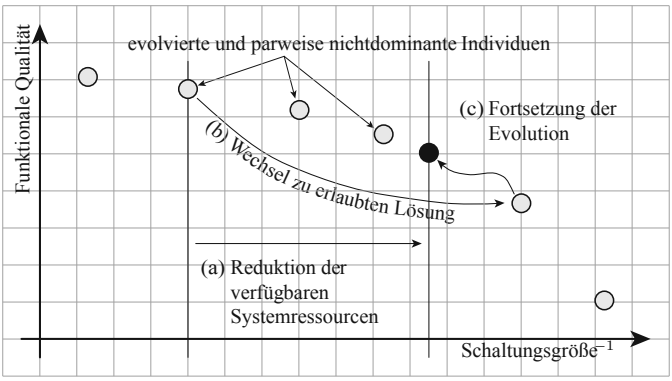


Abbildung 2: Kompensation rascher Änderungen mithilfe multikriterieller Algorithmen.

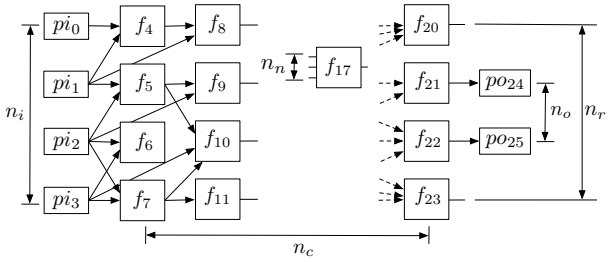


Abbildung 3: Cartesisches Genetische Programmieren.

Unsere Motivation für eine holistische Realisierung von Evolvable Hardware gründet in der Einsicht, dass es mehrere Zeitebenen geben kann, in denen sich ein autonomes System anpassen können muss. Während kontinuierliche Veränderungen mit niedrigen Veränderungsraten, üblicherweise Veränderungen in der Systemumwelt, durch kontinuierliche Optimierung kompensiert werden können, bedürfen rasche Veränderungen, oft Veränderungen in den Systemressourcen, präevolvierter Lösungen. Um einen autonomen Betrieb des adaptiven Systems zu gewährleisten, können in diesem Fall Paretobasierte multikriterielle evolutionäre Algorithmen eingesetzt werden. Diese können zur Laufzeit Lösungen evolvieren, die verschiedenartig bezüglich der sich rasch verändernden Zielfunktionen sind. Ein Beispiel ist in der Abbildung 2 dargestellt. Kreise symbolisieren Lösungen, die sich bezüglich der funktionalen Qualität und Größe unterschieden. Bei einer plötzlichen Reduktion der verfügbaren Systemressourcen (Abbildung 2(a)) wird die gerade operative Lösung durch eine Lösung ersetzt, die die neue Vorgaben einhält (Abbildung 2(b)). Danach wird der Optimierungsprozess fortgesetzt und es wird versucht, die funktionale Qualität unter Ausnutzung der noch verfügbaren Ressourcen zu maximieren (Abbildung 2(c)).

Der Fokus unserer Arbeit liegt auf digitaler Evolvable Hardware unter Verwendung von feldprogrammierbaren Logikbausteinen (FPGA). Dazu bedarf es zunächst einer Formalisierung des FPGA Berechnungsmodells. Wir verwenden hier die kartesische genetische Programmierung (CGP) nach Miller und Thomson [MT00], vorgestellt in Abbildung 3. Es besteht, ähnlich einem FPGA, aus gitterartig angeordneten Funktionsblöcken. Um die

Komplexität der Funktionsauswertung vertretbar zu halten, bildet CGP nur den kombinatorischen Fall ab. Dazu dürfen Funktionsblöcke keine Speicher und Signalpfade keine Zyklen enthalten. Leitungen dürfen nur Funktionselemente einer Spalte mit Funktionselementen in den darauf folgenden Spalten verbinden. Die maximale Leitungslänge l kann beschränkt werden.

In FPGA basierten Systemen können rasche Veränderungen durch Veränderungen in der Auslastung des FPGAs oder durch Neuzuteilung der FPGA Ressourcen für die aktiven Applikationen verursacht werden. Damit sind Schaltungsgröße und Verarbeitungsgeschwindigkeit Optimierungsziele, bezüglich derer verschiedenartige CGP Schaltung evolviert werden müssen.

In dieser Arbeit haben wir uns folgenden Herausforderungen gestellt: Zunächst benötigt das EHW Konzept einen effizienten, Paretobasierten multikriteriellen evolutionären Algorithmus (MOEA). Ein solcher Algorithmus gehört üblicherweise zu der Familie der globalen Optimierer. Dazu setzen diese Algorithmen den Rekombinationsoperator ein, um entfernte Bereiche des Suchraums erkunden und Informationsdrift zwischen den Individuen realisieren zu können. Der Suchraum der CGP Schaltungen ist aber hochgradig nichtlinear und begünstigt lokale Sucher, die sich ausschließlich auf den Mutationsoperator verlassen. Für den Einsatz Paretobasierter MOEAs für die Evolution von CGP Schaltungen ist somit ein effektiver Rekombinationsoperator notwendig. Unsere Arbeiten auf diesem Gebiet sowie zur Verbesserung der Skalierbarkeit der Evolution von CGP Schaltungen werden im Kapitel 2 zusammengefasst.

Des Weiteren lässt sich die Effizienz evolutionärer Methoden anhand synthetischer Testfunktionen abschätzen, die tatsächliche Verwendbarkeit und Nützlichkeit einer randomisierten Heuristik ist aber nur an realen Anwendungen demonstrierbar. Aus diesem Grund evaluieren wir im Kapitel 3 unsere Methoden an den Beispielen eines Mustererkenners für Prothesensteuerungen und an der Optimierung eines Prozessorcaches.

2 Effiziente Multikriterielle Evolutionäre Algorithmen für CGP

In diesem Kapitel fassen wir unsere Arbeiten zum CGP Rekombinationsoperator, der automatischen Akquise und Wiederverwendung von CGP Subfunktionen, der Priorisierung von Optimierungszielen und Periodisierung von Suchalgorithmen zusammen.

2.1 Strukturbasierter CGP Rekombinationsoperator

Im CGP Modell kann eine Funktion auf verschiedene Arten kodiert werden. Z.B. lässt sich die Funktion $\mathrm{po}_{24} = (\mathrm{pi}_0 \text{ AND } \mathrm{pi}_1)$ bei geeignet gewähltem Parameter l durch jeden der Funktionsblöcke im CGP Gitter in der Abbildung 3 berechnet werden. Die räumliche Anordnung der Funktionsblöcke, obwohl irrelevant für die berechnete Funktion, wird mitkodiert und mitevolviert und macht bei strukturellen Manipulationen einer Lösung, z.B. bei Austausch von Teillösungen, eine Revision der Abbildung der Funktion auf das CGP

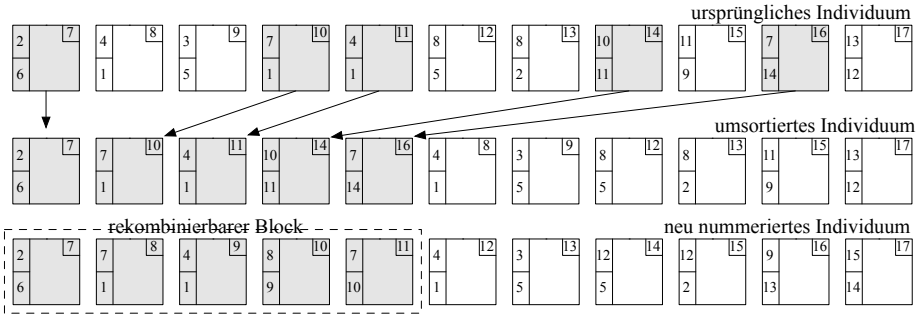


Abbildung 4: CGP Rekombination. Die Nummer in der rechten oberen Ecke eines Funktionsblocks ist seine laufende Nummer. Die Nummern auf der linken Seite sind Nummern funktional abhängiger Blöcke (z.B. grau schattiert).

Gitter notwendig. Da dieser Prozess nicht trivial ist, wurde er bisher nicht untersucht.

Möchte man nun einen Paretobasierten multikriteriellen Evolutionären Algorithmus auf CGP anwenden, ist die Implementierung eines Rekombinationsoperators, der für jede rekombinierte Funktion eine neue Abbildung auf das CGP Gitter berechnet, unumgänglich. Um dies zu bewerkstelligen, betrachten wir nur den eindimensionalen CGP Raum ($n_c = 1, l = \infty$). Für die Rekombination wird für ein Elterindividuum ein Funktionsblock (Wurzel) und die Anzahl der auszutauschenden Funktionsblöcke n zufällig gewählt. Eine Breitensuche markiert maximal n funktional abhängige Vorgängerknoten der Wurzel und das CGP Individuum wird so permutiert, dass die markierten Funktionsblöcke einen konsolidierten Block bilden (Abbildung 4). Das ist immer möglich und das resultierende Individuum ist stets eine gültige CGP Schaltung. Rekombination ist somit auf den Austausch solcher Blöcke zurückführbar.

Tabelle 1: Berechnungsaufwand für strukturbasierten Rekombinationsoperator sowie struktur- und altersbasierte Akquisition von Subfunktionen. Hervorgehobene Zahlen bedeuten eine Verbesserung. $1 + 4$ ES ist der Refernzalgorithmus.

	$1+4$ ES	Rekombination		Altersbasiert		Strukturbasiert	
	absolut	absolut	relativ	absolut	relativ	absolut	relativ
2×2 mul	66.623	64.111	$\mathbf{-3.8\%}$	51.961	$\mathbf{-22.0\%}$	49.052	$\mathbf{-26.4\%}$
3×3 mul	8.840.574	2.518.964	$\mathbf{-71.5\%}$	6.001.917	$\mathbf{-32.1\%}$	3.638.120	$\mathbf{-58.9\%}$
85% EMG	18.260	19.859	$+8.8\%$	14.743	$\mathbf{-19.3\%}$	23.855	$+30.7\%$
95% EMG	510.147	576.988	$+13.1\%$	314.311	$\mathbf{-38.4\%}$	873.319	$+71.2\%$

Die Auswertung des Rekombinationsoperators ist in der Tabelle 1 vorgestellt. Als Vergleichsmetrik dient der Berechnungsaufwand nach Koza [Koz94] und als Testfunktionen verwenden wir 2×2 und 3×3 Multiplizierer sowie Mustererkennungsfunktionen (EMG), deren Evolution beim Erreichen von 85% bzw. 95% der Trainingsgüte gestoppt und als Erfolg gewertet wird. Aus der Tabelle lässt sich schlussfolgern, dass für Multiplizierer, die viele repetitive Substrukturen besitzen, der strukturbasierte Rekombination sehr effizient ist wogegen für die Mustererkennungsfunktionen, die weniger ausgeprägte innere Struktur

aufweisen, der Operator im Vergleich zu lokaler Suche im Nachteil ist. Eine detailliertere Auswertung ist in [Kau13] zu finden.

2.2 Automatische Identifikation und Wiederverwendung von Subfunktionen

Die zuvor implementiere Funktion zur Extraktion funktional abhängiger Substrukturen kann dazu verwendet werden, die Skalierbarkeit evolutionärer Algorithmen zu verbessern. Die Idee ist es, häufig vorkommende Substrukturen automatisch zu identifizieren und als primitive Funktionale, Module, der Menge der erlaubten CGP Blockfunktionen hinzuzufügen und wiederzuverwenden [Koz94]. Dazu haben wir unseren strukturbasierten Rekombinationsoperators erweitert und anhand gleicher Testfunktionen wie im vorigen Experiment untersucht. Die ursprünglichen Ergebnisse werden auch in diesem Experiment bestätigt (Tabelle 1). Die Evolution von arithmetischen Schaltungen kann deutlich verbessert werden wogegen für Mustererkennugsfunktionen der strukturbasierte Operator wenig effektiv bleibt.

Um auch für nichtarithmetische Funktionen über einen gut skalierbaren evolutionären Operator zu verfügen, entwickelten wir ein altersbasierters Verfahren zur automatischen Identifikation von Modulen. Dieses Verfahren geht davon aus, dass, ähnlich der Entstehung von Organen in der biologischen Evolution, über viele Generationen unverändert gebliebene Substrukturen implizit zum Erfolg eines Individuums beitragen. Für die Umsetzung des Verfahrens bekommt jeder CGP Funktionsblock einen Zähler, der bei der Erzeugung einer neuen Generation inkrementiert wird. Wird ein Funktionsblock dagegen von einem Mutationsoperator verändert, wird sein Zähler auf Null gesetzt. Für strukturbehaftete Zielfunktionen, wie etwa Multiplizierer, ist die altersbasierte Modulerzeugung zwar besser als der Referenzalgorithmus, reicht aber an die Leistung der strukturbasierten Modulerstellung nicht heran (Tabelle 1). Dagegen ist die altersbasierte Modulerstellung deutlich besser als die strukturbasierte Modulerstellung und besser als der Referenzalgorithmus für die Evolution von Mustererkennungsfunktionen mit wenig innerer Struktur.

2.3 Priorisierung von Optimierungszielen

Im Suchraum der CGP Schaltungen ist die Skalierbarkeit Paretobasierter MOEAs oft dramatisch schlechter als die Skalierbarkeit lokaler Verfahren, wie z.B. der (1+4) Evolutionärer Strategien (ES). Um die Skalierbarkeit Paretobasierter MOEAs zu verbessern, haben wir zunächst versucht, die impliziten Abhängigkeiten der Optimierungsziele Schaltungsgröße, Verarbeitungsgeschwindigkeit und funktionale Qualität auszunutzen, ohne dabei die Diversität der evolvierten Lösungsmengen zu beeinträchtigen. Dazu haben wir den populären Algorithmus SPEA2 [Kau13] erweitert, indem wir den Selektionsdruck für die Evolution neuer Individuen abhängig von einer Linearkombination aller Optimierungsziele gemacht und die Gewichte der Linearkombination in vorhergehenden Experimenten bestimmt haben. Dabei blieb der Diversitätsmechanismus von SPEA2 unangetastet.

Mit diesem Ansatz lässt sich die multikriterielle Evolution von CGP Schaltungen bereits entscheidend verbessern. Der neue Algorithmus ist nicht nur in der Lage, häufiger Individuen mit hoher funktionaler Qualität zu finden, sondern auch Lösungsmengen zu evolvieren, die in ihrer Diversität den Lösungsmengen der Referenzalgorithmen in Nichts nachstehen und sogar oft besser sind [Kau13].

2.4 Periodisierung Lokaler und Globaler Suche

Die Priorisierung von Optimierungszielen definiert eine Suchrichtung. Für einen universelleren Optimierer, der ohne eine vorgegebene Suchrichtung verschiedenartige nichtdominante Lösungsmengen zuverlässig finden kann, greifen wir zu der Idee der Algorithmusperiodisierung. Dazu definieren wir eine Sequenz aus Algorithmen und eine Wiederholungsfunktion f. Die Algorithmussequenz wird zyklisch ausgeführt wobei die Ausgabepopulation eines Algorithmus in der Sequenz die Eingabepopulation des nächsten Algorithmus wird. Abhängig von f und der Historie des Optimierungslaufs, wird ein Algorithmus einmal, mehrmals oder auch gar nicht ausgeführt.

Bei der Periodisierung von Paretobasierten MOEAs ist zu beachten, dass die erzeugten Lösungsmengen eine innere Struktur haben. Nicht-Pareto basierte Algorithmen bedürfen deswegen einer Anpassung. Da sich (1+4) ES als sehr effizient für CGP gezeigt haben, entschieden wir uns dieses lokale Suche Verfahren für die Periodisierung anzupassen. Im Folgenden unter dem Namen hybride ES (hES) geführte Algorithmus führt für jedes Individuum der Eingabepopulation einen (1+4) ES Schritt aus und selektiert aus den vier neuen Individuen und dem Elterindividuum exakt ein Individuum aus, das dominant unter diesen fünf Individuen ist oder bezüglich einer Ähnlichkeitsmetrik über alle Individuen als möglichst verschiedenartig eingestuft wird. Um neutrale Suche zu ermöglichen, wird das Elterindividuum nur dann in die neue Generation kopiert, wenn es strikt besser als seine Kinderindividuen ist.

In Experimenten zeigte sich ein ähnliches Bild wie bei der Priorisierung von Optimierungszielen. Die Periodisierung von lokalen und globalen Suchern ist vorteilhaft bezüglich der Evolution von funktional qualitativen Schaltungen ohne dabei negativ für die Verschiedenartigkeit der evolvierten Lösungsmenge zu sein.

3 Evolvable Hardware Anwendungen

Die vorgestellten Methoden zur effizienten Evolution von CGP Schaltungen haben wir auf Mustererkennungsalgorithmen für Prothesensteuerungen und Prozessorcaches angewendet. Beide Applikationen können von Laufzeitadaption profitieren, um z.B. die Klassifikationsgüte aufrechtzuerhalten oder die Ausführungszeit eines Prozessors für neue Applikationen und Datensätze zu minimieren.

3.1 Evolvierbare Klassifizierer

In Abbildung 5 ist ein CGP basierter Ensembleklassifizierer vorgestellt. Jedes Klassifikationsmodul k besteht aus mehreren CGP Schaltungen, die darauf trainiert sind, für Eingabevektoren aus der k-ten Kategorie eine "1" und sonst eine "0" zu berechnen. Das Klassifikationsmodul mit der größten Anzahl aktivierter CGP Schaltungen bestimmt das Ergebnis des Klassifizierers.

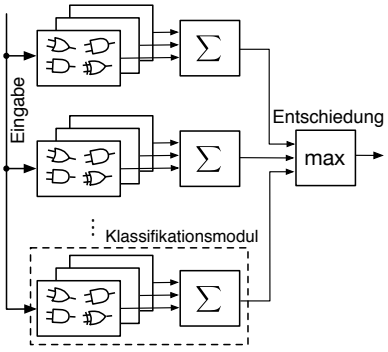


Abbildung 5: CGP basierter Ensembleklassifizierer.

Für einen realistischen Datensatz haben wir über 21 Tage in 121 Sitzungen Muskelspannungen des Unterarmes für elf Bewegungen aufgenommen. Dabei haben wir für die Merkmalsextraktion das gleitende Mittel verwendet, da dieses sich sehr effizient auf einem eingebetteten System implementieren lässt.

In initialen Experimenten konnten wir beobachten, dass die Klassifikationsgüte eines Mustererkennungsalgorithmus nachlässt, wenn dieser nur mit den Daten der ersten Sitzungen trainiert wird. Das macht ein periodisches Training mit aktuellen Daten notwendig. Ausgehend von dieser Beobachtung haben wir ein Evaluationsschema festgelegt, bei dem ein Algorithmus mit den Daten der Sitzungen $(i-5, \ldots, i-1)$ trainiert und mit den Datensatz i ausgewertet wird.

Die Tabelle 2 vergleicht unseren Ansatz mit den Methoden der k-nächsten-Nachbarn, Entscheidungsbäumen, neuronalen Netzen und Support Vektor Maschinen. Der Evolvable Hardware Klassifizierer reicht zwar nicht an die Leistung bester konventioneller Algorithmen heran, ist aber besser als die Entscheidungsbäume. Weiterhin können bei einer tatsächlichen Implementierung Eigenschaften wie effizientes Lernen, Kompaktheit und Energieeffizienz vor der absoluten (aber mindestens ausreichenden) Klassifikationsleitung ausschlaggebend sein.

Tabelle 2: Gemittelter Generalisierungsfehler.

	Fehlerrate
kNN	10.45
DT	17.91
MLP	10.44
SVM	**9.00**
CGP	16.48

In einem weiteren Experiment wurde die Laufzeitadaptionsfähigkeit des EHW Klassifizierers untersucht, indem man die Anzahl der Klassifikationsschaltungen in einem Klassifikationsmodul zur Laufzeit variierte und den Klassifizierer neu trainierte. Man konnte dabei die Beobachtung machen, dass der EHW Ensembleklassifizierer bereits mit sehr wenigen Klassifikationsschaltungen pro Klassifikationsmodul sehr gut diskriminieren kann und dass mehr Ressourcen zu einer schnelleren Wiederherstellung der Erkennungsraten führen können.

3.2 Evolvierbare Prozessorcaches

Eine Anwendung, die zunächst nicht strikt der Definition eines Evolvable Hardware System entspricht, ist das Optimieren einer Cachefunktion, die Speicheradressen eines Prozessors auf Cacheadressen abbildet. Führt man die Optimierung zur Laufzeit oder gar verteilt durch, entsteht ein Evolvable Hardware System das sich nicht wie bisher in der physikalischen Welt bewegt, sondern im globalen Computernetzwerk sein Habitat hat.

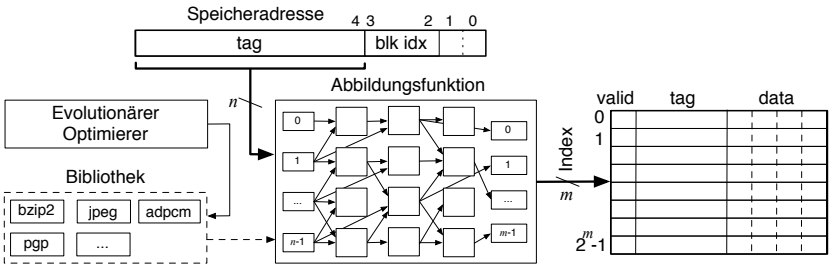


Abbildung 6: Das Prinzip eines evolvierbaren Prozessorcaches.

Die Idee evolvierbarer Cachefunktionen ist in der Abbildung 6 vorgestellt. Anstatt die Indexbits einer Speicheradresse für die Indizierung innerhalb eines Caches zu verwenden, werden die Tag- und Indexbits Eingaben einer rekonfigurierbaren CGP Schaltung, die mithilfe eines evolutionären Algorithmus optimiert wird. Die Schaltungsausgaben werden zur Cacheindizierung benutzt.

Tabelle 3: Verbesserung gegenüber einem konventionellen Cache in %.

	bzip2		jpeg	
	L1	L2	L1	L2
Ausführungszeit	5,84	7,95	14,31	12,96
Missrate	6,83	9,77	41,25	40,35
Energieverbrauch	5,71	7,83	16,43	14,46

Wir haben die Idee der evolvierbaren Caches an den Beispielen des bzip2 Komprimierers und des jpeg Kodierers evaluiert. Dazu haben wir in einer zyklengenauen Simulation für beide Anwendungen und vorher definierte Eingabedaten für entweder L1I und L1D oder L1I, L1D und L2 die Cachefunktionen evolviert und anschließend unter Benutzung von Testeingabedaten evaluiert. Die Speicherhierarchie des simulierten Systems bestand dabei aus einem Prozessor, einem L1 Instruktions- und Datencache (L1I, L1D), einem L2 Cache und dem Hauptspeicher. Die Ergebnisse sind in der Tabelle 3 zusammengefasst. Das wichtigste Ergebniss ist, dass die evolvierten Cachefunktionen sehr gut für unbekannte Eingabedaten generalisieren. Für bzip2 kann die Ausführungszeit und der Energieverbrauch um nahezu 8% gegenüber einem konventionellen System reduziert werden. Im jpeg Fall sind es sogar mehr als 14% und 16%. Die Trefferrate eines Caches kann bei bzip2 um 9,7% und bei um bis zu 41% verbessern werden. Die evolvierten Cachefunktionen sind im Durschnitt 13 bis 16 CGP Funktionsblöcke groß wobei der längste Pfad bis zu vier Funktionsblöcke enthalten kann.

4 Zusammenfassung und Ausblick

Das Ziel dieser Arbeit war die Entwicklung und Erprobung eines ganzheitlichen Konzepts für Evolvable Hardware. Die wesentlichen Ergebnisse hierbei sind effiziente multikriterielle Methoden für den automatischen Entwurf und die Optimierung Boolescher CGP Schaltungen sowie Anwendungen, die mithilfe dieser Methoden umgesetzt werden konnten und somit die Nützlichkeit des Evolvable Hardware Ansatzes verdeutlichen.

Im Kontext dieser Arbeit können folgende Schlussfolgerungen gemacht werden: Das Prinzip der nichtdeterministischen und kontinuierlichen Adaptation eines Systems beschränkt sich nicht nur auf autonome und adaptive eingebettete Systeme sondern kann für artfremde Anwendungen überraschend neuartige und innovative Lösungsansätze schaffen. Weiterhin kann das Evolvable Hardware Konzept im konventionellen Ingenieursbereich, trotz berechtigter Skepsis gegenüber nichtdeterministischer Adaptation, Lösungen schaffen und Anwendungen ermöglichen, die von traditionellen Adaptivitätsansätzen nicht mit heutiger Technologie und Materialien erreicht werden können.

Literatur

[HNT+93] Tetsuya Higuchi, Tatsuya Niwa, Toshio Tanaka, Hitoshi Iba, Hugo de Garis und Tatsumi Furuya. Evolving Hardware with Genetic Learning: a First Step Towards Building a Darwin Machine. In *From Animals to Animats*, Seiten 417–424. MIT Press, 1993.

[Kau13] Paul Kaufmann. *Adapting Hardware Systems by Means of Multi-Objective Evolution*. Logos Verlag, Berlin, 2013.

[Koz94] John R. Koza. *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, 1994.

[MT00] J. Miller und P. Thomson. Cartesian Genetic Programming. In *European Conf. on Genetic Programming (EuroGP)*, Seiten 121–132. Springer, 2000.

Paul Kaufmann studierte Informatik und Mathematik und schloss seine Dissertation auf dem Gebiet der Evolvierbaren Hardware an der Universität Paderborn im Jahr 2013 mit Auszeichnung ab. In den Jahren 2012 und 2013 war er des Weiteren Mitglied des Fraunhofer Institut für Windenergie und Energiesystemtechnik sowie der Gruppe für Energiemanagement und Betrieb Elektrischer Netze der Universität Kassel. Z.Zt. forscht Herr Kaufmann als Postdoktorand am Lehrstuhl für technische Informatik der Universität Paderborn an adaptiven eingebetteten Systemen, multikriteriellen evolutionären Algorithmen, hardwarebasierten Mustererkennungsalgorithmen und Optimierung intelligenter Stromnetze.

Die Support Feature Machine: Eine Odyssee in hochdimensionalen Räumen*

Sascha Klement

Institut für Neuro- und Bioinformatik
Universität zu Lübeck
klement@inb.uni-luebeck.de

Abstract: Wie finden wir die kleinste Menge von Merkmalen, die ein bestimmtes Verhalten am besten beschreiben, wenn es daneben zahllose irrelevante und irreführende Merkmale gibt? Als Beitrag zur Beantwortung dieser Frage haben wir die Support Feature Machine (SFM) entwickelt — ein effektives und effizientes Verfahren zur Merkmalsselektion in Klassifikationsproblemen. Die SFM basiert auf der Approximation der 0-Norm des Normalenvektors der trennenden Hyperebene durch Minimierung der 1-Norm. Am Beispiel von Datensätzen aus funktionaler Magnetresonanztomografie(fMRT) zeigen wir, dass die SFM mit wenigen Erweiterungen in der Lage ist, motorische Bewegungen und sogar emotionale Zustände probandenübergreifend vorherzusagen. Mit der SFM haben wir eine neuartige zukunftsweisende Methode entwickelt, die für die zukünftige Entwicklung von Brain Computer Interfaces eine wertvolle Grundlage bietet.

1 Präambel

Odysseus (engl. Ulysses, griech. Οδυσσέας) ist der wohl berühmteste Held der griechischen Sagenwelt; seine abenteuerliche Heimreise nach dem Trojanischen Krieg wird in Homers Odyssee erzählt. Motive, Charaktere und Analogien an die Odyssee finden sich zu allen Zeiten in vielfältiger Form in Kunst und Literatur.

Darüber hinaus wird der Begriff Odyssee mit einer langen Irrfahrt oder mit einer mit vielen Schwierigkeiten verbundenen, abenteuerlichen Reise gleichgesetzt und im metaphorischen Sinne für intellektuelle Herausforderungen verwendet. Nicht zuletzt passt der Begriff als Beschreibung für den wissenschaftlichen Fortschritt im Allgemeinen, aber auch das Verfassen einer Dissertation im Besonderen. Die vorliegende Odyssee hat annähernd sechs Jahre gedauert. Ich danke all den wunderbaren Menschen — allen voran meiner Frau Johanna — die mich auf diesem Weg umgeben und begleitet, die mein Denken nachhaltig beeinflusst und mich gelehrt haben, die Klippen der Wissenschaft geschickt zu umschiffen.

Darüber hinaus scheint die Odyssee — ganz im Sinne des Originals — weiter anzudauern. Beim Schreiben dieser Zeilen hat sich wieder ein Jahr zwischen mich und die Ergebnisse

*Englischer Titel der Dissertation: The Support Feature Machine: An Odyssey in High-dimensional Spaces

der Arbeit geschoben ohne mich losgelassen zu haben. In der Zwischenzeit habe ich ein Unternehmen gegründet und bin im Tagesgeschäft zumeist mit Projekt- und Personalmanagement, Business Development und Investoren beschäftigt, aber auch mit dem Transfer von jahrelanger Forschungsarbeit in ein marktreifes Produkt.

Die Ergebnisse dieser Arbeit haben das Potential sehr bald ebenfalls diesen Weg von der Forschung in eine Produkt zu nehmen. Konkret lassen sich die Ergebnisse dazu nutzen, Hirnzustände vorherzusagen — also Gedanken zu lesen und somit Brain-Computer-Interfaces zu ermöglichen. Damit liefert unsere Methode nicht nur theoretische und praktische Vorteile gegenüber vergleichbaren Verfahren, sondern kann auch in absehbarer Zeit wirtschaftlich genutzt werden.

2 Einleitung

In der Krebsforschung, Genomanalyse oder in den Neurowissenschaften stehen Wissenschaftlern und Anwendern komplexe Messgeräte zur Datenaufnahme zur Verfügung, die meist hochdimensionale Daten erzeugen: Eine einzelne Messung kann hunderte bis Millionen Merkmale liefern. Organisatorische, technische und finanzielle Rahmenbedingungen begrenzen die Anzahl der Messungen jedoch auf einige wenige und erfordern damit spezielle Methoden, um überhaupt valide und statistisch signifikante Vorhersagen treffen zu können.

Wie finden wir die kleinste Menge von Merkmalen, die ein bestimmtes Verhalten am besten beschreiben, wenn es daneben zahllose irrelevante und irreführende Merkmale gibt? Dies ist eine der zentralen Fragen in den Bereichen Künstliche Intelligenz, Maschinelles Lernen, Neuronale Netze, Support Vector Machines und Statistik. Dieses Lernen anhand von wenigen Beispielen mit vielen Freiheitsgraden ist auch deshalb anspruchsvoll, weil alle relevanten biologischen und medizinischen Daten genau von dieser Art sind.

Die hier zusammengefasste Arbeit [Kle13] liefert Antworten auf Fragen aus drei Bereichen: Zunächst vertieft sie das Verständnis der Andersartigkeit von hochdimensionalen Daten mit wenigen Beispielen — dazu zeigen wir, wann und warum maschinelle Lernverfahren, wie die Support Vector Machine (SVM), nicht in der Lage sind, valide Vorhersagen auf Basis derartiger Daten zu machen. Wir haben die Support Feature Machine (SFM) entwickelt, eine effektive und effiziente Methode zur Merkmalsselektion, die auf der Approximation der 0-Norm des Normalenvektors der trennenden Hyperebene durch Minimierung der 1-Norm basiert. Die Überlegenheit dieses Verfahrens gegenüber Support Vector Verfahren lässt sich sowohl theoretisch wie auch empirisch zeigen [KM11]. Mit wenigen Erweiterungen ist die SFM in der Lage, Bewegungen und sogar emotionale Zustände probandenübergreifend allein auf der Basis von funktioneller Magnetresonanztomografie (fMRT) vorherzusagen. Mit der SFM haben wir damit eine universelle Methode zur Merkmalsselektion entwickelt, die insbesondere zur Analyse von fMRT Daten geeignet erscheint.

Wir beginnen mit der Geometrie hochdimensionaler Daten, beschreiben die Grundzüge der Support Feature Machine und zeigen dann, welche Ergebnisse sie beim Dekodieren von Hirnaktivität liefert.

3 Geometrie hochdimensionaler Daten

Basierend auf der Statistischen Lerntheorie lassen sich Konvergenz- und Fehlerverhalten von Lernverfahren abschätzen. Voraussetzung für ihre Gültigkeit sind jedoch ausreichend große Datenmengen, die die zugrunde liegende Wahrscheinlichkeitsverteilung hinreichend genau approximieren. Das Fehlen dieser Voraussetzung wird gerne als *Fluch der Dimensionalität* (*curse of dimensionality*) bezeichnet.

Manche Eigenschaften hochdimensionaler Daten sind offensichtlich: Sie können weder manuell analysiert noch in einer übersichtlichen Form visualisiert werden; die geringe Anzahl an Datenpunkten kann mit Sicherheit nicht die gesamte Variabilität der Daten abbilden; schließlich erfüllen selbst moderne Hochleistungsrechner selten die enormen technischen Anforderungen bezüglich Rechenzeit und Speicherbedarf.

Daneben sind es weniger offensichtliche und damit leicht zu übersehende Eigenschaften, die die Auswertung erschweren: Die Geometrie hochdimensionaler Daten — wie die einzelnen Datenpunkte im Raum zueinander angeordnet sind — entspricht nicht im Entferntesten dem, was wir aus den uns bekannten 2-dimensionalen und 3-dimensionalen Räumen erwarten würden. Zufällig gewählte Datenpunkte beispielsweise verhalten sich in hochdimensionalen Räumen in einigen Aspekten zunehmend deterministisch — je größer die Dimension desto eher liegen sie auf den Ecken eines regulären Simplex: sie haben zueinander den gleichen Abstand, ebenso wie zum Ursprung und die Ortsvektoren stehen orthogonal zueinander (*distance concentration*, [HMN05]). Diese Geometrie lässt auf Abstandsmetriken basierende Lernverfahren unerwartete Ergebnisse liefern oder komplett fehlschlagen. Dies ist ein durchaus bekanntes Verhalten, wird jedoch in den meisten Fällen ignoriert, da unbekannt ist, wann sich zufällige endlich-dimensionale Datenmengen deterministisch wie unendlich-dimensionale verhalten. Im Falle von Leave-one-out-Kreuzvalidierung haben wir bewiesen [KMMM08], *dass* dieses Verhalten eintritt und wir haben handliche Charakteristiken entwickelt, die erlauben, zu entscheiden, *wann* es eintritt. Die Beweisführung besitzt eine gewisse Ästhetik — mit der Annahme endlicher Datenmengen aus einem unendlich-dimensionalen Eingaberaum ist die Support Vector Lösung für zufällige Daten exakt bestimmbar.

Jedes Lernverfahren, das auf Berechnung von Abständen beruht, hat im Prinzip die gleiche Schwäche. Allerdings wurde gezeigt, dass die Nutzung der 1-Norm oder der 0-Norm als Abstandsmaß anstatt der euklidischen Distanz zur Verbesserung beitragen kann [FWV07]. Dies ist ein Grund, warum die in dieser Arbeit vorgestellte Support Feature Machine — die auf 0-Norm Approximation beruht — gewisse inhärente Vorteile bringt.

Grundsätzlich steht aber die Erkenntnis im Vordergrund, dass irrelevante Merkmale verworfen werden sollten, wo immer es möglich ist. Die beste Vorgehensweise wäre es, diese Merkmale von Anfang an durch intuitives Vorwissen überhaupt nicht einzubeziehen. Wir haben gezeigt, dass die VC-Dimension — das wesentliche Maß der Mächtigkeit von Klassifikatoren — von kombinierter Merkmalsselektion und Klassifikation immer noch größer ist, als wenn wir ausschließlich anhand der tatsächlich relevanten Merkmale klassifizieren würden [Kle13]. Dies bedeutet, dass die zu erwartenden Fehler größer sind.

4 Support Feature Machine

Die Support Feature Machine als wesentlicher Beitrag der Arbeit ist eine Methode zur Merkmalsselektion und zielt darauf ab, die kleinste Anzahl von Merkmalen zu finden, so dass mit diesen Merkmalen eine lineare Trennung zweier Klassen ohne Fehler möglich ist [KM10a, KM10b, KM11, KAM13]. Die SFM kombiniert Merkmalsselektion und Klassifikation in ein iteratives Verfahren und minimiert das strukturelle Risiko (*Structural Risk Minimization*) durch Beschränkung der Klassifikationsfunktionen auf solche mit wenigen Parametern (Dimensionen). Viele Verfahren zur Merkmalsselektion sind eher konservativ, d.h. sie verwerfen ausschließlich Merkmale, die mit an Sicherheit grenzender Wahrscheinlichkeit irrelevant sind. Die SFM hingegen verwirft möglichst viele Merkmale, um nur tatsächlich relevante übrig zu behalten.

Grundsätzlich sucht die SFM eine Klassifikationsfunktion $f(\vec{x}) = \operatorname{sgn}(\vec{w}^{\mathrm{T}}\vec{x} + b)$, die für jeden Datenvektor $\vec{x}$ das korrekte Klassenlabel $f(\vec{x})$ vorhersagt. Die zu bestimmenden Parameter $\vec{w}$ und b beschreiben die Lage einer Hyperebene. Ziel ist die Trennung beider Klassen durch die Hyperebene wobei der Gewichtsvektor $\vec{w}$ möglichst wenige von 0 verschiedene Einträge haben soll. Je weniger solche Einträge er hat, desto geringer ist die Anzahl der relevanten Merkmale.

Die mathematische Formulierung der SFM wurde inspiriert durch die 0-Norm Support Vector Machine von Weston [WMC+00, WEST03] mit dem Unterschied, dass sie auf Minimierung der verwendeten Merkmale abzielt und dies nicht mit einer Maximierung des Abstandes beider Klassen verbindet:

$$\begin{array}{rl} \text{Minimiere} & \|\vec{w}\|_0^0 \\ \text{mit} & y_i\left(\vec{w}^{\mathrm{T}}\vec{x}_i + b\right) \geq 0 \quad \forall i \\ \text{und} & \frac{1}{n}\sum_{i=1}^{n} y_i\left(\vec{w}^{\mathrm{T}}\vec{x}_i + b\right) = 1 \ . \end{array} \tag{1}$$

wobei jedes $\vec{x}_i$ einen Datenpunkt darstellt, y_i das zugehörige Klassenlabel bezeichnet (+1 oder -1), $\vec{w}$ der Normalenvektor der trennenden Hyperebene ist und b ihr Bias, d.h. die Verschiebung bezüglich des Ursprungs des Koordinatensystems.

Die Lösung des nichtlinearen Optimierungsproblems lässt sich iterativ approximieren (Abbildung 1, die Software ist auf unserer Webseite verfügbar [1]) — die Wahl eines geeigneten linearen Optimierers ist jedoch alles andere als trivial. Wir haben dazu die Laufzeit für vier verschiedene Optimierer und zwei alternative mathematische Formulierungen empirisch evaluiert. Abhängig von der Anzahl der Merkmale und der Anzahl der Datenpunkte lässt sich das wahrscheinlich günstigste Verfahren ablesen — für hochdimensionale Daten ist das kommerzielle Optimierungswerkzeug CPLEX am geeignetsten.

Abbildung 2 zeigt die Unterschiede zwischen Support Vector Machine, Support Feature Machine und der verwandten SVM-basierten Methode von Weston.

Die SFM lässt sich mit den von der Support Vector Machine bekannten Methoden *weich machen*, also auf nicht linear trennbare Klassen erweitern [KM10a]. Dazu haben wir eine Formulierung entwickelt, die explizit auf unbalancierte Szenarien eingeht, bei denen eine

[1] `http://www.inb.uni-luebeck.de/tools-demos/support-feature-machine`

Eingabe : Datenpunkte $\vec{x}_i$ und zugehörige Klassenlabel y_i
Ausgabe : Gewichtsvektor $\vec{w}$ und Bias b

Setze $\vec{z} = (1, \ldots, 1)$
repeat
  Minimiere $|\vec{w}|$ unter $y_i \left(\vec{w}^{\mathrm{T}}(\vec{x}_i * \vec{z}) + b\right) \geq 0$ und $\frac{1}{n}\sum_{i=1}^{n} y_i \left(\vec{w}^{\mathrm{T}}\vec{x}_i + b\right) = 1$
  Setze $\vec{z} = \vec{z} * \vec{w}$
until *Konvergenz*

Abbildung 1: Iterativer SFM Algorithmus.

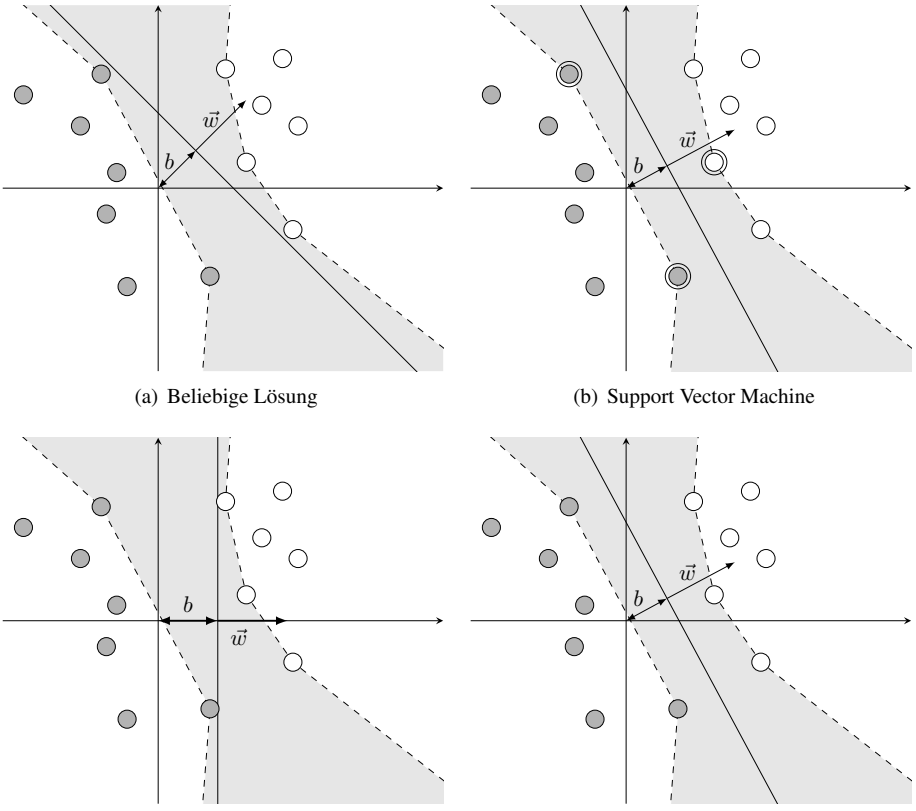
Klasse deutlich überrepräsentiert ist. Die Repetitive SFM bezeichnet schließlich ihre wiederholte Anwendung mit dem Ziel, nach und nach Mengen von Merkmalen mit abnehmender Relevanz zu finden [KAM13].

Experimente auf künstlichen und realen Daten zeigen, dass die SFM in der Tat relevante Merkmale sehr effektiv findet und die Klassifikationsleistung signifikant gegenüber einer Standard SVM verbessern kann. Selbst wenn exponentiell viele irrelevante Merkmale hinzugefügt werden, bleibt die Klassfikationsleistung annähernd gleich [KM10a].

5 Mindreading

In den kognitiven Neurowissenschaften haben sich in den letzten Jahren zunehmend Methoden etabliert, die basierend auf neuronaler Aktivität — gemessen über invasive Methoden, Elektroenzephalografie (EEG) oder funktionale Magnetresonanztomografie (fMRI) — neuartige Anwendungen wie Brain-Computer Interfaces, Neuromarketing oder Lügendetektoren ermöglichen [Hay11]. Die zu lösenden Herausforderungen reichen von informationstheoretischen Betrachtungen (wieviel Information wird durch neuronale Aktivität transportiert) über methodische (wie lässt sich diese Information extrahieren) bis hin zu praktischen Anforderungen (wie lassen sich die erforderlichen Geräte bezahlbar machen und miniaturisieren). Umso erstaunlicher ist es, dass multivariate Lernverfahren erst seit Kurzem überhaupt zur Analyse in Betracht gezogen werden; klassischerweise werden noch immer univariate voxelweise Statistiken verwendet (wie z.B. [FHW$^+$95]), obwohl bekannt ist, dass das Gehirn Informationen in komplexeren Strukturen verarbeitet. Die SFM bietet als inhärent multivariate Methode die Voraussetzung, um tatsächlich neue Erkenntnisse über die Verarbeitungskapazität des menschlichen Gehirns zu erlangen.

Die grundlegenden Betrachtungen der SFM haben die Annahme bestätigt, dass sie in der Lage ist, zwei Klassen mit der geringsten Anzahl von Merkmalen zu unterscheiden. Ihre Anwendung auf Hirndaten zielt nun auf die folgenden Fragen ab: Wie gut lassen sich Hirnzustände grundsätzlich unterscheiden? Wie viele Voxel sind mindestens notwendig, um funktionale Hirnzustände zu unterscheiden und wie groß sind die Regionen, die Information tragen? Wie viele Voxel tragen relevante Information und welches sind die am wenigsten



(a) Beliebige Lösung

(b) Support Vector Machine

(c) Support Feature Machine

(d) Westons Methode

Abbildung 2: Lineare Klassifikatoren. Der Datensatz (Klasse +1/weiß vs. Klasse -1/grau) ist linear trennbar — jede Gerade, die ausschließlich in der grauen Region verläuft und deren Gewichtsvektor $\vec{w}$ in Richtung der positiven Klasse zeigt, ist eine gültige Lösung z.B. der Klassifikator (a). Der Support Vektor Klassifikator (b) trennt die Klassen so, dass sie maximalen Abstand zueinander haben. Er ist eindeutig beschrieben durch die Support Vektoren (markiert durch zusätzliche Kreise), die alle den gleichen Abstand zur Entscheidungsgrenze haben. Im Gegensatz dazu minimiert die Support Feature Machine (c) die Anzahl der Merkmale — in diesem Fall reicht ein Merkmal (x-Achse) aus, um beide Klassen zu trennen. Die Lösung ist nicht eindeutig; sie kann leicht horizontal verschoben werden, ohne dass sich der Trainingsfehler ändern würde. Für diesen Datensatz liefert die SVM-basierte Methode von Weston (d) die gleiche Lösung wie die Standard SVM und ist damit nicht in der Lage, die minimale Anzahl von Merkmalen zu finden.

Abbildung 3: Univariate Statistik und Support Feature Machine liefern konsistente Ergebnisse bei der Unterscheidung von einfacher motorischer Hirnaktivität. Signifikante Voxel (rot, voxel-weiser t-Test, $p \leq 0.001$ entspricht 2.5% aller Voxel) und relevante Voxel (grün, repetitive SFM, 2.5% aller Voxel) überlappen sich in großen Teilen (gelb). Wie zu erwarten, liegen die ausgewählten Voxel hauptsächlich im *Gyrus praecentralis* und im *Gyrus postcentralis* (Motorcortex und somatosensorischer Cortex).

relevanten Voxel? Welche zeitlichen Muster lassen sich beobachten? Welche örtliche Auflösung ist mindestens erforderlich, um Hirnzustände zu unterscheiden?

Unseren Experimenten liegen zwei fMRT Datensätze zugrunde: Einer bestehend aus Daten motorischer Hirnaktivität beim Drücken eines Kopfes mit rechtem oder linkem Daumen (12 Probanden, 2 Klassen (rechts/links), 4 Datenpunkte je Klasse und Proband, 50989 Merkmale) und einer mit affektiver Hirnaktivität beim Empfinden von Emotionen (6 Probanden, 5 Klassen (Freude/Trauer/Furcht/Ekel/Ärger), 8 Datenpunkte je Klasse und Proband, 41642 Merkmale). Ein Datenpunkt repräsentiert die Aktivität im gesamten Gehirn; ein Voxel entspricht einem Merkmal. Die Daten wurden mit den üblichen in den Neurowissenschaften verwendeten Werkzeugen vorbereitet (SPM5, Wellcome Department of Imaging Neuroscience, London, UK). Alle Vorhersagen wurden stets probandenübergreifend getroffen, d.h. die Testdaten enthielten jeweils ausschließlich Messungen eines Probanden, dessen Daten nicht in den Trainingsdaten enthalten waren (*Leave-one-participant-out cross validation*).

Im Falle motorischer Hirnaktivität ist die SFM in der Lage, ähnliche Regionen wie die üblicherweise verwendeten univariaten Statistiken zu erkennen und erlaubt zusätzlich den Informationsgehalt zu quantifizieren. Bereits zwei Merkmale — also zwei Hirnvoxel der Größe 3 x 3 x 3 mm — reichen aus, um eine Klassifikationsrate von über 96% zu erzielen. Weiterhin konnten wir beobachten, dass bis zu 35% des Gehirns Informationen tragen, mit denen unterschieden werden kann, ob die Person einen Knopf mit dem linken oder dem rechten Daumen drückt (siehe Abbildung 3, [KAM13]).

Die SFM als multivariate Methode kann darüber hinaus auf komplexe affektive Hirnzustände angewendet werden, bei denen univariate Methoden nicht die mehrdimensionalen Abhängigkeiten zu dekodieren vermögen. Unsere Experimente zeigen, dass dies tatsächlich der Fall ist — Emotionen lassen sich vorhersagen, die Fehlerraten variieren jedoch stark; *Freude* scheint die am besten messbare Emotion zu sein; die SFM benötigt für die Klassifikation von *Freude* vs. *Trauer* nur 4 Merkmale/Hirnvoxel um eine Klassifikationsrate von 77% zu erreichen. Weiterhin lässt sich mit der SFM zeigen, dass affektive Informationen über große Bereiche des Gehirns verteilt und redundant vorliegen (siehe Abbildung 4). Für manche Paare von Emotionen enthält fast das gesamte Gehirn Informationen, um sie zu

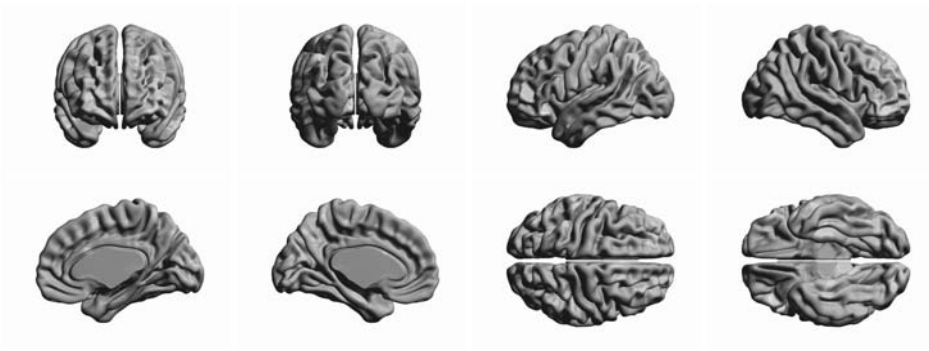


Abbildung 4: Hirnregionen, die eine Unterscheidung von *Freude/Trauer* (rot) und *Furcht/Trauer* (grün) ermöglichen. Überlappende Regionen sind gelb. Für beide Paar sind nur die 2.5% relevantesten Merkmale gezeigt.

unterscheiden. Darüber hinaus verbessert sich die Separierbarkeit von Emotionen, je länger ein Proband sich in diesem Zustand befindet. Gleichzeitig erhöht sich die Redundanz der affektiven Information [KAM13].

Zur genaueren Untersuchung dieser Redundanz haben wir die Experimente mit Hirndaten geringerer örtlicher Auflösung wiederholt (und dabei ein Methode aus der Bildverarbeitung basierend auf der Gauss-Pyramide angewandt, die ebenfalls im Rahmen dieser Arbeit entwickelt wurde [KTB11]). Selbst auf zweifach unterabgetasteten Hirndaten (< 700 Voxel) lassen sich Emotionen mit annähernd gleicher Performance wie auf den Originaldaten unterscheiden; bei motorischer Aktivität reichen sogar weniger als 100 Voxel aus.

Diese Ergebnisse sind umso beeindruckender, wenn man die möglichen Fehlerquellen betrachtet: Die Rohdaten bestehen aus wenigen Messungen einiger weniger Probanden; in der Vorverarbeitung werden die Hirndaten zwar auf ein Standardhirn registriert, individuelle Unterschiede in der Hirnstruktur können dadurch aber nicht vollständig kompensiert werden; auch die Support Feature Machine kann durch Ausreißer beeinflusst werden und liefert nicht notwendigerweise die optimale Lösung. Selbst unter diesen Bedingungen ist die SFM in der Lage, Hirnaktivität mit hoher Genauigkeit zu dekodieren und ermittelt eine hohe Redundanz der Information. Damit nehmen wir an, dass der wahre Informationsgehalt sogar noch größer, verteilter und redundanter sein kann, als von unseren Experimenten vorhergesagt.

6 Ausblick

Hochdimensionale Räume haben derartig ungünstige Eigenschaften, dass ihre Erforschung einer Odyssee gleicht — dennoch lassen sich bei Kenntnis und unter Berücksichtigung dieser Eigenschaften sehr wohl Methoden entwickeln, die trotzdem die relevanten Merkmale extrahieren und valide Vorhersagen treffen können.

Theoretisch und experimentell haben wir gezeigt, dass die Support Feature Machine wesentlich besser als andere Verfahren geeignet ist, in hochdimensionalen Daten anhand weniger Beispiele relevante Merkmale zu bestimmen. Dabei zeichnet sich die SFM nicht nur dadurch aus, tatsächlich die relevanten Merkmale mit hoher Wahrscheinlichkeit zu liefern, sondern auch dadurch, dass sich ihre Vorteile gegenüber anderen Verfahren beweisen lassen. Angewandt auf fMRT Daten stellt sich heraus, dass affektive Information im menschlichen Gehirn räumlich verteilt und mit einem hohen Grad an Redundanz gespeichert ist. Trotz weniger Messungen und großer individueller Varianz ist die SFM in der Lage, motorische Bewegungen und sogar Emotionen zu dekodieren.

Das Potential der SFM als universelle Methode zur Merkmalsextraktion eröffnet nun Forschungsansätze in eine Vielzahl von Richtungen: Wie kann die SFM weiter verbessert werden? Wie kann die Korrektheit der ermittelten Merkmale erhöht werden? Lässt sich die Laufzeit durch andere lineare Optimierer weiter verbessern? Kann die SFM einfacher ohne lineare Optimierer implementiert werden? Lässt sich beweisen oder widerlegen, dass die SFM optimal bezüglich der Auswahl der Merkmale ist? Lassen sich mit der SFM tatsächlich neue Erkenntnisse über die Informationsverarbeitung im menschlichen Gehirn erzielen oder lassen sich nur schon bekannte Fakten bestätigen? Wie lassen sich die Erkenntnisse durch alternative Methoden bestätigen? Kann die SFM auch komplexere zeitabhängige Gedanken lesen? Ist Gedankenlesen überhaupt eine wünschenswerte Technologie und was sind die ethischen Gefahren einer derart mental invasiven Methode?

Ungeachtet dieser theoretischen Überlegungen schreitet die Entwicklung von portablen Brain-Computer Interfaces fort — bisher fast ausschließlich basierend auf EEG Daten, wobei die Anzahl der Elektroden und damit das Auflösungsvermögen äußerst beschränkt ist (das Emotiv Headset besitzt z.B. 14 Elektroden). Portable MRT Scanner existieren allenfalls im Prototyp-Stadium, hätten aber deutlich höhere Ortsauflösung. Portable Devices (Smartphones, Tablets) sind bereits allgegenwärtig; Wearable Interfaces (Google Glass, Oculus Rift) machen rasante Fortschritte und werden in wenigen Jahren ebenfalls allgegenwärtig sein; Brain-Computer Interfaces sind dann der nächste logische Schritt. Wir sehen die SFM als einen wesentlichen Beitrag, die methodischen Voraussetzung zu schaffen, um derartige Geräte in wenigen Jahren Wirklichkeit werden zu lassen.

Literatur

[FHW+95] K.J. Friston, A.P. Holmes, K.J. Worsley, J.B. Poline, C. Frith und R.S.J. Frackowiak. Statistical Parametric Maps in Functional Imaging: A General Linear Approach. *Human Brain Mapping*, 2:189–210, 1995.

[FWV07] Damien François, Vincent Wertz und Michel Verleysen. The Concentration of Fractional Distances. *IEEE Transactions on Knowledge and Data Engineering*, 19:873–886, 2007.

[Hay11] John-Dylan Haynes. Special Issue on Multivariate Decoding and Brain Reading. *NeuroImage*, 56, 2011.

[HMN05] P. Hall, J. S. Marron und A. Neeman. Geometric representation of high dimension, low sample size data. *Journal of the Royal Statistical Society*, 67(3):427–444, 2005.

[KAM13] Sascha Klement, Silke Anders und Thomas Martinetz. The Support Feature Machine: Classification with the Least Number of Features and its Application to Neuroimaging Data. *Neural Computation*, 25:1548–1584, 2013.

[Kle13] Sascha Klement. *The Support Feature Machine: An Odyssey in High-dimensional Spaces*. Dissertation, Universität zu Lübeck, 2013.

[KM10a] Sascha Klement und Thomas Martinetz. A new approach to classification with the least number of features. In *9th International Conference on Machine Learning and Applications*, Seiten 141–146. IEEE Computer Society, 2010.

[KM10b] Sascha Klement und Thomas Martinetz. The Support Feature Machine for Classifying with the Least Number of Features. In Konstantinos I. Diamantaras, Wlodek Duch und Lazaros S. Iliadis, Hrsg., *ICANN (2)*, Jgg. 6353 of *Lecture Notes in Computer Science*, Seiten 88–93. Springer, 2010.

[KM11] Sascha Klement und Thomas Martinetz. On the Problem of Finding the Least Number of Features by L1-norm Minimisation. In Timo Honkela, Hrsg., *Proceedings of the 21st International Conference on Artificial Neural Networks*, Lecture Notes in Computer Science 6791, Seiten 315–322. Springer, Heidelberg, 2011.

[KMMM08] Sascha Klement, Amir Madany Mamlouk und Thomas Martinetz. Reliability of Cross-Validation for SVMs in High-Dimensional, Low Sample Size Scenarios. In *Proceedings of the 18th International Conference on Artificial Neural Networks*, Seiten 41–50, Berlin, Heidelberg, 2008. Springer-Verlag.

[KTB11] Sascha Klement, Fabian Timm und Erhardt Barth. Illumination correction for image stitching. In *Proceedings of the International Conference on Imaging Theory and Applications*, Jgg. 1, Seiten 81–86. INSTICC, 2011.

[WEST03] Jason Weston, Andr'e Elisseeff, Bernhard Schölkopf und Mike Tipping. Use of the Zero-Norm with Linear Models and Kernel Methods. *Journal of Machine Learning Research*, 3:1439–1461, 2003.

[WMC+00] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio und V. Vapnik. Feature Selection for SVMs. In *Advances in Neural Information Processing Systems*, 2000.

Sascha Klement hat in Lübeck Informatik studiert mit Schwerpunkten in Neuroinformatik, Maschinellem Lernen und Bildverarbeitung. Am Institut für Neuro- und Bioinformatik der Universität zu Lübeck hat er promoviert und als Projektleiter der Pattern Recognition Company GmbH Forschungs- und Entwicklungsprojekte geleitet — von der Optimierung von Walzvorgängen in Stahlwerken über Hochauflösende Scanner bis hin zu Human Machine Interfaces. Seit 2011 ist er Unternehmer und hat die gestigon GmbH als Managing Director/CTO gegründet und aufgebaut. Hier entwickelt er mit seinem Team Technologien zur berührungslosen Steuerung von Geräten allein durch Gesten für die Märkte Automotive und Consumer Electronics. Er ist IKT-Innovativ Preisträger, KfW Gründerchampion und hat den Weconomy Award gewonnen.

Modulare Modellspezifikation auf der Dokumentationsebene – Konzepte und Anwendung in einer webbasierten Modellierungsumgebung*

Stefan Kuntsche

KWT-9
Technische Universität Berlin
Straße des 17. Juni 135
10623 Berlin
stefan.kuntsche@alumni.tu-berlin.de

Abstract:
In diesem Beitrag werden neue Ansätze zur Spezifikation von gleichungssystembasierten Modellen vorgestellt, die in einer neuen Modellierungsumgebung mit dem Namen MOSAIC umgesetzt wurden. Der Grundgedanke ist es dabei, die Beschreibung der Modelle in einer Form vorzunehmen, wie sie in Fachbüchern und Artikeln vorgefunden wird. Dies schließt Betrachtungen über ausdruckstarke und eindeutige mathematische Notationen und die möglichen dokumentationsorientierten Speicherformate ein. Mit Hilfe von automatischer Code-Generierung können die Modelle schließlich in unterschiedlichen numerischen Umgebungen eingesetzt werden. Durch den Fokus auf die Darstellung der Modelle in der Literatur oder Dokumentation wird ein neuer Ansatz der Modularisierung ermöglicht. Sowohl bei der Modularisierung als auch bei der Verwendung der Modelle durch Code-Generierung steht die gemeinsame Nutzung und Wiederverwendung der Modelle und Modellteile im Vorderund. Die entwickelte Modellierungsumgebung ist als Internetanwendung implementiert und erlaubt die zentralisierte Zusammenarbeit von vierschiedenen Standorten aus.

1 Einleitung

Die hier zusammengefasste Arbeit [Kun13] entstand innerhalb des Exzelenz-Clusters UniCat (www.unicat.tu-berlin.de). Die interdisziplinäre, interkulturelle und örtlich verteilte Zusammenarbeit innerhalb dieses Forschungsprojekts ist stellvertretend für eine moderne Art des Teamworks in der heutigen globalisierten Welt. Daher wurde die Verbesserung der Modellierungswerkzeuge hinsichtlich der Zusammenarbeit zu einem Teil der Forschungsaufgabe.

In mathematischen Modellen der Verfahrenstechnik spielen Systeme aus algebraischen Gleichungen und Differentialgleichungen eine bedeutende Rolle. Die Grundlage bilden

*Englischer Titel der Dissertation: “Modular Model Specification on the Documentation Level – Concepts and Application in a Web-Based Modeling Environment”

dabei in der Regel die Erhaltungssätze von Masse, Energie und Impuls sowie Zustandsgleichungen, Teilmodelle für Stoffeigenschaften und vieles mehr. Diese Gleichungssysteme müssen aufgrund ihrer Größe numerisch gelöst und daher in einer geeigneten Programmiersprache implementiert werden. Nun stellt das Gleichungssystem mit den darin verwendeten, besonders auf das gegebene Problem zugeschnittenen Formeln den eigentlichen Wissenschatz dar, den es einerseits zu erhalten und - im Falle von öffentlicher Forschung - zu verbreiten gilt. Die Beschreibung des Modells in mathematischen Formeln in einer Spezifikation oder einer Veröffentlichung (also auf der Dokumentationsebene) ist daher von großer Bedeutung. Dagegen ist das implementierte Modell für den Austausch wenig geeignet, unter anderem da es spezifisch für nur eine Plattform, d.h. genau eine Kombination aus Programmiersprache und numerischer Umgebung geschrieben wurde. Auch für den Erhalt des Wissensschatzes sind implementierte Modelle langfristig ungeeignet, da sie an den Lebenszyklus der Plattform gebunden sind [FLM98]. Das in der Dokumentationsebene korrekt und vollständig beschriebene Modell ist demnach der wichtigere Wissensträger. Nichtsdestoweniger ist die fehlende, falsche oder unvollständige Dokumentation ein kontinuierlich diskutiertes Thema [Gas78, FLM98, KM09]. Die Erarbeitung der Modelle direkt auf der Dokumentationsebene in Verbindung mit Code-Erzeugung garantiert eine fehlerfreie Programmierung und zugleich vollständige und korrekte Dokumentation. Gleichzeitig entfallen dabei die manuelle Programmierung sowie die fortlaufende Aktualisierung der Dokumentation.

Die Wiederverwendbarkeit von Modellen ist von großer Bedeutung für die Produktivität. Die Code-Genierung für verschiedene Plattformen auf Basis der dokumentierten Gleichungssysteme ist ein wesentlicher Anwendungsfall der Wiederverwendung. Im Modellierungsprozess bezieht sich die Wiederverwendung darüber hinaus auf die Zugänglichkeit zu Teilmodellen sowie auf die Verwendung von vollständigen Modellen in unterschiedlichen Kontexten [BD06]. Die Beschreibung der Modelle auf der Dokumentationsebene muss daher mit einem Modellierungskonzept einhergehen, das diese beiden Anforderungen erfüllt. Dies schließt die Definition von Teilmodellen und den bestmöglichen Zugriff auf diese ein.

Vier Interessengebiete Aus diesen Betrachtungen lassen sich vier Interessengebiete ableiten:

1. Spezifikation der Modelle auf der Dokumentationsebene, d.h. Beschreibung der Modelle in einer mathematischen symbolischen Notation, welche eindeutig hinsichtlich ihres mathematischen Inhalts und gleichzeitig ausdrücklich für die direkte Verwendung in Artikeln, Handbüchern usw. geeignet ist. Dies geht über die in der Regel nicht übertragbaren Notationen, welche in Computer-Algebra-Software verwendet werden, hinaus und schließt eine Betrachtung der verwendbaren Speicherformate (z.B. Markup-Sprachen und Formate der Echtbilddarstellung (WYSIWYG)) ein.

2. Entwickeln eines stark modularen Modellierungskonzepts, welches die Verwendung herkömmlicher Schnittstellen zwischen Modulen vermeidet und stattdessen Techniken anwendet, die im Umgang mit Büchern, Artikeln, usw. vorgefunden werden.

3. Code-Erzeugung zur Verwendung der Modelle in den existierenden leistungsfähigen Sprachen und Software-Werkzeugen, wobei eine größtmögliche Abdeckung der Programmiersprachen angestrebt wird.
4. Einbetten der oben genannten Techniken in eine webbasierte Modellierungssoftware, welche die zentralisierte Zusammenarbeit über das Internet oder - im Falle von Industrieunternehmen - das Intranet unterstützt

Diese Punkte wurden im Rahmen der Dissertation bearbeitet. Auf Punkte 1 bis 3 wird in den folgenden Abschnitten näher eingegangen.

2 Eindeutige Modelldefinition auf der Dokumentationsebene

2.1 Problemstellung und Ziel

Die grundsätzliche Zielstellung dieses Abschnittes ist es, Gleichungen in Dokumenten zu speichern, die gleichzeitig als Dokumentation dienen, und aus diesen Gleichungen den Quellcode für die numerische Lösung des Problems zu generieren. Die verwendete symbolische Notation soll ausdrucksstark sein. Insbesondere soll ein Variablenname Zeichen in verschiedenen Positionierungen enthalten können, z.B. $\gamma_{o,i}^{L,V}$. Neben toolspezifischen Ansätzen (MathCad, Mathematica, MapleSim,[Sam72, KKG92]) gibt es Ansätze in denen spezielle Datenformate als Träger des mathematischen Inhalts verwendet werden (z.B. Content-MathML, OpenMath), wobei die visuelle Darstellung nicht im Vordergrund steht und je nach verwendeter Visualisierungsmethode auch uneindeutig sein kann [KB06, SW06]. Im Gegensatz dazu existieren die Speicherformate der Dokumentationsebene, zu denen hier Markup-Sprachen wie Latex oder WYSIWYG-Formate wie DOCX zusammengefasst werden. Allerdings sind diese Formate grundsätzlich nicht für die Speicherung mathematischen Inhalts vorgesehen.

Die Fragestellung ist, ob und unter welchen Bedingungen eine Ablage und Übertragung von mathematischem Inhalt in Speicherformaten der Dokumentationsebene möglich ist.

2.2 Konzept

Im Fokus des hier vorgeschlagenen Ansatzes steht die Eindeutigkeit der symbolischen Notation, mit folgender Hypothese: Wenn der mathematische Inhalt durch einen Ausdruck korrekt und eindeutig *angezeigt* wird, dann lässt sich dieser Inhalt aus der Darstellung mit Hilfe einer Maschine ermitteln. Allerdings sind weder die gebräuchlichen noch die in Standardwerken ([Int09, BSMM07]) definierten Regeln für diesen Schritt eindeutig genug. Der hier verfolgte Ansatz geht daher über das zusätzliche Einführen von Notationsregeln. Abgesehen von der Notation spielen die Fähigkeiten der Speicherformate eine wichtige Rolle. Für dieses Zusammenspiel wurde ein theoretisches Rahmenwerk erstellt bestehend aus:

1. Charakterisierung der im Rahmenwerk behandelten Notationen
2. Spezifikation der für die Definition von Notationen notwendigen Punkte
3. Bedingungen für die Notationen
4. Bedingungen für die Speicherformate
5. Beiweise, in denen Punkte 1 bis 4 als Voraussetzungen gelten und die folgendes zeigen:
   - Es ist möglich, den mathematischen Inhalt aus einem Ausdruck zu ermitteln der in einem Speicherformat der Dokumentationsebene abgelegt ist.
   - Der Inhalt des Ausdrucks bleibt bei der Übetragung zwischen verschiedenen Speicherformaten erhalten, wenn die Darstellung erhalten bleibt. Dies gilt insbesondere auch dann, wenn der mathematische Inhalt des Ausdrucks bei der Übertragung nicht berücksichtigt wird (kein Parsen).

2.3 Anwendung

Die in der vorgestellten Modellierungsumgebung verwendete mathematische Notation wurde so entwickelt, dass sie die Bedingungen für Notationen der obigen Systematik erfüllt (Punkt 3).

Um die Übertragbarkeit des mathematischen Inhalts zwischen Speicherformaten der Dokumentationsebene zu veranschaulichen wurde folgender Test vorgenommen: Zunächst wurden Gleichungen in verschiedenen Speicherformaten der Dokumentationsebene erzeugt (unter anderem Gleichung (1)). Diese wurden dann über mehrere Schritte in andere Speicherformate übertragen (vgl. Abbildung 1).

$$K_{j,i}^{L,V} = \frac{\gamma_{j,i} \cdot p_{o,j,i}^{L,V}}{p_j} \tag{1}$$

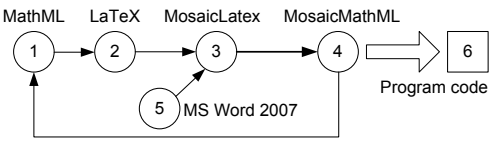


Abbildung 1: Übertragungsschritte zwischen Speicherformaten der Dokumentationsebene

Der mathematische Inhalt wird nur in Übertragungsschritt (3-4) ausgewertet [1] . Alle anderen Übertragungsschritte sind nur auf die korrekte Übertragung der Darstellung ausgelegt.

[1] Die in Abb. 1 erwähnten Sprachen MosaicLatex und MosaicMathML sind gültige Untermengen der Sprachen Latex und MathML im Präsentations-Markup. Sie wurden für die Anwendung in der Modellierungsumgebung MOSAIC entwickelt, um die Komplexität der Eingabeverifikation und des Parsens zu reduzieren.

Für Schritt (1-2) wurde die unabhängige Software MathParser [WW04] verwendet, Schritte (2-3) und (5-3) beruhen auf dem Ersetzen der für die Darstellung relevanten Muster von Zeichenfolgen. Die Erzeugung der Gleichungen erfolgte in unabhängiger, auf die Darstellung fokussierter Software: Firemath [Bon12] (MathML im Präsentations-Markup), MathType (LaTeX) und Microsoft Word 2007. Die Übertragung verlief für die getesteten Formeln fehlerfrei, sodass der mathematisch korrekte Programm-Code erzeugt werden konnte (4-6).

Die Verwendung unabhängiger, darstellungsorientierter Werkzeuge unterstreicht dabei folgendes:

- Wird eine eindeutige Notation verwendet, dann kann die visuelle Darstellung von Formeln als Träger des mathematischen Inhalts verwendet werden.
- Die Erhaltung des Inhalts ist *unabhängig* vom Speicherformat der Dokumentationsebene.
- Die Übertragung des Inhalts zwischen Speicherformaten ist unabhängig von der verwendeten Software, sofern die visuelle Darstellung korrekt übertragen wird.
- Der Lebenszyklus des Inhalts ist daher losgelöst von dem der erzeugenden oder übertragenden Software sowie der Speicherformate.

3 Ein modulares Modellierungskonzept mit hohem Wiederverwendungsgrad

Das Modellierungskonzept verfolgt zwei Hauptziele: Erstens soll ein hoher Wiederverwendungsgrad erreicht werden und zweitens soll sich die Modellspezifikation an der Präsentation in beschreibenden Dokumenten wie Artikeln und Büchern orientieren.

Ein häufig angewandtes Mittel zur Erhöhung des Wiederverwendungsgrades ist die Erzeugung von Modellbausteinen [FLM98]. Für den Fall der hier betrachteten Gleichungssysteme ist das kleinst-mögliche Element die Gleichung. Für Programmiersprachen ist eine derart hohe Granularität keineswegs ratsam, da die notwendige Verbindung durch Modulschnittstellen oder zusätzliche triviale Gleichungen einen hohen Aufwand bedeuten würde. Auf der Dokumentationsebene ergibt sich jedoch ein anderes Bild: Gleichungen können einem System zugeordnet werden, wenn sie die gleiche Namenskonvention verwenden. Diese Namenskonvention spiegelt sich in der Nomenklatur wider, welche die Variablen erklärt und den Leser gleichzeitig bei der korrekten Anwendung der Gleichungen unterstützt. Daraus ergeben sich zunächst folgede Grundprinzipien:

- Gleichungen werden einzeln definiert und abgelegt.
  - Die Definition erfolgt in mathematischer symbolischer Notation.
  - Die Menge der Variablen wird automatisch auf der Grundlage der Unterscheidbarkeit der Variablennamen ermittelt.

- Gleichungssysteme werden als eine Menge von zu verwendenden Gleichungen definiert.
  - Die Definition erfolgt als Menge von Referenzen auf Gleichungen oder andere Gleichungssysteme (d.h. es werden keine Gleichungen neu definiert).
  - Die Menge der Variablen in einem Gleichungssystem ist die Vereinigungsmenge der in den referenzierten Gleichungen enthaltenen Variablen.
  - Die Verkopplung der Gleichungen ergibt sich aus aus der Vergleichbarkeit von Variablennamen, auf die nun näher eingegangen wird.

Auf der Ebene der Dokumentation bestehen Variablennamen häufig aus einer Kombination von Zeichen, deren Bedeutung einer Systematik folgt, die in einer Nomenklatur erklärt wird, z. B. $p_{o,i}^{LV}$ mit p - Druck, o - Referenz, i - i-te Komponente, usw. Diese Vorgehensweise wird im vorgestellten Modellierungskonzept berücksichtigt. Die Elemente von Variablennamen sind in Abb. 2 dargestellt. Da Variablennamen auf der Dokumenationsebene nur dann sinnvoll vergleichbar sind, wenn sie die gleiche Nomenklatur verwenden, wird diese als modulares Modellelement eingeführt, welches die Beschreibung der verwendeten Zeichen enthält und gleichzeitig als notwendiges Kriterium für die Vergleichbarkeit wie folgt herangezogen wird.

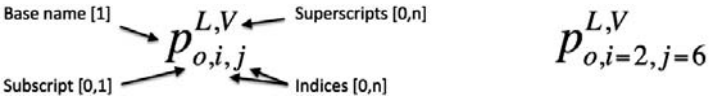


Abbildung 2: Links: Elemente von Variablennamen in MOSAIC, rechts: Instanz eines generischen Variablennamens

- Nomenklaturen werden unabhängig definiert und abgelegt.
- Jede Gleichung und jedes Gleichungssystem muss eine Nomenklatur referenzieren, die alle im System enthaltenen Variablennamen vollständig beschreibt.
- Variablennamen werden genau dann als ‚gleich' betrachtet, wenn sie sich auf die gleiche Nomenklatur beziehen und die gleichen Namenselemente enthalten.
- Um Gleichungssysteme verbinden zu können, die verschiedene Nomenklaturen referenzieren, werden Konnektoren verwendet. Ein Konnektor ist ein weiteres modulares Modellelement, welches die Gleichsetzung von Variablennamen spezifiziert.

Auf der Ebene der Dokumentation ist es üblich Gleichungen allgemein darzustellen, wie z.B. die Komponentenbilanz einer Kolonnentrennstufe:

$$F_j \cdot z_{j,i} + L_{j+1} \cdot x_{j+1,i} + V_{j-1} \cdot y_{j-1} = L_j \cdot x_{j,i} + V_j \cdot y_{j,i} \tag{2}$$

In Gleichung 2 sind i und j generische Indizes, die für Komponenten und Stufen stehen. Die tatsächliche Anzahl von Gleichungen und Variablen, die durch Gl. 2 repräsentiert werden, steht erst fest, nachdem die Anzahl der Stufen und der Komponenten (also die Maximalwerte der Indizes) festgelegt wurden[2]. Diese typische Form der allgemeinen Beschreibung wird im vorgestellten Modellierungskonzept unterstützt. Um die Wiederverwendbarkeit zu erhalten, werden die Maximalwerte der Indizes dabei nicht im Gleichungssystem festgelegt, da jeder verschiedene Satz von Maximalwerten einen eigenen Anwendungsfall für ein gegebenes Gleichungssystem darstellt.

Mit Hilfe der oben genannten Prinzipien lassen sich zusammengehörige, verkoppelte Sätze von Gleichungen definieren. Zu einer vollständigen Problemdefinition fehlen noch die *Klassifikation* der Variablen und die *Wertvorgabe*, d.h. die Einordnung der Variablen in Design und Iterations- bzw. Zustandsvariablen sowie die Festlegung der für das Problem geltenden Werte. Für einen gegebenen Satz von Gleichungen lassen sich damit verschiedene numerische Probleme definieren. Im Sinne der Wiederverwendbarkeit wird dieser Informationsbaustein daher nicht im Gleichungssystem sondern als modulares Modellelement (‚VariableSpecification') gehalten.

Die für die Problemdefinition notwendigen modularen Modellelemente werden im Modellelement 'Evaluation' zusammengeführt, welche folgende Informationen enthält:

- Referenz auf ein Gleichungssystem,
- Spezifikation der Maximalwerte der im referenzierten Gleichungssystem enthaltenen generischen Indizes,
- Referenz auf eine ‚VariableSpecification', welche die Klassifikation und die Werte der Variablen enthält.

Auf Grundlage dieser Problembeschreibung kann der Anwender Programmcode für die von ihm bevorzugte numerische Umgebung erzeugen und dann das Problem lösen.

4 Code-Generierung

4.1 Konzept

Die auf der Ebene der Dokumentation vorliegende Modellinformation (siehe Abschn. 2 und 3) dient als plattformunabhängiges Modell für die Code-Generierung. Der verwendete Ansatz verläuft analog zu dem der modellgetriebenen Architektur [MM03]. Dabei werden plattformspezifische Modelle aus plattformunabhängigen Modellen generiert. Die für die Transformation notwendige Information wird separat in Plattformmodellen gehalten.

Die vorgestellte Modellierungsumgebung unterstützt die folgenden Gleichungssystemtypen: Nichtlinear algebraisch (NLE), gewöhnliche Differentialgleichungssysteme (ODE),

[2] Dabei werden Instanzen der Gleichungen und Variablen erzeugt, vgl. Abb. 2 rechts

Systeme aus gewöhnlichen Differentialgleichungen und nichtlinearen algebraischen Gleichungen (DAE). Die für die Beschreibung solcher Probleme entwickelten Sprachen verarbeiten dabei grundsätzlich die gleichen Modellinformationen. Daher reduziert sich die Code-Erzeugung auf die Ausgabe der für die jeweilige Sprache korrekten Syntax zur Problemdefinition und das Einsetzen der modellspezifischen Informationsbestandteile. Die korrekte Übertragung der algebraischen Ausdrücke beruht dabei auf den Regeln zur Eindeutigkeit der symbolischen Notation (vgl. Abschnitt 2).

4.2 Umsetzung

Die Ablage der Modellelemente erfolgt in XML und MathML. Für die Darstellung und interne Verarbeitung der mathematischen Ausdrücke wird MosaicMathML verwendet, welches das Auswerten des mathematischen Inhalts erleichtert. MosaicMathML ist eine gültige Untermenge von MathML im Präsentations-Markup, was die Anzeige der abgelegten Ausdrücke durch existierende Softwarepakete ermöglicht. Die spezifischen Erweiterungen in MosaicMathML beinhalten lediglich Designregeln innerhalb des gültigen MathML.

Der für dieses Projekt verwendete Code-Generator und die Plattformmodelle wurden in Java implementiert.[3] Zu den Zielplattformen zählen C++/BzzMath [MDBFP09], C/GSL, Matlab, Scilab, Aspen Custom Modeler, GAMS, gPROMS und C++/sDACL [BKWAG11].

Da der zu erzeugende Code das Hauptanliegen des Anwenders ist, beinhaltet der vorgestellte Ansatz zusätzlich die Möglichkeit, Plattformmodelle frei und ausschließlich auf der Grundlage der Kenntnis des Ziel-Codes zu definieren. Dafür wurde ein Benutzerinterface entwickelt, das es erlaubt, die Übersetzung aller Grundelemente zu beeinflussen und diese anschließend nach Bedarf zusammenzusetzen. Die so erstellten Plattformmodelle werden in XML abgelegt und stehen wie jedes andere Modellelement zur geteilten Nutzung, Kopie und Bearbeitung zur Verfügung.

5 Beispiel

Ein im Exzelenz-Cluster UniCat entwickelter Reaktor [JGAG+10] wurde in der vorgestellten Modellierungsumgebung modelliert. Der für die Spezifikation der Gleichungen verwendete Latex-Code ist hier eingebunden:

$$\frac{d\,C_i^t}{d\,l} = r_i \cdot w \cdot \frac{A^t}{v^t} - \frac{D_i}{\delta^m} \cdot (C_i^t - C_i^s) \cdot \frac{p^{circ,t}}{v^t} \tag{3}$$

$$\frac{d\,C_i^s}{d\,l} = \frac{D_i}{\delta^m} \cdot (C_i^t - C_i^s) \cdot \frac{p^{circ,t}}{v^s} \tag{4}$$

[3] Auf die Verwendung von XSLT wurde verzichtet, da mit Java (worin die gesamte Modellierungsumgebung geschrieben ist) eine mächtige und flexible Sprache für die Aufgabe der Transformation zur Verfügung steht.

Es wurde Programm-Code mit den vordefinierten Plattformmodellen für C++/BzzMath, Matlab und Aspen Custom Modeller erstellt. Der Code stellt jeweils eine vollständige Problembeschreibung dar und ist ohne manuelle Eingriffe lauffähig. Der Code in C++/BzzMath wurde direkt auf dem Modelierungsserver ausgeführt. Die Simulationsresultate sind in Abb. 3 dargestellt.

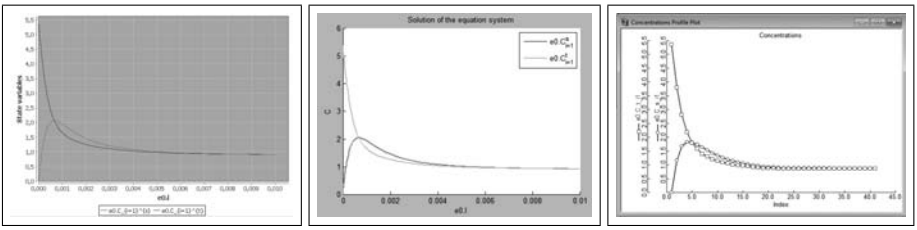


Abbildung 3: Plots der Simulationsergebnisse in MOSAIC, Matlab, und Aspen Custom Modeler.

6 Zusammenfassung

Bei der Implementierung mathematischer Modelle stellen die individuelle fachliche Spezialisierung einerseits und die Unterschiede zwischen den je nach Anwender und Anwendungsfall variierenden numerischen Werkzeugen andererseits Grenzen auf. Die vorliegende Arbeit hatte das Ziel, diese Grenzen zu überwinden. Die entwickelte webbasierte Modellierungssoftware MOSAIC hat ihren Anwenderkreis gefunden und wird derzeit am Institut für Dynamik und Betrieb technischer Anlagen der TU-Berlin weiterentwickelt; derzeit sind über 300 Nutzer registriert.

Diese Arbeit entstand innerhalb des Exzelenz-Clusters UniCat und war gefördert durch die DFG.

Literatur

[BD06] Jens Bausa und Guido Dünnebier. Life cycle modelling in the chemical industries: Is there any reuse of models in automation and control? *Computer Aided Chemical Engineering*, 21:3–8, 2006.

[BKWAG11] Tilman Barz, Stefan Kuntsche, Günter Wozny und Harvey Arellano-Garcia. An efficient sparse approach to sensitivity generation for large-scale dynamic optimization. *Computers & Chemical Engineering*, 35(10):2053–2065, 2011.

[Bon12] Claas Bontus. Firemath. *http://www.firemath.info/*, 2012.

[BSMM07] Ilja Bronshtein, Konstantin Semendyayev, Gerhard Musiol und Heiner Mühlig. *Handbook of Mathematics*. Springer, Berlin and Heidelberg and Germany, 2007.

[FLM98] B. A. Foss, B. Lohmann und Wolfgang Marquardt. A field study of the industrial modeling process. *Journal of Process Control*, 8(5-6):325–338, 1998.

[Gas78] Saul I. Gass. Computer model documentation. *Proceedings of the conference on winter simulation*, 10:281–287, 1978.

[Int09] International organization for standardization. *80000: Quantities and units*. 2009.

[JGAG+10] S. Jašo, H.R Godini, H. Arellano-Garcia, M. Omidkhah und G. Wozny. Analysis of attainable reactor performance for the oxidative methane coupling process. *Chemical Engineering Science*, 65(24):6341–6352, 2010.

[KB06] Michael Kohlhase und Alan Bundy. *OMDoc - an open markup format for mathematical documents: [version 1.2]*. Springer, Berlin [u.a.], 2006.

[KKG92] Robert J. Klerer, Melvin Klerer und Fred Grossman. A language for automated programming of mathematical applications. *Computer Languages*, 19(3):169–184, 1992.

[KM09] Karsten-Ulrich Klatt und Wolfgang Marquardt. Perspectives for process systems engineering–Personal views from academia and industry. *Computers & Chemical Engineering*, 33:536–550, 2009.

[Kun13] Stefan Kuntsche. *Modular model specification on the documentation level : concepts and application in a web-based modeling environment*. Dissertation, Technische Universität Berlin, 2013.

[MDBFP09] Flavio Manenti, Ivan Dones, Guido Buzzi-Ferraris und Heinz A. Preisig. Efficient Numerical Solver for Partially Structured Differential and Algebraic Equation Systems. *Industrial & Engineering Chemistry Research*, 48(22):9979–9984, 2009.

[MM03] Joaquin Miller und Jishnu Mukerji. Technical Guide to Model Driven Architecture: The MDA Guide v1.0.1. *http://www.omg.org/mda/presentations.htm*, 2003.

[Sam72] Jean E. Sammet. Programming languages: history and future. *Communications of the ACM*, 15(7):601–610, 1972.

[SW06] Elena Smirnova und Stephen M. Watt. Notation selection in mathematical computing environments. *In Proc. Transgressive Computing*, Seiten 339–355, 2006.

[WW04] Tilman Walther und Martin Wilke. MathParser: parses MathML to Latex. *http://www.tilman.de/programme/mathparser/*, 2004.

Stefan Kuntsche, geboren am 7. August 1976 in Neustrelitz, erlernte zunächst den Beruf des Fremdsprachenkorrespondenten und übte diesen bis zu seinem Studium aus. Er studierte Informationstechnik im Maschinenwesen mit dem Schwerpunkt Prozess-Systemtechnik an der TU-Berlin. In den Abschlussarbeiten befasste er sich mit der detaillierten Simulation und der Optimierung von verfahrenstechnischen Anlagen. Sein Industriepraktikum absolvierte er bei der Firma ProSim in Toulouse. Anschließend promovierte er am Institut für Dynamik und Betrieb technischer Anlagen an der TU-Berlin. Er interessiert sich für Modellierung, Softwaretechnik und Regelungstechnik. Derzeit arbeitet er als Softwareentwickler bei der PSI AG in Berlin.

Schnelle Algorithmen für zerlegbare Graphen*

Alexander Langer

Lehr- und Forschungsgebiet Theoretische Informatik
RWTH Aachen University
langer@cs.rwth-aachen.de

Abstract: Ein berühmter Satz von Courcelle besagt, dass jedes MSO-definierbare Problem auf Graphen beschränkter Baumweite in Linearzeit gelöst werden kann. Obwohl dies erklärt, warum sich viele Graphprobleme auf solchen Graphen effizient lösen lassen, scheitern in der praktischen Umsetzung von Courcelles Satz die herkömmlichen Verfahren am benötigten Speicherbedarf, der sehr schnell die Grenzen dessen übersteigt, was wir in der Praxis handhaben können. In dieser Arbeit stellen wir einen neuen Ansatz vor, der auf modelltheoretischen Spielen basiert. Mit Hilfe dynamischer Programmierung auf der Baumzerlegung wird in Linearzeit ein Spiel aufgebaut, in welchem sich schnell testen lässt, ob die Formel auf dem Eingabegraphen erfüllt ist. Eine Reihe von Experimenten legt nahe, dass unser Ansatz für manche Problemstellungen als Alternative zur ganzzahligen linearen Optimierung in Betracht kommt.

1 Einführung

Im Jahr 1990 hat Courcelle bewiesen [Cou90], dass jedes Graphproblem, das sich in Monadischer Prädikatenlogik zweiter Stufe (MSO) ausdrücken lässt, auf Graphen mit beschränkter Baumweite in Linearzeit lösbar ist. Dieses Ergebnis wurde wenig später auf viele weitere MSO-definierbare Probleme erweitert, darunter eine große Klasse von Aufzählungs- und Optimierungsproblemen [ALS91, CM93]. Beispiele sind etwa die klassischen Probleme VERTEX COVER, INDEPENDENT SET, DOMINATING SET oder 3-COLORABILITY.

Die Baumweite [Hal76, RS86] eines Graphens ist eine ganzzahlige Zahl, die im Wesentlichen beschreibt, wie ähnlich ein Graph einem Baum ist: Bäume und Wälder haben Baumweite eins, Gitter der Größe $n \times n$ haben Baumweite n; ein vollständiger Graph mit n Knoten hat Baumweite $n - 1$. Ein Graph mit Baumweite k lässt sich rekursiv in kleinere Graphen zerlegen, die ebenfalls nur Baumweite k haben. Eine solche *Baumzerlegung* kann algorithmisch berechnet werden. Probleme lassen sich häufig effizient lösen, wenn wir die in der Baumzerlegung beschriebene Struktur ausnutzen und dynamische Programmierung verwenden. Viele Instanzen aus der täglichen Praxis haben eine kleine Baumweite. Beispielsweise ist die Baumweite der Kontrollflussgraphen von C-Programmen ohne `goto` höchstens sechs [Tho98]; für Java [GMT02] und Ada [BBS04] gibt es ähnliche Schran-

*Englischer Titel der Dissertation: "Fast algorithms for decomposable graphs"

ken. Auch Schienen- oder andere Verkehrsnetze haben in der Regel eine kleine Baumweite. Viele weitere Beispiele finden sich in [Bod93, Bod98, BK08b], auf die wir auch für die formalen Definitionen verweisen.

Den Satz von Courcelle nennen wir auch ein *Meta-Theorem*, weil er nicht nur einige vereinzelte Probleme, sondern direkt eine ganze Klasse von Problemen betrifft. Es gilt jedoch gemeinhin als bekannt, dass der Satz von Courcelle nicht unmittelbar zu einem effizienten Algorithmus führt, der in der Praxis auch tatsächlich nutzbar ist. Beispielsweise schreibt Niedermeier [Nie06] in seinem bekannten Werk über parametrisierte Algorithmen:

> *It must be emphasized, however, that the now described methodology is of purely theoretical interest because the associated running times suffer from huge constant factors and combinatorial explosions with respect to the parameter treewidth.* [...] *After establishing fixed-parameter tractability in this way, as a second step one should then head for a concrete, problem-specific algorithm with improved efficiency* [Nie06, p. 169f].

Ähnliche Aussagen von weiteren Autoren findet man mitunter in [GPW10b, Gro99]. Dies hat unter anderem die folgenden zwei Gründe: Erstens wird im Standardbeweis von Courcelles Satz ein endlicher Baumautomat konstruiert, der die Baumzerlegung genau dann akzeptiert, wenn die Formel auf dem Eingabegraph erfüllt ist. Zwar ist die Konstruktion eines endlichen Baumautomaten aus einer MSO-Formel in der Theorie sehr gut verstanden – in der Praxis jedoch bleibt dies eine herausfordernde Aufgabe wegen des Problems der „Zustandsexplosion“, siehe z.B. [KMS01, Mar06]. Zweitens gibt es unter der Voraussetzung $\mathsf{P} \neq \mathsf{NP}$ riesige untere Schranken [FG04] für die Konstanten in der Laufzeitabschätzung, so dass bessere Laufzeiten im Allgemeinen auch überhaupt nicht möglich sind. Damit ist die Automatenmethode – zumindest im Hinblick auf die Laufzeitanalyse – beweisbar die effizienteste Möglichkeit.

2 Stand der Forschung

In der Praxis konzentriert man sich daher hauptsächlich darauf, der Platzproblematik beizukommen. Dementsprechend betreffen viele der “*implementation secrets*” [KMS01] der bekannten MONA-Software [KM01] den Speicherbedarf, etwa die Darstellung der Automaten durch BDDs oder die Einführung sogenannter *guides*. Möchte man jedoch die Baumautomaten nach dem Satz von Courcelle konstruieren, scheitert dies selbst bei vielen Trivialproblemen an dem dafür benötigten Platzbedarf – und zwar unabhängig davon, ob man die Algorithmen selbst implementiert oder bestehende Software wie MONA oder `Autowrite` [Dur02, Dur05] dafür verwendet, vgl. [Kep05, Sog08, GPW10b, GPW10a, Mar06, CD12, CE12].

Als Abhilfe schlagen Courcelle und Durand [CD10a, CD10b, CD11, CD12] vor, die problematische Potenzmengenkonstruktion zu vermeiden, indem nicht-deterministische anstelle deterministischer Baumautomaten verwendet werden und indem Quantorenwechsel in der Formel verboten werden (existentielles Fragment von MSO). Dem damit verbunde-

nen Verlust an Ausdrucksstärke der Logik soll teilweise entgegengewirkt werden, indem für häufig verwendete Konstrukte (etwa für Zusammenhang oder für Mengen und Partionen) vordefinierte Prädikate zur Verfügung stehen. Desweiteren ist die von ihnen verwendete Software `Autowrite` in der Lage, die Transitionen des Automaten während der Simulation "*on-the-fly*" zu berechnen, so dass keine vollständige Repräsentation des Automaten im Speicher vorgehalten werden muss. Stattdessen werden sie nur implizit durch eine kleine Anzahl „Meta-Regeln" dargestellt und die jeweils möglichen Transitionen in jedem Schritt daraus abgeleitet. Durch eine syntaktische Erweiterung kann `Autowrite` auch Entscheidungsprobleme lösen, bei denen eine Zahl Teil der Eingabe ist. Leider ist bisher keine Implementierung für Baumzerlegungen verfügbar (verwendet wurde stattdessen das Maß der Cliquenweite), durchgeführte Experimente [CD10b, CD10a, CD12, CE12] deuten jedoch darauf hin, dass ihre Vorgehensweise durchaus Potential hat, wenn man nicht auf die volle Ausdrucksstärke von MSO angewiesen ist.

Einen völlig anderen Weg schlagen Gottlob, Pichler und Wei [GPW10b, GPW10a, Pic11] vor, deren ursprüngliche Motivation in der Antwortmengenprogrammierung (*Answer Set Programming*) liegt: Anstatt einen Baumautomaten zu konstruieren, beschreiben sie eine Übersetzung der MSO-Formel in ein Datalog Programm, in dem nur monadische Prädikate verwendet werden (*Monadic Datalog*). Ähnlich wie die ganzzahlige Optimierung ist Datalog schon seit Jahren Gegenstand intensiver Forschung, so dass bereits eine Reihe hochoptimierter, nachweisbar praxistauglicher Software zur Verfügung steht, deren Stärke man so nutzen kann. Experimente, die in verschiedenen Arbeiten zu verschiedenen Problemstellungen durchgeführt wurden [GPW10b, GPW10a, JPW09, JPRW08], zeigen, dass der Umweg über Datalog in der Tat praktikabel sein kann. Leider steht bisher noch keine Implementierung einer *automatischen* Übersetzung der MSO-Formeln in ein Datalog-Programm zur Verfügung, und so wurden die Datalog-Programme für obige Experimente von den Autoren von Hand geschrieben. Daher bleibt auch offen, ob die nach [GPW10b] automatisch generierten Programme ähnlich effizient lösbar sind.

3 Ein spieltheoretischer Ansatz

In dieser Arbeit schlagen wir einen neuen Ansatz vor, der auf einem modelltheoretischen Spiel basiert, das unter dem Namen Model-Checking-Spiel bekannt ist, vgl. [Hin73, Grä07, Grä11]. Das hier vorgestellte Verfahren funktioniert für MSO auf beliebigen Strukturen, wobei nicht nur Entscheidungsprobleme gelöst werden können, sondern wie in [CM93] auch eine große Anzahl Optimierungs- und Aufzählungsprobleme.

Das Model-Checking-Spiel $\mathscr{G} = \mathscr{G}(\varphi, G)$ wird jeweils für eine MSO-Formel φ und einen Graphen $G = (V, E)$ gespielt. Es gibt zwei Spieler, den Verifizierer und den Falsifizierer. Der Verifizierer möchte nachweisen, dass die Formel auf dem Graphen erfüllt ist, der Falsifizierer möchte das Gegenteil beweisen. Das Spielbrett ist ein gerichteter, azyklischer Graph, dessen Positionen (ψ, α) jeweils aus einer MSO-Formel ψ und einer Variablenbelegung α bestehen. Von jeder Position (ψ, α) gibt es eine Menge von Kanten zu Nachfolgepositionen (ψ', α'), jeweils mit einer Unterformel ψ' von ψ samt passender Variablenbelegung α'. Ist beispielsweise $\psi = \forall X \psi'$, dann gibt es für jedes $U \subseteq V$ eine Kante

von (ψ, α) nach $(\psi', \alpha[X/U])$, wobei $\alpha[X/U]$ bedeutet, dass in α nun die Variable X der Formel mit dem Wert U belegt wurde. Ist beispielsweise aber $\psi = \psi_1 \vee \psi_2$, dann gibt es von (ψ, α) zwei ausgehende Kanten zu (ψ_1, α) und (ψ_2, α). Von Positionen mit atomaren Formeln gibt es keine ausgehenden Kanten. Das Spiel beginnt in (φ, ε), wobei ε die leere Variablenbelegung ist. In jeder Position darf genau einer der beiden Spieler ziehen und eine Nachfolgeposition anhand einer ausgehenden Kante auswählen – bei universellen Formeln $(\wedge, \forall)$ der Falsifizierer, bei existentiellen Formeln $(\vee, \exists)$ der Verifizierer. Da der Spielgraph azyklisch und gerichtet ist, endet das Spiel nach einer endlichen Anzahl von Schritten in einer Position (ψ, α), wobei ψ eine atomare Formel ist. Wenn nun die Formel mit der Variablenbelegung α auf G erfüllt ist (kurz $G \models \psi[\alpha]$), gewinnt der Verifizierer, ansonsten der Falsifizierer. Nun kann man durch Induktion über den Formelaufbau zeigen, dass φ auf G genau dann erfüllt ist, wenn der Verifizierer eine Gewinnstrategie für das Spiel $\mathscr{G}(\varphi, G)$ hat. Für weiterführende Informationen verweisen wir auf die Literatur, z.B. [Grä07, Grä11].

Ist das Spiel $\mathscr{G}$ gegeben, kann man effizient testen, welcher der beiden Spieler eine Gewinnstrategie hat, vgl. [Grä07, Grä11]. Jedoch ist die Darstellung von $\mathscr{G}$ bis zu exponentiell größer als die Darstellung von φ und G. Es ist für uns daher nicht sinnvoll, $\mathscr{G} = (G, \varphi)$ zu konstruieren. Zu jedem Model-Checking-Spiel $\mathscr{G}$ gibt es allerdings ein äquivalentes Spiel $\mathscr{G}'$ konstanter Größe. Hier seien $\mathscr{G}$ und $\mathscr{G}'$ *äquivalent*, wenn gilt, dass der Verifizierer genau dann eine Gewinnstrategie für $\mathscr{G}'$ hat, wenn er eine für $\mathscr{G}$ hat. Wenn $G \models \varphi[\varepsilon]$ gilt, ist ein äquivalentes Spiel zu $\mathscr{G}(\varphi, G)$ beispielsweise das Spiel, welches nur die Position $(\mathit{true}, \varepsilon)$ enthält.

Unser Ziel ist es nun, dieses Spiel $\mathscr{G}'$ zu konstruieren. Ignorieren wir für einen Augenblick einmal die Laufzeit und Platzbedarf, kann man dazu prinzipiell wie folgt vorgehen: Wir verwenden dynamische Programmierung auf der Baumzerlegung, die ohne Beschränkung der Allgemeinheit eine gewisse Form hat, die wir „nett“ nennen. Zu jedem Knoten t der Baumzerlegung sei V_t die Menge der Knoten in G, die unterhalb von t in der Baumzerlegung vorkommen. Dann sei $G_t = G[V_t]$ der entsprechende induzierte Untergraph von G, der die gleiche Baumweite wie G hat. Die dynamische Programmierung beginnt in den Blättern t der Baumzerlegung, auf denen die $G[V_t]$ nur konstante Größe haben – damit ist es möglich, das Spiel $\mathscr{G}_t = \mathscr{G}(\varphi, G_t)$ in konstanter Zeit zu konstruieren. Für innere Knoten der Baumzerlegung wird das Spiel $\mathscr{G}_t = \mathscr{G}(\varphi, G_t)$ nun aus den bereits konstruierten Spielen der Kindsknoten von t konstruiert. Ist beispielsweise t ein Knoten der Baumzerlegung, in welchem die Untergraphen G_l und G_r *verschmolzen* werden, geht man grob gesagt so vor, dass für jede Position in $\mathscr{G}_t$ zwei bestimmte Positionen aus $\mathscr{G}_l$ und $\mathscr{G}_r$ auch geeignet miteinander verschmolzen werden. In der Wurzel t der Baumzerlegung gilt $G = G_t$ und somit $\mathscr{G} = \mathscr{G}_t$. Nun testen wir, welcher der beiden Spieler eine Gewinnstrategie hat und erzeugen das gesuchte Spiel $\mathscr{G}'$ mit nur einer einzigen Position – entweder $(\mathit{true}, \varepsilon)$ oder $(\mathit{false}, \varepsilon)$.

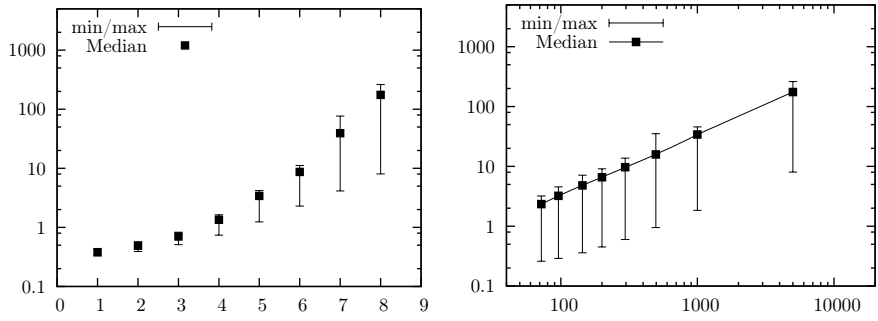
Wenn G ein Graph mit beschränkter Baumweite ist, soll dieses Verfahren aber in Linearzeit möglich und zudem auch noch praxistauglich sein. Dazu führen wir zwei Verbesserungen im Verfahren ein, die in [KLR11] bzw. [LRS11] erstmalig vorgestellt wurden. Die erste Verbesserung besteht im Zusammenfassen von äquivalenten Positionen im Spiel. Die genaue Definition ist sehr technisch, deshalb möchten wir es bei einem ein-

fachen Beispiel zur Veranschaulichung belassen: Wenn $u, v, w \in V$ drei Knoten mit $\{u, v\} \in E$ und $\{u, w\} \in E$ sind, dann kann die Formel $\mathrm{adj}(x, y)$ diese Knotenpaare nicht voneinander unterscheiden. Auch im Spiel sind aus Sicht der Spieler die Positionen $(\mathrm{adj}(x, y), \varepsilon[x/u, y/v])$ und $(\mathrm{adj}(x, y), \varepsilon[x/u, y/w])$ äquivalent, da in beiden Fällen der Verifizierer gewinnen wird. Diese Positionen können also zu einer gemeinsamen Position zusammengefasst werden, ohne den Ausgang des Spiels zu beeinflussen. Fassen wir sukzessive alle äquivalenten Positionen zusammen, erhalten wir irgendwann ein zu $\mathscr{G}$ äquivalentes Spiel $\mathscr{R}$ reduzierter Größe. In einem gewissen Sinne verhält sich das Spiel $\mathscr{R}$ damit zu $\mathscr{G}$ so wie der Bisimulationsquotient eines Transitionssystems zum ursprünglichen Transitionssystem (siehe z.B. [BK08a]). Wir können nun zeigen, dass die Größe von $\mathscr{R}$ ausschließlich von φ und der Baumweite abhängt – sind Baumweite und Formel Konstanten, hat $\mathscr{R}$ also konstante Größe. Bereits allein durch diese erste Verbesserung des Verfahrens ist es möglich, einen spieltheoretischen Beweis für den Satz von Courcelle zu führen [LRS11]. Dieses Verfahren ist allerdings noch nicht praktisch anwendbar: Die Größe von $\mathscr{R}$ entspricht den bekannten unteren Schranken für Courcelles Theorem [FG04] und ist ein exponentieller „Turm“, dessen Höhe von der Formel abhängt.

Die zweite Verbesserung besteht darin, Positionen aus dem Spiel zu entfernen, wenn klar ist, dass diese sowieso niemals besucht werden oder durch eine äquivalente Position ersetzt werden können. Ist beispielsweise die Formel $\exists v \forall u \psi$, und hat der Falsifizierer eine Gewinnstrategie, sobald im Spiel die Position $p' = (\forall u \psi, \alpha')$ erreicht wird, dann wird der optimal spielende Verifizierer ausgehend von der Position $p = (\exists v \forall u \psi, \alpha)$ niemals auf p' ziehen, es sei denn, er verliert bei *allen* Nachfolgepositionen von p. In beiden Fällen können wir nun Positionen aus dem Spiel entfernen: Entweder löschen wir p' ersatzlos, oder wir ersetzen p durch die äquivalente Position $(\mathit{false}, \alpha')$; nicht mehr erreichbare Nachfolgepositionen werden anschließend ebenfalls entfernt.

Im Verlauf der dynamischen Programmierung ist es jedoch mit dem einfachen Model-Checking-Spiel von oben nicht möglich, zu entscheiden, welche Positionen entfernt werden können, da die Spiele (φ, G) und $(\varphi, G[U])$ für einen Teilgraphen $G[U]$ von G nicht notwendigerweise äquivalent sind – man betrachte etwa die Formel $\exists x(x = x)$ und $U = \emptyset$. Daher führen wir eine Erweiterung des klassischen Model-Checking-Spiels ein, in welchem es möglich ist, in der Variablenbelegung α Platzhalter zu lassen, die in der dynamischen Programmierung erst später belegt werden sollen. In diesem Spiel $\mathscr{E}_t = \mathscr{E}(\varphi, G_t)$ gibt es nun auch die zusätzliche Möglichkeit, dass das Spiel *unentschieden* endet, nämlich genau dann, wenn in einer Position (ψ, α) die Formel eine Variable verwendet wird, die in α nicht belegt ist.

Wir können nun zeigen, dass gilt: Wenn für einen Knoten t der Baumzerlegung einer der beiden Spieler eine Gewinnstrategie für $\mathscr{E}_t$ hat, dann hat er auch eine Gewinnstrategie für $\mathscr{G}$. Wir nennen dies daher auch „frühe Determinierung“. Da diese Eigenschaft beweisbar auch für jede Position des Spiels $\mathscr{E}_t$ gilt, sind wir bereits im Knoten t der Baumzerlegung in der Lage, Positionen (ψ, α) von $\mathscr{E}_t$ zu entfernen bzw. durch (false, α) oder (true, α) zu ersetzen und so die Größe des Spiels $\mathscr{E}_t$ erheblich zu verkleinern. In der Wurzel der Baumzerlegung schließlich entfernen wir alle Positionen mit Platzhaltern in der Variablenbelegung. Dadurch werden alle verbleibenden Positionen in $\mathscr{E}_t$ determiniert, so dass wir schließlich das gesuchte Spiel $\mathscr{G}'$ mit der einzigen Position $(\mathit{true}, \varepsilon)$ bzw.



(a) MINIMUM DOMINATING SET, 5000 Knoten (b) MINIMUM DOMINATING SET, Breite 8

Abbildung 1: Laufzeiten (in Sekunden) unserer MSO-Software für Gitter variabler Breite bei fester Anzahl Knoten sowie variabler Anzahl Knoten bei fester Breite.

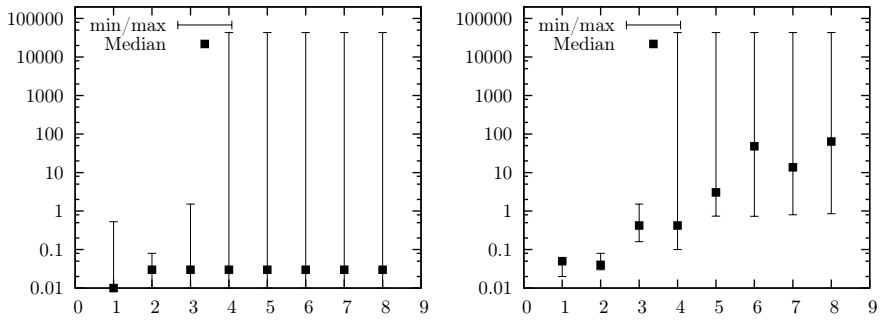
$\mathscr{G}' = (false, \varepsilon)$ erhalten. In diesem können wir den Gewinner des Spiels schnell bestimmen und die Lösung ausgeben.

Wenn wir diese beiden Techniken miteinander kombinieren, wird in jedem Knoten t der Baumzerlegung ein Spiel $\mathscr{E}'_t$ betrachtet, welches nur konstante Größe hat. Die gesamte Laufzeit des Verfahrens ist daher linear in der Größe der Baumzerlegung und auch des Eingabegraphens. Im Allgemeinen sind die Konstanten in der Laufzeitabschätzung wegen der bekannten unteren Schranken wieder riesig – für konkrete Probleme können wir jedoch beweisen, dass die Laufzeit unseres allgemeinen Verfahrens der Laufzeit spezieller Algorithmen für Graphen mit Baumweite w erstaunlich nahe kommt: Für MINIMUM VERTEX COVER ist diese beispielsweise $O(2^w poly(w)n)$ gegenüber den $O(2^w n)$ des speziellen Algorithmus [AP89]. Für 3-COLORABILITY ist sie $O(3^w poly(w)n)$ gegenüber den $O(3^w n)$ aus [AP89]. Beides ist unter einer bestimmten komplexitätstheoretischen Annahme sogar bis auf den kleinen polynomiellen Faktor $poly(w)$ optimal [LMS11]. Für MINIMUM DOMINATING SET hat unser Algorithmus eine Laufzeit von $O(5^w poly(w)n)$. Dies ist nicht optimal, allerdings nutzen sowohl der $O(3^w poly(w)n)$-Algorithmus aus [vRBR09] als auch der $O(4^w n)$-Algorithmus aus [AN02] eine spezielle Monotonie-Eigenschaft von Dominierungsproblemen aus, die für MSO-definierbare Probleme in der Regel nicht gilt.

4 Experimente

Das hier vorgestellte Verfahren wurde von uns in C++ implementiert und unter dem Namen *Sequoia* als Open-Source-Software veröffentlicht[1]. Um seine Praxistauglichkeit zu ermitteln, haben wir sie in einer Reihe von Experimenten mit IBM ILOG CPLEX [CPL] verglichen, einer bekannten kommerziellen Software für die ganzzahlige und lineare Programmierung. Verwendet wurde CPLEX Academic Research Edition 12.4.0.0. Eine kleine Auswahl der Ergebnisse zeigen die Abbildungen 1 und 2 sowie Tabelle 1.

[1] Verfügbar auf `https://github.com/sequoia-mso`



(a) alle Graphen (b) nur dichter besetzte Graphen

Abbildung 2: Laufzeiten (in Sekunden) von CPLEX für MINIMUM DOMINATING SET, alle Instanzen gegenüber den dichter besetzten Teilgraphen (Kantenwahrscheinlichkeit $p \geq 0.9$) von Gittern mit ungefähr 1000 Knoten bei variabler Breite. Bei der Mehrheit der dünnbesetzten Gittern war bereits die LP-Relaxation optimal oder nahezu optimal. Das Problem wird deutlich schwerer auf dichtbesetzten Instanzen: Obwohl CPLEX sogar beim Erreichen von potentiell nicht-optimalen Lösungen bei einem verbleibenden *integrality gap* von 5% stoppen durfte, benötigt CPLEX deutlich mehr Zeit auf diesen Instanzen als unsere MSO-Software.

			Sequoia		*CPLEX*		
Graph	*Größe*	*Bw.*	*Zeit*	*Lösung*	*Zeit*	*Lösung*	*gap*
Hannover, klein	673	8	$3''$	319	*Time-Out*	327	41 %
Hannover, groß	956	9	$9''$	376	*Time-Out*	385	42 %
Berlin	2599	11	$197''$	1259	*Time-Out*	1342	35 %

Tabelle 1: Laufzeiten für das MINIMUM CONNECTED DOMINATING SET-Problem auf drei Graphen, die aus echten Schienennetzen erzeugt wurden. *Größe:* Anzahl der Knoten; *Bw.:* Baumweite der Instanz; *Lösung:* beste gefundene Lösung innerhalb der angegebenen *Zeit*; *gap:* das verbleibende *integrality gap* von CPLEX; das *Time-Out* ist nach 43 200 Sekunden (12 Stunden).

5 Fazit

In dieser Arbeit haben wir einen neuartigen Beweis für den Satz von Courcelle beschrieben, der zu einem allgemeinen und praxistauglichen Verfahren für Graphprobleme auf Instanzen kleiner Baumweite führt. Überraschenderweise kann die von uns entwickelte Software sogar mit bestehenden kommerziellen Lösungen mithalten, wenn es Optimierungsprobleme auf Graphen mit kleiner Baumweite zu lösen gilt. Der immense Fortschritt in anderen Bereichen wie das Lösen von Entscheidungsproblemen Boolscher Formeln oder der Logikprogrammierung stimmt uns zuversichtlich, dass durch weitere Forschung und neue Ideen die Geschwindigkeit von MSO-Model-Checking-Software noch deutlich weiter gesteigert werden kann.

Literatur

[ALS91] S. Arnborg, J. Lagergren und D. Seese. Easy Problems for Tree-Decomposable Graphs. *J. Algorithms*, 12(2):308–340, 1991.

[AN02] J. Alber und R. Niedermeier. Improved tree decomposition based algorithms for domination-like problems. In *Proceedings of the 5th Symposium on Latin American Theoretical Informatics (LATIN)*, number 2286 in Lecture Notes in Computer Science, Seiten 613–627, Cancun, Mexico, 2002. Springer.

[AP89] S. Arnborg und A. Proskurowski. Linear time algorithms for NP-hard problems restricted to partial k-trees. *Discrete Appl. Math.*, 23(1):11–24, 1989.

[BBS04] B. Burgstaller, J. Blieberger und B. Scholz. On the Tree Width of Ada Programs. In *9th Ada-Europe International Conference on Reliable Software Technologies*, number 3063 in Lecture Notes in Computer Science, Seiten 78–90. Springer, 2004.

[BK08a] C. Baier und J.-P. Katoen. *Principles of Model Checking (Representation and Mind Series)*. The MIT Press, 2008.

[BK08b] H.L. Bodlaender und A. M. C. A. Koster. Combinatorial Optimization on Graphs of Bounded Treewidth. *The Computer Journal*, 51(3):255–269, 2008.

[Bod93] H. L. Bodlaender. A tourist guide through treewidth. *Acta Cybernetica*, 11:1–21, 1993.

[Bod98] H. L. Bodlaender. A partial k-arboretum of graphs with bounded treewidth. *Theoretical Computer Science*, 209:1–45, 1998.

[CD10a] B. Courcelle und I. A. Durand. Tractable constructions of finite automata from monadic second-order formulas, 2010. Presented at *Logical Approaches to Barriers in Computing and Complexity*, Greifswald, Germany.

[CD10b] B. Courcelle und I. A. Durand. Verifying monadic second-order graph properties with tree automata. In *3rd European Lisp Symposium*, Seiten 7–21, 2010. Informal proceedings edited by C. Rhodes.

[CD11] B. Courcelle und I. A. Durand. Fly-Automata, Their Properties and Applications. In *Implementation and Application of Automata - 16th International Conference, CIAA 2011*, number 6807 in Lecture Notes in Computer Science, Seiten 264–272. Springer, 2011.

[CD12] B. Courcelle und I. A. Durand. Automata for the verification of monadic second-order graph properties. *J. Applied Logic*, 10(4):368–409, 2012.

[CE12] B. Courcelle und J. Engelfriet. *Graph Structure and Monadic Second-Order Logic: A Language Theoretic Approach*. Number 138 in Encyclopedia of Mathematics and its Applications. Cambridge University Press, Juni 2012.

[CM93] B. Courcelle und M. Mosbah. Monadic second-order evaluations on tree-decomposable graphs. *Theor. Comput. Sci.*, 109(1-2):49–82, 1993.

[Cou90] B. Courcelle. The Monadic Second-Order Theory of Graphs. I. Recognizable Sets of Finite graphs. *Information and Computation*, 85:12–75, 1990.

[CPL] IBM ILOG CPLEX Optimizer, Version 12.4.0.0. `http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/`. Aufgerufen am 2012-09-17.

[Dur02] I. Durand. Autowrite: A Tool for Checking Properties of Term Rewriting Systems. In Sophie Tison, Hrsg., *Proceedings of the 13th International Conference on Rewriting Techniques and Applications (RTA)*, number 2378 in Lecture Notes in Computer Science, Seiten 371–375. Springer, 2002.

[Dur05] I. Durand. A Tool for Term Rewrite Systems and Tree Automata. *Electr. Notes Theor. Comput. Sci.*, 124(2):29–49, 2005.

[FG04] M. Frick und M. Grohe. The complexity of first-order and monadic second-order logic revisited. *Annals of Pure and Applied Logic*, 130(1–3):3–31, 2004.

[GMT02] J. Gustedt, O. A. Mæhle und J. A. Telle. The Treewidth of Java Programs. In D. M. Mount und C. Stein, Hrsg., *ALENEX*, number 2409 in Lecture Notes in Computer Science, Seiten 86–97. Springer, 2002.

[GPW10a] G. Gottlob, R. Pichler und F. Wei. Bounded treewidth as a key to tractability of knowledge representation and reasoning. *Artif. Intell.*, 174(1):105–132, 2010.

[GPW10b] G. Gottlob, R. Pichler und F. Wei. Monadic datalog over finite structures of bounded treewidth. *ACM Trans. Comput. Logic*, 12(1):3:1–3:48, 2010.

[Grä07] E. Grädel. Finite Model Theory and Descriptive Complexity. In *Finite Model Theory and Its Applications*, Seiten 125–230. Springer, 2007.

[Grä11] E. Grädel. Back and Forth Between Logics and Games. In K. R. Apt und E. Grädel, Hrsg., *Lectures in Game Theory for Computer Scientists*, Seiten 99–145. Cambridge University Press, 2011.

[Gro99] M. Grohe. Descriptive and Parameterized Complexity. In J. Flum und M. Rodríguez-Artalejo, Hrsg., *Proceedings of the 13th international Workshop on Computer Science Logic (CSL)*, number 1683 in Lecture Notes in Computer Science, Seiten 14–31. Springer, 1999.

[Hal76] R. Halin. S-functions for graphs. *Journal of Geometry*, 8:171–186, 1976.

[Hin73] J. Hintikka. *Logic, Language-Games and Information: Kantian Themes in the Philosophy of Logic*. Clarendon Press, 1973.

[JPRW08] M. Jakl, R. Pichler, S. Rümmele und S. Woltran. Fast Counting with Bounded Treewidth. In *Proceedings of the 15th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, number 5330 in Lecture Notes in Computer Science, Seiten 436–450. Springer, 2008.

[JPW09] M. Jakl, R. Pichler und S. Woltran. Answer-Set Programming with Bounded Treewidth. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, Seiten 816–822, 2009.

[Kep05] S. Kepser. Using MONA for Querying Linguistic Treebanks. In *Proceedings of HLT/EMNLP 2005*. The Association for Computational Linguistics, 2005.

[KLR11] J. Kneis, A. Langer und P. Rossmanith. Courcelle's Theorem – A Game-Theoretic Approach. *Discrete Optimization*, 8(4):568–594, 2011.

[KM01] N. Klarlund und A. Møller. *MONA Version 1.4 User Manual*. BRICS, Dept. of Comp. Sc., University of Aarhus, January 2001. Verfügbar auf `http://www.brics.dk/mona/`.

[KMS01] N. Klarlund, A. Møller und M. I. Schwartzbach. MONA Implementation Secrets. In *Proc. of CIAA'00*, Seiten 182–194. Springer-Verlag, 2001.

[Lan13] A. Langer. *Fast algorithms for decomposable graphs*. Dissertation, RWTH Aachen University, 2013.

[LMS11] D. Lokshtanov, D. Marx und S. Saurabh. Slightly Superexponential Parameterized Problems. In D. Randall, Hrsg., *Proceedings of the 22nd ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Seiten 760–776. SIAM, 2011.

[LRS11] A. Langer, P. Rossmanith und S. Sikdar. Linear-Time Algorithms for Graphs of Bounded Rankwidth: A Fresh Look Using Game Theory (Extended Abstract). In *TAMC'11*, number 6648 in LNCS, Seiten 505–516. Springer, 2011.

[Mar06] H. Maryns. On the Implementation of Tree Automata: Limitations of the Naive Approach. In *Proc. 5th Int. Treebanks and Linguistic Theories Conference (TLT 2006)*, Seiten 235–246, 2006.

[Nie06] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

[Pic11] R. Pichler. Exploiting Bounded Treewidth with Datalog (A Survey). In *Datalog Reloaded - First International Workshop*, number 6702 in Lecture Notes in Computer Science, Seiten 88–105. Springer, 2011.

[RS86] N. Robertson und P. D. Seymour. Graph minors II. Algorithmic aspects of tree-width. *Journal of Algorithms*, 7:309–322, 1986.

[Sog08] D. Soguet. *Génération automatique d'algorithmes linéaires*. Doctoral dissertation, University Paris-Sud, 2008.

[Tho98] M. Thorup. All Structured Programs have Small Tree-Width and Good Register Allocation. *Inf. Comput.*, 142(2):159–181, 1998.

[vRBR09] J. M. M. van Rooij, H. L. Bodlaender und P. Rossmanith. Dynamic Programming on Tree Decompositions Using Generalised Fast Subset Convolution. In *ESA*, number 5757 in Lecture Notes in Computer Science, Seiten 566–577. Springer, 2009.

Alexander Langer hat von 2001–2008 an der RWTH Aachen Informatik studiert und war von 2008–2012 wissenschaftlicher Mitarbeiter am Lehr- und Forschungsgebiet Theoretische Informatik der RWTH Aachen. Nach seiner Diplomarbeit, in der es bereits um MSO-definierbare Probleme für Graphen mit kleiner Baumweite ging, hat er im Rahmen seiner Promotion, die er 2013 abschloss, an der Entwicklung und Implementierung eines praxistauglichen Verfahrens für solche Probleme geforscht.

Robuste Klassifikationsverfahren für hochdimensionale Datensätze*

Ludwig Maximilian Lausser

Medizinische Systembiologie, Institut für Neuroinformatik, Universität Ulm
ludwig.lausser@uni-ulm.de

Abstract: Die hier dargestellten neuen Methoden zur Klassifikation hochdimensionaler molekularbiologischer Daten werden bezüglich ihrer theoretischen Eigenschaften und ihrer Praxistauglichkeit untersucht. Eine wichtige Eigenschaft in diesem Zusammenhang ist die Interpretierbarkeit der Klassifikationsmodelle und deren Möglichkeit neue Hypothesen datengetrieben zu generieren. Weitere Aspekte sind die oftmals geringe Anzahl von Lernbeispielen (small sample size) und die Anforderung daraus sinnvolle Schlüsse zu ziehen.

1 Einleitung

In ihrer ursprünglichen Bedeutung ist die Tätigkeit der Klassifikation ein allgegenwärtiger Bestandteil des täglichen Lebens und der wissenschaftlichen Disziplinen. Sie ist der Vorgang, einem Objekt, anhand von Sinneseindrücken oder Messungen eine vorher bekannte Kategorie, eine Klasse oder eine Bedeutung zuzuweisen. Die Klassifikation erlaubt es damit, eine Situation genauer einzuschätzen und geeignet auf diese zu reagieren.

Nach welchen Kriterien Objekte einzustufen sind, muss zuvor erlernt werden. Sind bereits von Anderen Entscheidungsregeln angelegt worden, können wir diese nutzen, insofern wir in der Lage sind, sie zu interpretieren oder Schlussfolgerungen aus ihnen zu ziehen. Die Voraussetzungen dafür sind allerdings nicht immer gegeben. Speziell in frühen Jahren stehen uns die notwendigen sprachlichen Fähigkeiten noch nicht zur Verfügung. Das Erlernen von Begriffen geschieht hier anhand von Beispielen. Dies kann z.B. mit Hilfe eines Lehrers geschehen, der die Bedeutung eines Begriffs oder einer Kategorie kennt (überwachtes Lernen). Er präsentiert dem Lernenden mehrere Beispiele und Gegenbeispiele. Der Lernende selbst versucht nun, die Bedeutung des Begriffs von diesen Beispielen zu abstrahieren. Ist kein Lehrer zugegen, kann der Lernende die Beispiele nur nach ihrer Ähnlichkeit oder Distanz gruppieren (unüberwachtes Lernen). Ob die gefundenen Gruppen mit der gesuchten Bedeutung übereinstimmen, muss allerdings im Nachhinein überprüft werden.

Unsere Fähigkeit, Objekte zu klassifizieren, ist stark an unsere Wahrnehmung gekoppelt. Sind Objekte oder deren Datenrepräsentation nicht für die menschlichen Sinne geeignet,

*Englischer Titel der Arbeit: Sparse and Robust Classification methods for High-Dimensional Datasets [Lau13]

fällt es uns schwer, eine korrekte Einteilung zu treffen. In diesem Fall werden die Objekte über technische Messungen beschrieben und das Erlernen einer Entscheidungsregel automatischen Lernverfahren überlassen.

Aus Sicht der Informatik zählt die Klassifikation zu den grundlegenden Aufgabenstellungen des maschinellen Lernens und ist im weiteren Sinne eine Disziplin der Neuroinformatik, der Statistik oder der künstlichen Intelligenz. Ihre relativ gute Charakterisierung macht die Klassifikationsaufgabe zudem zu einem modelhaften Szenario für andere Lernaufgaben, wie etwa dem Bewegungs- und Handlungslernen oder dem Reinforcement Lernen.

Der vermehrte Einsatz von automatischen Klassifikationsverfahren zeigte, dass sich die Verfügbarkeit und Art der Daten in verschiedenen Disziplinen stark unterscheiden können. Vorhandene Algorithmen konnten nicht ohne Weiteres in jedem Szenario angewandt werden. Eine besondere Rolle spielen dabei Daten, deren Dimensionalität n größer ist als die Anzahl der Trainingsbeispiele m ($m < n$). Hier können klassische Techniken, wie etwa das Schätzen von Verteilungen, instabil werden. Verstärkt werden diese Effekte in Fragestellungen, bei denen nur im geringen Umfang Trainingsbeispiele zur Verfügung stehen ($m \ll n$). In diesen Szenarien reichen die Daten meist nicht mehr aus, um herkömmliche Klassifikationssysteme zu adaptieren.

Der Rest dieses Artikels ist wie folgt organisiert. Die nächsten zwei Abschnitte sollen einen Einblick in die analysierten Daten (Abschnitt 2) und die mathematische Formulierung des Klassifikationsproblems geben (Abschnitt 3). Die Themen, die im Rahmen der Arbeit behandelt worden sind, werden in Abschnitten 4, 5 und 6 aufgeführt und in Abschnitt 7 zusammengefasst.

2 Molekularbiologische Daten

Ein Modell für den Fall ($m \ll n$) stellen Fragestellungen aus dem Bereich der molekularen Medizin und Biologie dar. In diesem Feld ist es von Interesse, auf Basis von molekularen Messungen Unterschiede zwischen biologischen Phänotypen zu diagnostizieren oder den Verlauf von Krankheiten zu prognostizieren. Untersucht wird dabei das Verhalten von Zell- oder Gewebetypen. Dieses wird über die Anzahl und Art der Proteine bestimmt, die sich in einer Zelle befinden. Das biologische Programm, das die Produktion der Proteine veranlasst, ist in der DNA der Zellkerne abgelegt. Es wird über Botenmoleküle, so genannten mRNA-Moleküle, an die Produktionsorganellen, die Ribosome, übermittelt. Dieser Informationsfluss ist als das zentrale Dogma der Molekularbiologie bekannt geworden.

$$\text{DNA} \xrightarrow{\text{Transkription}} \text{mRNA} \xrightarrow{\text{Translation}} \text{Protein}$$

Eine Implikation des Dogmas ist es, dass die Konzentration verschiedener Proteine auch indirekt über die Konzentration der mRNA Moleküle gemessen werden kann. Die Analyse solcher molekularbiologischer Daten ist heutzutage geprägt von so genannten Hochdurchsatzverfahren. Sie erlauben es, mehrere tausend Genexpressionslevel simultan zu messen. Die Idee dabei ist es, einen möglichst vollständigen Überblick über alle Aktivitäten in

einer Zelle zu erhalten. Je nach verwendeter Technologie können hier mehrere zehn- bis hunderttausende simultane Messungen aus einer Probe entnommen werden.

Im Gegensatz dazu ist die Anzahl der zur Verfügung stehenden Beispiele oft sehr gering. Die Generierung von Beispielen und die Verifizierung von Phänotypen sind oft mit hohen finanziellen und zeitlichen Kosten verbunden. Zudem können Beispiele für medizinisch relevante Klassen (z.B. Krankheiten) sehr selten sein. Gesammelte Datensätze sind oft das Ergebnis von nationalen oder internationalen Studien. Eine grobe Schlagzahl sind in 100 Beispiele, die für eine Fragestellung zur Verfügung stehen.

3 Klassifikation

Die Idee des überwachten Lernens ist auf verschiedene Art und Weise in automatische Lernverfahren umgesetzt worden. Das häufigste Lernschema ist dabei das des induktiven Lernens. Hier geht man davon aus, dass von Anfang an eine fixe Trainingsmenge $\mathcal{S}_{tr} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m}$ aus m Beispielen $\mathbf{x}_i \in \mathcal{X}$ und den dazugehöhrigen Klassenlabels $y_i \in \mathcal{Y}$ vorliegt. Die Beispiele werden als Vektoren aus Messungen $\mathbf{x} = (x^{(1)}, \ldots, x^{(n)})^T$ aufgefasst. Die Trainingsmenge wird dazu genutzt einen Klassifikator $c : \mathcal{X} \rightarrow \mathcal{Y}$ mit Hilfe eines Lernalgorithmus l an die aktuelle Aufgabenstellung anzupassen

$$l(\mathcal{S}_{tr}, \mathcal{C}) \mapsto c \in \mathcal{C}. \tag{1}$$

Es wird hierbei zuvor festgelegt, aus welcher Funktionen- oder Konzeptklasse $\mathcal{C}$ ein Klassifikator zu wählen ist. Die Konzeptklasse bestimmt strukturelle Eigenschaften eines Klassifikators. Unter anderem trägt sie maßgelblich dazu bei, wie flexibel sich ein Klassifikator an seine Trainingsdaten anpassen kann. Die Flexibilität eines Modells hängt von der Dimensionaliät n und Anzahl m der Trainingsdaten ab. Thomas Cover zeigt zum Beispiel, dass im Falle ($m < n$) linearen Klassifikatoren mit fast sicherer Wahrscheinlichkeit in der Lage sind, beliebig gelabelte Beispiele zu trennen [Cov65].

Generalisierungssleistung. Die wichtigste Eigenschaft eines Klassifikators ist seine Fähigkeit, die Klassenzugehörigkeit von neuen unbekannten Beispielen korrekt vorherzusagen (Generalisierungsleistung). Diese kann stark von seinen Leistungen auf den Trainingsdaten abweichen. Ist ein Klassifikationsmodell zu flexibel, kann es einen Trainingsdatenssatz sehr leicht per Zufall separieren. Die gewonnenen Erkenntnisse sind allerdings nicht auf weitere Beispiele übertragbar (Überanpassung).

Die Generalisierungsleistung wird daher mit Hilfe einer unabhängigen Testmenge $\mathcal{S}_{te}$ geschätzt. Ist nichts über die Wichtigkeit oder die Wahrscheinlichkeit der einzelnen Klassen bekannt, ist die Fehlerrate R_{emp} ein Maß, das häufig für die Abschätzung der Klassifikationsleistung verwendet wird

$$\mathrm{R}_{emp}(c, \mathcal{S}_{te}) = \frac{1}{|\mathcal{S}_{te}|} \sum_{(\mathbf{x},y)\in\mathcal{S}_{te}} \mathbb{I}_{[c(\mathbf{x})\neq y]}. \tag{2}$$

Im Fall $(m \ll n)$ stehen oft nicht genügend Beispiele zur Verfügung, um sowohl eine Trainings- als auch eine Testmenge einer geeigneten Größe bereitzustellen. In diesem Fall, wird die Leistung eines Klassifikators in Re- oder Subsampling Experimenten geschätzt. Ein Beispiel dafür ist die $r \times f$ Kreuzvalidierung. Die Daten werden hierfür in f gleiche Teile aufgeteilt. Davon werden $f-1$ Teile zum Trainieren des Klassifikationsmodells verwendet. Auf dem letzten Teil wird die Leistung des Klassifikators getestet. Der Vorgang wird für alle f Konstellationen wiederholt, so dass eine Fehlerschätzung über den ganzen Datensatz erfolgt. Die Kreuzvaliderung wird auf r Permutationen der Daten wiederholt um möglichen systematischen Fehlern zu entgegenzuwirken.

Interpretierbarkeit. Ein weiteres Gütekriterium kann die Interpretierbarkeit eines Klassifikators sein. Ist es möglich, Informationen aus einem trainierten Klassifikationsmodell abzuleiten, kann der Klassifikator selbst dazu beitragen, Eigenschaften der Daten besser zu charakterisieren oder Hypothesen zu generieren. Im Fall von molekularbiologischen Daten könnten dies Hinweise auf Ursachen einer Krankheit oder Ansatzpunkte für neue Medikamente sein. Ein weiters Beispiel ist die Identifizierung von Datenpunkten, die an der Schwelle zur Entstehung eine Krankheit stehen [SK02]. Ob und wie ein Klassifikator interpretierbar ist, hängt von seiner Struktur ab. Wichtige Schritte sind hierbei die Merkmalsauswahl eines Klassifikators oder die Wahl repräsentativer Prototypen.

4 Konzeptklassen

Eine Möglichkeit komplexere Entscheidungsregeln zu entwickeln ist es, sie aus einfachen Basisklassifikatoren zusammenzusetzen. Man spricht in diesem Zusammenhang von einem Ensemble aus Klassifikatoren $\mathcal{E} = \{c_e\}_{e=1}^{|\mathcal{E}|}$, dessen Vorhersagen mittels einer Fusionsarchitektur oder Ensembleentscheidung h kombiniert werden

$$h\left(c_1(\mathbf{x}), \ldots, c_{|\mathcal{E}|}(\mathbf{x})\right) \mapsto \mathcal{Y} \tag{3}$$

Ein Beispiel sind Abstimmungsverfahren, bei denen jeder Basisklassifikator für eine der zur Auswahl stehenden Klassen stimmt. Diese Art der Entscheidungsregeln umfasst Konzepte wie etwa den einfachen Mehrheitsentscheid oder die Einstimmigkeitsentscheidung (logischen Konjunktion). Im Rahmen der Arbeit wurden Abstimmungsverfahren aus verschiedenen Basisklassifikatoren entwickelt und analysiert [KLLP11, LK14].

Schwellwertklassifikatoren (STC): Der STC stellt einen der am einfachsten zu interpretierenden Klassifikatoren dar. Er trifft seine Entscheidung, indem er eine ausgewählte Messung $x^{(i)}$ mit eine Schwellwert t vergleicht

$$c(\mathbf{x}) = \mathbb{I}_{\left[x^{(i)} > t\right]}.$$

Diese Entscheidung ist unabhängig von allen anderen Messungen. STCs sind nicht in der Lage, komplexere Entscheidungsgrenzen abzubilden. In [KLLP11] wurden sie im Verband von größeren Voting-Ensembles analysiert. Es zeigte sich, dass die Leistung von

Mehrheitsentscheide aus wenigen STCs vergleichbar ist mit denen von komplexere Klassifikatoren. Einheitsentscheide aus STCs erwiesen sich als gut geeignet für Aufgaben mit unbalancierten Klassenverhältnissen.

Die Struktur der so entstandenen Klassifikatoren erlaubte die Entwicklung von Datenkompressionsschranken. Diese können dazu genutzt werden, schon während des Trainings eines Klassifikators Schranken für dessen späteren Fehler zu erhalten. Sie geben zudem eine Leitlinie für den Entwurf von Klassifikatoren vor: Erhalten zwei Klassifikatoren des gleichen Typs im Training den gleichen Fehler, wird derjenige, der auf weniger Beispielen oder Messungen beruht, mit hoher Wahrscheinlichkeit eine geringere Fehlerwahrscheinlichkeit aufweisen. Diese Leitlinie wurde in weitergehenden Arbeiten verfolgt [LMMH14, LMK12].

Fold-Change-Klassifikatoren (FCC): Im Rahmen der Arbeit wurde der FCC als Alternative zum STC vorgestellt [LK14]. Im Gegensatz zum Schwellwertvergleich des STC werden beim FCC zwei Merkmale eines Datenpunktes $x^{(i)}$ und $x^{(j)}$ miteinander verglichen

$$c(\mathbf{x}) = \mathbb{I}_{\left[x^{(i)} > a x^{(j)}\right]}.$$

Als Entscheidungskriterium verwendet der Klassifikator das Verhältnis (Fold-Change) der beiden Messungen. Im biologischen Kontext entspricht dies dem Vergleich zweier Konzentrationen zweier unterschiedlicher Moleküle. Obwohl Fold-Change-Klassifikatoren für den Einsatz in Ensembles gedacht waren, zeigen schon einzelne trainierte Modelle vergleichbare oder bessere Klassifikationsleistungen als wesentlich komplexere Modellklassen. Eine obere Schranke für die Fehlerwahrscheinlichkeit von FCCs und deren Ensembles konnte mit der Hilfe von Datenkompressionsansätzen gesetzt werden.

Die Konzeptklasse der FCC ist invariant gegen die globale Skalierung von Datenpunkten und wird von dieser weder im Training noch bei der Vorhersage beeinflußt. Diese Eigenschaft kann auf Ensembles aus FCCs übertragen werden. Durch diese Eigenschaft ist er robuster gegen störende globale Effekte, wie etwa Labor- oder Standortunterschiede. Effekte dieser Art können zum Beispiel bei klinischen multizentrischen Studien auftreten.

5 Merkmalsselektion

In Falle von hochdimensionalen Daten ist eine initiale Auswahl von geeigneten Messungen ein häufig verwendeter Vorverarbeitungsschritt [GGNZ06]. Diese Merkmalsselektion kann entweder implizit durch die gewählte Konzeptklasse $\mathcal{C}$ oder den gewählten Lernalgorithmus l vorgegeben sein oder explizit durch ein allgemeines Selektionsverfahren f erfolgen

$$f : \mathcal{C} \times \mathcal{S} \longrightarrow \mathbf{i} \in \mathcal{I} = \{\mathbf{i} \in \mathbb{N}^{\hat{n} \leq n} | i_k < i_{k+1}, 1 \leq i_k \leq n\}. \tag{4}$$

Die Merkmalsauswahl wird hierbei durch einen geordneten und wiederholungsfreien In-

dexvektor $\mathbf{i} = (i_1, \ldots, i_{\hat{n}})^T$ dargestellt. Der eigentliche Klassifikator wird später nur auf den Messungen $(x^{(i_1)}, \ldots, x^{(i_{\hat{n}})})^T$ arbeiten. Eine Merkmalsselektion verringert nicht nur die Dimensionalität der Daten, sondern kann gleichzeitig dazu genutzt werden, überflüssige und verrauschte Messungen auszusortieren. Die gewählten Merkmale können Hinweise auf Eigenschaften der einzelnen Klassen geben.

Unabhänging von dem gewählten Methode muss eine Merkmalsselektion immer als Bestandteil des Trainings eines Klassifikators angesehen werden. Wird eine Merkmalsselektion fälschlicher Weise auf allen zur Verfügung stehenden Trainings- und Testdaten ermittelt, wird implizit Wissen über die Testdaten weitergegeben. Dies kann zu einer Überschätzung der Generalisierungsleitung eines Klassifikators führen.

Gleiches gilt für den Einfluss gewählter Merkmale. Werden die Merkmale zuvor auf den gleichen Daten gewählt, kann deren Nutzen zum Beispiel nicht durch eine Kreuzvalidierung auf diesen Daten bestimmt werden. Die Merkmalsselektion muss hier auf jeder Trainingsmenge neu durchgeführt werden. Pro Einzelexperiment kann dies zu unterschiedlichen Auswahlen führen, was die Interpretation der Ergebnisse erschwert [EDZD06].

Stability Score: Im Rahmen der Arbeit wurde einen Maßzahl entwickelt, die es erlaubt, die Möglichkeit einzuschätzen, durch eine gewählte Merkmalselektionsmethode biologisch relevante Ergebnisse zu erhalten [LMK13]. Grundlage dafür ist die Stabilität der gefundenen Merkmalsselektionen. Ist eine Messung für eine Einstufung relevant, sollte sie stabil in allen Einzelexperimenten gefunden werden. Verbessert sie dagegen nur durch Zufall die Klassifikationsleistung, sollte dies nur in wenigen Fällen möglich sein. Für ein vorgegebenes Profil von Messungen bedeutet das, dass die meisten Messungen eher selten oder gar nicht gewählt werden sollten, während einige wenige in allen Experimenten verwendet werden.

Der Stability Score S_{stab} lässt sich für eine gewählte Merkmalsanzahl $\hat{n}$ und ein gewähltes Setting aus R Einzelexperimenten wie folgt berechnen

$$S_{stab} = \frac{1}{R^2 \hat{n}} \sum_{j=1}^{R} j^2 a_{\hat{n}}^{(j)}. \tag{5}$$

Hierbei ist $a_{\hat{n}}^{(j)}$ die Anzahl der Merkmale, die in j Experimenten gewählt worden sind

$$a_{\hat{n}}^{(j)} = \left| \left\{ s_{\hat{n}}^{(i)} = j | i \in 1, \ldots, n \right\} \right| \quad \text{mit} \quad s_{\hat{n}}^{(i)} = |\{i \in \mathbf{I}_r | r = 1, \ldots, R\}| \tag{6}$$

und $\mathbf{I}_r$ die Indexmenge des $r$ten Experiments. Für den Score konnte bewiesen werden, dass er monoton mit einer zunehmenden Stabilität der Merkmale wächst. Der komponentenweise Aufbau des Scores erlaubt zudem eine genauere Inspektion der Stabilität einer Merkmalsselektionsmethode und deren Visualisierung.

Für die finale Einstufung eines Klassifikationsmodels ist die Stabilität der Merkmalsselektion nur eine sekundäre Eigenschaft, die in Relation zur erreichten Klassifikationsleistung gesehen werden muss. Die genaue Gewichtung der beiden Ziele ist oft nicht direkt ersichtlich, sodass sie mit der Hilfe von Algorithmen zur Mehrzieloptimierung eingestellt werden muss [MLMK12].

6 Selektion von Beispielen

Leistung und Interpretierbarkeit eines Klassifikators hängen nicht allein von seiner finalen Entscheidungsgrenze ab. Auch die Vorgänge während des Trainings können Rückschlüsse über die zugrundeliegenden Daten geben. Schwierigkeiten bei der korrekten Einstufung eines Trainingsbeispiels können Hinweise auf eine falsche Zuordnung oder verrauschte Proben sein. Diese Art der Information bezieht sich stärker auf Datenpunkte als auf einzelne Messungen. Sie legt es nahe den Trainingsprozess durch eine Beispiel- oder Labelselektion zu beeinflußen.

6.1 Prototyp-basierte Klassifikatoren

Ein Klassifikationschema, das eine hohe Interpretierbarkeit der Datenpunkte zulässt, ist das der prototyp-basierten Klassifikatoren. Diese Verfahren wählen eine Menge von Prototypen $\mathcal{P} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{|\mathcal{P}|}$ aus, die sie für repräsentativ für den gegebenen Datensatz halten. Werden diese Prototypen direkt aus der Trainingsmenge gewählt $\mathcal{P} \subseteq \mathcal{S}_{tr}$, können diese Datenpunkte über ihren Nutzen charakterisiert werden. Prototyp-basierte Verfahren bestimmen die Klassenzugehörigkeit eines neuen Datenpunktes $\mathbf{v}$, indem sie ihn mit den Prototypen in einer vorher definierten Nachbarschaft $\mathcal{N}(\mathbf{v}, \mathcal{P})$ vergleichen. Der neue Datenpunkt wird der Klasse zugeordnet, die am häufigsten in dieser Nachbarschaft vertreten ist

$$c(\mathbf{v}, \mathcal{P}) = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \left| \{y | (\mathbf{x}, y) \in \mathcal{N}(\mathbf{v}, \mathcal{P})\} \right|. \tag{7}$$

Ein Beispiel aus dieser Gruppe ist der k Nächste Nachbar Algorithmus (k-NN) [FH51]. Dieser Klassifikator benutzt alle gegebenen Trainingsbeispiele als Prototypen $\mathcal{P} = \mathcal{S}_{tr}$. Die Klasse eines weiteren Datenpunktes wird über die k nächstliegenden Prototypen bestimmt. Eine Eigenschaft dieser Methode ist es, dass alle Trainingsdatenpunkte als gleichwichtig angesehen werden. Mögliche Ausreißer, verrauschte oder falsch klassifizierte Beispiele können das Klassifikationsergebnis genauso beeinflussen wie besonders gut geeignete Prototypen. Die Klassifikationsleistung kann dadurch verringert werden. Im Rahmen der Dissertation wurden zwei Modifikationen dieses Algorithmus vorgeschlagen, die versuchen, die Anzahl der Prototypen auf geeignete Weise zu reduzieren [LMMH14, LMK12].

Predictive Hubs k Nearest Neighbor (PH-k-NN): Der erste Algorithmus selektiert seine Prototypen (Hubs) anhand eines neu entwickelten Optimierungskriteriums [LMMH14]. Es setzt die Qualität der einzelnen Hubs mit der Gesamtleistung des Klassifikators in Beziehung. Die Qualität eines Hubs wird dabei über seinen Einfluss auf benachbarte Datenpunkte bestimmt. Befindet er sich in der Nachbarschaft vieler gefährdeter Datenpunkte der gleichen Klasse, ist sein Einfluss förderlich. Ist er umgeben von Datenpunkten der anderen Klasse, ist sein Einfluss schädlich. Die Klassifikationsleistung der Hubs wird über ein neu entwickeltes Gütemaß bestimmt. Bewertet wird der Anteil der korrekt klassifizierten Beispiele aus der Menge der nicht selektierten Beispiele. Je weniger Hubs gewählt wurden,

desto größer kann dieser Anteil werden. Dieses Gütemaß erzeugt einen Selektionsdruck, der zur Ausdünnung des Datensatzes führt.

Die Experimente mit PH-k-NN zeigen eine Auffälligkeit des Klassifikators. Die selektierten Hubs sind keineswegs prototypisch für ihre Klassen. Sie sind in der Nähe der Entscheidungsgrenzen zu finden und nicht in homogenen oder dichten Gebietes ihrer Klasse. Das Ergebnis legt nahe, dass echt prototypische Datenpunkte nur bis zu einem gewissen Grad in der Lage sind, eine Entscheidungsgrenze zu modellieren. Zur genaueren Bestimmung sind auffällige „Anti-Prototypen“ notwendig, die die Eigenschaften der Randregionen genauer charakterisieren.

Representative Prototype Sets (RPS): Dieser Ansatz ist experimenteller als PH-k-NN [LMK12]. In ihm wird jede Klasse durch einen einzelnen Prototypen dargestellt. Dieser kann aus allen gegebenen Trainingsbeispielen seiner Klasse gewählt werden. Im Sinne der oben erwähnten Datenkompressionsschranken stellt der RPS-Klassifikator den einfachsten aller prototyp-basierten Klassifikatoren dar. Er erzeugt einfachere Entscheidungsebenen und ist damit weniger flexibel. Von seinen Möglichkeiten her ist der RPS-Klassifikator eher für den Einsatz in Ensembles gedacht. Es konnte jedoch experimentell gezeigt werden, dass schon Kombinationen aus wenigen Klassifikatoren die Leistung von komplexeren Modellen erreichen können [LMK12].

Der RPS-Klassifikator hat Eigenschaften, die es ihm ermöglichen, Datenpunkte auf eine neue Art und Weise zu interpretieren. Die Anzahl aller möglichen Klassifikatoren ist relativ klein und kann deshalb komplett aufgezählt werden. Ein einzelner Datenpunkt kann nun dadurch beschrieben werden, wie er sich in allen möglichen Klassifikatoren als Prototyp verhält. Dieses Profil kann nun mit denen der anderen Datenpunkten seiner Klasse verglichen werden. Auf diese Weise können Ausreißer, Untergruppen oder besonders gut geeignete Datenpunkte erkannt werden. Es wurde ein Darstellungsschema entwickelt, das es erlaubt, diese Eigenschaften visuell zu erfassen.

6.2 Halbüberwachtes Lernen:

Überwachtes und unüberwachtes Lernen können als Eckpunkte einer ganzen Reihe von Lernschemata angesehen werden. Dazwischen befinden sich Verfahren, die sowohl bekannte (zugeordnete) Trainingsbeispiele $\mathcal{S}_{tr} = \{(\mathbf{x}, y)\}_{i=1}^{m}$ als auch unbekannte (nicht zugeordnete) Trainingsbeispiele $\mathcal{U} = \{\mathbf{x}_i\}_{i=m+1}^{m+m'}$ verwenden [CSZ06].

$$l(\mathcal{S}_{tr}, \mathcal{U}, \mathcal{C}) \mapsto c \in \mathcal{C} \tag{8}$$

Solches halbüberwachtes Lernen stammt aus Anwendungsbereichen, in denen die Datenerhebung sehr leicht zu bewerkstelligen ist, der Labelprozess hingegen sehr zeit- oder kostenintensiv sein kann. In diesem Fall, stehen meist ausreichend Trainingsbeispiele zur Verfügung. Die Klassenzugehörigkeit ist allerdings nur von sehr wenigen Beispielen bekannt ($|\mathcal{S}_{tr}| \ll |\mathcal{U}|$).

Genexpressionsdaten stellen für diese Methoden in zweifacher Hinsicht eine Herausforderung dar, da sowohl die Gesamtanzahl der Beispiele als auch die Anzahl der bekannten

Klassenzuordnungen kleiner ist als in den üblichen Anwendungsgebieten. Nichtsdestotrotz scheinen halbüberwachte Verfahren für gewisse prognostische Fragestellungen geeigneter zu sein. Liegt ein langer Zeitraum zwischen den Messungen und der einer Diagnose (z.B. frühes oder spätes Rezidiv) könnten die entsprechenden Studien früher analysiert werden.

Als Teil der Arbeit wurden ein Resampling-Experiment für halbüberwachte Lernverfahren entwickelt [LSK11] und eine vergleichende Studie auf Genexpressiondaten durchgeführt [LSSH14]. Die Art der Experimente erlaubte die Simulation unterschiedlicher Verhältnisse von gelabelten zu ungelabelten Daten. Unter den in der Studie getesteten Kandidaten konnten drei Algorithmen identifiziert werden, die trotz der geringen Anzahl von Trainingsbeispielen eine hohe Klassifikationsleistung erlaubten.

Von theoretischer Seite ist es überraschend zu sehen, dass die untersuchten Klassifikatoren ihre besten Ergebnisse nicht in den Experimenten mit dem höchsten Labelanteilen erzielt haben. Das zufällige Weglassen von Klassenlabels konnte konsistent zu einer Verbesserung der Klassifikationsleistung führen. Ein nächster Schritt wäre hier die Beantwortung der Frage, ob dieser Effekt durch ein gezieltes Weglassen von Klassenlabels rekonstruiert oder verbessert werden kann.

7 Zusammenfassung

Eine hohe Dimensionalität gepaart mit einer niedrigen Kardinalität stellen eine Herausforderung für herkömmliche Klassifikationsmodelle dar. Während die hohe Dimensionaliät den Suchraum für geeignete Entscheidungsgrenzen vergrößert, verringert eine niedrige Kardinaliät die Möglichkeit, sich an vorhandenen Beispielen zu orientieren oder zu evaluieren. In der molekuarbiologischen Diagnostik ist jedoch eine gute Interpretierbarkeit der Klassifikatoren von höchster Wichtigkeit. Die hier vorgestellten Klassikationsmodelle zeichnen sich durch eine große Einfachheit und Interpretierbarkeit aus. Sie können von medizinischen und biologischen Experten auf ihre Plausibilität geprüft oder zur Hypothesengenerierung herangezogen werden. Ihre strukturellen Eigenschaften erlauben eine theoretische Abschätzung ihrer Generalisierungsleistung mittels Datenkompressionsschranken. Unterstützt werden diese eher theoretischen Ergebnisse durch deren sehr gute Klassifikationsleistung in den durchgeführten empirischen Studien.

Literatur

[Cov65] T. M. Cover. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. *IEEE Transactions on Electronic Computers*, 14(3):326–334, 1965.

[CSZ06] O. Chapelle, B. Schölkopf, A. Zien, Hrsg. *Semi-Supervised Learning*. MIT Press, 2006.

[EDZD06] L. Ein-Dor, O. Zuk, E. Domany. Thousands of samples are needed to generate a robust gene list for predicting outcome in cancer. *PNAS*, 103(15):5923–5928, 2006.

[FH51] E. Fix, J. L. Hodges, Jr. Discriminatory Analysis: Nonparametric Discrimination: Consistency Properties. Bericht Project 21-49-004, Report Number 4, USAF School of Aviation Medicine, Randolf Field, Texas, 1951.

[GGNZ06] I. Guyon, S. Gunn, M. Nikravesh, L.A. Zadeh. *Feature Extraction: Foundations and Applications (Studies in Fuzziness and Soft Computing)*. Springer, 2006.

[KLLP11] H.A. Kestler, L. Lausser, W. Lindner, G. Palm. On the fusion of threshold classifiers for categorization and dimensionality reduction. *Computational Statistics*, 26(2):321–340, 2011.

[Lau13] L. Lausser. *Sparse and Robust Classification Methods for High-Dimensional Datasets*. Dissertation, Universität Ulm, 2013.

[LK14] L. Lausser, H.A. Kestler. Fold change classifiers for the analysis for the analysis of gene expression profiles. In W. Gaul, A. Geyer-Schulz, Y. Baba, A. Okada, Hrsg., *proceedings volume of the German/Japanese Workshops in 2010 (Karlsruhe) and 2012 (Kyoto), Studies in Classification, Data Analysis, and Knowledge Organization*, Seiten 193–202, 2014.

[LMK12] L. Lausser, C. Müssel, H.A. Kestler. Representative Prototype Sets for Data Characterization and Classification. In N. Mana, F. Schwenker, E. Trentin, Hrsg., *Artificial Neural Networks in Pattern Recognition (ANNPR12), volume LNAI 7477*, Seiten 36–47. Springer, 2012.

[LMK13] L. Lausser, C. Müssel, H.A. Kestler. Measuring and visualizing the stability of biomarker selection techniques. *Computational Statistics*, 28(1), 2013.

[LMMH14] L. Lausser, C. Müssel, A. Melkozerov, Kestler H.A. Identifying predictive hubs to condense the training set of k-nearest neighbour classifiers. *Computational Statistics*, 29:81–95, 2014.

[LSK11] L. Lausser, F. Schmid, H.A. Kestler. On the Utility of Partially Labeled Data for Classification of Microarray Data. In F. Schwenker, E. Trentin, Hrsg., *Partially Supervised Learning, volume LNCS 7081*, Seiten 96–109, 2011.

[LSSH14] L. Lausser, F. Schmid, M. Schmid, Kestler H.A. Unlabeling data can improve classification accuracy. *Pattern Recognition Letters*, 37:15–23, 2014.

[MLMK12] C. Müssel, L. Lausser, M. Maucher, H.A. Kestler. Multi-Objective Parameter Selection for Classifiers. *Journal of Statistical Software*, 46(5):1–27, 2012.

[SK02] F. Schwenker, H. A. Kestler. Analysis of Support Vectors Helps to Identify Borderline Patients in Classification Studies. In *Computers in Cardiology*, Seiten 305–308, Piscataway, NJ, USA, 2002. IEEE Press.

Ludwig Maximilian Lausser

- Studium der Informatik an der Universität Ulm (Dipl.-Inf. 2007, Note 1,0).
- Seit 2007 Forschung im Bereich Klassifikation, Mustererkennung und Merkmalsselektion molekularbiologischer Daten (AG Kestler, Institut für Neuroinformatik und Medizinische Fakultät, Universität Ulm) .
- 2013 Promotion in Informatik (Dr.rer.nat., Summa cum laude, Universität Ulm).

Algorithmen und Werkzeuge zur Analyse von Hochdurchsatz-DNA-Sequenzierungsdaten*

Marcel Martin

Science for Life Laboratory (SciLifeLab), Stockholm
marcel.martin@scilifelab.se

Abstract: Hochdurchsatz-DNA-Sequenzierung hat in den letzten Jahren die biologische und medizinische Forschung revolutioniert. Gestiegene Datenmengen und völlig neue Arten der Sequenzierung haben auch zu neuen Herausforderungen in der Informatik geführt. Wir beschreiben hier Algorithmen und Methoden aus vier verschiedenen Gebieten der Hochdurchsatzsequenzierung. Es geht hierbei um das Entfernen von Verunreinigungen durch Adaptersequenzen, eine effiziente Methode um Methylierungsmuster in DNA zu entdecken, um eine neuartige Methode zum interaktiven Aufspüren von krankheitsverursachenden Mutationen und schließlich um eine Reduktion von Sequenzierfehlern durch einen neuen Alignment-Algorithmus.

1 Einführung

Eine auch heute noch gebräuchliche Methode zur DNA-Sequenzierung wurde bereits 1977 vorgestellt [SNC77] und über die Jahre so verbessert, dass 2001 die Sequenz des menschlichen Genoms veröffentlicht werden konnte [LLB+01]. Die *Sanger-Seqenzierung* liefert bis zu ca. 1000 Basen pro DNA-Fragment, allerdings zu vergleichsweise hohen Kosten: Das Humangenom mit seinen drei Milliarden Basen konnte nur durch eine internationale Anstrengung, die Milliarden von Dollarn kostete, sequenziert werden. Sequenzierungsgeräte der zweiten Generation, die *Hochdurchsatzsequenzierungsmethoden* nutzen, kamen wenige Jahre später auf den Markt [MEA+05]. In ihnen laufen die Reaktionen hochparallel und miniaturisiert ab, sodass die Kosten pro Base drastisch reduziert werden. Sie veränderten und verändern die biomedizinische Forschung daher grundlegend. Die rasante Entwicklung hält immer noch an und das 1000-Dollar-Genom ist in Reichweite.

Damit einhergehend kamen aber auch völlig neue informatische Probleme. Zum einen macht das schiere Datenvolumen immer effizientere Algorithmen erforderlich: Ein typischer Lauf des im Jahr 2013 aktuellsten Sequenziergeräts der Firma Illumina produziert 600 GB pro Woche. Sequenzierzentren, in denen dutzende von Geräten parallel betrieben werden, müssen proportional mehr bewältigen. Zum anderen erfordert die spezielle Art der Daten die Entwicklung neuer Verfahren, denn die neuen Geräte liefern zwar Milliar-

*Englischer Titel der Dissertation: "Algorithms and Tools for the Analysis of High-Throughput DNA Sequencing Data"

den von Sequenzen gleichzeitig, aber jeder dieser sogenannten *Reads* ist sehr kurz, typischerweise nur 100 Basen, und mit einer im Vergleich zur Sanger-Sequenzierung hohen Fehlerrate von – je nach Technologie – bis zu 1% behaftet.

Die hier zusammengefasste Dissertation [Mar13] liefert einen entscheidenden Beitrag dabei, vier der durch die neuen Technologien auftretenden Probleme zu lösen. Zuerst widmen wir uns dem Problem, eine durch den Sequenzierungsprozess bedingte häufig vorkommende Verunreinigung der Reads durch sogenannte *Adaptoren* zu entfernen. Erst hierdurch werden die Daten in nachfolgenden Schritten nutzbar. Als zweites geht es um das *Read Mapping*, d. h. darum, die Reads durch fehlertolerante Textsuchalgorithmen in einem Referenzgenom wiederzufinden, wobei wir uns jedoch auf einen speziellen und biologisch wichtigen Anwendungsfall konzentrieren, in dem die DNA vor der Sequenzierung einer chemischen Veränderung unterzogen wurde, um zusätzliche Informationen über ihren Aufbau zu erhalten. Die chemischen Veränderungen führen zu bestimmten Basenersetzungen, die das Read Mapping verkomplizieren, aber mit einer neuen von uns vorgeschlagenen Datenstruktur berücksichtigt werden können. Im dritten Abschnitt zeigen wir, wie ein System aussehen kann, mit dem die Forschung in der Humangenetik stark vereinfacht werden kann, wenn es unter anderem darum geht, die Ursachen für seltene Krankheiten zu finden oder auch die Mutationen zu identifizieren, die in einem menschlichen Tumor auftreten. Zum Schluss geht es wieder zurück zu den Grundlagen des Sequenzierens und wir beschreiben ein völlig neues Verfahren, um Fehler in Sequenzierungsdaten zu beheben, die von Geräten stammen, welche nach dem Prinzip der *Flowgramm-Sequenzierung* arbeiten.

Molekularbiologische Grundlagen Das Erbgut in jeder biologischen Zelle ist in Form von DNA gespeichert, die als Doppelstrang vorliegt. Die beiden Stränge sind jeweils Kettenmoleküle, welche aus den vier Bausteinen Adenin, Cytosin, Guanin und Thymin (A, C, G, T) aufgebaut sind, wobei sich immer A mit T und C mit G des jeweils anderen Strangs verbinden (Basenpaare). Die Informationen liegen somit redundant vor, da sich die Basenabfolge des einen Strang aus dem anderen ableiten lässt. Einzelne Abschnitte der DNA (Gene) werden von der Zelle ausgelesen und in Proteine übersetzt, die alle möglichen Funktionen im Organismus übernehmen.

2 Entfernen von Adaptersequenzen

Zur Sequenzierung wird die zunächst einsträngig vorliegende DNA im Laufe des Sequenzierungsprozess schrittweise zu einem Doppelstrang ergänzt (siehe Abb. 1). Die dabei jeweils eingebauten Basen werden detektiert und resultieren im Read, der somit eine Zeichenkette über dem Alphabet $\Sigma = \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$ ist. Dieser Prozess ist mit Fehlern behaftet und typische Fehlerraten von 1% oder mehr müssen in den Algorithmen berücksichtigt werden. Beim Sequenzieren von derart kurzen Molekülen, dass sie die fest vorgegebene Readlänge nicht ausnutzen, erscheinen die Sequenzen der für den Sequenzierprozess notwendigen Adaptersequenzen als Teil des Reads und müssen entfernt werden. Solche

Abbildung 1: Vor und nach dem Sequenzierungsprozess. Die schwarzen Buchstaben gehören zum Adapter, die konturierten gehören zum uns interessierenden DNA-Fragment. Oben: Das zum Sequenzieren vorbereitete DNA-Molekül wurde mit Hilfe des Adapters AGGTT innerhalb des Geräts auf einer Oberfläche befestigt. Unten: Nach Abschluss des Sequenzierprozesses ist der DNA-Doppelstrang vollständig aufgebaut. Der Read lautet AAT...CAAGGA und schließt somit die Adaptersequenz mit ein.

kurzen Moleküle können z. B. microRNA-Moleküle sein, welche in der Zelle genutzt werden um Genexpression zu regulieren.

Das sich ergebende und von uns definierte *Adapter Trimming Problem* ist eine Variante des bereits von Gusfield [Gus97] beschriebenen Problems, fehlertolerant Überlappungen zweier Zeichenketten zu finden und kann prinzipiell mit semiglobalen Alignment gelöst werden. Herausforderungen bestehen in den nötigen Anpassungen: Zum einen gibt es nicht nur den beschriebenen Typ von Adaptoren, sondern mindestens drei weitere. Zum anderen gilt es eine Höchstfehlerrate als Parameter zuzulassen.

2.1 Höchstfehlerrate

Ein semiglobales Alignment zwischen dem Read `FRAGMENTADOPT` und der Adaptersequenz `ADAPTER` kann so veranschaulicht werden:

```
      ADAPTER
      ||X||
FRAGMENTADOPT
```

Da eine Substitution vorliegt (Insertionen und Deletionen sind ebenso möglich) und fünf Basen des Adapters betroffen sind, ist die Fehlerrate $r = \frac{1}{5}$. Problematisch ist nun, dass eine Minimierung der Fehlerrate immer zu Alignments mit null Fehlern führen würde, in denen sich die Sequenzen aber nicht überlappen. Bisher wurde in semiglobalen Alignments daher die Summe der Punkte, die für Substitutionen, Übereinstimmungen, Insertionen und Deletionen vergeben wurde, maximiert. Problematisch hierbei ist, dass diese

Punkte ohne tiefere Bedeutung sind und die Ergebnisse daher nicht intuitiv verständlich sind. Wir verändern daher das Optimierungskriterium folgendermaßen: Betrachte zu allen erlaubten Überlappungen alle optimale Alignments und nimm dann unter denjenigen, deren Fehlerrate unter der vorgegebenen Höchstfehlerrate liegt, dasjenige mit der höchsten Anzahl Übereinstimmungen. Der zugehörige Algorithmus kann mit einigen Verbesserungen Adaptoren der Länge m in Reads der Länge n in Zeit $O(n\varepsilon m)$ finden, wobei ε die Höchstfehlerrate ist. Dieser Ansatz, so können wir zeigen, funktioniert in der Praxis ausgesprochen gut und wurde in der Form des sehr erfolgreichen Programms *cutadapt*[1] der Bioinformatik-Gemeinschaft zur Verfügung gestellt.

Weitere in der Dissertation betrachtete Aspekte sind u. a. theoretische Überlegungen zu Sensitivität und Spezifizität und eine Erweiterung des Algorithmus auf sogenannte *Colorspace*-Reads, in denen die Zeichen eines Reads als "Farben" $0, \ldots, 3$ kodiert sind, die aber jeweils von zwei aufeinanderfolgenden Basen im DNA-Molekül abhängen.

3 Bisulfit-Read-Mapping

Manche Basen der DNA sind chemisch so modifiziert, dass an ihnen eine CH_3-Gruppe heftet; sie sind *methyliert*. In Wirbeltier-DNA, also auch menschlicher, können nur Cytosine vor Guaninen (auch CpG genannt) methyliert sein. Die Methylierung hat für die Zelle eine wichtige Funktion, denn es gibt Bereiche in der Nähe von Genen, in denen CpGs besonders häufig auftreten, genannt CpG-Inseln, die als Schalter fungieren: Ist eine Insel methyliert, so wird das zugehörige Gen inaktiviert. Auf diese Weise können die Zellen eines mehrzelligen Organismus, die ja alle die gleiche DNA enthalten, dennoch ganz unterschiedlich ausgeprägt sein. Zum Beispiel können so in Leberzellen andere Gene als in Nervenzellen aktiv sein.

Um Methylierungsmuster herauszufinden, wird die DNA vor dem Sequenzieren chemisch mit (Natrium-)Bisulfit behandelt. Dies führt zu einer vom Methylierungszustand abhängigen Veränderung im Read: Unmethylierte C werden zu T umgewandelt und methylierte C bleiben unverändert. Durch den Vergleich zur Referenz kann dann festgestellt werden, welche Basen methyliert waren.

Wenn ein Referenzgenom bekannt ist, werden Reads (evtl. nach Entfernen der Adaptoren) im allerersten Verarbeitungsschritt fehlertolerant ihrer Position in der Referenz zugeordnet. Dies wird auch *Read Mapping* genannt und es existieren verschiedene Algorithmen, die das Problem lösen [LD09]. Als wir DNA-Methylierung in menschlichen X-Chromosomen untersuchten [ZMB+09], gab es keine zuverlässigen Algorithmen, um bisulfitbehandelte Reads aus Hochdurchsatzexperimenten zu mappen. Wir haben daher eine neue Datenstruktur entwickelt, die beim Mappen alle derartigen Bisulfit-Ersatzungsregeln berücksichtigen kann. Zum Mappen nutzt unser Algorithmus den *Seed-and-extend*-Ansatz: Zuerst werden kurze, exakte Übereinstimmungen zwischen Referenz und Read gefunden (*seeds*), die dann in einem zweiten Schritt fehlertolerant auf den vollen Read erweitert werden (*extend*). Beide Phasen werden von uns an die Bisulfit-Regeln angepasst.

[1] https://github.com/marcelm/cutadapt/

Um kurze Übereinstimmungen zu finden, erweitern wir den *q-Gramm-Index*. Ein solcher Index zu einem Text s ist eine Funktion, die alle Zeichenketten der Länge q auf die Liste ihrer Vorkommen in s abbildet. Implementiert wird der Index als 4^q Arrays, was speicheraufwändig ist, aber $O(1)$ Zugriff auf jedes Vorkommen garantiert. Unsere *Bisulfit-q-Gramm-Index* ist eine Erweiterung, die die Frage beantwortet: Gegeben einen String der Länge q, wo kommt dieser in der Referenz s vor unter der Annahme, dass Bisulfitveränderungen durchgeführt worden? Wird beispielsweise `TTG` (mit $q = 3$) gesucht, so werden auch Positionen, die `TCG` lauten, zurückgeliefert. Aber auch: Wird nach `CAG` gesucht, so werden nur Positionen zurückgeliefert, an denen `TAG` steht, denn ein C, welches nicht vor G steht, ist nie methyliert und wird daher immer in T umgewandelt. Erschwerend kommt hinzu, dass die DNA nach der Bisulfitbehandlung kopiert wird und dadurch aufgrund der Doppelsträngigkeit der DNA letztendlich auch Ersetzungen der Form G→A auftreten können, wenn das G nach einem C steht. Auch diese Form von Ersetzungen sowohl den Fall, dass die Bisulfitbehandlung unvollständig war und somit teilweise gar keine Ersetzungen vorkommen, wird von dem neuen Index behandelt, indem die Bisulfitbehandlung bei der Indexerstellung simuliert wird. Auch für die Erweiterung der kurzen Treffer müssen Anpassungen vorgenommen werden und wir zeigen, dass dies besonders effizient mit einem bitparallelen Algorithmus möglich ist, wenn die DNA als Binärstring dargestellt ist, in dem je zwei Bits eine Base kodieren.

In der Dissertation wird weiterhin untersucht, wie viele unterschiedliche Strings der Länge n es gibt, die aus Bisulfitbehandlung herrühren. Es lässt sich zeigen, dass die Anzahl dieser Bisulfitstrings näherungsweise $1.19 \cdot 3.3^n$ ist. Eine weiterer Aspekt ist die Frage, ob sich der Index komprimieren lässt, da insbesondere auch die Bisulfitvariante den Speicher erhöht. Wir können zeigen, dass eine Platzreduktion von 25% ohne Verlust von Laufzeiteffizienz möglich ist.

4 Exomsequenzierung

Das *Exom* ist der Teil des Genoms, der aus allen *exprimierten Sequenzen* besteht, also vereinfacht gesagt aus allen Genen. Das Exom macht mit ca. 50 Mio. Basen nur ca. 2% des menschlichen Genoms aus, ist aber der Teil, in dem sich Mutationen am stärksten auswirken. Ein großer Teil der Erbkrankheiten und Tumorarten entstehen durch Veränderungen im Exom. Exomsequenzierung besteht darin, vor dem Sequenzieren das Exom aus der vorliegenden Gewebeprobe zu extrahieren. Da es durch den so reduzierten Sequenzieraufwand möglich ist, zu gleichen Kosten mehr Exome und somit Patienten zu untersuchen, hat dieses Vorgehen in letzter Zeit an besonderer Bedeutung gewonnen [NTR+09].

Wir stellen hier eine neue Methode zur effizienten Analyse von Exomen vor, die in der Software *Exomate*[2] implementiert wurde. Das Ziel solch einer Analyse ist es üblicherweise, die kranhkeitsverursachenden Mutationen zu finden. Im einfachen Fall ist nur ein Gen betroffen, bei Tumoren sind es mehrere, deren Zusammenspiel zur Erkrankung führt. Die Schwierigkeit besteht darin, dass es immer tausende bis zehntausende Varianten zwischen

[2] https://bitbucket.org/marcelm/exomate

der untersuchten Probe und dem Referenzgenom gibt, die aber nicht schädlich sind, sondern durch die natürlichen, normalen Unterschiede zwischen Individuuen begründet sind (Haarfarbe etc.). Es ist daher elementar, geschickte Filterungsstrategien einzusetzen. Diese Strategien müssen jedoch für jede Studie angepasst werden, wobei es von Vorteil ist, wenn die Mediziner oder Genetiker, die die Studie durchführen, dies direkt und interaktiv im Webbrowser tun können. Exomate besteht daher aus drei Teilen: Einem Pipeline-Modul, einer Datenbank und einer Weboberfläche.

Pipeline. Das Pipeline-Modul von Exomate übernimmt alle nötigen Berechnungen, um von den Rohdaten, also den Reads, zu zuverlässigen Mengen von Varianten zu gelangen. Hier sind mehrere Zwischenschritte erforderlich, u. a. Read Mapping, Entfernen von doppelten Sequenzen (PCR-Duplikate) und Rekalibrierung der Qualitätswerte, die mit jeder Base verknüpft sind. Die Pipeline geht hierbei nach den *Best practices* des GATK-Framework [DBP+11] vor und ruft Standardprogramme auf. Die Schritte sind jedoch vollständig automatisiert, sodass alle gefundenen Varianten einer Probe am Ende automatisch in die Datenbank eingetragen werden. Hier findet nur sehr schwache Filterung statt, um keine Sensitivität zu verlieren. Jede Variante ist mit mehreren Qualitätswerten versehen, die dabei helfen zu beurteilen, ob die Variante reell ist oder evtl. von systematischen Sequenzierfehlern verursacht wurde.

Datenbank. Die Postgres-Datenbank enthält zum einen Metadaten über Familien, Patienten und sequenzierte Proben. Erfasst werden z. B. Datum des Sequenzierlaufs, Name des Sequenziergeräts, Statistiken zur Qualität, welche Krankheit mit einer Probe verknüpft ist und das Geschlecht des Patienten. Patienten werden anhand von anonymen Bezeichnern identifiziert, wobei die Zuordnung zwischen Patient und Bezeichner nicht im System gespeichert ist.

Den weitaus größeren Teil der Datenbank machen aber die Varianten selbst aus, die aus zwei Quellen kommen: Sequenzierte Proben und öffenliche Datenbanken, wie z. B. dbSNP. Die sequenzierten Proben stammen auch von gesunden Kontrollpatienten und dienen zusammen mit den dbSNP-Varianten der Filterung. Alle Varianten haben eine eindeutige Identifikationsnummer y und in der Liste aller Varianten wird nur gespeichert, dass Patient x Variante y hat. Dies ist eine Form der Normalisierung, die hier eine besonders schnelle Filterung ermöglicht, da der Datenbank-Server im Grunde nur Mengenoperationen auf Mengen von Identifikationsnummern durchführen muss. Die Varianten selbst sind in einer separaten Tabelle gespeichert und beschreiben jeweils Chromosom, Koordinate auf dem Chromosom und die genaue Veränderung, also z. B. „Chr. 6, 157 406 006, C→T“.

Weboberfläche. Die Weboberfläche dient im Gegensatz zu anderen Systemen nicht nur der Anzeige von Tabellen, sondern enthält einen entscheidenden Teil der Intelligenz des Systems, die wie erwähnt hauptsächlich im geschickten Filtern der Varianten besteht. Über die Python-Bibliothek SQLalchemy[3] ist es möglich, auf vergleichsweise einfache und lesbare Art aufwändige SQL-Abfragen automatisch erstellen zu lassen, die darüberhinaus

[3] http://www.sqlalchemy.org/

	Gene	Sample	Transcripts	Nucleotide	Codon	Amino acid	Type	Region	Quality	Location
1	ARID1A	M024	001, 201, 002, more	C → T	CGA→ TGA	R → *	Nonsense	Coding	75.19	1:27106354
2	ARID1B	M026	201, 203, 009, more	C → T	CGA→ TGA	R → *	Nonsense	Coding	37.98	6:157406006
3	SMARCB1	M021	005, 001, 002, 003	G → A	CGG→ CAG	R → Q	Missense	Coding	224.79	22:24176330

Abbildung 2: Diese von Exomate gelieferte Ergebnistabelle zeigt alle Mutationen in von uns untersuchten Coffin-Siris-Patienten (eine seltene Erbkrankheit), die nach dem Filtern von nicht relevanten Mutationen übrig bleiben. Für jede Mutation wird u. a. der Genname und Chromosomenposition angegeben, welche Nukleotidveränderung im Detail aufgetreten ist und welche Aminosäureveranderung herbeigeführt wurde.

noch dynamisch sind, das bedeutet zum Beispiel, dass in einer Version der Abfrage ein JOIN durchgeführt wird und in der anderen nicht. Auf diese Art werden Filter, die von uns mit einem LEFT OUTER JOIN realisiert wurden, an- und abgeschaltet. Neben der wichtigsten Einstellung, welche Proben untersucht werden sollen, beeinflussen weitere Parameter die Suche: Schwellwerte für verschiedenste Qualitätswerte, welche dbSNP-Varianten gefiltert werden sollen, ob auch synonyme Varianten angezeigt werden sollen (diese verändern normalerweise das entstehende Protein nicht), welche Gene von Interesse sind, welche Proben ausgefiltert werden sollen usw. Eine beispielhaftes Ergebnis ist in Abb. 2 zu sehen.

5 Flowgramm-Alignment

Die Sequenziergeräte der Firmen 454 Life Sciences und Ion Torrent liefern ihre Rohdaten als *Flowgramme*, aus denen dann die Reads in der gewohnten Darstellung als Zeichenkette über dem DNA-Alphabet gewonnen werden. Um die bei der Konversion enstehenden Fehler zu verringern, ist es sinnvoll, direkt mit den Flowgrammen zu arbeiten. 454 und Ion Torrent detektieren nicht einzelne Basen, sondern messen die Länge von *Homopolymeren*, also Läufe identischer Basen. Diese Längenmessung ist fehlerbehaftet. Ein einzelner *Flow* ist ein Paar $(c, f) \in \Sigma \times \mathbb{R}_0^+$, wobei wir dies als c^f schreiben und f als *Intensität* bezeichnen. Ein Flowgramm ist eine Sequenz von Flows. Wird beispielsweise das DNA-Molekül `TTCGG` sequenziert, so ist ein mögliches gemessenes Flowgramm $\mathtt{T}^{2.3}\mathtt{A}^{0.1}\mathtt{C}^{0.9}\mathtt{G}^{1.9}$.

Üblicherweise werden die reellwertigen Intensitäten gerundet, um eine normale Zeichenkette zu erhalten. Dies führt zu Informationsverlust: Zum Beispiel werden sowohl $\mathtt{C}^{3.50}$ als auch $\mathtt{C}^{4.49}$ zu `CCCC`, obwohl sie beim Vergleich mit einer Referenz, an der `CCC` steht, anders bewertet werden sollten. Ein ähnliches Problem gibt es, wenn $\mathtt{C}^{4.49}$ und $\mathtt{C}^{4.50}$ einmal zu `CCCC` und einmal zu `CCCCC` werden. Hier geht die Information verloren, dass beide Intensitäten sehr ähnlich sind. Statt also Flowgramme erst zu Strings zu konvertieren, um dann Standard-Alignment-Algorithmen zu nutzen, um sie gegen eine Referenz zu alignieren, ist es vorteilhaft, die Flowgramme ohne Informationsverlust direkt an die Referenz zu alignieren. Hierfür wurden bereits Verfahren vorgeschlagen [VJZL08], die aber entweder die Referenz in ein künstliches Flowgramm oder den Read in ein Flowgramm/String-

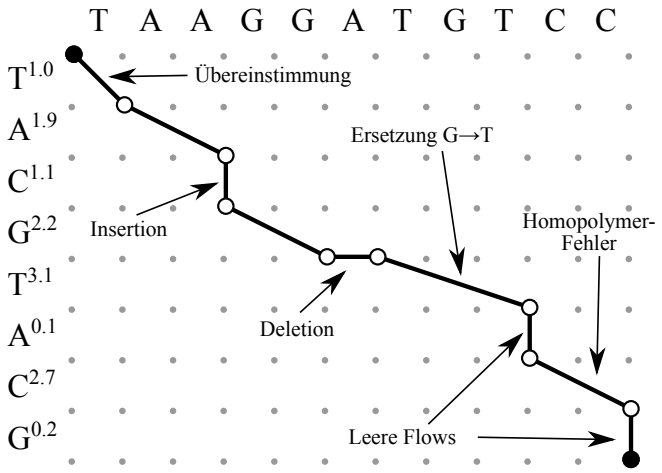


Abbildung 3: In diesem Flowgramm-Alignment-Graph ist ein Pfad eingezeichnet, der für ein mögliches Flowgramm-Alignment steht. Der Übersichtlichkeit halber sind nur die tatsächlich gewählten Kanten eingezeichnet, nicht alle möglichen.

Hybrid konvertieren, wodurch sie nicht in der Lage sind, alle Arten von Veränderungen korrekt zu modellieren. Unsere Methode ist hingegen ein *direktes* Alignment zwischen Flowgramm und Referenz und reduziert so die durch das Runden verursachten Sequenzierfehler.

Die Idee ist, jeden Flow mit einem Teilstring der Referenz zu paaren, wobei die Konkatenation aller Teilstrings die Referenz ergibt und einzelne Teilstrings leer sein dürfen (ε). Außerdem dürfen Teilstrings der Referenz mit einem „Gap“($-$) gepaart werden. Ein Flowgramm-Alignment sieht beispielsweise so aus (auch in Abb. 3 zu sehen):

$\mathtt{T}^{1.0}$	$\mathtt{A}^{2.1}$	$\mathtt{C}^{1.1}$	$\mathtt{G}^{2.2}$	$-$	$\mathtt{T}^{3.1}$	$\mathtt{A}^{0.1}$	$\mathtt{C}^{2.7}$	$\mathtt{G}^{0.2}$
T	AA	ε	GG	A	TGT	ε	CC	ε

Es sind also alle Arten von Veränderungen darstellbar: $\mathtt{T}^{3.1}$ aligniert an TGT enthält einen Substitution; $\mathtt{C}^{1.1}$ aligniert an ε ist eine Insertion und der Gap aligniert an A ist eine Deletion. Wie andere Alignments auch lässt sich ein Flowgramm-Alignment als ein Pfad in einem Alignment-Graphen darstellen (siehe Abb. 3). Jeder Knoten in Spalte i (Randknoten ignorierend) hat eine Kante zu seinem linken Nachbarn und zu allen Knoten $1, \ldots, i$ in der Zeile darüber. Es stellen sich nun zwei Fragen: Wie kann ein optimales Alignment, also der optimale Pfad gefunden werden und welche Scorefunktion sollte für die Kanten gewählt werden?

Optimaler Pfad. Der optimale Pfad lässt sich, wie in regulären Alignments auch, durch dynamische Programmierung finden, indem an jedem Knoten über den Score des Vorgängers plus den Score entlang der Kante zum Vorgänger maximiert wird.

Kantenscore. Ein Kantenscore muss bewerten, wie gut ein Flow zu einem Teilstring der Referenz passt, also z. B. $\texttt{T}^{3.1}$ zu `TGT`. Genau genommen sind hier zwei unterschiedliche Prozesse involviert: Zum einen gibt es Unterschiede zwischen der Referenz und der zu untersuchenden Probe, die dazu führen, dass ein Homopolymer entsteht. In diesem Beispiel wurde (augenscheinlich) aus `TGT` so `TTT`. Zum anderen gibt es Messfehler bei der Intensitätsmessung. Hier ist also die Messung von 3.1 statt 3.0 zu bewerten. Wir setzen daher den Score aus zwei Termen s_{edit} und s_{measure} zusammen.

s_{edit} ist der Score für ein Alignment des Teilstrings der Referenz an ein Homopolymer einer vorgegebenen Länge ℓ. Ein Alignment erfordert normalerweise quadratische Laufzeit, aber aufgrund der speziellen Struktur des Problems können wir eine geschlossene Formel für diesen Teilscore angeben, sodass er sich effektiv in $O(1)$ berechnen lässt. s_{measure} ist der Score, der die Messung bewertet und wir lernen die Scorefunktion, wie auch schon Vacic et al. [VJZL08], aus vorliegenden, zuverlässigen Alignments (log-odds-Scores).

Entscheidend ist noch ein weiteres Detail: Die tatsächliche Länge ℓ des Homopolymers ist nicht bekannt, so muss also auch z. B. die Möglichkeit berücksichtigt werden, dass `TGT` mit einer Deletion zu `TT` wurde ($\ell = 2$) und dann mit Intensität 3.1 gemessen wurde. Dies erfordert für eine einzelne Scoreberechnung theoretisch eine Maximierung über alle möglichen Längen ℓ. In der Dissertation können wir zeigen, dass sich dies mit wenig Speicherbedarf einfach vorberechnen lässt, sodass ein einzelner Kantenscore dennoch in $O(1)$ berechnet werden kann.

Wir können schließlich zeigen, dass mit unserer neuen Methode des direkten Flowgramm-Alignment die Sequenzierfehlerrate im Vergleich zu der naiven Rundungsmethode deutlich sinkt.

6 Abschlussbemerkungen

Bioinformatik-Forschung ist ein Feld der Informatik, in dem die Nützlichkeit für die Praxis besonders im Vordergrund steht, also letztlich ob mit einer neuen Methode Fragen der Biologie besser oder eventuell überhaupt erst beantwortet werden können. Diese Arbeit hält sich an diesen Grundsatz und neben einer soliden theoretischen Grundlage der Algorithmen wurde auf gute Benutzbarkeit der Implementierung geachtet. Insbesondere *Exomate* und *cutadapt* haben sich im praktischen Einsatz bewährt, letzteres ist weltweit in vielen Bioinformatik-Gruppen im praktischen Einsatz.

Literatur

[DBP+11] Mark A DePristo, Eric Banks, Ryan Poplin, Kiran V Garimella, Jared R Maguire, Christopher Hartl, Anthony A Philippakis, Guillermo Del Angel, Manuel A Rivas, Matt Hanna et al. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–498, Mai 2011.

[Gus97] Dan Gusfield. *Algorithms on Strings, Trees and Sequences*. Cambridge University

Press, Mai 1997.

[LD09] Heng Li und Richard Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760, Juli 2009.

[LLB+01] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh et al. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, Feb 2001.

[Mar13] Marcel Martin. *Algorithms and Tools for the Analysis of High-Throughput DNA Sequencing Data*. Dissertation, TU Dortmund, 2013.

[MEA+05] Marcel Margulies, Michael Egholm, William E Altman, Said Attiya, Joel S Bader, Lisa A Bemben, Jan Berka, Michael S Braverman, Yi-Ju Chen, Zhoutao Chen et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–380, September 2005.

[NTR+09] Sarah B. Ng, Emily H. Turner, Peggy D. Robertson, Steven D. Flygare, Abigail W. Bigham, Choli Lee, Tristan Shaffer, Michelle Wong, Arindam Bhattacharjee, Evan E. Eichler et al. Targeted capture and massively parallel sequencing of 12 human exomes. *Nature*, 461(7261):272–276, September 2009.

[SNC77] F. Sanger, S. Nicklen und A. R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12):5463–5467, Dezember 1977.

[VJZL08] Vladimir Vacic, Hailing Jin, Jian-Kang Zhu und Stefano Lonardi. A probabilistic method for small RNA flowgram matching. *Pacific Symposium on Biocomputing*, Seiten 75–86, 2008.

[ZMB+09] Michael Zeschnigk, Marcel Martin, Gisela Betzl, Andreas Kalbe, Caroline Sirsch, Karin Buiting, Stephanie Gross, Epameinondas Fritzilas, Bruno Frey, Sven Rahmann et al. Massive parallel bisulfite sequencing of CG-rich DNA fragments reveals that methylation of many X-chromosomal CpG islands in female blood DNA is incomplete. *Human Molecular Genetics*, 18(8):1439–1448, April 2009.

Marcel Martin studierte Naturwissenschaftliche Informatik an der Universität Bielefeld. Mit der Bioinformatik kam er in Kontakt, als er dort in der AG Genominformatik als studentische Hilfskraft arbeitete. Seine Diplomarbeit schrieb er über Clusteringalgorithmen für Graphen, wie sie beim Vergleich von bakteriellen Proteinsequenzen entstehen. Für die Promotion wechselte er zur TU Dortmund in die dort gerade entstandene Arbeitsgruppe *Bioinformatik für Hochdurchsatztechnologien*. Seit Januar 2014 ist er als Postdoc am SciLifeLab in Stockholm.

Geometrische Formen in Vektorfeldern*

Kontinuierliche Deformationen und Oberflächen-basierte Strömungsvisualisierungen

Janick Martinez Esturo
Max Planck Institut für Informatik, Saarbrücken
janick@mpi-inf.mpg.de

Abstract: *Geometrischen Formen* haben sich zu einem unverzichtbaren Kernbestandteil in einer Vielzahl von Anwendungsgebieten entwickelt. Hierzu zählen die digitale Entwicklung und Fertigung industrieller Produkte sowie Anwendungen in der Medizin, Architektur und der Unterhaltungsindustrie, um nur einige Beispiele zu nennen. Das Forschungsfeld der Geometrieverarbeitung beschäftigt sich als Teilgebiet der Informatik mit der effektiven computergestützten Verarbeitung von geometrischen Formen.

In der vorgestellten Dissertation werden neue Lösungen für offene Probleme der Geometrieverarbeitung über *kontinuierliche* Beschreibungen mit Hilfe von *Vektorfeldern* vorgeschlagen und untersucht. Die Arbeit gliedert sich dabei in zwei Teile: Im ersten Teil werden Vektorfelder zur effektiven *Manipulation* von geometrischen Formen genutzt. Dabei werden kontinuierliche Deformationen sowohl zur interaktiven Modellierung als auch zur Optimierung von geometrischen Formen genutzt. In dem zweiten Teil der Dissertation werden Vektorfelder nicht mehr zur Repräsentation von Deformationen genutzt. Stattdessen werden sie als Strömungsfelder interpretiert, die charakteristische geometrische Formen, wie beispielsweise Stromflächen, *definieren*, und zur Visualisierung dieser komplexen Vektorfelder genutzt werden.

Die in diesem Artikel vorgestellte Dissertation [Mar13] wurde von der Fakultät für Informatik der Universität Magdeburg angenommen und die vorgeschlagenen Beiträge in begutachteten internationalen Konferenzbänden und Zeitschriften veröffentlicht.

1 Einleitung

Jahrzehntelang wurden Medien ausschließlich von Text dominiert. Danach wurde das Konzept der Multi-Media in Form von Bildern, Tönen und Videos eingeführt. Erst kürzlich haben sich Modelle für *geometrischen Formen* von Objekten zu einer wichtigen neuen Form von digitalen Medien entwickelt. Digitale Repräsentationen von geometrischen Formen sind allgegenwärtig. Ihre Relevanz für eine Vielzahl von wirtschaftlichen Anwendungsbereichen steigt stetig an. Beispielsweise tragen anspruchsvolle Methoden auf Basis geometrischer Daten im industriellen Design dazu bei Produkte zu verbessern und gleichzeitig deren Kosten zu senken. Auf die gleiche Weise beruhen ganze wissenschaftliche Teilbereiche, wie bspw. die Visualisierung oder die computergestützte Medizin, maßgebliche auf der Erforschung verschiedener Objektrepräsentationen von realen oder simulierten Geometrien für die Entdeckung und Entwicklung neuer wissenschaftlicher Erkenntnisse.

Die Bedeutung von geometrischen Formen ist eng mit der Notwendigkeit von entsprechenden Algorithmen und Datenstrukturen verbunden, die für deren effektive Verarbeitung erforderlich sind. In der Informatik studiert der Bereich der *digitale Geometrieverarbeitung* als Teilgebiet der Computergrafik Methoden für die Verarbeitung von geometrischen Formen, die sowohl effizient als auch skalierbar sind. Die numerische Verarbeitung von geometrischen Formen bildet den Kern der geometrischen Modellierung. Eine Vielzahl an

*Englischer Titel der Dissertation: "*Shapes in Vector Fields* – Methods for Continuous Deformations and Surface-based Flow Visualizations"

effektiven Methoden sind bspw. für die Erfassung, Analyse und Speicherung von geometrischen Formen bekannt [BKP+10].

Die in diesem Artikel vorgestellte Dissertation [Mar13] befasst sich mit der *Manipulation* von geometrischen Formen, d.h. mit der zielgerichteten Modifikation der Geometrie von Formen. Modifikationen können dabei entweder interaktiv durch den Benutzer zur persistenten *Modellierung* von Formen genutzt werden, oder zur *Optimierung* von Formen durch automatische Methoden, die keine Nutzerinteraktion benötigen. Die Modellierung von Formen ist ein klassisches und gleichzeitig sehr anspruchsvolles Problem in der digitalen Geometrieverarbeitung, da die oftmals komplexen Deformationsmethoden interaktiv ausgeführt und hinter intuitiv zu bedienenden Nutzerschnittstellen verborgen sein müssen. Auf der anderen Seite ist die Optimierung von Formen in der Regel ein Offline-Prozess, der jedoch ein noch höheres Maß an Robustheit und Genauigkeit der Lösung garantieren muss.

Die Manipulation von Formen ist eng verbunden mit ihren digitalen Repräsentationen und dem verwendeten Deformationsparadima. Geometrische Formen werden typischerweise als Punktmengen interpretiert und in Abhängigkeit der vorhandenen Daten und der beabsichtigten Anwendung repräsentiert, wobei explizite und implizite Formrepräsentationen am üblichsten sind. Unabhängig von der Formrepräsentationen ist das verwendete Deformationsparadima: Klassischerweise werden Deformationen als Abbildungen der Form (oder deren Umgebenen Raum) auf eine deformierte Variante modelliert, wobei diese Deformationen in einem einzelnen Schritt berechnet werden. Lineare und nichtlineare Methoden dieses “Einzelschritt”-Paradigmas sind gut erforscht.

Im Kontrast klassischen Einzelschritt-Modellen werden in diese Arbeit *kontinuierliche* Manipulationen studiert, d.h. Abbildungen der Formen, die durch eine kontinuierlich parametrisierte Familie von Deformationen repräsentiert sind. Bis jetzt haben kontinuierliche Deformationen wenig Aufmerksamkeit in der Forschung erhalten, obwohl eine Reihe von klassischerweise schwierigen nichtlinearen Problemen, wie bspw. die Berechnung von volumenerhaltenden Deformationen, in diesem Framework sehr natürliche Lösungen besitzen. Ein eleganter Weg um kontinuierliche Deformationen zu definieren ist die Verwendung von *Vektorfeldern*, die die Deformationen durch Integration der Formen entlang dieser Geschwindigkeitsfelder leiten. Im Rahmen der Dissertation werden Vektorfeldbasierte Deformationen von explizit und implizit definierten Formen studiert und sowohl zur Modellierung als auch zur Optimierung von geometrischen Formen eingesetzt. Es wird gezeigt, wie mit Hilfe der neu entwickelten Methoden Lösungen bisher offener Probleme der geometrischen Modellierung erhalten werden können und die Qualität bisheriger Deformationsansätze noch weiter gesteigert werden kann. Kontinuierliche Deformationen sind der Schwerpunkt des ersten Teils dieser Dissertation, in dem sowohl neue Vektorfeld-Energien und korrespondierende Typen von Deformationen als auch neue Anwendungen für kontinuierliche Deformationen vorgestellt werden.

In der Regel sind Vektorfelder von kontinuierlichen Deformation nicht von vornherein gegeben, sondern müssen auf eine problemspezifische Weise berechnet werden, etwa um möglichst wenig Verzerrung zu induzieren. Darüber hinaus kann das abstrakte Vektorfeld-Konzept zudem noch weitere Typen von Feldern beschreiben: Vektorfelder eignen sich bspw. besonders gut zur Repräsentation der Geschwindigkeitsfelder von komplexen Strömungen. Die Analyse der Eigenschaften dieser Strömungsfelder ist von besonderem Interesse für verschiedenen wissenschaftlichen Disziplinen, etwa im Maschinenbau. Im Gegensatz zum

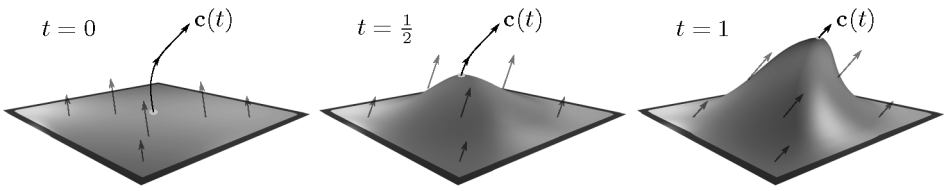


Abbildung 1: Kontinuierliche Vektorfeld-basierte Deformation. Eine kleine Griffregion (•) wird kontinuierlich entlang einer vom Benutzer vorgegebene Leitkurve $\mathbf{c}(t)$ bewegt, während der Rand (•) der Oberfläche fixiert bleibt. Für jeden Integrationsschritt zum Zeitpunkt t ist die aktuelle Oberflächengeometrie und die erzwungenen Vektoren (•) gegeben, während das übrige Vektorfeld (•) berechnet werden muss, bspw. durch eine globale Energieminimierung.

ersten Teil der Dissertation sind Strömungsfelder entweder gemessen bzw. simuliert. Insbesondere sind sie a priori für die Analyse gegeben und werden im Zuge dessen nicht, wie im ersten Teil der Arbeit, modifiziert. Als Teilgebiet der Computergrafik konzentriert sich die *Strömungsvisualisierung* auf die Berechnung von abstrakteren und charakteristischeren Repräsentationen von Strömungsfeldern um die Analyse von komplexen Flussphänomenen zu vereinfachen. Unter den verschiedenen Typen von Strömungsvisualisierungen beruhen die Geometrie-basierten Methoden auf geometrischen Formen zur Visualisierung von Flusseigenschaften. Im Allgemeinen sind diese Formen durch die Strömungsfelder *definiert*. Durch Vektorfelder definierte geometrische Formen werden im zweiten Teil der Dissertation untersucht. Dabei wird gezeigt, wie bekannte Methoden aus dem verwandten Gebiet der Geometrieverarbeitung zur Verbesserung von Geometrie-basierten Strömungsvisualisierungen eingesetzt werden können.

In den folgenden Abschnitten werden die wissenschaftliche Hauptbeiträge der vorgestellten Dissertation kurz erläutert und mit anschaulichen Beispielen illustriert.

2 Vektorfeld-basierte *Manipulation* Geometrischer Formen

Im Rahmen des ersten Teiles der vorgestellten Dissertation werden kontinuierliche Deformationen für verschiedene Typen von geometrischen Formen und für verschiedene Manipulationsziele entwickelt. Die wichtigsten Beiträge werden im Folgenden dargestellt.

2.1 Generalisierte Vektorfeld-Energien

Jede Vektorfeld-basierte Deformation wird über eine kontinuierliche Raumzeit-Integration einer initialen geometrischen Form beschrieben. Abbildung 1 illustriert dieses Konzept. In der Regel werden Vektorfelder hierfür durch globalen Energieminimierungen berechnet und mit Hilfe von numerischen Verfahren integriert (siehe bspw. [SBBG11a]).

In Abhängigkeit der spezifischen Formrepräsentation und des konkreten Modellierungsproblems werden verschiedene Typen von Vektorfeldern benötigt: Hierfür wird im Rahmen der Dissertation eine Familie *generalisierter Vektorfeldenergien* eingeführt, die Approximationen von *isometrischen*, *konformen*, sowie *volumenerhaltenden* kontinuierlichen Deformationen von planaren und volumetrischen Formen ermöglicht [MRT13]. Diese generalisierte Energie ist die erste Formulierung plastischer kontinuierlicher Deformationen in der Modellierung, die das gesamte Spektrum an geometrischen Deformationen beschreiben kann. Interessanterweise kann diese Energie über einen einzelnen Freiheits-

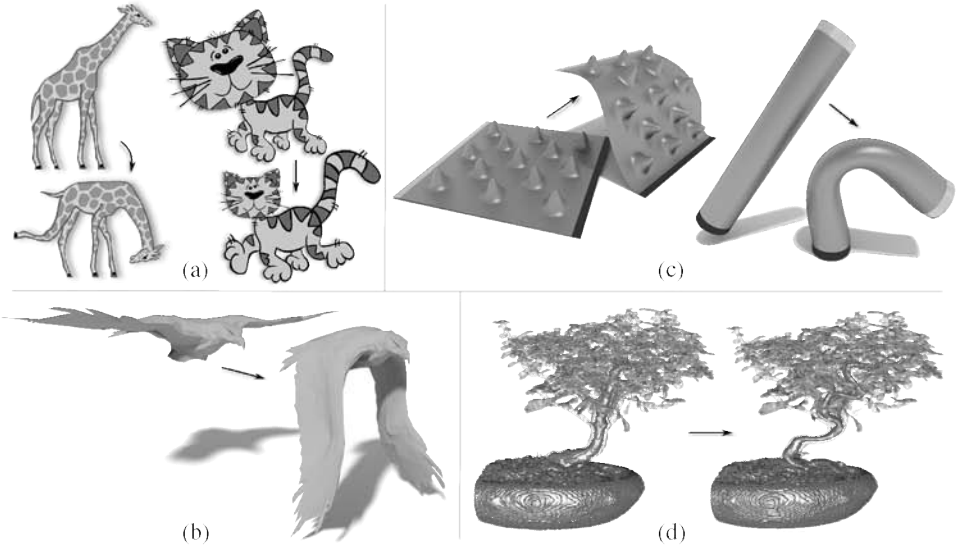


Abbildung 2: *Kontinuierliche* Deformationen verschiedener Formrepräsentationen. (a) Isometrische und konforme Deformationen von $2d$ Formen [MRT13]. (b) Isometrische Deformationen von volumetrischen $3d$ Körpern [MRT13]. (c) Isometrische Deformationen von Oberflächen [MRT12]. (d) Volumenerhaltende Deformationen von Isoflächen [MRT10].

grad parametrisiert werden und unterstützt darüber hinaus inhomogene und anisotrope Materialeigenschaften. Im Kontrast zu elastischen Deformationen klassischer Einzelschritt-Verfahren müssen keine globalen nichtlinearen Optimierungen ausgeführt werden, da sich die Vektorfeldoptimierungen als globale lineare Probleme formulieren lassen. Durch die Einführung des neuen Konzeptes der Energieglattheit wird eine hohe Qualität der Deformationen erreicht, die selbst modernste nichtlineare Standardverfahren übertreffen kann. Um Interaktivität und Skalierbarkeit zu gewährleisten werden darüber hinaus große Teile der Berechnungen parallelisiert auf der Grafikkarte (GPU) ausgeführt. Abbildung 2 (a) zeigt Beispiele für planare isometrische und konforme $2d$ Deformationen und Abbildung 2 (b) zeigt ein Beispiel einer isometrischen Deformation eines volumetrischen $3d$ Körpers. Planare Formen werden dabei durch Netze von Dreiecken und volumetrische Körper durch Netze von Tetraedern diskretisiert.

2.2 Isometrische Deformationen von Oberflächen

Über die generalisierten Vektorfeldenergien lassen sich Deformation hoher Qualität berechnen. Unglücklicherweise unterstützt die mathematische Formulierung dieser Energien nur planare und volumetrische Formen, jedoch keine Deformationen von Oberflächen. Da durch Netze von Dreiecken diskretisierte Oberflächen von besonderer praktischer Bedeutung sind wird in der Dissertation eine neue Energie vorgeschlagen, die auch diesen Typ von geometrischer Form unterstützt. Diese Energie beschreibt den wichtigen isometrischen Deformationstyp indem Vektorfelder optimiert werden, die lokal so nah wie möglich zu *starren Transformationen* sind und global eine glatte Variation besitzen [MRT12]. Wie zuvor bedarf es nur linearer Minimierungen um global optimale Vektorfelder zu erhalten.

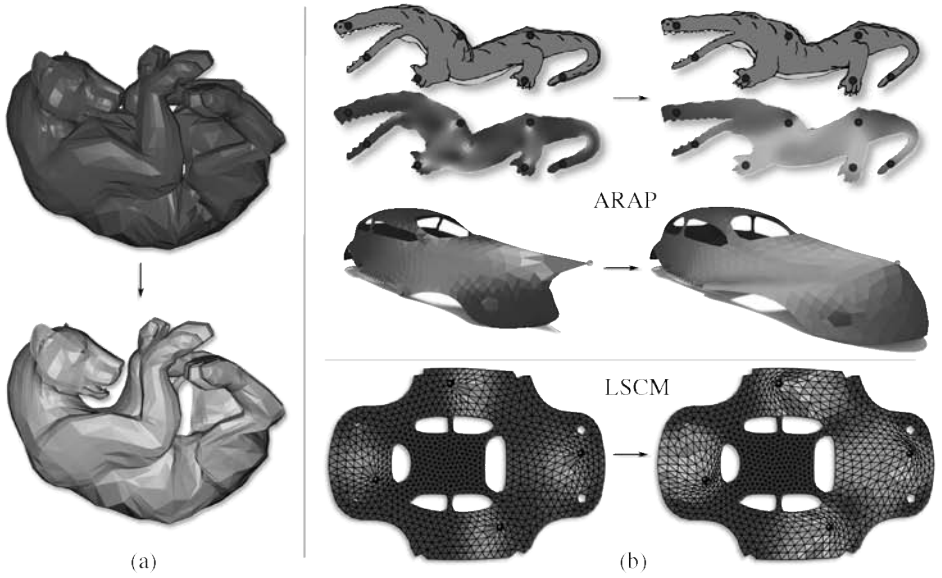


Abbildung 3: Optimierung von Formrepräsentationen und Geometrieverarbeitungsmethoden. (a) Kontinuierliche Deformationen werden zur *Korrektur* von artefaktbehafteten Posen (●) genutzt: die korrigierte Pose (●) ist frei von *Selbstüberschneidungen* [MRF+11]. (b) Ein generischer und problemspezifischer linearer *Regularisierungsansatz* erhöht die Effektivität einer Vielzahl von Geometrieverarbeitungsmethoden. Beispielsweise können Artefakte in etablierten und weit verbreiteten ARAP Deformationen und in konformen LSCM Parametrisierungen verhindert werden [MRT14].

Die Berechnungen können wieder durch die GPU parallelisiert werden, wobei sich der vorgeschlagene generische Ansatz zum parallelisierten Aufsetzen der linearen Systeme auch zur Beschleunigung anderer zellbasierter Verfahren anwenden lässt, bspw. für Simulationsmethoden basierend auf finiten Elementen. Die Qualität der Ergebnisse kann lineare sowie nichtlineare State-of-the-Art Methoden durchgängig übertreffen (vgl. [BPGK06, BS08]). Abbildung 2 (c) zeigt Beispiele für isometrische Oberflächendeformationen, die durch diese Energieformulierung erhalten werden.

2.3 Volumenerhaltende Deformationen von Isoflächen

Die zuvor beschriebenen kontinuierlichen Deformationen unterstützen nur *explizit* definierte planare und volumetrische Formen sowie Oberflächen. Für viele Probleme in der Computergrafik sind jedoch *implizit* definierte Formen besser geeignet, bspw. eignen sich Isoflächen, die als Isokonturen eines Skalarfeldes definiert sind, besonders zur Rekonstruktion von gescannten Oberflächen. Auch gegenüber klassischen Deformationsverfahren, die in der Regel nur eine einzelne Isofläche manipulieren können, haben Vektorfeld-basierte Deformationen eine Reihe von Vorteile für diesen Formtyp: Im Rahmen der Dissertation wird eine neue Deformationsmethode vorgeschlagen, die unter Verwendung von global *divergenzfreien Vektorfeldern* garantieren kann, dass sich das eingeschlossene Volumen einer Isofläche nicht ändert [MRT10]. Dies gilt nicht nur für eine einzelne Isofläche, sondern für *jede* in dem Volumen vorhandene Isofläche. Da viele gebräuchliche Materialien ihr Volumen unter Deformationen approximativ erhalten ist die Eigenschaft der Volumenerhaltung

geeignet plausible Deformationen zu beschreiben. Darüber hinaus kann die Erhaltung der Topologie jeder Isofläche garantiert werden. Da die genutzten divergenzfreien Vektorfelder unabhängig vom zugrundeliegenden Skalarfeld sind kann zudem ein neu entwickeltes, sehr effizientes und genaues Integrationsschema namens "rückwärts-Lagrangian Integration" verwendet werden. Dieses wird durch die GPU parallelisiert ausgeführt und garantiert damit Interaktivität der Deformationen. Zudem können Isoflächen-Diskretisierungen hoher Qualität jederzeit durch eine Offline-Rekonstruktion berechnet werden. Experimentelle Ergebnisse belegen die theoretisch gezeigten Eigenschaften der Volumen- und Topologieerhaltung selbst für extreme Deformationen. Abbildung 2 (d) zeigt ein Beispiel für eine volumenerhaltende Isoflächendeformation des Stammes eines gescannten Bonsai-Baumes.

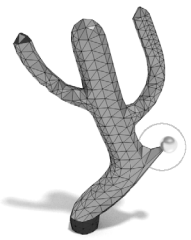
2.4 Posen-Korrektur durch Raumzeit-Integration

Eine Vielzahl von Modellierungsmethoden nutzt Beispieldaten in Form von verschieden deformierten Posen ein und desselben Objektes um neue Ergebnisse abzuleiten (siehe bspw. [FB11]). Die Effektivität dieser Verfahren leidet jedoch stark unter Posen mit geometrischen Inkonsistenzen, z.B. in Form von lokalen Selbstüberschneidungen. Da Posen eine feste paarweise Konnektivität besitzen müssen können keine Standardverfahren zur Korrektur der Artefakte eingesetzt werden, da diese die Konnektivität modifizieren würden. Dieses offene Problem der Modellierung kann dabei elegant durch einen Vektorfeld-basierten Ansatz gelöst werden indem eine grundlegende Eigenschaft von integrations-basierten Systemen ausgenutzt wird: Die Fronten von Pfadlinien schneiden sich zu keinem Zeitpunkt in Raumzeit. In der Dissertation erlaubt dies die Formulierung des ersten deformationsbasierten Ansatzes zur *Optimierung* von Artefakten in Posendatenbanken, der Konnektivität nicht modifiziert [MRF+11]. Dazu wird ein *Raumzeit-Vektorfeld* so an die Geometrie gefittet, dass Integration der Referenzpose in diesem Feld *korrigierte Posen* beschreibt, die frei von geometrischen Artefakten sind. Die korrigierten Posen werden dabei nur lokal modifiziert um frei von Selbstüberschneidungen zu sein und reproduzieren global die originale Posengeometrie. Abbildung 3 (a) zeigt das Ergebnis einer automatischen Korrektur einer Pose eines Löwenmodells, das initial eine hohe Anzahl an Selbstüberschneidungen aufweist. Technisch werden die Raumzeit-Vektorfelder dabei durch 4-dimensionale radiale Basisfunktionen (RBFs) repräsentiert, deren Berechnung zum einen durch Auslagerung von Operationen auf die GPU beschleunigt wird. Zum anderen wird zusätzlich ein neues Selektionsschema für RBF-Zentren vorgestellt, welches eine höhere Konvergenzrate und numerische Stabilität im Vergleich zu bisherigen Standardverfahren bietet. Gekoppelt mit einer neuen Formulierung für die Aktualisierung der Faktorisierungen der linearen RBF-Systeme ergibt sich so ein neuer generischer RBF-basierter Approximationsansatz, der effizient und effektiv ist. Dieses Schema eignet sich dabei nicht nur zur Vektorfeld-basierten Posenkorrektur sondern ist zur Approximation *beliebiger* hochdimensional verstreuter Daten einsetzbar.

2.5 Geglättete Energien für die Geometrieverarbeitung

Eine Großteil an Methoden der Geometrieverarbeitung beruht auf dem Prinzip der Energieminimierung: Problemspezifische Energien werden so formuliert, dass Minimierer dieser Funktionale die gesuchten global optimalen Lösungen der Probleme sind. Oftmals kommen dabei quadratische Energien zum Einsatz, die zu effizient zu berechnenden linearen

Optimierungsproblemen führen. Einige Beispiele sind die Berechnung von Einzelschritt-Deformationen verschiedener Formtypen, Parametrisierungen von Oberflächen oder die oben beschriebenen Vektorfeld-basierten kontinuierlichen Deformationen.



Vieler dieser Methoden generieren dabei Resultate, die in der Nähe der Nutzerbedingungen Artefakte wie Diskontinuitäten der Lösungen aufweisen, da die Energien in der Regel keine Glattheitseigenschaften beschreiben (siehe Einschub rechts). Doch selbst wenn Glattheit der Lösungen über Standardregularisierungen in der Optimierung erzwungen wird können viele der Artefakte nicht korrigiert werden, da diese Art von Regularisierung unabhängig von der konkret optimierten Energie ist. Um dieses sehr allgemeine und weitgreifende Problem der Geometrieverarbeitung zu lösen wird im Rahmen der Dissertation die erste generische Regularisierung vorgeschlagen, die nicht auf der Glättung der Lösung, sondern auf der *Glättung der optimierten Energie* selbst beruht [MRT14]. Diese neue Form von Regularisierung ist daher direkt abhängig von der betrachteten Energie, sie ist somit *problemspezifisch*. Für eine Vielzahl von verschiedenen Problemen kann dabei die Effektivität dieses einfachen aber mächtigen Konzeptes demonstriert werden. Abbildung 3 (b) zeigt mit problemspezifischen Regularisierungen optimierte Ergebnisse von planaren und Oberflächen-basierten Deformationen etablierter nichtlinearer *as-rigid-as-possible* (ARAP) Deformationen und konformen *least-squares conformal maps* (LSCM) Parametrisierungen (siehe [LPRM02, LZX+08]). Dabei ist diese Form der linearen Regularisierung sehr einfach und generisch zu implementieren und zeigt keine signifikanten Steigerungen der Laufzeiten. In vielen Fällen weist sie die gleiche Qualität komplexer zu berechnender nichtlinearer Regularisierungen auf und kann diese oftmals sogar übertreffen. Da diese Eigenschaften der vorgestellten generischen Regularisierung potentiell die Effektivität eines Großteils der etablierten Geometrieverarbeitungsmethoden (und die Methoden weiterer Forschungsfelder, die Energieminimierungen zur Problemlösung nutzen) erhöhen kann ist sie eine der wichtigsten Beiträge der vorgestellten Dissertation.

3 Vektorfeld-basierte *Definition* Geometrischer Formen

Nachdem im ersten Teil der Dissertation Manipulationen von Formen durch Vektorfelder betrachtet wurden fokussiert sich der zweite Teil der Arbeit auf Probleme der Strömungsvisualisierung in denen geometrische Formen durch Vektorfelder definiert werden. Im Folgenden werden die wichtigsten Beiträge zur Strömungsvisualisierung zusammengefasst.

3.1 Poisson-basierte Methoden für Oberflächen-basierte Strömungsvisualisierung

Geometrie-basierte Ansätze der Strömungsvisualisierung nutzen unterschiedliche Typen von *integrierten* geometrischen Formen verschiedener Dimensionalität, darunter Partikel, Integralkurven sowie Integralflächen. Auf Grund ihrer höheren intrinsischen Dimensionalität kann bereits eine geringe Anzahl von Integralflächen selbst komplexe Strömungsphänomene gut beschreiben. Integralflächen in stationären Strömungen werden *Stromflächen* genannt. Klassischerweise werden sie aus dem Vektorfeld extrahiert in dem sie von definierenden *Saatkurven* aus entlang des Feldes integriert werden. Nutzer können Saatkurven manipulieren um interessante Stromflächen zu selektieren. Leider ist jedoch gerade die

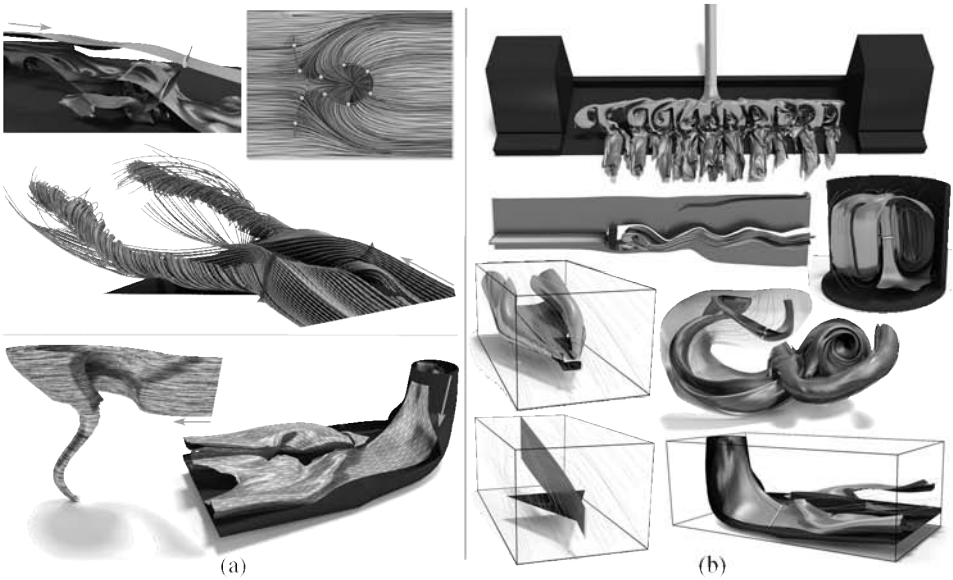


Abbildung 4: Oberflächen-basierte Flussvisualisierung. (a) Über *Poisson-basierte Deformationen* werden interaktiv Oberflächen positioniert, die entlang oder orthogonal zum Fluss ausgerichtet sein können. Orthogonale Flächen (●) eignen sich besonders zur Definition von Saatstrukturen (●) von exakt integrierten Geometrien (●) (oben). *Poisson-basierte Parametrisierungen* ermöglichen zudem effiziente illustrative Visualisierungen (unten) [MSRT13b]. (b) Die Einführung des ersten globalen und *differentialgeometrischen Qualitätsmaßes* für Stromflächen ermöglicht darüber hinaus die vollkommen *automatische Selektion* [MSRT13a].

Selektion von repräsentativen Stromflächen eine zeitaufwändige und mühsame Aufgabe, da die Integration der Flächen nur schwer vorherzusagen ist.

Um Nutzer bei der interaktiven Selektion von Stromflächen zu unterstützen wird daher in der Dissertation die erste Extraktionsmethode für Strom-ausgerichteten Flächen vorgeschlagen, die *nicht* wie klassische Ansätze integrations-basiert ist. Stattdessen wird ein neuer *deformations-basierter* Ansatz zur Extraktion Strom-ausgerichteter Flächen vorgeschlagen [MSRT13b]: Nutzer können direkt mit ganzen Flächen interagieren, die sich automatisch deformieren und interaktiv an der lokalen Strömung ausrichten. Erstmalig werden dabei aus der Geometrieverarbeitung wohlbekannte Poisson-basierte Problembeschreibungen auf Probleme der Strömungsvisualisierung angewendet. Mit diesen Techniken können sowohl klassische Stromflächen als auch i.A. nicht integrierbare Strom-orthogonale Flächen extrahiert werden, die sich besonders zur Definition von Saatstrukturen eignen. Darüber hinaus werden mit dem gleichen Poisson-basierten Framework effizient Parametrisierungen von Stromflächen berechnet, die illustrative Visualisierungen der Strömung auf der Fläche ermöglichen. Abbildung 4 (a) zeigt Beispiele für interaktiv positionierte strom-ausgerichtete Flächen in verschiedenen komplexen Strömungsdatensätzen sowie Ergebnisse für parametrisierungs-basierte illustrative Strömungsvisualisierungen.

3.2 Automatische Globale Selektion von Stromflächen

Die zuvor beschriebenen Poisson-basierten Methode vereinfachen die manuelle Exploration von Strömungen. Dieses interaktive Verfahren wird nun durch einen neuen Ansatz komplementiert, der erstmals eine vollkommen *automatische Selektion* von global charakteristischen Stromflächen erlaubt. Die automatische Selektion von Stromflächen ist ein wichtiges jedoch bis dahin ungelöstes Problem der Visualisierung, welches bspw. die unbeaufsichtigte Analyse von simulierten Strömungen ermöglicht. Dieses Problem ist komplex, da der Raum der möglichen Stromflächen extrem groß ist und bis dahin kein Selektionskriterium für relevante Stromflächen in der Literatur existierte. Im Rahmen der Dissertation wird daher das erste *differentialgeometrische Qualitätsmaßes* für Stromflächen eingeführt, das auf Wahrnehmungseigenschaften der Flächen und der in ihnen verlaufenden Strömung basiert [MSRT13a]. Differentialgeometrische Ansätze sind in der Geometrieverarbeitung weit verbreitet, wurden jedoch bis jetzt sehr selten zur Lösung von Visualisierungsproblemen eingesetzt. Für verschiedenste Datensätze kann gezeigt werden, dass dieses Maß deren charakteristischen Stromflächen beschreibt. Zur Optimierung wird auf Grundlage dieses Selektionskriteriums eine globale Diskretisierung des Raumes der möglichen Stromflächen vorgestellt, die durch einen globalen gewichteten Graphen repräsentiert wird. Dadurch können Saatkurven durch simple Pfade in dem globalen Graphen dargestellt werden. Die Diskretisierung des Suchraumes wird von einem neuen Selektionsalgorithmus zur globalen Optimierung der Saatkurve der global optimalen Stromfläche genutzt, der die Lösung dieses NP-schweren Problems dabei durch simulierte Abkühlung approximiert. In Abbildung 4 (b) werden für verschiedene Datensätze automatisch selektierte charakteristische Stromflächen gezeigt. Diese automatisch selektierten Stromflächen sind sehr ähnlich zu von Visualisierungsexperten manuell selektierten Flächen, was die Relevanz der erzielten Ergebnisse bekräftigt.

4 Zusammenfassung

Die vorgestellte Dissertation hat eine Reihe von Beiträgen auf den Gebieten der geometrischen Modellierung und der Strömungsvisualisierung geleistet. Im Kern wird die Beziehung zwischen geometrischen Formen und Vektorfeldern untersucht. Daraus wurden eine Reihe von neuen Ansätzen für offene Probleme beider Disziplinen entwickelt, indem etablierte Konzepte eines Bereiches erfolgreich zur Lösung offener Fragen des anderen Bereiches übertragen werden konnten: Hierzu zählen zum einen der integrations-basierte Ansatz zur Beschreibung kontinuierlicher Deformation verschiedener geometrischer Formen, wobei planare, volumetrische, Oberflächen- und Isoflächen-Deformationen sowie Vektorfeld-basierte Korrekturen von Posen vorgeschlagen wurden. Zum anderen wurden erstmalig etablierter Deformationsverfahren und differentialgeometrische Konzepte zur Flächen-basierten Strömungsvisualisierung angewendet. Dies hat erstmals Poisson-basierte interaktive Explorationen von Strömungen sowie Qualitätsmaße zur automatischen Selektion von Stromflächen ermöglicht. Darüber hinaus wurde mit dem generischen Konzept der Energieregularisierung eine einfache aber mächtige Technik vorgestellt, die erfolgreich zur Verbesserung einer Vielzahl von etablierten Methoden der Modellierung eingesetzt worden ist und auf Grund seiner Universalität auch in anderen Disziplinen anwendbar ist.

Publikationen

[Mar13] J. Martinez Esturo. *Shapes in Vector Fields*. Dissertation, University of Magdeburg, 2013.

[MRF+11] J. Martinez Esturo, C. Rössl, S. Fröhlich, M. Botsch und H. Theisel. Pose Correction by Space-Time Integration. In *Proc. VMV*, Seiten 33–40. EG, 2011.

[MRT10] J. Martinez Esturo, C. Rössl und H. Theisel. Continuous Deformations of Implicit Surfaces. In *Proc. VMV*, Seiten 219–226. EG, 2010.

[MRT12] J. Martinez Esturo, C. Rössl und H. Theisel. Continuous Deformations by Isometry Preserving Shape Integration. *Springer LNCS (Proc. Curves and Surfaces 2010)*, 6920(1):456–472, 2012.

[MRT13] J. Martinez Esturo, C. Rössl und H. Theisel. Generalized Metric Energies for Continuous Shape Deformation. *Springer LNCS (Proc. Curves and Surfaces 2012)*, 8177(1):135–157, 2013.

[MRT14] J. Martinez Esturo, C. Rössl und H. Theisel. Smoothed Quadratic Energies on Meshes. *ACM Trans. Graph.*, 2014. To appear.

[MSRT13a] J. Martinez Esturo, M. Schulze, C. Rössl und H. Theisel. Global Selection of Stream Surfaces. *Comput. Graph. Forum (Proc. Eurographics)*, 32(2):113–122, 2013.

[MSRT13b] J. Martinez Esturo, M. Schulze, C. Rössl und H. Theisel. Poisson-based Tools for Flow Visualization. In *Proc. PacificVis*, Seiten 241–248. IEEE, 2013.

Literatur

[BKP+10] M. Botsch, L. Kobbelt, M. Pauly, P. Alliez und B. Levy. *Polygon Mesh Processing*. AK Peters, 2010.

[BPGK06] Mario Botsch, Mark Pauly, Markus Gross und Leif Kobbelt. PriMo: coupled prisms for intuitive surface modeling. In *Proc. SGP*, Seiten 11–20, 2006.

[BS08] Mario Botsch und Olga Sorkine. On Linear Variational Surface Deformation Methods. *IEEE TVCG*, 14(1):213–230, 2008.

[FB11] Stefan Fröhlich und Mario Botsch. Example-Driven Deformations Based on Discrete Shells. *Comput. Graph. Forum*, 30(8):2246–2257, 2011.

[LPRM02] Bruno Lévy, Sylvain Petitjean, Nicolas Ray und Jérome Maillot. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 21(3):362–371, 2002.

[LZX+08] Ligang Liu, Lei Zhang, Yin Xu, Craig Gotsman und Steven J. Gortler. A local/global approach to mesh parameterization. *Comput. Graph. Forum (Proc. SGP)*, 27(5):1495–1504, 2008.

[SBBG11a] Justin Solomon, Mirela Ben-Chen, Adrian Butscher und Leonidas Guibas. As-Killing-As-Possible Vector Fields for Planar Deformation. *Comput. Graph. Forum (Proc. SGP)*, 30(5):1543–1552, 2011.

Janick Martinez Esturo wurde am 29. Juli 1983 in Bünde geboren. Er studierte Informatik an der Universität Bielefeld mit den Schwerpunkten Computergraphik sowie Robotik und erhielt sein Diplom im Jahre 2008. Von 2008 bis 2013 arbeitete er als wissenschaftlicher Mitarbeiter an der Otto-von-Guericke Universität Magdeburg, an der er seine Promotion im Oktober 2013 erfolgreich abschloss. Seit Juni 2013 forscht er als PostDoc am Max Planck Institut für Informatik in Saarbrücken. Sowohl sein Studium als auch seine Promotion wurden durch Exzellenzstipendien der "Studienstiftung des deutschen Volkes" unterstützt. Seine Forschungsinteressen liegen in den Bereichen der geometrischen Modellierung und geometriebasiert Strömungsvisualisierungen.

Beziehungsstrukturen in Evidenznetzwerken*

Folke Mitzlaff
Knowledge and Data Engineering Group
Universität Kassel
folke@mitzlaff.org

Abstract: Soziale Web-Anwendungen sind dank mobiler Internetzugangsgeräte fester Bestandteil des täglichen Lebens geworden. In vielen Lebensbereichen kommunizieren und interagieren Anwender dieser Applikationen digital und hinterlassen *Datenspuren* in den Datenbanken und Logdateien der beteiligten Server. Aus diesen Datenspuren lassen sich explizit hergestellte, aber auch implizit angesammelte Netzwerkstrukturen extrahieren, welche im Rahmen dieser Arbeit als *Evidenznetzwerke* bezeichnet werden.

Im ersten Teil der Arbeit wird mit der *Social Distributional Hypothesis* eine kausalitätsneutrale Arbeitshypothese als Grundlage der Nutzung von Evidenznetzwerken zur Analyse von Beziehungsstrukturen in Anwendungen des Social Web vorgestellt. Basierend auf dieser Arbeitshypothese wird ein Verfahren zur relativen *Bewertung von Nutzergruppen* in solchen Web-Anwendungen erarbeitet und evaluiert.

Im zweiten Teil der Arbeit wird schließlich gezeigt, wie sich die Analysewerkzeuge des ersten Teils nutzen lassen, um Beziehungen zwischen Vornamen aus solch angesammelten Datenspuren zu analysieren. Es wird vorgestellt, wie sich die so erlangten Vornamensstrukturen zur Implementierung einer Vornamenssuchmaschine verwenden lassen und personalisierte Vornamensempfehlungssysteme realisiert werden können.

1 Einleitung

Einhergehend mit der Entwicklung und zunehmenden Verfügbarkeit des Internets, hat sich die Art der Informationsbereitstellung und der Informationsbeschaffung deutlich geändert. Die einstmalige Trennung zwischen Publizist und Konsument wird durch kollaborative Anwendungen des sogenannten Web 2.0 aufgehoben, wo jeder Teilnehmer gleichsam Informationen bereitstellen und konsumieren kann. Zudem können Einträge anderer Teilnehmer erweitert, kommentiert oder diskutiert werden. Mit dem Social Web treten schließlich die sozialen Beziehungen und Interaktionen der Teilnehmer in den Vordergrund. Dank mobiler Endgeräte können zu jeder Zeit und an nahezu jedem Ort Nachrichten verschickt und gelesen werden, neue Bekannschaften gemacht oder der aktuelle Status dem virtuellen Freundeskreis mitgeteilt werden.

Damit findet ein immer größerer Teil des täglichen Lebens digital statt und wird durch entsprechende Applikationen des Social Web begleitet und aufgezeichnet. "*We live our life in the network*" schrieben hierzu David Lazer et al., als sie *Computational Social Science* als

*Englischer Titel der Dissertation: "Relatedness in Evidence Networks"

neuen interdisziplinären Wissenschaftszweig zwischen Sozialwissenschaften und Informatik proklamierten [LPA+09]. Im Kern der Computational Social Science stehen die aggregierten Daten, welche beim Betrieb sozialer Webapplikationen in Form von Datenbankeinträgen und digitalen Spuren innerhalb der Logdateien von beteiligten Servern anfallen. Lazer et al. sehen in diesen Daten die Möglichkeit, ein umfassendes Bild des Einzelnen, sowie Dynamiken von Gesellschaften zu analysieren und zu verstehen [LPA+09]. Hierbei gilt es, Fragestellungen der Sozialwissenschaften und Methoden der Informatik zusammenzuführen, einhergehend mit einem Paradigmenwechsel der Wissenschaftsmethodik, von der hypothesengetriebenen zur datengetriebenen Forschung [CGB+12, CKK13]. In diesem Kontext leistet diese Arbeit einen methodischen Beitrag zur Nutzbarmachung der im Social Web angesammelten Daten.

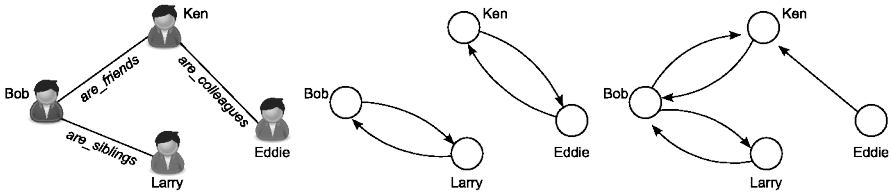
2 Die "Social Distributional Hypothesis"

Im ersten Teil der Arbeit wird untersucht, ob beobachtete Interaktionsmuster, welche aus implizit angesammelten Datenspuren abgeleitet sind, Rückschlüsse auf Beziehungsstrukturen zwischen Anwendern einer sozialen Webapplikation zulassen und greift mit *Homophily* [MSLC01] ein grundlegendes Konzept der Soziologie auf.

Homophily bezeichnet dabei das in vielen Bereichen beobachtete Phänomen, dass Menschen dazu neigen, mit solchen anderen in Beziehung zu treten, die als "ähnlich" wahrgenommen werden. Prominente Ausprägungen von Homophily basieren auf Wohlstand, sozialem Status, Alter – aber auch komplexeren Ähnlichkeiten, wie z.B. gemeinsamen politische Ansichten oder moralischen und religiösen Wertevorstellungen. Verschiedene Erklärungsansätze untersuchen Homophily und greifen dabei auf psychologische und soziale Theorien zurück [MC03].

Da bei Untersuchungen auf implizit angesammelten Daten ein expliziter Versuchsaufbau im Nachhinein nicht möglich ist, sind kausale Rückschlüsse im Allgemeinen nich zulässig. Die in dieser Arbeit vorgestellte *Social Distributional Hypothesis* dient als kausalitätsneutrale Arbeitshypothese zur Nutzung von implizit angesammelten Interaktionsdaten. Diese Arbeitshypothese baut auf dem Konzept der *Distributional Hypothesis* [Har54, MR01] aus der Linguistik auf, welche besagt, dass Wörter mit ähnlichen Verteilungsmustern (Syntax) in der Tendenz eine ähnliche Bedeutung (Semantik) haben, dass also z.B. zwei Wörter, welche häufig gemeinsam in Sätzen verwendet werden, wahrscheinlich eine verwandte Bedeutung haben.

Die Social Distributional Hypothesis wendet diesen Ansatz auf beobachtete Interaktionen zwischen Nutzern in sozialen Webapplikationen an und postuliert, dass sich Nutzer mit ähnlichen Interaktionscharakteristiken in der Tendenz auch ähnlich sind. Damit stellt die Social Distributional Hypothesis einen Zusammenhang zwischen Interaktionsmustern in sozialen Webapplikationen und semantischen Beziehungen zwischen den beteiligten Nutzern her.



(a) Soziale Grundgesamtheit (b) Follower Graph (c) ReTweet Graph

Abbildung 1: Beispiel für eine soziale Grundgesamtheit mit entsprechenden Evidenznetzwerken in Twitter, insbesondere dem Follower-Graph und dem ReTweet-Graph.

Evidenznetzwerke Die Arbeit verfolgt dabei einen graphentheoretischen Ansatz und baut somit auf Konzepte der *Sozialen Netzwerkanalyse* aus der Soziologie und *Complex Networks* aus der Physik auf. Hierzu werden *Evidenznetzerke* zur formalen Untersuchung (implizit) beobachteter Interaktionen zwischen Nutzern einer sozialen Webapplikation eingeführt, motiviert durch die Annahme, dass ein beobachtetes Interaktionsnetzwerk als Stichprobe einer zugrundeliegenden sozialen Grundgesamtheit aufgefasst werden kann.

Abbildung 1 zeigt hierzu ein exemplarisches Beispiel für eine fiktive Nutzermenge *Bob, Ken, Larry* und *Eddie* [1]. In diesem Beispiel sind für die betrachteten Nutzer die realen Beziehungsrelationen "*sind Geschwister*", "*sind befreundet*" und "*sind Arbeitskollegen*" angegeben (Abb. 1a). Innerhalb eines Beobachtungszeitraums werden Followship-Beziehungen [2] bzw. ReTweets [3] in Twitter aufgezeichnet (Abb. 1b bzw. Abb. 1c). Für die verschiedenen realen Beziehungen werden in den betrachteten Netzwerken Interaktionen unterschiedlicher Ausprägung beobachtet. So werden z.B. für Bob und Larry entsprechend ihrer Geschwisterbeziehung bidirektionale Interaktionen sowohl im Follower-Graph, als auch im ReTweet-Graph beobachtet. Die Freundschaftsbeziehung zwischen Bob und Ken dagegen spiegelt sich jedoch nur im ReTweet-Graphen wider. Genauso könnten Interaktionen zwischen Benutzern beobachtet werden, welche in keiner bekannten Beziehung zueinander stehen. In diesem Sinne wird ein beobachtetes Interaktionsnetzwerk als eine Stichprobe der zugrundeliegenden sozialen Grundgesamtheit aufgefasst.

Zur Unterstützung der Social Distributional Hypothesis werden verschiedene Interaktionsnetzwerke aus Twitter [4], Flickr [5] und BibSonomy [6] untersucht und in Relation zu semantischen Beziehungen zwischen Nutzern gestellt. Exemplarisch zeigt Abbildung 2 das Ergebnis zur Untersuchung des Zusammenhangs zwischen Interaktionshäufigkeit in BibSonomy und einem semantisch motivierten Ähnlichkeitsmaß der jeweiligen Nutzerpaare.

[1] Die Nutzernamen sind frei erfunden und wurden mittels der im zweiten Teil der Arbeit vorgestellten Vornamenssuchmaschine *Nameling* ausgewählt.

[2] In Twitter kann ein Nutzer über eine *Follower*-Relation den Status-Updates eines anderen Nutzers "folgen", d.h. über neue Einträge informiert werden.

[3] Ein ReTweet in Twitter ist eine (in Auszügen) wiedergegebene Status-Nachricht eines anderen Nutzers.

[4] http://twitter.com

[5] http://flickr.com

[6] http://www.bibsonomy.org

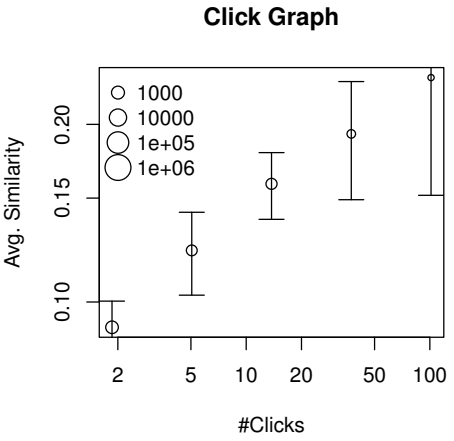


Abbildung 2: Zusammenhang zwischen Interaktionshäufigkeit und Ähnlichkeit von Nutzern in BibSonomy: Je mehr Informationen eines anderen Nutzers angeklickt wurden, desto ähnlicher sind sich in der Tendenz die beteiligten Nutzerpaare.

Konkret zeigt die Abbildung Ergebnisse für den *Click*-Graphen in BibSonomy. Dieser enthält eine gerichtete Kante von Nutzer u zu Nutzer v, falls u mit der Maus auf einen Eintrag von v in BibSonomy geklickt hat [7]. Zur Bestimmung der Ähnlichkeit zwischen zwei Nutzern in BibSonomy werden Nutzer anhand der von Ihnen verwendeten Schlagworte (Tags) dargestellt. Hierzu wird ein Nutzer u durch einen m-dimensionalen Vektor $\vec{u} \in \mathbb{N}^m$ repräsentiert, wobei m der Anzahl der verschiedenen in BibSonomy verwendeten Schlagworte bezeichnet und u_i komponentenweise angibt, wie häufig der Nutzer u das i-te Schlagwort verwendet hat ($i = 1, \ldots, n$). Als Maß für die Ähnlichkeit zwischen zwei Nutzern u und v wird die *Kosinus-Ähnlichkeit*

$$\mathrm{COS}(u, v) := \frac{\vec{u} \cdot \vec{v}}{\|\vec{u}\|\|\vec{v}\|} = \frac{\sum_{i=1}^{m} u_i v_i}{\sqrt{\sum_{i=1}^{m} (u_i)^2} \sqrt{\sum_{i=1}^{m} (v_i)^2}}$$

zwischen deren entsprechenden Vektorrepräsentationen verwendet, motiviert durch entsprechende Verwendung der Kosinus-Ähnlichkeit im Bereich Distributional Semantics für Tagging-Systeme [CBHS08].

In Abbildung 2 wird die *durchschnittliche paarweise Kosinus-Ähnlichkeit* relativ zur Interaktionshäufigkeit (d.h. wie oft hat Nutzer u auf Einträge eines anderen Nutzers v geklickt) dargestellt, wobei die Interaktionshäufigkeiten in Bins mit exponentiell wachsender Bin-Größe zusammengefasst wurden, bedingt durch die Heavy-tailed-Verteilung der Interaktionshäufigkeiten. In der Abbildung zeigt sich die Tendenz, dass eine gesteigerte Interaktionshäufigkeit in BibSonomy mit einer höheren Nutzerähnlichkeit einhergeht.

In Abbildung 3 wird schließlich für BibSonomy's Click-Graphen der Zusammenhang zwischen der Ähnlichkeit von Interaktionscharakteristiken und der Ähnlichkeit von Nutzern

[7] In BibSonomy verwalten Nutzer Internet-Lesezeichen sowie Publikationsmetadaten und stellen diese anderen Nutzern zur Verfügung. Ein Nutzer kann in BibSonomy z.B. per Mausklick Publikationsdetails anzeigen [BHJ+10].

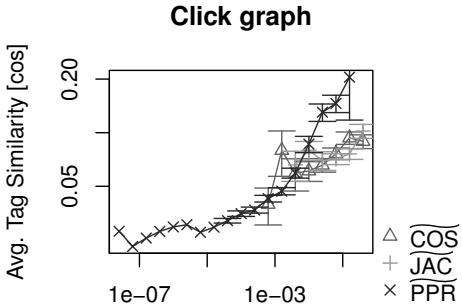


Abbildung 3: Zusammenhang zwischen der Ähnlichkeit von Interaktionscharakteristiken und Ähnlichkeit von Nutzern in BibSonomy: Je ähnlicher sich zwei Nutzer strukturell im betrachteten Interaktionsnetzwerk sind, desto ähnlicher sind sich in der Tendenz die beteiligten Nutzerpaare. Betrachtet wurden die gewichteten Varianten der Kosinus-Ähnlichkeit $\widetilde{\text{COS}}$, Jaccard-Koeffizient $\widetilde{\text{JAC}}$ [BYRN+99] und PageRank mit Präferenz $\widetilde{\text{PPR}}$ [HJSS06b].

anhand deren strukturellen Ähnlichkeit im entsprechenden Evidenznetzwerk betrachtet. Dabei wird die strukturelle Ähnlichkeit von Nutzern anhand von Nachbarschaftsbasierten Maßen (Kosinus-Ähnlichkeit bzw. Jaccard-Ähnlichkeit der entsprechenden Spalten in der Adjazenzmatrix des Graphens) sowie der PageRank-Ähnlichkeit mit Präferenz [HJSS06b] bestimmt, welche durch die globale Graphstruktur beeinflusst ist. Auch bestätigt sich eine positive Korrelation zwischen der strukturellen Ähnlichkeit im Click-Graphen sowie der semantisch motivierten Tag-Ähnlichkeit von Nutzern in BibSonomy.

Bewertung von Nutzergruppenstrukturen Die Social Distributional Hypothesis dient als Arbeitshypothese zur Unterstützung der Analyse von Beziehungsstrukturen zwischen Nutzern, basierend auf beobachteten Interaktionsdaten. Dabei ist die Untersuchung und Bewertung von Benutzergruppen (*Communities*) [MAB+11] von besonderer praktischer Relevanz, da in vielen Fällen eine Referenz zur Beurteilung der Güte einer Einteilung von Nutzern in Gruppen fehlt [FC07].

Hierzu wird ein methodischer Ansatz zur vergleichenden Bewertung von verschiedenen Benutzergruppenmodellen vorgestellt und ausführlich analysiert. Die Grundidee des vorgestellten Ansatzes basiert auf der Bewertung einer Benutzergruppe relativ zur Community-Struktur innerhalb eines beobachteten Interaktionsnetzwerks, wie in Abbildung 4 schematisch dargestellt: Ist eine Einteilung der Nutzermenge einer Anwendung des Social Web gegeben (z.B. als Ergebnis der Anwendung eines Inhaltsbasierten Clusterverfahrens auf Merkmalen der Nutzer, wie z.B. die Tag-Clouds der Nutzer in BibSonomy), so wird dies als Graph-Clustering eines Evidenznetzwerks übertragen, welches aus Daten des laufenden Betriebs der betrachteten Anwendung extrahiert wurde. Die Güte dieses Graph-Clusterings wird anhand etablierter Maße zur Bewertung von Communitystrukturen in Graphen, insbesondere Modularity [New06] bestimmt.

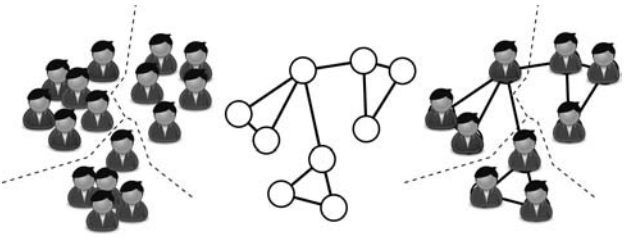


Abbildung 4: Illustration des Ansatzes zur Bewertung von Nutzergruppen mit Hilfe von Evidenz-Netzwerken

3 Onomastik 2.0

Im zweiten Teil werden die gleichen methodischen Ansätze der Untersuchungen zur Social Distributional Hypothesis auf Fragestellungen eines weiteren nicht-technischen Fachgebiets, der *Namenforschung*, übertragen. Die Namenforschung ist eine interdisziplinäre Wissenschaft [Zie11], eingebettet zwischen Sprachwissenschaften, Soziologie und Psychologie. Dabei ist die Untersuchung der Enwicklung und Verbreitung von Personennamen eine klassische Problemstellung der Namenforschung. Aus methodischer Sicht unterscheidet sich dabei die Untersuchung von *historischen* Entwicklungen und Zusammenhängen von der Untersuchung *aktueller* Entwicklungen und deren Einflussfaktoren.

Mit den Daten des Social Web steht der Namenforschung eine neue Basis zur Verfügung, die es ermöglicht, zeitliche und räumliche Dynamiken von Namen in *Echtzeit* zu beobachten und auszuwerten. Aktuelle Entwicklungen, die sich ansonsten nur schwer aus der Beobachtung heraus belegen lassen, können nun messbar gemacht und quantifiziert werden.

Auch zur Untersuchung von Namensbeziehungen wird ein graphentheoretischer Ansatz verfolgt. Hierzu werden, basierend auf gesammelten Daten aus Wikipedia und Twitter, Namensbeziehungsnetzwerke konstruiert und analysiert. In Anlehnung an die Distributional Hypothesis, werden Kookkurrenznetzwerk für Vornamen erzeugt, basierend auf gemeinsamen Vorkommen in Sätzen in Artikeln der Wikipedia, bzw. in Tweets in Twitter. Für die Untersuchungen zu Namensbeziehungsnetzwerken, werden für Wikipedia die Englische, Deutsche und Französische Sprachvariante separat betrachtet.

Strukturelle Beziehungen zwischen Namen in diesen Netzwerken werden mit semantischen Ähnlichkeiten zwischen Namen in Relation gesetzt. Eine semantische Ähnlichkeit zwischen Vornamen wird dabei basierend auf der Kategorisierung von Vornamen in Wiktionary [8] bestimmt, insbesondere werden Vornamen als binäre Vektoren dargestellt, deren Komponenten jeweils einer möglichen Kategorie in Wiktionary entsprechen und angeben, ob ein Name in Wiktionary enstprechend kategorisiert wurde. Die Ähnlichkeit

[8] http://www.wiktionary.org

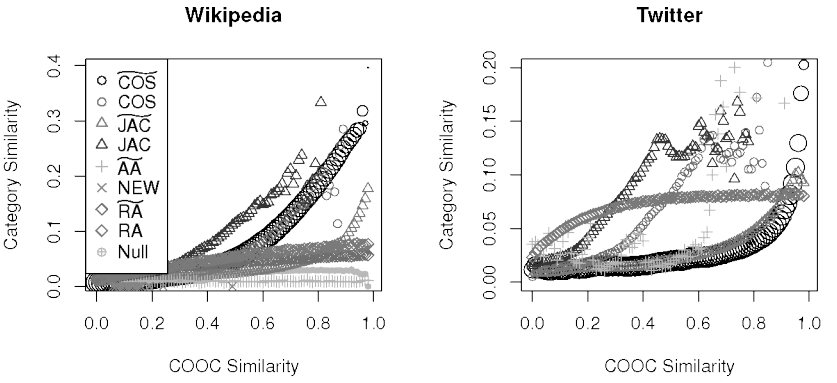


Abbildung 5: Vergleich verschiedener Maße für *strukturelle Ähnlichkeit* von Vornamen mit deren *semantischen* Ähnlichkeit basierend auf der Kategorisierung von Vornamen in Wiktionary.

zwischen Namen wird dann wieder als (ungewichtete) Kosinus-Ähnlichkeit zwischen den entsprechenden Kategorie-Vektoren berechnet.

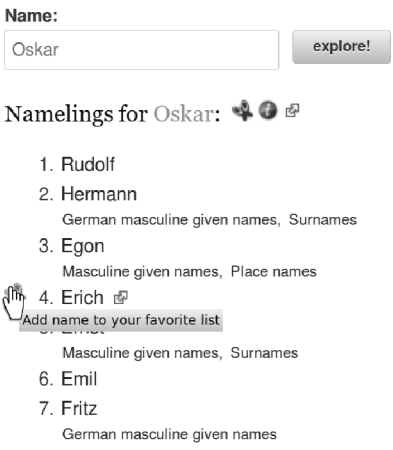
Hierzu zeigt Abbildung 5 die Ergebnisse für verschiedene Maße struktureller Ähnlichkeit in den Kookkurrenznetzwerken aus Wikipedia und Twitter im Vergleich zu dem oben beschriebenen Maß für semantische Ähnlichkeit zwischen Vornamen. Insbesondere für die ungewichtete Jaccard-Ähnlichkeit, sowie der gewichteten Kosinus-Ähnlichkeit zwischen der Nachbarschaft von Vornamen im Wikipedia-basierten Kookkurrenznetzwerk zeigt sich eine positive Korrelation mit der betrachteten semantischen Ähnlichkeit von Vornamen. In den aus Twitter extrahierten Vornamensnetzwerken lassen sich ähnliche Korrelationen beobachten, jedoch weniger stark ausgeprägt und stabil.

Zusätzlich wird eine semantische Ähnlichkeit von Städtenamen basierend auf der paarweisen geographischen Distanz betrachtet. Im Ergebnis zeigt sich ein Zusammenhang zwischen Struktureller Ähnlichkeit von Namen in Wikipedia-basierten Netzwerken und den betrachteten semantischen Namensähnlichkeiten.

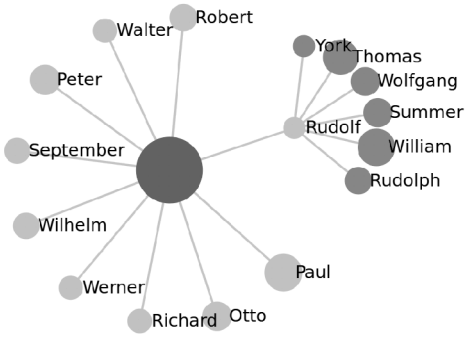
Im Rahmen der Arbeit wurden diese Ergebnisse praktisch in Form der Vornamensuchmaschine Nameling [9] [MS12] umgesetzt, welche es erlaubt, gezielt nach passenden Vornamen zu suchen und in einer persönlichen Favoritenliste zu verwalten. Diese Favoritenliste kann auch mit Freunden geteilt werden, um z.B. gemeinsam nach einem Namen für das eigene Kind zu suchen.

Zusätzlich werden aus Wikipedia extrahierte Attribute von Namen angezeigt und zur Navigation zur Verfügung gestellt, um z.B. nach allen skandinavischen Vornamen zu suchen. Weiterhin werden popularitätswerte basierend auf Auszählungen von Twitter, Wikipedia und den eigenen Zugriffsstatistiken angezeigt (siehe Abbildung 6).

[9] http://nameling.net



(a) Ähnliche Namen



(b) Häufige Kookkurrenzen

Abbildung 6: Benutzerinterface von Nameling bei der Suche nach dem Namen "Oskar".

Tabelle 1: Ähnliche Namen für den Vornamen "Kevin", berechnet auf Wikipedias Kookkurrenznetzwerk (links) und basierend auf den Nutzungsdaten in Nameling (rechts).

Wikipedia	Nameling
Tim	Chantal
Danny	Justin
Jason	Jaqueline
Jeremy	Sky
Nick	Chantalle

Innerhalb der ersten sechs Monate wurde Nameling von über 35.000 Anwendern genutzt. Die dabei angefallenen Nutzungsdaten werden im Rahmen dieser Arbeit ausgewertet und in Beziehung zu den betrachteten Namensbeziehungsnetzwerken gesetzt. Insbesondere wird dabei untersucht, in wie weit die betrachteten strukturellen Ähnlichkeitsmaße für Namen dem Suchverhalten der Nutzer entsprechen und wie mehrere Namen zu einem Suchvektor verknüpft werden können. Zwar kann dabei festgestellt werden, dass einzelne Ähnlichkeitsmaße bessere Ergebnisse erzielen, jedoch sind die erreichten Ergebnisse im Sinne der im Bereich *Information Retrieval* üblichen Maße sehr niedrig.

Dies deutet an, dass Nutzer einer Vornamensuchmaschine nicht entsprechend semantisch klar definierter Ähnlichkeiten von Vornamen (wie sie sich aus Wikipedia ergeben) suche, sondern entsprechend persönlicher Präferenzen. Hierzu zeigt Tabelle 1 exemplarisch die ähnlichsten Vornamen zu "Kevin", basierend auf Wikipedias Kookkurrenznetzwerk sowie basierend auf einem Vornamensnetzwerk welches aus den Nutzungsdaten in Nameling extrahiert wurde. Während entsprechend der Wikipedia-basierten Ähnlichkeit weitere im amerikanischen Sprachraum verbreitete, kurze männliche Vornamen aufgelistet werden, besteht die Nameling-basierte Liste aus sowohl englischen und französischen, so-

wie männlichen und weiblichen Vornamen. Jedoch besteht zwischen diesen semantisch verschiedenen Vornamen ein Zusammenhang, entsprechend der sozio-onomastischen Assoziation von Vornamen im Deutschen Sprachraum des betrachteten Beobachtungszeitraums.

Entsprechend werden personalisierte Vornamensempfehlungen als neues Anwendungsgebiet für *Empfehlungssysteme* vorgestellt. Hierzu werden zunächst grundlegende Evaluierungsprotokolle definiert, welche darauf abzielen, unterschiedliche Aspekte der Qualität von Vornamensempfehlungssystemen zu untersuchen. Verschiedene etablierte Empfehlungssysteme werden angewendet, evaluiert und verglichen. Zusätzlich wird mit NameRank ein neues Empfehlungsverfahren als Adaption des am Fachgebiet Wissensverarbeitung der Universität Kassel entwickelten FolkRanks [HJSS06a] vorgestellt und bewertet. Im Vergleich zu den anderen betrachteten Verfahren zeigt NameRank einen guten Kompromiss zwischen Laufzeiteigenschaften und Empfehlungsgüte.

Literatur

[BHJ+10] Dominik Benz, Andreas Hotho, Robert Jäschke, Beate Krause, Folke Mitzlaff, Christoph Schmitz und Gerd Stumme. The social bookmark and publication management system BibSonomy. *The VLDB Journal*, 19(6):849–875, 2010.

[BYRN+99] Ricardo Baeza-Yates, Berthier Ribeiro-Neto et al. *Modern information retrieval*, Jgg. 463. ACM press New York, 1999.

[CBHS08] Ciro Cattuto, Dominik Benz, Andreas Hotho und Gerd Stumme. Semantic Grounding of Tag Relatedness in Social Bookmarking Systems. In Amit P. Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy W. Finin und Krishnaprasad Thirunarayan, Hrsg., *The Semantic Web – ISWC 2008, Proc.Intl. Semantic Web Conference 2008*, Jgg. 5318 of *LNAI*, Seiten 615–631, Heidelberg, 2008. Springer.

[CGB+12] Rosaria Conte, Nigel Gilbert, Giulia Bonelli, Claudio Cioffi-Revilla, Guillaume Deffuant, Janos Kertesz, Vittorio Loreto, Suzy Moat, J-P Nadal, Anxo Sanchez et al. Manifesto of computational social science. *The European Physical Journal Special Topics*, 214(1):325–346, 2012.

[CKK13] Ray M. Chang, Robert J. Kauffman und YoungOk Kwon. Understanding the paradigm shift to computational social science in the presence of big data. *Decision Support Systems*, 2013. in press.

[FC07] Santo Fortunato und Claudio Castellano. Community Structure in Graphs, 2007. arxiv:0712.2716.

[Har54] Zellig S Harris. Distributional structure. *Word*, 1954.

[HJSS06a] Andreas Hotho, Robert Jäschke, Christoph Schmitz und Gerd Stumme. Information Retrieval in Folksonomies: Search and Ranking. In York Sure und John Domingue, Hrsg., *The Semantic Web: Research and Applications*, Jgg. 4011 of *Lecture Notes in Computer Science*, Seiten 411–426, Heidelberg, Juni 2006. Springer.

[HJSS06b] Andreas Hotho, Robert Jäschke, Christoph Schmitz und Gerd Stumme. Information Retrieval in Folksonomies: Search and Ranking. In York Sure und John Domingue,

Hrsg., *The Semantic Web: Research and Applications*, Jgg. 4011 of *Lecture Notes in Computer Science*, Seiten 411–426, Heidelberg, Juni 2006. Springer.

[LPA+09] David Lazer, Alex Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, Tony Jebara, Gary King, Michael Macy, Deb Roy und Marshall Van Alstyne. Computational Social Science. *Science*, 323(5915):721–723, 2009.

[MAB+11] Folke Mitzlaff, Martin Atzmueller, Dominik Benz, Andreas Hotho und Gerd Stumme. Community Assessment using Evidence Networks. In *Analysis of Social Media and Ubiquitous Data*, Jgg. 6904 of *LNAI*, Seiten 79–98, 2011.

[MC03] Peter Monge und Noshir Contractor. *Theories of Communication Networks*. Oxford University Press, New York, Oxford / New York, 2003.

[MR01] Scott McDonald und Michael Ramscar. Testing the distributional hypothesis: The influence of context on judgements of semantic similarity. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society*, Seiten 611–616, 2001.

[MS12] Folke Mitzlaff und Gerd Stumme. Namelings - Discover Given Name Relatedness Based on Data from the Social Web. In Karl Aberer, Andreas Flache, Wander Jager, Ling Liu, Jie Tang und Christophe Guéret, Hrsg., *SocInfo*, Jgg. 7710 of *Lecture Notes in Computer Science*, Seiten 531–534. Springer, 2012.

[MSLC01] Miller McPherson, Lynn Smith-Lovin und James M. Cook. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology*, 27:pp. 415–444, 2001.

[New06] MEJ Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[Zie11] Arne Ziegler. *Methoden der Namenforschung : Methodologie, Methodik und Praxis*. Akademie Verlag Berlin, Berlin, 2011.

Folke Mitzlaff wurde 1979 in Hamburg geboren und studierte Informatik mit Nebenfach Mathematik und Schwerpunkt Theoretische Informatik an der *Technischen Universität Ilmenau*. Er studierte zwei Semester am *Moskauer Energetischem Institut* am Lehrstuhl für angewandte Mathematik.
Bereits mit der Diplomarbeit zum Thema "*Statische Analyse der Komplexität von Programmen*" wandte er graphentheoretische Konzepte als Werkzeug für die betrachteten Fragestellungen an.
Nach dem Studium arbeitete er als wissenschaftlicher Mitarbeiter in der Fachgruppe für Mathematische Logik und Theoretische Informatik der Universität Siegen, wechselte aber im Jahr 2009 zur Fachgruppe "*Knowledge and Data Engineering*" der Universität Kassel, wo er sich insbesondere mit der Analyse von Netzwerkstrukturen in sozialen Webapplikationen beschäftigte. Im Rahmen seiner Arbeit entstand dabei unter anderem die Vornamenssuchmaschine "*Nameling* " in deren Kontext auch die 15. ECML PKDD Discovery Challenge zum Thema Vornamensempfehlungssysteme organisiert wurde.
Im Jahr 2013 verteidigte er seine Dissertation "*Relatedness in Evidence Networks*" am Fachbereich Elektrotechnik/Informatik der Universität Kassel und arbeitet seitdem als Entwicklungsingenieur im Bereich erneuerbare Energien.

Rendern von Unterteilungsflächen mittels Hardware Tessellierung*

Matthias Nießner

Friedrich-Alexander-Universität Erlangen-Nürnberg
niessner@cs.stanford.edu

Abstract: Computergenerierte Bilder haben sich längst zu einem festen Bestandteil unseres alltäglichen Lebens entwickelt. Neben Computerspielen und Filmen sind synthetische Bilder auch ein wesentlicher Bestandteil von Illustrationen in Zeitschriften und Plakaten - meist ohne, dass wir es bewusst wahrnehmen. Die Qualität erzeugter Bilder hängt dabei maßgeblich von der verwendeten Geometriebeschreibung der zugrunde liegenden 3D Umgebungen ab. Unterteilungsflächen (*Subdivision Surfaces*) haben sich hierbei aufgrund ihrer besonderen Oberflächeneigenschaften als sehr nützlich erwiesen. Letztendlich haben sich Unterteilungsflächen wegen hochwertiger Resultate - speziell in der Filmbrache - als absoluter Industriestandard durchgesetzt und kommen heutzutage in nahezu allen Produktionen zum Einsatz. Ein gravierender Nachteil ist allerdings die Komplexität zugrunde liegender Berechnungen, was entsprechend lange Rechenzeiten zur Folge hat. In dieser Dissertation befassen wir uns mit Algorithmen, welche die Auswertung dieser Oberflächen um mehrere Größenordnungen beschleunigen. Dadurch können hochwertige Filminhalte binnen weniger Millisekunden auf handelsüblichen Computern dargestellt werden und somit in Echtzeitanwendungen, wie Spielen, eingesetzt werden. Die Ergebnisse dieser Arbeit wurden unter anderem im Rahmen des OpenSource Projekts *OpenSubdiv* in Zusammenarbeit mit Pixar Animation Studios veröffentlicht und finden bereits breite Anwendung in der Industrie, wie zum Beispiel in der Modellierungssoftware von Branchenprimus Autodesk (z.B. Maya, Mudbox) und anderer Hersteller. Darüber hinaus werden unsere Algorithmen in Computerspielen wie *Call of Duty: Ghosts* von Activision verwendet, die dadurch die Oberflächenqualität von Filmen erreichen.

Abbildung 1: Unsere Algorithmen ermöglichen es qualitativ hochwertige Filminhalte auf handelsüblichen Computern in Echtzeit darzustellen. Die Abbildung zeigt wie unser adaptiver Unterteilungsalgorithmus das Automodell aus *Pixar's Cars* in Echtzeit verarbeitet. Die Grundidee dabei ist es, die Oberfläche - falls möglich - analytisch auszuwerten und nur dort wo es auch wirklich nötig ist, aufwändige Unterteilungen zu berechnen. Im linken Bild sehen wir die Eingabedaten, welche die Oberfläche definieren. In der Mitte werden die adaptiven Unterteilungen dargestellt (eine Farbe je Unterteilungslevel), und rechts sehen wir das Ergebnisbild. © Disney/Pixar

*Englischer Titel der Dissertation: "Rendering Subdivision Surfaces using Hardware Tessellation" [Nie13]

1 Einführung

Computergenerierte Bilder werden immer wichtiger im alltäglichen Leben. Mit steigender Relevanz erhöhen sich auch die Anforderungen, die an die Bildqualität gestellt werden. Um eine möglichst wirklichkeitsgetreue Abbildung zu garantieren, benötigen Bilder deshalb zunehmend mehr visuelles Detail. Dieses Detail basiert dabei maßgeblich auf der Oberflächenrepräsentation der zugrunde liegenden Szenengeometrie. Während in der Echtzeitgrafik vor allem Dreiecksnetze etabliert sind, haben sich in der Filmbrache Unterteilungsflächen (*Subdivision Surfaces* [CC78]) zu einem Industriestandard entwickelt (cf. Abbildung 1, 2). Unterteilungsflächen bieten Grafikdesignern im Gegensatz zu polygonbasierten Ansätzen einen weit höheren Grad an Flexibilität beim Modellieren und liefern dank ihrer Oberflächeneigenschaften sehr hochwertige Bildergebnisse.

Ein gravierender Nachteil von Unterteilungsflächen ist die einhergehende Rechenkomplexität beim Auswerten dieser Oberflächendarstellung. Daher werden in Filmproduktionen typischerweise teure Hochleistungsrechner mit entsprechender Rechenleistung zur Bildgenerierung (*Rendering*) eingesetzt. In Echtzeitanwendungen wie Computerspielen oder Modellierungssoftware, müssen allerdings mehrere Bilder pro Sekunde generiert werden (mindestens 30), was die Anwendung von hochwertigen Filminhalten deutlich erschwert.

Abbildung 2: Generiertes Bild von *Woody* aus *Pixar's Toy Story* mittels unserem Verfahren. Dank unserer neuartigen Technik können Bilder aus Filmszenen in unter 10 Millisekunden auf handelsüblichen Computern gerendert werden. © Disney/Pixar

Im Rahmen dieser Dissertation haben wir daher Verfahren entwickelt, um Inhalte bestehend aus Unterteilungsflächen, in Echtzeitanwendungen auf handelsüblichen Desktop-Computern einzusetzen. Wir greifen dabei auf die Rechenleistung moderner Grafikkarten zurück und entwickeln unsere Algorithmen speziell für die massiv parallele GPU Prozessorarchitektur. Neben der hohen Prozessoranzahl, besitzen Grafikkarten außerdem eine Hardware Tessellierungseinheit, welche besonders beim Rendern von parametrischen Flächenstücken nützlich ist. Der entscheidende Vorteil der Hardware Tessellierung liegt darin, dass die Oberflächengeometrie auf den jeweiligen Berechnungseinheiten ausgewertet und direkt weiterverarbeitet wird. Dadurch können Oberflächen dynamisch (bzgl. der Kamera) verfeinert und resultierende Dreiecke direkt und ohne zusätzliche Speicherzugriffe rasterisiert werden. Dies ist auf parallelen Plattformen wie Grafikkarten entscheidend, da Speicherbandbreite und Latenz typischerweise die Leistungsfähigkeit limitieren. Mit der Tessellierungseinheit lassen sich Objekte außerdem

sehr einfach animieren, da lediglich die Kontrollpunkte der entsprechenden Flächenstücke angepasst werden müssen.

Ein entscheidender Beitrag der Arbeit ist die (parallele) Formulierung unserer Algorithmen, welche es erlauben die Fähigkeiten der Tessellierungseinheit optimal zu nutzen. Wir zerlegen daher Unterteilungsflächen zunächst in einzelne Flächenstücke, welche anschließend vom Tessellierer verarbeitet werden. Entgegen der naiven rekursiven Definition von Unterteilungsflächen, können wir dadurch die Oberfläche direkt auswerten, was auf Grafikkarten weitaus schneller und speicherfreundlicher ist. Darüber hinaus kann unser Verfahren effizient hierarchisch definiertes Oberflächendetail (*hierarchical edits*) und halbweiche Knicke (*semi-sharp creases*) benutzen und anwenden. Um feine und hochfrequente Oberflächeneigenschaften darzustellen, führen wir außerdem eine analytische Verschiebungsfunktion (*Displacements*) ein, die Oberflächennormalen implizit definiert. Im Gegensatz zu traditionellen Repräsentationen wird dadurch der Speicherbedarf massiv reduziert, was zu einer deutlich verbesserten Renderzeit führt. Unsere Repräsentation erlaubt es zudem Oberflächendetail dynamisch zu verändern ohne zusätzliche Berechnungskosten zu verursachen. Das macht unser Verfahren sowohl schneller als auch deutlich flexibler - und vor allem praktikabel. Bei der Bildgenerierung werden allerdings immer noch viele unnötige Berechnungen durchgeführt. Beispielsweise werden viele Flächenstücke ausgewertet, obwohl sie verdeckt und nicht sichtbar sind. Um dies zu vermeiden, identifizieren wir Flächenstücke die zum einen von der Kamera abgewandt sind und zum anderen durch andere Flächenstücke verdeckt sind [NL12]. Diese werden anschließend aus der Renderliste entfernt und müssen nicht mehr weiter verarbeitet werden. Der Berechnungsaufwand kann dadurch signifikant reduziert werden und Renderzeiten um effektiv die Hälfte gesenkt werden. Wir befassen uns ebenso mit der Kollisionserkennung für die Hardware Tessellierung in Echtzeitanwendungen [NSSL13]. Speziell bei Unterteilungsflächen mit Verschiebungsfunktionen und dynamischer Tessellierung ist dies sehr anspruchsvoll. Im Rahmen dieser Kurzfassung der Dissertation legen wir allerdings den Fokus auf das Auswerten und Anzeigen (siehe Abschnitt 3) von Unterteilungsflächen, und auf die Repräsentation von feinem Oberflächendetail (siehe Abschnitt 4).

Letztlich wurde in der zugrundeliegenden Dissertation eine umfassende Lösung - bestehend aus zahlreichen neuartigen Algorithmen - entwickelt, um Unterteilungsflächen erstmals in Echtzeitanwendungen einzusetzen. Die Forschungsergebnisse bilden die Grundlage von Pixar's OpenSource Projekt *OpenSubdiv* und finden bereits - ein Jahr nach Veröffentlichung - breite Anwendung in der Industrie (z.B. Autodesk Maya & Mudbox, Call of Duty Ghosts, etc.). Siehe Abbildungen 1, 2, 8.

2 Grundlagen: Unterteilungsflächen und Moderne Grafikhardware

Unterteilungsflächen In folgendem Abschnitt geben wir eine kurze Einführung zu Unterteilungsflächen, um das Verständnis unserer Algorithmen zu erleichtern. Unterteilungsflächen wurden von Catmull und Clark [CC78] eingeführt und sind eine Verallgemeinerung von Tensorproduktflächen, welche eine glatte Grenzfläche definieren. Wir nehmen dabei hauptsächlich auf Catmull-Clark Oberflächen Bezug, welche bi-kubische B-

Splineflächen auf Topologien beliebiger Konnektivität erweitern und ein reines Quadmesh nach einem Unterteilungsschritt erzeugen. Neben Catmull-Clark Flächen gibt es noch eine Vielfalt anderer Unterteilungsverfahren, die zwar von unseren Algorithmen verarbeitet werden können, allerdings in der Praxis eine untergeordnete Rolle einnehmen. Die Grundidee von Unterteilungsflächen besteht darin, ein grobes Basismesh iterativ mittels Unterteilungsregeln zu verfeinern. Diese Regeln beschreiben Linearkombinationen der Kontrollpunkte des aktuellen Unterteilungslevels i, woraus sich die Kontrollpunkte des nächst feineren Levels $i+1$ ableiten lassen. Die glatte Grenzfläche ergibt sich nach einer unendlichen Anzahl von Unterteilungen und ist C^2, abgesehen von außergewöhnlichen Gitterpunkten (Vertices mit Valenz ungleich 4), wo sie C^1 ist. Die zugrundeliegenden Catmull-Clark Unterteilungsregeln sind definiert durch Flächenpunkte (f_j), Kantenpunkte (e_j) und Vertexpunkte (v_j) mit jeweiliger Valenz n (entsprechend in Abbildung 3 gekennzeichnet). Die zugehörigen Regeln sind gewichtete Mittelwerte des vorherigen Unterteilungslevels:

- Flächenregel: f^{i+1} ist der Schwerpunkt der Vertices der umgebenden Fläche,
- Kantenregel: $e_j^{i+1} = \frac{1}{4}(v^i + e_j^i + f_{j-1}^{i+1} + f_j^{i+1})$,
- Vertexregel: $v^{i+1} = \frac{n-2}{n} v^i + \frac{1}{n^2} \sum_j e_j^i + \frac{1}{n^2} \sum_j f_j^{i+1}$.

Es gibt zudem zahlreiche Erweiterungen zu Catmull-Clark Flächen, wie harte Kanten, halbweiche Knicke und hierarchisch definiertes Oberflächendetail. Diese können problemlos von unseren Algorithmen verarbeitet werden, allerdings gehen wir hier im Rahmen dieser Kurzfassung nicht näher darauf ein.

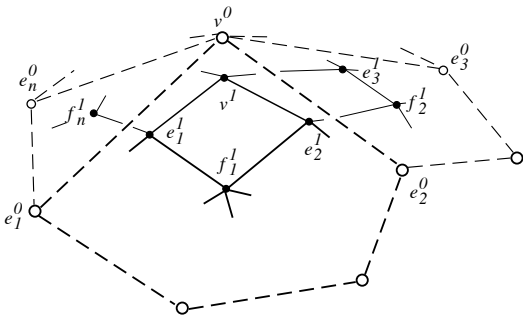


Abbildung 3: Kennzeichnung von Vertices des Basismeshes (Unterteilungslevel 0) um den Punkt v^0 mit Valenz n und des nächsten Unterteilungslevels 1.

Moderne Grafikhardware Moderne Grafikkarten sind massiv parallele Prozessorarchitekturen. Sie bestehen aus mehreren *Streaming Multiprozessoren* (SMs), wobei jeder SM eine eigene Vektoreinheit ist und somit viele Daten mit gleicher Instruktion parallel verarbeiten kann, *Single Instruction, Multiple Data* (SIMD). Im Rahmen dieser Arbeit verwenden wir eine NVIDIA GTX 480, die aus 15 SMs mit einer SIMD-Breite von 32 besteht und folglich 480 Threads parallel ausführen kann. Im Verhältnis zu CPUs, wird bei GPUs weit mehr Chipfläche für Berechnungseinheiten als für Zwischenspeicher verwendet. Dadurch steht zwar sehr viel Rechenleistung zur Verfügung, allerdings ist die Speicherbandbreite typischerweise ein Flaschenhals. Daher ist es sehr wichtig dies im Entwurf unserer Algorithmen zu berücksichtigen. Dies kann speziell mit der Hardware Tessellierungseinheit erreicht werden, welche seit der Einführung der XBox 360 auf moderner Grafikhardware verfügbar ist. Dabei werden einzelne Flächenstücke (*Patches*) jeweils durch ein unabhängiges Set von Kontrollpunkten definiert, parallel ausgewertet und gerendert. Die Patch Kontrollpunkte

werden zunächst von der programmierbaren Hullshadereinheit (je 1 Thread pro Kontrollpunkt) verwendet um sogenannte Tessellierungsfaktoren zu bestimmen (je Patchkante und Patchinnenfläche), welche die Tessellierungsdichte bestimmen. Die Hardware generiert anschließend für jeden Patch Abtastpunkte mit uv-Patchparameterwerten, an welchen die Oberfläche ausgewertet werden muss - dies geschieht in der programmierbaren Domainshadereinheit. Die Herausforderung dabei ist es die Oberfläche *direkt* auszuwerten, was vor allem bei Unterteilungsflächen durch die rekursive Definition sehr anspruchsvoll ist (speziell an außergewöhnlichen Gitterpunkten).

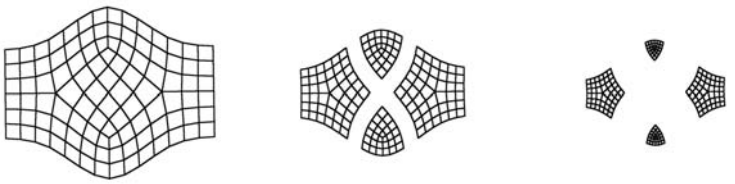


Abbildung 4: Unser adaptives Unterteilungsverfahren angewandt auf einem Kontrollgitter mit vier außergewöhnlichen Gitterpunkten. Wir unterteilen nur in Bereichen in welchen direkte Auswertung nicht möglich ist; i.e., an außergewöhnlichen Punkten.

3 Adaptive Unterteilung von Unterteilungsflächen auf Moderner Grafikhardware

Die Grundidee adaptiver Unterteilung ist es nur dort zu unterteilen, wo es auch wirklich nötig ist. Bei Catmull-Clark Flächen kann daher ausgenutzt werden, dass alle regulären Patches als bi-kubische B-Splines definiert sind. Diese können an beliebigen Parameterstellen ausgewertet und somit direkt vom Hardware Tessellierer verarbeitet werden. Alle anderen Flächen konvertieren wir durch sukzessives Unterteilen in (großteils) reguläre Patches, um sie direkt auswerten zu können. Unser adaptives Unterteilungsverfahren, unterteilt daher zunächst nicht-reguläre Regionen mittels GPGPU-Kernel. Anschließend werden alle Teilflächen so zusammengefügt und mit dem Hardware Tessellierer gerendert, dass dabei keine Löcher entstehen. Das Verfahren ist in Abbildung 1 visualisiert.

3.1 Datenparallele Unterteilung basierend auf Tabellen

Unterteilungsflächen können anhand ihrer rekursiven Definition sehr einfach auf der CPU ausgewertet werden. Eine effiziente GPU Auswertung ist dagegen schwieriger, da Berechnungen parallelisiert werden müssen und zugehörige Nachbarschaftsinformationen benötigt werden. Um dies zu vereinfachen, nehmen wir an, dass die Konnektivität der Unterteilungsfläche statisch ist. Dadurch kann unser Algorithmus vorberechnete Tabellen verwenden, in welchen die Unterteilungsregeln der jeweiligen Kind-Vertices bezüglich ihrer Eltern kodiert sind. Zur Laufzeit müssen nur noch die entsprechenden Linearkombinationen mit den Vertexdaten des vorausgehenden Unterteilungslevels angewandt werden. Dazu verwenden wir drei GPGPU-Kernel je Unterteilungslevel - jeweils für Flächen, Kanten und Vertices - die auf einem gemeinsamen Vertexpuffer arbeiten und einen Thread pro Ausga-

bevertex ausführen. Die Kernel werden ausgehend vom Basismesh sukzessive ausgeführt. Dadurch werden Vertexupdates entsprechend vom gröbsten zum feinsten Unterteilungslevel propagiert (z.B. unter Animation). In den Unterteilungstabellen sind außerdem Features wie zum Beispiel Meshgrenzen und halbweiche Knicke, kodiert. Für weiter Details bezüglich des Tabellenaufbaus verweisen wir auf die entsprechende Veröffentlichung [NLMD12]. Unsere kompakte Darstellung der Unterteilungsregeln ist das derzeit schnellste GPU Unterteilungsverfahren.

3.2 Feature-adaptive Unterteilung

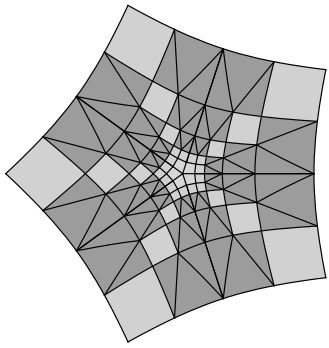


Abbildung 5: Die Patchanordnung um einen außergewöhnlichen Gitterpunkt. Parametergebiete von UPs sind rot, reguläre VPs sind blau, und irregulare VPs sind grün (Zentrum) gekennzeichnet.

Die uniforme Unterteilung einer Catmull-Clark Oberfläche, wie im vorherigen Abschnitt beschrieben, kann zwar in Echtzeit auf Grafikkarten ausgeführt werden, ist jedoch relativ aufwändig. Vor allem der Speicherbedarf wächst exponentiell mit der Anzahl der Unterteilungslevel. Um dies zu vermeiden, unterteilen wir adaptiv; das heißt es werden nur Regionen unterteilt, die wir nicht direkt auswerten können. Im Falle von Catmull-Clark sind dies alle Flächen, die an einen außergewöhnlichen Vertex angrenzen (*irregulärer Patch*). Alle regulären Patches sind definiert als bi-kubische B-Splineflächen und können daher direkt (ohne weitere Unterteilung) ausgewertet werden. Nachdem ein irregulärer Patch unterteilt wurde, entstehen daraus vier Kind-Flächen. Drei davon enthalten keinen außergewöhnlichen Vertex mehr und sind demzufolge regulär und somit direkt auswertbar. Als Ergebnis der adaptiven Unterteilen erhalten wir eine verschachtelte Anordnung von bi-kubischen Flächen und ein sehr kleines irreguläres Rest-Flächenstück auf dem feinsten Unterteilungslevel. Wir können dieses Flächenstück zwar nicht direkt an beliebigen Parameterpositionen auswerten, allerdings ist es möglich die Grenzflächenpositionen exakt an der Stelle des außergewöhnlichen Vertex zu bestimmen. Dafür werden sogenannte Grenzschablonen verwendet, welche eine Linearkombination des 1-Rings um den entsprechenden Punkt definieren.

Für die feature-adaptive Unterteilung verwenden wir Unterteilungstabellen wie in Abschnitt 3.1 beschrieben. Die Tabellen sind allerdings wesentlich kleiner, da im Gegensatz zur uniformen Unterteilung, nur irreguläre Patches unterteilt werden. Bei einem Modell mit v außergewöhnlichen Gitterpunkten, und F Patches, werden ca. $12k \cdot v$ Kind-Flächen nach k Unterteilungsleveln generiert. Bei uniformer Unterteilung sind es ca. $4^k \cdot F$, wobei $F >> k$. Da die regulären bi-kubischen Flächen jeweils aus 16 Kontrollpunkten bestehen, muss zusätzlich zum eigentlichen irregulären Patch auch noch dessen 1-Ring mit unterteilt werden. Ein Beispiel einer drei-Level adaptiven Unterteilung eines Kontrollnetzes mit vier außergewöhnlichen Gitterpunkten ist in Abbildung 4 dargestellt. Da die Konnektivität statisch ist, kann die entsprechende Logik für die Tabellenberechnung in einem Vorverarbeitungsschritt auf der CPU durchgeführt werden, was in unserer unoptimierten Implementierung ca. 50 Millisekunden (je nach Modell) dauert.

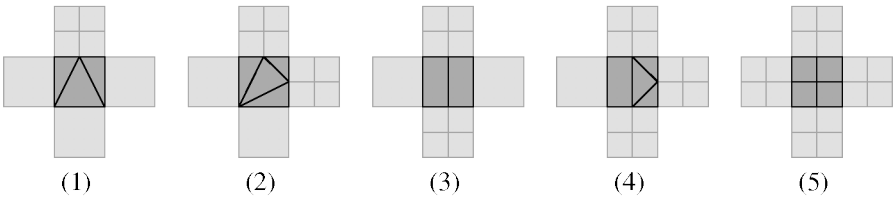


Abbildung 6: Es gibt fünf verschiedene Konstellationen von Übergangs-Patches (**UPs**). UPs sind rot, der aktuelle Unterteilungslevel blau, und der nächste Level grün gekennzeichnet. Das Aufteilen des Parametergebietes eines UPs in mehrere Teil-Patches erzwingt gleiche Länge für gemeinsame Kanten. Dadurch kann die Tessellierungsrate beliebig verändert werden ohne, dass Löcher entstehen.

3.3 Patchkonstruktion

Nach dem Schritt der feature-adaptive Unterteilung (siehe vorigen Abschnitt), verwenden wir die GPU hardware Tessellierung um die generierten Patches adaptiv zu triangulieren und anzuzeigen. Die Anzahl der Geometrie-Abtastpunkte auf Patchkanten kann dabei beliebig über die Tessellierungsfaktorn gesetzt werden. Für jeden Unterteilungslevel unterscheiden wir zwischen Voll-Patches und Übergangs-Patches.

Voll-Patches Voll-Patches (**VPs**) sind Flächen die nur mit Patches des gleichen Unterteilungslevels gemeinsame Kanten haben. Das können sowohl reguläre, als auch irreguläre Patches sein. Reguläre VPs können direkt von der Hardware Tessellierung verarbeitet und als bi-kubische B-Splineflächen gerendert werden. Durch die feature-adaptive Unterteilung stellen wir sicher, dass irreguläre VPs nur an den Eckpunkten ausgewertet werden müssen (dort kann die Grenzposition berechnet werden). Das bedeutet, für einen gegebenen Tessellierungsfaktor TF sind $\lceil \log_2 TF \rceil$ adaptive Unterteilungsschritte nötig um die selbe Tessellierungsdichte zu erzielen. Da auf aktueller Hardware der Tessellierungsfaktor auf 64 limitiert ist, müssen maximal 6 Schritte durchgeführt werden. Um die Grenzpositionen irregulärer Punkte zu bestimmen, verwenden wir einen separaten Vertexshader, welcher die Grenzschablonen anwendet.

Übergangs-Patches Bei der feature-adaptiven Unterteilung in Abschnitt 3.2 stellen wir sicher, dass generierte bi-kubische Patches nur an Flächen des selben oder um maximal eins verschiedenen Unterteilungslevels angrenzen. Patches, welche an Flächen des nächst feineren Unterteilungslevels angrenzen nennen wir Übergangs-Patches (**UPs**). Wir stellen zudem sicher, dass UPs immer regulär sind. Um Löcher beim Rendern zu vermeiden, müssen gemeinsame Kanten von benachbarten Patches an den selben Parameterpositionen ausgewertet werden. Dazu teilen wir UPs in verschiedene Teil-Patches auf und gewährleisten damit nahtlose Übergänge an T-Abzweigungen. Unter Ausnutzung von Rotationssymmetrie, ergeben sich fünf verschiedene Fälle (siehe Abbildung 6). Jedes Teil-Parametergebiet gehört dabei zu separaten Teil-Patches, welche allerdings alle durch die gleichen 16 Kontrollpunkte definiert sind. Die Unterteilung hat also lediglich auf das Parametergebiet, nicht aber auf die Oberfläche, Einfluss. Durch die geschickte Anordnung

der verschiedenen UPs werden T-Abzweigungen eliminiert und die Tessellierungsrate an gemeinsamen Kanten kann beliebig festgesetzt werden, ohne dass Löcher entstehen. Beim Setzen der Tessellierungsfaktorn sollte allerdings der aktuelle Unterteilungslevel beachtet werden um eine gleichmäßige Oberflächenabtastung zu gewährleisten. Die verschachtelte Anordnung von Patches ist in Abbildung 5 dargestellt. Generierte Bilder mit unserem Verfahren sind in den Abbildungen 1, 2, 8 zu finden.

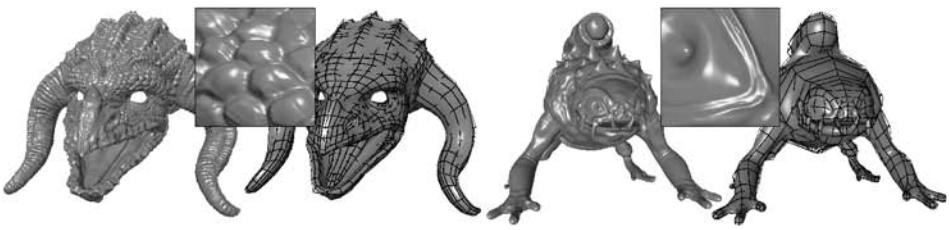


Abbildung 7: Feines Oberflächendetail wird mit unserer analytischen Verschiebungsfunktion sehr effizient auf Unterteilungsbasisflächen (siehe jeweils links) hinzugefügt. Die Renderzeiten von links nach rechts auf einer NVIDIA GTX 480: 1.7 ms, 1.2 ms, 1.3 ms, 0.85 ms. Unter 1 MB Speicher.

4 Analytische Verschiebungsfunktionen für Unterteilungsflächen

Unterteilungsflächen definieren (in der Regel) eine glatte Grenzfläche, meist ohne feines Obeflächendetail. Wir führen daher eine Verschiebungsfunktion für hochfrequentes detail auf Unterteilungsflächen ein. Im Gegensatz zu vorherigen Methoden ist unsere Verschiebungsfunktion analytisch und kommt daher ohne zusätzlichen Speicher für Oberflächennormalen aus. Wir stellen unsere verschobene Oberfläche dar als

$$f(u,v) = s(u,v) + N_s(u,v)D(u,v),$$

wobei $s(u,v)$ eine Catmull-Clark Fläche ist, $N_s(u,v)$ die Funktion der zugehörigen Oberflächennormalen, und $D(u,v)$ eine skalarwertige Oberflächenfunktion. Dadurch, dass die Grundfläche C^2 ist, gilt für $N_s(u,v)$ folglich C^1. Wir konstruieren die Verschiebungsfunktion $D(u,v)$ so, dass diese ebenfalls C^1 ist, indem wir skalare bi-quadratische B-Splines verwenden. Folglich ist auch $f(u,v)$ glatt und differenzierbar - d.h. es ist ein kontinuierliches Normalenfeld $N_f(u,v)$ auf $f(u,v)$ definiert. Für Details zur Behandlung außergewöhnlicher Gitterpunkte siehe [NL13].

Oberflächenauswertung Wir werten die Oberfläche inklusive Verschiebungsfunktion anhand der u,v Patchkoordinaten und der zugehörigen Flächenindizes aus. Für die Catmull-Clark Grundfläche $s(u,v)$ und deren Oberflächennormale $N_s(u,v)$ verwenden wir feature-adaptive Unterteilung (siehe Abschnitt 3). Die skalare Verschiebungsfunktion $D(u,v)$ wird ausgewertet indem ein lokales 3×3 Fenster der bi-quadratischen B-Spline Koeffizienten betrachtet wird. Dafür müssen die Patchparameter u,v in das Unterparametergebiet $(\hat{u},\hat{v})$ mittels eine linearen Transformation T gebracht werden:

$$\hat{u} = T(u) = u - \lfloor u \rfloor + \tfrac{1}{2}, \quad \text{and} \quad \hat{v} = T(v) = v - \lfloor v \rfloor + \tfrac{1}{2}.$$

Dadurch kann die skalare Verschiebungsfunktion ausgewertet werden

$$D(u,v) = \sum_{i=0}^{2}\sum_{j=0}^{2} B_i^2(T(u))B_j^2(T(v))d_{i,j},$$

wobei $d_{i,j}$ die ausgewählten Koeffizienten der Verschiebungsfunktion, und $B_i^2(u)$ die quadratischen B-Spline Basisfunktionen sind.

Um die Normale der Verschiebungsfunktion $f(u,v)$, auszuwerten, werden deren partielle Ableitungen benötigt:

$$\frac{\partial}{\partial u} f(u,v) = \frac{\partial}{\partial u} s(u,v) + \frac{\partial}{\partial u} N_s(u,v) D(u,v) + N_s(u,v) \frac{\partial}{\partial u} D(u,v).$$

$\frac{\partial}{\partial v} f(u,v)$ folgt analog. $\frac{\partial}{\partial u} s(u,v)$ ist direkt durch die Definition der Basisfläche gegeben. Um die Ableitungen von $N_s(u,v)$ zu bestimmen, berechnen wir zunächst die Ableitung der un-normalisierten normale $N_s^*(u,v)$ mit der Weingarten Gleichung (E, F, G und e, f, g sind die Koeffizienten der ersten und zweiten Fundamentalform):

$$\frac{\partial}{\partial u} N_s^*(u,v) = \frac{\partial}{\partial u} s(u,v) \frac{fF - eG}{EG - F^2} + \frac{\partial}{\partial v} s(u,v) \frac{eF - fE}{EG - F^2},$$

$\frac{\partial}{\partial v} N_s^*(u,v)$ folgt analog. Durch Nachdifferenzieren bestimmen wir die normalisierten partiellen Ableitungen der Normalen $\frac{\partial}{\partial u} N_s(u,v)$ und $\frac{\partial}{\partial v} N_s(u,v)$, womit sich $N_f(u,v)$ berechnen lässt.

Das Rendern der Oberfläche kann ebenso, wie in Abschnitt 3 beschrieben, mit dem Hardware Tessllierer durchgeführt werden. Oberflächennormalen werden auf Pixelebene ausgewertet, was zu sehr hochwertigen Ergebnissen führt (siehe Abbildung 7).

5 Zusammenfassung

Mit unseren Methoden ist es erstmals möglich Unterteilungsflächen in Echtzeitanwendungen einzusetzen. Letztendlich erreichen dadurch Computerspiele und Modellierungssoftware die Oberflächenqualität die bisher nur aufwändigen Filmproduktionen vorbehalten war. Bereits nach kurzer Zeit der Veröffentlichung unserer Algorithmen, wurden die Forschungsergebnisse in kommerziellen Anwendungen integriert und verwendet, wie es zum Beispiel in Abbildung 8 zu sehen ist. Während unsere Arbeit den Fokus speziell auf die geometrischen Aspekte des Echtzeitrenderings legt, erwarten wir in den nächsten Jahren ähnliche Fortschritte im Bereich der Beleuchtungsberechnung und Physiksimulation. Dadurch können visuell noch ansprechendere und von der Realität kaum noch zu unterscheidende Ergebnisse erzielt werden.

In dieser Kurzfassung haben wir einen kleinen Ausschnitt unserer Algorithmen vorgestellt, die während der Promotion entwickelt wurden [Nie13]. Für eine detaillierte Darstellung verweisen wir auf die entsprechenden Publikationen und die Ausarbeitung der Dissertation. Wir hoffen jedoch, dass wir einen kleinen Einblick in die technischen Grundlagen und den akademischen Beitrag unserer Forschung geben konnten.

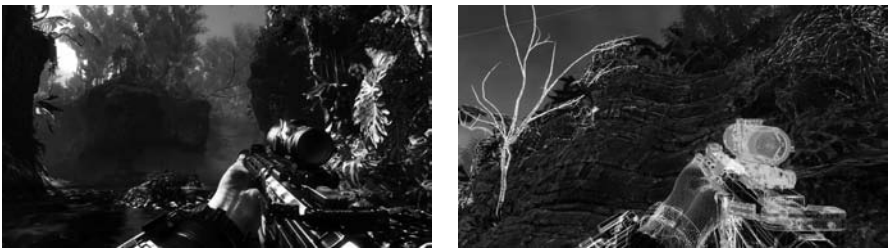


Abbildung 8: Unterteilungsflächen im Bestseller *Call of Duty: Ghosts*, gerendert mit unserem feature-adaptiven Unterteilungsverfahren. Das Drahtgittermodel und zugehörige Unterteilungslevel sind rechts visualisiert. © Activision Blizzard

Literatur

[CC78] Edwin Catmull und James Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-aided design*, 10(6):350–355, 1978.

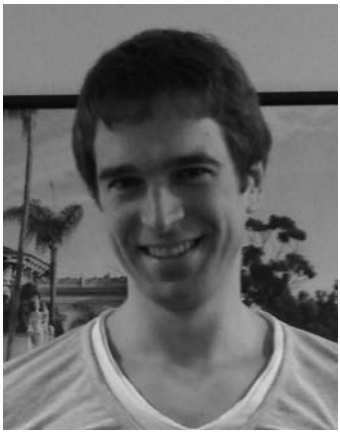
[Nie13] M. Nießner. *Rendering Subdivision Surfaces using Hardware Tessellation*. Dissertation. Dr. Hut, 2013.

[NL12] M. Nießner und C. Loop. Patch-based Occlusion Culling for Hardware Tessellation. In *Computer Graphics International*, Jgg. 2, 2012.

[NL13] M. Nießner und C. Loop. Analytic Displacement Mapping using Hardware Tessellation. *ACM Transactions on Graphics (TOG)*, 32(3):26, 2013.

[NLMD12] M. Nießner, C. Loop, M. Meyer und T. DeRose. Feature-adaptive GPU Rendering of Catmull-Clark Subdivision Surfaces. *ACM Transactions on Graphics (TOG)*, 31(1):6, 2012.

[NSSL13] M. Nießner, C. Siegl, S. Schäfer und C. Loop. Real-time Collision Detection for Dynamic Hardware Tessellated Objects. *Eurographics*, 2013.



Matthias Nießner, geboren 1986, hat an der Universität Erlangen-Nürnberg Informatik studiert und 2009 sein Diplom erhalten (Betreuer: Prof. Dr.-Ing. Marc Stamminger). Wegen exzellenter Studienleistungen (Bestnote), sehr kurzer Studienzeit, und seiner herausragenden Diplomarbeit wurde er mit dem ASQF Förderpreis ausgezeichnet. Anschließend hat er unter der Betreuung von Prof. Dr. Günther Greiner im Bereich der Computergrafik seine Promotion begonnen (ebenfalls Universität Erlangen-Nürnberg). 2013 hat er im Alter von 26 Jahren seine Doktorarbeit abgeben, welche sowohl mit Bestnote als auch mit Auszeichnung bestanden wurde (Externer Gutachter: Prof. Dr. Hans-Peter Seidel). Seit September 2013 ist er Visiting Assistant Professor an der Stanford Universität und arbeitet in der Gruppe von Prof. Pat Hanrahan an aktuellen Forschungsthemen im Bereich der Informatik - *the next big thing*.

Konnektionskalküle für automatisches Beweisen in klassischen und nicht-klassischen Logiken*

Jens Otten

Institut für Informatik, University of Potsdam
August-Bebel-Str. 89, 14482 Potsdam-Babelsberg, Germany
E-Mail: jeotten@cs.uni-potsdam.de

Abstract: Die Automatisierung des formalen logischen Beweisens ist ein grundlegendes Forschungsgebiet innerhalb der Künstlichen Intelligenz. Der vorliegende Artikel stellt Konnektionskalküle vor, die zur automatischen Beweissuche in klassischer, intuitionistischer und modaler Logik verwendet werden können. Dabei wird der Konnektionskalkül für klassische Logik in einer eleganten und uniformen Art erweitert. Es werden sehr kompakte Implementierungen dieser Kalküle vorgestellt, die zu den leistungsfähigsten Beweissystemen in ihrer Klasse zählen und bereits mehrfach erfolgreich an internationalen Wettbewerben teilgenommen haben.

1 Einleitung

Logisches Schlussfolgern ist nicht nur eine wichtige Grundlage menschlichen Handelns im Alltag, sondern ein elementares Prinzip, das in allen wissenschaftlichen Disziplinen angewendet wird. So werden zum Beispiel in der Mathematik neue Sätze und Theoreme durch Anwendung formaler logischer Regeln aus Axiomen und bereits bekannten Sätzen hergeleitet. Der *Beweis* eines Satzes oder Theorems ist dann eine Folge von logischen Schlüssen. Das Forschungsgebiet, das sich mit der Automatisierung dieses logischen Schlussfolgerns beschäftigt, wird daher auch *Automatisches Beweisen* genannt. Ziel dabei ist es, den Beweis eines Theorems vollautomatisch durch den Computer finden zu lassen. Ein solches Computerprogramm wird auch *Beweiser* genannt.

Die Sprache der *Prädikatenlogik* wird dabei benutzt, um Aussagen zu repräsentieren und das logische Schließen über diese Aussagen zu formalisieren. In der Sprache der Prädikatenlogik werden zum Beispiel die Aussagen (a) *Plato ist ein Mensch* und (b) *für alle* X *gilt, wenn* X *ein Mensch ist, dann ist* X *sterblich*, wie folgt formalisiert:

$$\left.\begin{array}{ll}(a) & \mathit{Mensch(Plato)} \\ (b) & \forall X\ (\mathit{Mensch}(X) \Rightarrow \mathit{sterblich}(X))\end{array}\right\} \Rightarrow \quad (c) \quad \mathit{sterblich(Plato)}\,.$$

Aus diesen beiden Aussagen lässt sich logisch Schlussfolgern, dass (c) *Plato sterblich ist*. Die Aussage (c) ist also eine *logische Konsequenz* der Aussagen (a) und (b).

*Englischer Titel der Dissertation: "Connection Calculi for Automated Theorem Proving in Classical and Non-Classical Logics".

Das *Deduktionstheorem* besagt, dass sich das logische Schlussfolgern auf das Bestimmen der Gültigkeit einer Aussage reduzieren lässt. Eine Aussage ist genau dann *gültig*, wenn sie unabhängig davon wahr ist, welche Bedeutung den einzelnen Wörtern zugewiesen wird. Eine Aussage A ist genau dann eine logische Konsequenz der Aussagen $A_1, \ldots, A_n$, falls die Aussage $(A_1 \wedge \ldots \wedge A_n) \Rightarrow A$ gültig ist. So ist zum Beispiel

$$(d) \quad (\mathit{Mensch(Plato)} \wedge \forall X(\mathit{Mensch}(X) \Rightarrow \mathit{sterblich}(X))) \Rightarrow \mathit{sterblich(Plato)}$$

gültig, das heißt die Aussage (c) folgt tatsächlich logisch aus den Aussagen (a) und (b).

Die Aufgabe eines Beweisers ist es daher herauszufinden, ob eine gegebene Aussage oder *Formel* gültig ist. Da der Suchraum bei der Beweissuche bereits für relativ einfache Theoreme sehr groß ist, stellt das automatische Beweisen eine immense Herausforderung dar. Dieses Forschungsgebiet zählt daher zum Kerngebiet der *Künstlichen Intelligenz*.

Neben der häufig verwendeten *klassischen Logik*, gehören die intuitionistische Logik und die modale Logik zu den wichtigsten *nicht-klassischen Logiken*. Sie haben zahlreiche Anwendungen, zum Beispiel in der Programmsynthese, der Programmverifikation, beim Planen, der Modellierung von Kommunikation und bei der Verarbeitung natürlicher Sprache.

Die *intuitionistische Logik* [Dal01] stellt höhere Anforderungen an die Gültigkeit einer Formel und ist daher eine Einschränkung der klassischen Logik. Die Formel

$$(e) \quad \mathit{Mensch(Sokrates)} \vee \neg \mathit{Mensch(Sokrates)}$$

formalisiert die Aussage *Sokrates ist ein Mensch oder Sokrates ist kein Mensch* und ist in klassischer, aber *nicht* in intuitionistischer Logik gültig. Die *modale Logik* [BBW06] ist eine Erweiterung der klassischen Logik um die Modaloperatoren $\Box$ und $\Diamond$, die als "notwendigerweise" und "möglicherweise" interpretiert werden. Die gültige Formel

$$(f) \quad (\Box \mathit{Mensch(Plato)}) \Rightarrow (\Diamond \mathit{Mensch(Plato)})$$

formalisiert also die Aussage *falls Plato notwendigerweise ein Mensch ist, dann ist er auch möglicherweise ein Mensch*. Abhängig davon, wie die Modaloperatoren interpretiert werden, gibt es nicht nur eine modale Logik, sondern eine ganze Reihe von modalen Logiken.

Während die Entwicklung von effizienten Beweisern für klassische Logik in den letzten Jahrzehnten große Fortschritte gemacht hat, steht die Entwicklung von leistungsfähigen Beweisern für viele nicht-klassische Logiken noch am Anfang. Die Beweissuche in nicht-klassischen Logiken ist erheblich schwieriger als in klassischer Logik. So ist die Beweissuche in klassischer *Aussagenlogik* bereits *co-$\mathcal{NP}$-vollständig*, d.h. alle bekannten Algorithmen haben eine exponentielle Zeitkomplexität. Die Beweissuche in intuitionistischer und den (meisten) modalen Aussagenlogiken ist sogar *$\mathcal{PSPACE}$-vollständig*.

Die formale Beschreibung der Beweissuche dabei häufig mit Hilfe eines *Kalküls*, der aus *Axiomen* und *Regeln* besteht. Ein Beweiser implementiert dann einen Kalkül mit einer Strategie, die beschreibt, in welcher Reihenfolge die Regeln bei der Beweissuche angewendet werden. Bekannte Kalküle zur Beweissuche sind zum Beispiel *Sequenzenkalküle*, *Tableaukalküle*, *Konnektionskalküle*, *Resolutionskalküle* und *instanzenbasierte Kalküle*.

Dieser Artikel ist die Zusammenfassung einer Dissertation [Ott13], in der sowohl neue Konnektionskalküle als auch deren Implementierung für die automatische Beweissuche in klassischer Logik und wichtigen nicht-klassischen Logiken vorgestellt werden.

2 Klausel-Konnektionskalküle

2.1 Konnektionskalkül für klassische Logik

Eine *Konnektion* ist ein Paar von (atomaren) Formeln der Form $\{A, \neg A\}$. Zum Beispiel ist $\{$*Mensch(Plato)*, $\neg$*Mensch(Plato)*$\}$ eine Konnektion; *Mensch(Plato)* und $\neg$*Mensch(Plato)* werden als *Literale* bezeichnet. Bei Konnektionskalkülen [Bib87, LS01] wird die Beweissuche durch Konnektionen gesteuert, was eine zielgerichtete Suche ermöglicht.

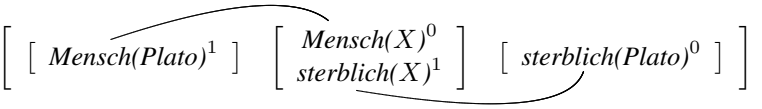
Die Formeln, deren Gültigkeit bewiesen werden sollen, müssen im Konnektionskalkül in Klauselform sein. Eine Formel in *Klauselform* hat die Form $\exists x_1 \ldots \exists x_n (C_1 \vee \ldots \vee C_n)$, wobei jede *Klausel* C_i die Form $L_1 \wedge \ldots \wedge L_m$ hat und jedes L_i ein Literal ist. In klassischer Logik kann jede Formel F in eine äquivalente Formel F' in Klauselform übersetzt werden, so dass F genau dann gültig ist, wenn F' gültig ist. Eine *Matrix* stellt ein Formel in Klauselform als eine Menge von Klauseln $\{C_1, \ldots, C_n\}$ dar, wobei jede Klausel C_i eine Menge von Literalen $\{L_1, \ldots, L_m\}$ ist, und Literale der Form A bzw. $\neg A$ durch A^0 bzw. A^1 dargestellt werden; eine Konnektion hat dann die Form $\{A^0, A^1\}$.

Die Formel (d) aus Abschnitt 1 hat die Klauselform

$$\exists X (\neg \textit{sterblich(Plato)} \vee (\textit{Mensch}(X) \wedge \neg \textit{sterblich}(X)) \vee \textit{sterblich(Plato)})$$

und $M' = \{\{\textit{Mensch(Plato)}^1\}, \{\textit{Mensch}(X)^0, \textit{sterblich}(X)^1\}, \{\textit{sterblich(Plato)}^0\}\}$ ist die zugehörige Matrix. Damit $\{\textit{Mensch}(X)^0, \textit{Mensch(Plato)}^1\}$ eine Konnektion ist, muss die Variable X durch *Plato* ersetzt werden. Dies wird durch eine *Termsubstitution* σ realisiert, d.h. $\sigma(X) = \textit{Plato}$. Die Konnektion wird dann als *σ-komplementär* bezeichnet.

Bei der graphischen Darstellung eines Beweises im Konnektionskalkül werden die Klauseln der Matrix nebeneinander, Literale jeder Klausel übereinander angeordnet; Literale, die eine Konnektion bilden, werden durch einen Linie miteinander verbunden. Der *graphische Konnektionsbeweis* der Matrix M' mit $\sigma(X) = \textit{Plato}$ sieht dann wir folgt aus:



Ein formaler *Klausel-Konnektionskalkül* wurde von Otten und Bibel [OB03] angegeben und ist in Abbildung 1 dargestellt. Die Wörter des Kalküls haben die Form "$C, M, Path$"; M ist eine Matrix, C und $Path$ sind Mengen von Literalen oder ε und werden als *Zielklausel* bzw. *aktiver Pfad* bezeichnet. Ein *Klausel-Konnektionsbeweis* für eine Formel F ist eine Ableitung von $\varepsilon, M, \varepsilon$ im Klausel-Konnektionskalkül, wobei M die Matrix von F ist und jeder Zweig der Ableitung durch ein Axiom abgeschlossen ist. Die folgende Ableitung ist ein Beweis der Matrix M', und damit der Formel (d), im Klausel-Konnektionskalkül.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\{\}, M', \{\textit{sterblich(Plato)}^0, \textit{Mensch}(X')^0\}}\,A \quad \dfrac{}{\{\}, M', \{\textit{sterblich(Plato)}^0\}}\,A}{\{\textit{Mensch}(X')^0\}, \{\{\textit{Mensch(Plato)}^1\}, \ldots\}, \{\textit{sterblich(Plato)}^0\}}\,E \quad \dfrac{}{\{\}, M', \{\}}\,A}{\{\textit{sterblich(Plato)}^0\}, \{\{\textit{Mensch(Plato)}^1\}, \{\textit{Mensch}(X)^0, \textit{sterblich}(X)^1\}, \ldots\}, \{\}}\,E}{\varepsilon, \{\{\textit{Mensch(Plato)}^1\}, \{\textit{Mensch}(X)^0, \textit{sterblich}(X)^1\}, \{\textit{sterblich(Plato)}^0\}\}, \varepsilon}\,S$$

Axiom (A)	$\dfrac{}{\{\}, M, Path}$	
Start (S)	$\dfrac{C_2, M, \{\}}{\varepsilon,\ M,\ \varepsilon}$	und C_2 ist Kopie von $C_1 \in M$
Reduktion (R)	$\dfrac{C, M, Path \cup \{L_2\}}{C \cup \{L_1\}, M, Path \cup \{L_2\}}$	und $\{L_1, L_2\}$ ist σ-komplementär
Extension (E)	$\dfrac{C_2 \backslash \{L_2\}, M, Path \cup \{L_1\} \quad C, M, Path}{C \cup \{L_1\}, M, Path}$	und C_2 ist eine Kopie von $C_1 \in M,\ L_2 \in C_2$, $\{L_1, L_2\}$ ist σ-kompl.

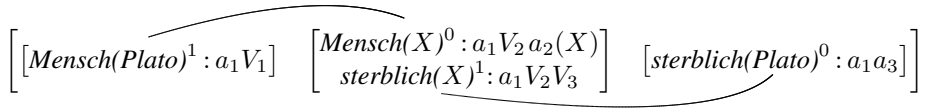
Abbildung 1: Der Klausel-Konnektionskalkül für klassische Logik

Der Beweis der *Korrektheit* und der *Vollständigkeit* des Kalküls beruht auf der *Matrixcharakterisierung* [Bib87] für klassische Logik. Die *Beweissuche* im Klausel-Konnektionskalkül startet mit $\varepsilon, M, \varepsilon$ und die Regeln werden analytisch, d.h. von unten nach oben, angewendet. Falls es bei der Anwendung von Regeln oder Regelinstanzen mehrere Alternativen gibt, muss durch *Backtracking* sichergestellt werden, dass gegebenfalls alle alternative Regeln oder Regelinstanzen berücksichtigt werden. Die Termsubstitution σ wird dabei schrittweise durch einen der bekannten Algorithmen zur *Termunifikation* berechnet.

2.2 Konnektionskalkül für intuitionistische Logik

Um den Konnektionskalkül für klassische Logik auf intuitionistische Logik zu erweitern, wird jedem Literal ein Präfix zugeordnet. Ein *Präfix* ist eine Zeichenkette, die aus *Präfixvariablen* und *Präfixkonstanten* besteht. Er wird bestimmt durch den Kontext, in dem das Literal in der Formel vorkommt. Eine Konnektion hat dann die Form $\{A(t_1)^0 : p_1,\ A(t_2)^1 : p_2\}$, in der die Literale den Präfix p_1 bzw. p_2 haben. Aufgrund der Matrixcharakterisierung für intuitionistische Logik [Wal90] ist diese Konnektion genau dann σ-komplementär, wenn es zusätzlich zu der Termsubstitution σ_T eine *Präfixsubstitution* σ_P gibt, so dass $\sigma_T(t_1) = \sigma_T(t_2)$ und $\sigma_P(p_1) = \sigma_P(p_2)$. Da es für intuitionistische Logik keine Klauselform gibt, wird die Matrixcharakterisierung angepasst. Dazu wird die für klassische Logik bekannte *Skolemisierungstechnik* auf Präfixkonstanten erweitert [Ott05].

Für die Formel (d) ist $\{\{$*Mensch(Plato)*$^1 : a_1 V_1\}, \{$*Mensch(X)*$^0 : a_1 V_2\, a_2(X)$, *sterblich(X)*$^1 : a_1 V_2 V_3\}, \{$*sterblich(Plato)*$^0 : a_1 a_3\}\}$ die *intuitionistische Matrix*, wobei a_1, $a_2(X)$, a_3 Präfixkonstanten und V_1, V_2, V_3 Präfixvariablen sind. Der graphische intuitionistische Konnektionsbeweis dieser Matrix mit $\sigma_T(X) = Plato$, $\sigma_P(V_1) = a_2(Plato)$, $\sigma_P(V_2) = \varepsilon$ (wobei ε die leere Zeichenkette ist) und $\sigma_J(V_3) = a_3$ sieht dann wie folgt aus:



Axiom (A) $\dfrac{}{\{\}, M, Path}$

Start (S) $\dfrac{C_2, M, \{\}}{\varepsilon,\ M,\ \varepsilon}$ und C_2 ist Kopie von $C_1 \in M$

Reduktion (R) $\dfrac{C, M, Path \cup \{L_2 : p_2\}}{C \cup \{L_1 : p_1\}, M, Path \cup \{L_2 : p_2\}}$ und $\{L_1 : p_1, L_2 : p_2\}$ ist σ-komplementär

Extension (E) $\dfrac{C_2 \backslash \{L_2 : p_2\}, M, Path \cup \{L_1 : p_1\} \quad C, M, Path}{C \cup \{L_1 : p_1\}, M, Path}$

und C_2 ist ein Kopie von $C_1 \in M$, $L_2{:}p_2 \in C_2$, $\{L_1{:}p_1, L_2{:}p_2\}$ ist σ-kompl.

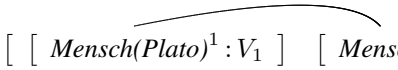
Abbildung 2: Der Klausel-Konnektionskalkül für intuitionistische und modale Logik

Für die Formel (e) ergibt sich die intuitionistische Matrix $\{\{$*Mensch(Sokrates)*$^0 : a_1\}$, $\{$*Mensch(Sokrates)*$^1 : a_2\}\}$. Da es keine Substitution σ_P gibt, die die beiden Präfixe a_1 und a_2 der Konnektion gleichmacht, ist die Formel intuitionistisch *nicht* gültig.

Der formale *Klausel-Konnektionskalkül für intuitionistische Logik* [Ott05] ist in Abbildung 2 dargestellt, wobei M eine intuitionistische Matrix ist. Der Beweis des Korrektheit und der Vollständigkeit des Kalküls basiert auf der angepassten Matrixcharakterisierung. Die Präfixsubstitution σ_P wird von einem speziellen *Präfixunifikations-Algorithmus* berechnet [Ott05]. Der Algorithmus berechnet aus einer Menge von Präfix-Gleichungen mit Hilfe von Ersetzungsregeln eine *minimale* Menge *allgemeinster* Präfixsubstitutionen.

2.3 Konnektionskalkül für modale Logiken

Der *Klausel-Konnektionskalkül für modale Logik* [Ott12, BOR12] ist im Wesentlichen identisch mit dem Kalkül für intuitionistische Logik in Abbildung 2. Die Definition der Präfixe ist nun abhängig von den Modaloperatoren, in deren Kontext ein Literal in der Formel vorkommt [Wal90]. Der Kalkül verwendet dabei die *modale Matrix*, in der wiederum eine angepasste Skolemisierungstechnik benutzt wird [Ott12]. Zum Beispiel ist die modale Matrix der Formel (f) aus Abschnitt 1 $\{\{$*Mensch(Plato)*$^1{:}V_1\}$, $\{$*Mensch(Plato)*$^0{:}V_2\}\}$, wobei V_1 und V_2 Präfixvariablen sind. Der graphische modale Konnektionsbeweis dieser Matrix mit $\sigma_P(V_1) = V_2$ sieht wie folgt aus:



Der Algorithmus zur Präfixunifikation wird angepasst und berücksichtigt nun die *Erreichbarkeitsrelation* der *Kripke-Semantik* der gewählten modalen Logik. Dabei wurden entsprechende Algorithmen für die modalen Logiken D, T, S4 und S5 entwickelt [Ott12]. Eine zusätzliche *Domänen-Bedingung* spezifiziert, ob *konstante*, *kumulative* oder *variierende Domäne* betrachtet werden.

3 Implementierung von Klausel-Konnektionskalkülen

3.1 leanCoP für klassische Logik

leanCoP ist ein Beweiser für klassische Prädikatenlogik [OB03, Ott08] und stellt eine sehr kompakte Prolog-Implementierung des Klausel-Konnektionskalküls aus Abschnitt 2.1 dar. Während leanCoP 1.0 [OB03] im Wesentlichen den Basis-Kalkül implementiert, verwendet leanCoP 2.0 zusätzliche Techniken, die die Beweissuche optimieren [Ott08, Ott10]. Der Quellcode des Kernbeweisers ist in Abbildung 3 dargestellt.

Lean-Prolog-Technologie ist eine Technik, die die Klauseln der gegebenen Matrix mit Hilfe des Prädikats `lit/3` repräsentiert und vor der Beweissuche in der Prolog-Datenbank speichert. Die Beweissuche wird dann mit `prove(I,S)` gestartet; `I` ist die maximale Länge des aktiven Pfades und wird für die iterative Tiefensuche benutzt, `S` ist eine Menge von Optionen, um die Beweissuche zu steuern [Ott08, Ott10]. Die *Start*-Regel wird in den ersten beiden Zeilen implementiert, die nächsten beiden Zeilen realisieren die iterative Tiefensuche. Die *Reduktions-* und *Extensions*-Regel werden durch das Prädikat `prove(C,P,I,Q,S)` implementiert; `C` ist die Zielklausel, `P` ist der aktive Pfad und `Q` ist die Menge der bereits bewiesenen Literale und wird für eine zusätzliche *Lemma*-Regel verwendet. Das *Axiom* wird in Zeile fünf implementiert. Die Termsubstitution wird implizit durch Prolog gespeichert. Weitere Techniken sind *Regularität* (Zeile 6/7), die *kontrollierte Tiefensuche* (Zeile 9/10), sowie ein *eingeschränktes Backtracking* (Zeile 11). Das eingeschränkte Backtracking ist die zur Zeit effektivste Technik, um die Beweissuche im Konnektionskalkül zu optimieren [Ott10]. Weiterhin wird eine *definitorische Klauselform-Transformation* benutzt und ein *Strategie-Scheduling*, das nacheinander verschiedene Strategien bei der Beweissuche anwendet [Ott08, Ott10].

Diese zusätzlichen Techniken in leanCoP 2.0 führen zu einer Verbesserung der Leistung um etwa den Faktor 10^5 bezüglich der international etablierten *TPTP-Problemsammlung*. leanCoP 2.1 gibt zusätzlich einen lesbaren Konnektionsbeweis aus.

Der komplette Beweiser mit den zusätzlichen Komponenten ist auf der leanCoP-Webseite unter der Adresse `http://www.leancop.de` verfügbar.

```
prove(I,S) :- \+member(scut,S) -> prove([-(#)],[],I,[],S) ;
    lit(#,C,_) -> prove(C,[-(#)],I,[],S).
prove(I,S) :- member(comp(L),S), I=L -> prove(1,[]) ;
    (member(comp(_),S);retract(p)) -> J is I+1, prove(J,S).
prove([],_,_,_,_).
prove([L|C],P,I,Q,S) :- \+ (member(A,[L|C]), member(B,P),
    A==B), (-N=L;-L=N) -> ( member(D,Q), L==D ;
    member(E,P), unify_with_occurs_check(E,N) ; lit(N,F,H),
    (H=g -> true ; length(P,K), K<I -> true ;
    \+p -> assert(p), fail), prove(F,[L|P],I,Q,S) ),
    (member(cut,S) -> ! ; true), prove(C,P,I,[L|Q],S).
```

Abbildung 3: Der Quellcode des Kernbeweisers von leanCoP 2.0 für klassische Logik

```
prove(I,S) :- ( \+member(scut,S) ->
    prove([(-(#)):(-[])],[],I,[],[Z,T],S) ;
    lit(#):_,G:C,_) -> prove(C,[(-(#)):(-[])],I,[],[Z,R],S),
    append(R,G,T) ), check_addco(T), prefix_unify(Z).
prove(I,S) :- member(comp(L),S), I=L -> prove(1,[]) ;
    (member(comp(_),S);retract(p)) -> J is I+1, prove(J,S).
prove([],_,_,_,[[],[]],_).
prove([L:U|C],P,I,Q,[Z,T],S) :- \+(member(A,[L:U|C]), member(B,P),
    A==B), (-N=L;-L=N) -> ( member(D,Q), L:U==D, X=[], O=[] ;
    member(E:V,P), unify_with_occurs_check(E,N),
    \+ \+ prefix_unify([U=V]), X=[U=V], O=[] ;
    lit(N:V,M:F,H), \+ \+ prefix_unify([U=V]),
    (H=g -> true ; length(P,K), K<I -> true ;
    \+p -> assert(p), fail), prove(F,[L:U|P],I,Q,[W,R],S),
    X=[U=V|W], append(R,M,O) ), (member(cut,S) -> ! ; true),
    prove(C,P,I,[L:U|Q],[Y,J],S), append(X,Y,Z), append(J,O,T).
```

Abbildung 4: Der Quellcode der Kernbeweiser von ileanCoP 1.2 und MleanCoP 1.2

3.2 ileanCoP für intuitionistische Logik

ileanCoP ist ein Beweiser für intuitionistische Prädikatenlogik [Ott05, Ott08] und eine sehr kompakte Implementierung des intuitionistischen Kalküls aus Abschnitt 2.2. Der Quellcode des Kernbeweisers ist in Abbildung 4 dargestellt. Lediglich die unterstrichenen Zeichen wurden dem Quellcode von leanCoP 2.0 in Abbildung 3 hinzugefügt. Es wurden Präfixe hinzugefügt, als auch eine Präfixunifikation (`prefix_unify/1`), die weitere 18 Zeilen an Prolog-Code benötigt. Alle Optimierungen von leanCoP 2.0 wurden angepasst und integriert. ileanCoP 1.2 beweist erheblich mehr Probleme aus der TPTP-Problemsammlung und der *ILTP-Problemsammlung* [ROK07] als alle anderen Beweiser für intuitionistische Logik [Ott08]. Der komplette Beweiser ist auf der ileanCoP-Webseite unter der Adresse `http://www.leancop.de/ileancop/` verfügbar.

3.3 MleanCoP für modale Logiken

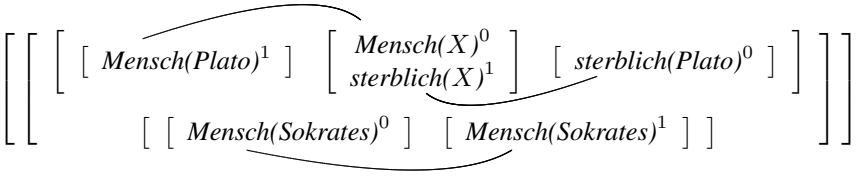
MleanCoP ist ein Beweiser für die modalen Prädikatenlogiken D, T, S4 und S5 mit konstanten, kumulativen und variierenden Domäne [Ott12] und eine sehr kompakte Implementierung des modalen Kalküls aus Abschnitt 2.3. Der Quellcode des Kernbeweisers ist im Wesentlichen identisch mit dem für ileanCoP in Abbildung 4. Dabei wird die Definition der Präfixe an die modalen Logiken angepasst. Weiterhin müssen die Präfixunifikationen für die modalen Logiken D, T, S4 und S5 hinzugefügt und die *Domänen-Bedingung* (`check_addco/1`) angepasst werden. MleanCoP 1.2 beweist erheblich mehr Probleme aus der *QMLTP-Problemsammlung* [RO12] als alle anderen Beweiser für modale Logik [BOR12]. Der komplette Beweiser ist auf der MleanCoP-Webseite unter der Adresse `http://www.leancop.de/mleancop/` verfügbar.

4 Der Nicht-Klausel-Konnektionskalkül

Die für die Klausel-Konnektionskalküle notwendige Transformation in Klauselform verändert die Struktur der Formel und kann dazu führen, dass sich die Formel deutlich vergrößert, was die Beweissuche erheblich erschweren kann. Dies gilt auch für die definitorische Klauselform-Transformation [Ott10]. Der *Nicht-Klausel-Konnektionskalkül* [Ott11] hat diese Nachteile nicht, da der Kalkül keine Klauselform benötigt und stattdessen die originale Struktur der gegebenen Formel erhält. Dafür muss die Definition der Matrix verallgemeinert werden. Eine *Nicht-Klausel-Matrix* ist eine Menge von Klauseln, wobei jede Klausel eine Menge von Literalen und (Unter-)Matrizen ist. Zum Beispiel wird die Formel

$$\begin{array}{c}((\mathit{Mensch(Plato)} \wedge \forall X(\mathit{Mensch}(X) \Rightarrow \mathit{sterblich}(X))) \Rightarrow \\ \mathit{sterblich(Plato)}) \ \wedge \ (\mathit{Mensch(Sokrates)} \vee \neg \mathit{Mensch(Sokrates)})\end{array}$$

durch die Nicht-Klausel-Matrix $\{\{\{\{\mathit{Mensch(Plato)}^1\}, \{\mathit{Mensch}(X)^0, \mathit{sterblich}(X)^1\}, \{\mathit{sterblich(Plato)}^0\}\}, \{\{\mathit{Mensch(Sokrates)}^0\}, \{\mathit{Mensch(Sokrates)}^1\}\}\}$ repräsentiert, die 6 Literale enthält. Die entsprechende *Klausel*form enthält mindestens 14 Literale. Der graphische *Nicht-Klausel-Konnektionsbeweis* mit $\sigma(X) = \mathit{Plato}$ hat die folgende Form:



Der *formale* Nicht-Klausel-Konnektionskalkül hat das gleiche Axiom und die gleiche Start- und Reduktions-Regel wie der Klausel-Konnektionskalkül. Die Extensions-Regel wird angepasst und eine *Dekompositions*-Regel hinzugefügt [Ott11]. Die vereinfachte Version des Kalküls, bei dem Start- und Reduktions-Regel subsumiert werden, ist in Abbildung 5 dargestellt. Für die Nicht-Klausel-Matrix M startet die Suche mit $\{M\}, M, \{\}$.

Der Beweis der Korrektheit und der Vollständigkeit des Nicht-Klausel-Konnektionskalküls basiert auf einer Nicht-Klausel-Matrixcharakterisierung und der Einbettung des Nicht-Klausel-Kalküls in den Klausel-Konnektionskalkül [Ott11].

Axiom (A) $\dfrac{}{\{\}, M, Path}$

Extension (E) $\dfrac{C_3, M[C_1 \backslash C_2], Path \cup \{L_1\} \qquad C, M, Path}{C \cup \{L_1\}, M, Path}$

und $C_3 := \beta\text{-}clause_{L_2}(C_2)$, C_2 ist Kopie von C_1, C_1 ist e-Klausel von M bzgl. $Path \cup \{L_1\}$, C_2 enthält L_2 mit $\{L_1, L_2\}$ σ-kompl.

Dekomposition (D) $\dfrac{C \cup C_1, M, Path}{C \cup \{M_1\}, M, Path}$ und $C_1 \in M_1$

Abbildung 5: Der (vereinfachte) Nicht-Klausel-Konnektionskalkül

5 Zusammenfassung

Formales Beweisen ist eine fundamentale Tätigkeit in der Informatik und vielen verwandten Wissenschaften. Seit Frege seine *Begriffsschrift* [Fre79] im Jahr 1879 veröffentlicht hat, gilt die Entwicklung von effizienten Kalkülen zur Automatisierung des formalen Beweisens als eines der wichtigsten Forschungsziele innerhalb der Künstlichen Intelligenz. Die Entwicklung von Kalkülen und Beweisern hat in den letzten Jahrzehnten große Fortschritte gemacht, und Beweiser werden heutzutage in industriellen Anwendungen, wie etwa der Verifikation von Hard- und Software, eingesetzt.

Die vorliegende Arbeit stellt leistungsfähige Beweiser für klassische und einige wichtige nicht-klassische Logiken vor. Alle Beweiser basieren auf einem uniformen Klausel-Konnektionskalkül, der durch die Verwendung von Präfixen, einer Präfixunifikation sowie einer erweiterten Skolemisierung auf intuitionistische und modale Logiken übertragen wurde. Daneben enthalten die Beweiser weitere Techniken um die Beweissuche zu optimieren, darunter eine optimierte definitorische Klauselform-Transformation, Lean-Prolog-Technologie, Regularität und das sehr erfolgreiche "eingeschränkte Backtracking".

leanCoP ist der zur Zeit schnellste Beweiser, der auf einem Klausel-Konnektionskalkül basiert. Der Kernbeweiser besteht aus nur wenigen Zeilen Prolog-Code. leanCoP hat bereits mehrfach erfolgreich an den internationalen CASC-Wettbewerben teilgenommen. Darunter sind nicht nur Platzierungen unter den Top 3 der schnellsten Beweiser mit Beweisausgabe, sondern auch der "Best Newcomer"-Preis für leanCoP 2.0 [Sut08], der dritte Preis in der SUMO-Kategorie, die sehr große Formeln enthält, für leanCoP-SInE [Sut10], und der Gewinn der arithmetischen TFA-Kategorie durch leanCoP-Ω [Sut11].

ileanCoP und MleanCoP sind die zur Zeit leistungsfähigsten Beweiser für intuitionistische und modale Prädikatenlogik. Sie basieren auf leanCoP und erweitern diesen durch das Hinzufügen von Präfixen und einer Präfixunifikation. Die spezielle Logik wird dabei lediglich durch das Austauschen der Unifikationskomponente bestimmt. Dadurch lassen sich Optimierungstechniken, die bereits für klassische Logik benutzt werden, auf einfache Art übertragen. Die Benutzung einer zusätzlichen Präfixunifikation bei der Beweissuche in nicht-klassischen Logiken ähnelt der Benutzung der Termunifikation bei der Beweissuche in klassischer *Prädikaten*logik. Dieser Sachverhalt lässt sich wie folgt darstellen:

$$\begin{array}{lcl} \text{Prädikatenlogik} & = & \text{Aussagenlogik} \;+\; \text{Termunifikation}\,, \\ \text{intuitionistische/modale Logik} & = & \text{klassische Logik} \;+\; \text{Präfixunifikation}\,. \end{array}$$

Der Nicht-Klausel-Konnektionskalkül verallgemeinert den Klausel-Konnektionskalkül. Es wird lediglich die Extensions-Regel leicht angepasst und eine Dekompositions-Regel hinzugefügt. Dies kombiniert die Vorteile von (Nicht-Klausel-)Sequenzen- und Tableaukalkülen mit der zielgerichteten Beweisführung von Klausel-Konnektionskalkülen.

Zukünftige Arbeiten beinhalten die Erweiterung der Kalküle und Beweiser auf weitere nicht-klassische Logiken, wie etwa multimodale Logiken und Beschreibungslogiken, als auch die Implementierung des Nicht-Klausel-Konnektionskalküls und dessen Erweiterung auf nicht-klassische Logiken. Diese Arbeiten werden weitere Belege dafür geben, dass Konnektionskalküle eine solide und erfolgreiche Grundlage für die Automatisierung des formalen Beweisens in klassischen und nicht-klassischen Logiken sind.

Literatur

[BBW06] P. Blackburn, J. van Bentham, F. Wolter. *Handbook of Modal Logic*. Elsevier, Amsterdam, 2006.

[Bib87] W. Bibel. *Automated Theorem Proving*. Vieweg, Wiesbaden, 2. Aufl., 1987.

[BOR12] C. Benzmüller, J. Otten, T. Raths. Implementing and Evaluating Provers for First-order Modal Logics. In L. De Raedt et al., Ed., *20th European Conference on Artificial Intelligence (ECAI 2012)*, S. 163–168. IOS Press, Amsterdam, 2012.

[Dal01] D. van Dalen. Intuitionistic Logic. In L. Goble, Ed., *The Blackwell Guide to Philosophical Logic*. Blackwell, Oxford, 2001.

[Fre79] G. Frege. *Begriffsschrift: Eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. L. Nebert, Halle, 1879.

[LS01] R. Letz, G. Stenz. Model Elimination and Connection Tableau Procedures. In A. Robinson, A. Voronkov, Ed., *Handbook of Automated Reasoning*, S. 2015–2114. Elsevier, Amsterdam, 2001.

[OB03] J. Otten, W. Bibel. leanCoP: lean connection-based theorem proving. *Journal of Symbolic Computation*, 36:139–161, 2003.

[Ott05] J. Otten. Clausal Connection-Based Theorem Proving in Intuitionistic First-Order Logic. In B. Beckert, Ed., *TABLEAUX 2005*, LNAI 3702, S. 245–261. Springer, Heidelberg, 2005.

[Ott08] J. Otten. leanCoP 2.0 and ileanCoP 1.2: High Performance Lean Theorem Proving in Classical and Intuitionistic Logic. In A. Armando, P. Baumgartner, G. Dowek, Ed., *IJCAR 2008*, LNCS 5195, S. 283–291. Springer, Heidelberg, 2008.

[Ott10] J. Otten. Restricting backtracking in connection calculi. *AI Communications*, 23:159–182, 2010.

[Ott11] J. Otten. A Non-clausal Connection Calculus. In K. Brünnler, G. Metcalfe, Ed., *TABLEAUX 2011*, LNAI 6793, S. 226–241. Springer, Heidelberg, 2011.

[Ott12] J. Otten. Implementing Connection Calculi for First-order Modal Logics. In E. Ternovska, K. Korovin, S. Schulz, Ed., *IWIL 2012*, EPiC, volume 22, S. 18–32. EasyChair, 2012.

[Ott13] J. Otten. Connection Calculi for Automated Theorem Proving in Classical and Non-Classical Logics. Dissertation, Institut für Informatik, University of Potsdam, 2013.

[RO12] T. Raths, J. Otten. The QMLTP Problem Library for First-order Modal Logics. In B. Gramlich et al., Ed., *IJCAR 2012*, LNAI 7364, S. 454–461. Springer, Heidelberg, 2012.

[ROK07] T. Raths, J. Otten, C. Kreitz. The ILTP problem library for intuitionistic logic. *Journal of Automated Reasoning*, 38:261–271, 2007.

[Sut08] G. Sutcliffe. The CADE-21 automated theorem proving system competition. *AI Communications*, 21:71–81, 2008.

[Sut10] G. Sutcliffe. The CADE-22 automated theorem proving system competition – CASC-22. *AI Communications*, 23:47–59, 2010.

[Sut11] G. Sutcliffe. The 5th IJCAR automated theorem proving system competition – CASC-J5. *AI Communications*, 24:75–89, 2011.

[Wal90] L. A. Wallen. *Automated Deduction in Nonclassical Logics*. MIT Press, Cambridge, 1990.

Jens Otten studierte Informatik an der Technischen Universität Darmstadt und arbeitete dort anschließend als Stipendiat und wissenschaftlicher Mitarbeiter. Aufenthalte als Gastwissenschaftler führten ihn an die Oxford Universität in England und an die Cornell Universität in Ithaca, New York. Nach mehrjähriger Tätigkeit als Technical Alliance Manager bei Hewlett-Packard in England, nahm er eine Stelle am Institut für Informatik der Universität Potsdam an, wo er in Theoretischer Informatik promovierte.

Seit mehr als zehn Jahren forscht er an der Automatisierung des formalen Beweisens, einem Kerngebiet der Künstlichen Intelligenz. Der Schwerpunkt liegt dabei auf der Entwicklung und Implementierung von Kalkülen für klassische und nicht-klassische Logiken. Er hat mehr als 30 Artikel in Büchern, Fachzeitschriften, Konferenz- und Workshop-Tagungsbänden veröffentlicht. Neben Gutachtertätigkeiten für internationale Fachzeitschriften und Konferenzen ist er Mitglied in Programmkomitees von mehreren Konferenzen und Workshops. Er ist Mitglied im Vorstand der "International Joint Conference on Automated Reasoning" (IJCAR) und Vorstandsvorsitzender der "International Conference on Automated Reasoning with Analytic Tableaux and Related Methods" (TABLEAUX).

Analyse einer praktischen fundamentalen Schranke der anonymen Kommunikation*

Vinh Pham

vinh.pham@ur.de

Abstract: Existierende Systeme zur anonymen Kommunikation können bei offenen Nutzergruppen, wie sie im Internet vorherrschen, keine informationstheoretisch perfekte Anonymität gewährleisten. Daher ist es notwendig die Schranke der Anonymität in diesem Fall zu bestimmen. Diese Arbeit analysiert diese Schranke anhand eines konkreten Anonymitätssystems, welches Chaum Mix genannt wird. Ein *Mix* ist ein System zur anonymen Netzwerkkommunikation, welches in jeder Kommunikationsrunde die Sender (Subjekte) von Nachrichten in eine *Anonymitätsmenge* einbettet, um die Zuordnung zu ihren Empfängern (Attributen) zu verdecken. Anoymitätsmengen bilden die Grundlage aller Anonymitätsysteme, um die Zuordnung zu sensiblen Attributen zu verschleiern. Die fundamentale Schranke der Anonymität wird daher durch die Schwierigkeit bestimmt aus den beobachtbaren Anonymitätsmengen und Attributen die Zuordnungen eindeutig zu rekonstruieren. In Anlehnung an Shannons Unicity-Distance bestimmen wir die Anonymitätsschranke über die minimale Anzahl an Anonymitätsmengen, die beobachtet werden muss, um die Zuordnung eindeutig aufzudecken. Wir zeigen, dass diese Aufdeckung in vielen realistischen Fällen sogar effizient berechenbar ist, obwohl sie das Lösen eines NP-vollständigen Problems erfordert.

1 Einleitung

In dieser Arbeit [Pha13] geht es um die Bestimmung (bzw. Messung) der Sicherheit von Anonymitätssystemen in Kommunikationsnetzen. Dazu analysieren wir die Schranke, ab der die Anonymität eines Subjekts durch ein System nicht mehr gewährleistet werden kann, so dass eine Deanonymisierung möglich wird. Die Erforschung dieser Anonymitätsschranke und ihrer Einflussfaktoren deckt allgemeine Schwächen der Anonymitätssysteme auf und unterstützt daher die Entwicklung von Systemen mit höherer Sicherheit.

Wir bestimmen die Anonymitätsschranke anschaulich anhand eines konkreten, aber einfachen Anonymitätssystems, das Chaum Mix [Cha81] genannt wird. Der Chaum Mix ermöglicht die Anonymiserung der Kommunikationsbeziehung zwischen Sendern und Empfängern von Nachrichten in einem unsicheren Netzwerk wie dem Internet. Abbildung 1 skizziert das Funktionsprinzip eines Mixes, das einer Wahlurne ähnelt. In jeder Kommunikationsrunde sammelt der Mix von b Sendern ($b = 3$ in Abbildung 1) ein Datenpaket gleicher Größe. Diese Pakete sind an den Mix adressiert und enthalten

*Englischer Titel der Dissertation: "Towards Practical and Fundamental Limits of Anonymity Protection"

für diesen verschlüsselt jeweils die eigentliche (mit x bezeichnete) Nachricht, sowie die Empfängeradresse. Der Mix entfernt die Verschlüsselung in den b Paketen und sortiert die daraus entnommenen (mit x gekennzeichneten) Nachrichten alphabetisch. Die Nachrichten werden in derselben Runde an die entschlüsselten Empfängeradressen weitergeleitet. Dieser Vorgang randomisiert das Aussehen und die Reihenfolge der am Mix eingehenden und ausgehenden Pakete. Ein beobachtender Angreifer (*passiver Angreifer*)[1] kann durch diese Maßnahme nur die *Anonymitätsmenge* der Sender S' und der Empfänger R' beobachten, aber nicht wer in dieser Runde mit wem kommuniziert hat.

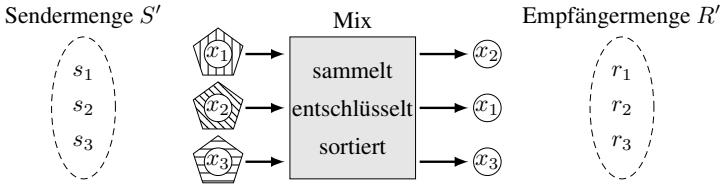


Abbildung 1: Mix Model: Jedes Muster um eine Nachricht x ist eine Verschlüsselungsebene. Mit s bzw. r wird ein Sender bzw. Empfänger bezeichnet.

Basis Modell der Anonymitätsanalyse Die Mengen (S', R') stellen die einzigen Informationen dar, die bei einer idealen Umsetzung[2] des Chaum Mixes in jeder Runde im Netzwerk beobachtbar sind. Die Anonymitätsschranke des Chaum Mixes ist daher fundamental durch die Schwierigkeit bestimmt, aus den Beobachtungen von (S', R') die Kommunikationsbeziehungen eines Senders zu identifizieren. Daher kann ohne Beeinträchtigung der Allgemeinheit die Ermittlung dieser Schranke auf die Analyse der Beobachtungen von (S', R') reduziert und vereinfacht werden. Sei *Alice* ein beliebiger Sender, dann müssen sogar nur die Mengen (S', R') betrachtet werden, in denen Alice als Sender in S' vorkommt, um ihre Kommunikationsbeziehung zu deanonymisieren. Wir nennen die Empfängermenge R' in diesem Fall eine *Observation*, so dass jede Observation wie in Abbildung 2 mindestens einen Empfänger von Alice enthält. Zur Hervorhebung wird ein Empfänger von Alice als ein *Freund* bezeichnet und mit der Variablen a adressiert, während die Variable r verwendet wird, wenn keine Unterscheidung erforderlich ist.

Erweitertes Modell der Anonymitätsanalyse Das Basis Modell ist der initiale Ansatz für die Anonymitätsanalyse in dieser Arbeit. Wir erweitern dieses Modell um die Möglichkeit von *fehlerhaften Observationen*, welche entgegen der Erwartung des Angreifers keine Freunde enthalten. Fehlerhafte Observation können entstehen, wenn z.B. der Angreifer zu schwach ist, um die Anonymitätsmengen (S', R') vollständig zu beobachten. Des Weiteren können sie induziert werden, wenn das Kommunikationsprotokoll

[1] Wir betrachten einen passiven Angreifer, weil seine Angriffe schwer erkennbar und schwer vermeidbar sind. Angriffe von aktiven Angreifer können hingegen durch die CUVE [KP06] Anforderung des Chaum Mixes erkannt werden.

[2] Diese schließt Schwächen aus, die auf die Implementierung, oder z.B. kryptographische Protokolle zurückzuführen sind, um ausschließlich die Anonymitätsfunktion zu bewerten.

Scheinnachrichten vorsieht, oder Mix Varianten wie z.B. Mixmaster [DDM03] eingesetzt werden, die Nachrichten indeterministisch weiterleiten[3].

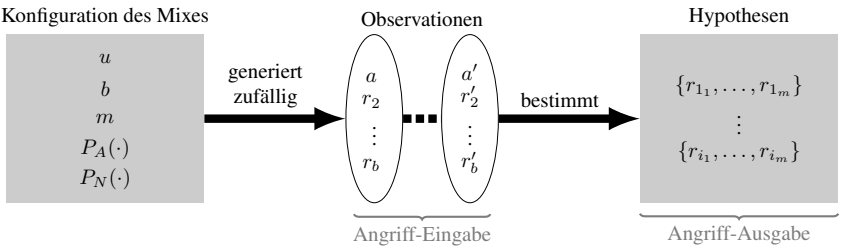


Abbildung 2: Anonymitätsanalyse: Mit r bzw. a wird ein Empfänger bzw. ein Freund bezeichnet. Aus Observationen werden Hypothesen für die möglichen m Freunde von Alice gebildet.

Wahl des Modells Die Arbeit betrachtet den Chaum Mix, weil es ein einfaches Anonymitätssystem darstellt. Dadurch sind die Einflussfaktoren auf die Anonymitätsmengen (S', R') übersichtlich. Dieses vereinfacht die Analyse der Sicherheit für unterschiedliche Einstellungen der Anonymitätsmengen. Zugleich ist der Chaum Mix die theoretische Grundlage vieler anonymer Kommunikationssysteme [EY09, DD08], so dass auch eine praktische Anwendung der Forschungsergebnisse begünstigt wird.

Erwartungswert der Anonymitätsschranke Abbildung 2 modelliert über die *Konfiguration des Mixes* die Einflussfaktoren auf die Anonymität. Damit studieren wir die erwartete Wirkung beliebiger Konfigurationen auf die Anonymitätsschranke, um den Entwurf sichererer Systeme zu unterstützen. Die Konfiguration des Mixes enthält die Gesamtzahl aller möglichen Empfänger u im System (d.h. $u = |\{r \mid r \text{ kann Nachrichten empfangen}\}|$), die Größe der Anonymitätsmenge b und die Anzahl der Freunde von Alice m. Des Weiteren sind die Wahrscheinlichkeitsfunktionen $P_A(a)$ und $P_N(r)$ enthalten, um die Verteilung der einzelnen Empfänger a, r von Alice und von den der anderen Sender in den Observationen zu modellieren. Da das Sammeln von Nachrichten aufwendig ist, nehmen wir an, dass in jeder Runde nur ein kleiner Teil (d.h. $b \ll u$) aller möglichen Empfänger kontaktiert werden.

- Wir ermitteln den Erwartungswert der theoretischen Anonymitätsschranke über die erwartete minimale Anzahl von Observationen, bis der Mix die Anonymität von Alice nicht mehr gewährleistet.[4] Diese ähnelt Shannons Unicity-Distance [Sha49].
- Zusätzlich analysieren wir den erwarteten Rechenaufwand, den diese Deanonymiserung erfordern würde. Dadurch wird ersichtlich, welche Angreifer in der Praxis diese Ressourcen aufbringen können.

[3] Im Mixmaster wird ein Paket in einer Runde zufällig weitergeleitet, oder zurückgehalten. Die Weiterleitung von Paketen ist daher indeterministisch.

[4] Diese Schranke quantifziert wie die Unicity-Distance die Information, aber nicht den Rechenaufwand für eine Deanonymisierung einer Kommunikationsbeziehung.

Diese theoretische und praktische Analyse basiert auf dem Hitting-Set Angriff (HS-Attack) [KP04]. Der Angriff erhält, wie in Abbildung 2 illustriert, die Observationen als Eingabe und leitet daraus die möglichen *Hypothesen* für die Menge der m Freunde von Alice ab. Alice ist deanonymisiert, wenn der HS-Attack genau eine Hypothese liefert. Es wurde in [KAPR06] bewiesen, dass der HS-Attack die minimale Anzahl an Observationen für eine Deanonymisierung benötigt. Die erwartete minimale Anzahl an Beobachtungen, um Alice in einer gegebenen Mix Konfiguration eindeutig zu identifizieren, entspricht dem Erwartungswert der Anonymitätsschranke. Allerdings erfordert der HS-Attack das Lösen eines NP-vollständigen Problems.

1.1 Beitrag

Der Beitrag dieser Dissertation kann in drei Bereiche unterteilt werden. Erstens werden wir den Erwartungswert der theoretischen Schranke der Anonymität für ein beliebiges Subjekt, das wir Alice nennen, durch eine geschlossene Formel approximieren. Diese Schranke ist von der Konfiguration des Chaum Mixes abhängig. Unsere Aprroximation zeigt, dass die Konfiguration des Mixes für gewöhnlich nur einen polynomiellen Einfluss auf die theoretische Schranke der Anonymität hat.

Zweitens bestimmen wir anhand eines konkreten Angriffs den mittleren Rechenaufwand, um die Freunde von Alice zu identifizieren. Dies zeigt, dass die Identifizierung für viele realistische Parameter im Mittel effizient berechenbar ist, obwohl es das Lösen eines NP-vollständigen Problems erfordert.

Drittens schlagen wir eine Erweiterung unserer Analysen auf indeterministische Mix Strategien vor, in der fehlerhafte Observationen möglich sind. Es wird gezeigt, dass Alice mit hoher Wahrscheinlichkeit deanonymisiert werden kann, wenn die Fehlerwahrscheinlichkeit der Observationen unter einer bestimmten Schranke liegt. Strategien, die durch die Induzierung fehlerhafter Observationen die Anonymität stärken wollen, müssen daher diese Schranke berücksichtigen.

2 Theoretische Schranke der Anonymität

Die theoretische Schranke der Anonymität bestimmt, ab wann der Chaum Mix die Anonymität eines Senders nicht mehr schützen kann. Wir messen diese Schranke über die minimale Anzahl an Observationen, die ein Angriff benötigt, um die Freunde von Alice eindeutig zu identifizieren. Die Schranke ist theoretisch, da sie die für einen solchen Angriff erforderliche Zeit- und Speicherkapazitäten nicht berücksichtigt.

Sei ${}_A\mathcal{H} = \{a_1, \ldots, a_m\}$ die Menge aller Freunde, die Alice während der Observationen des Angreifers wiederholt kontaktiert und $m = |{}_A\mathcal{H}|$ die Anzahl dieser Freunde. In der Basis Analyse betrachten wir den Fall, dass Alice während des Angriffs nur Empfänger in ${}_A\mathcal{H}$ kontaktiert. Diese Betrachtung wird in Kapitel 4 um fehlerhafte Observationen erweitert.

2.1 Hitting-Set Angriff

Wie in Abbildung 2 illustriert, bestimmt ein Angriff aus Observationen Hypothesen für die Menge der Freunde von Alice. Da jede Observation mindestens einen Freund von Alice enthält, ist jede Hypothese ein Hitting-Set der Größe m. Ein *Hitting-Set* ist in unserem Kontext eine Menge, die sich mit jeder Observation des Angreifers schneidet. Der Hitting-Set ist ein *Minimal-Hitting-Set*, wenn keine echte Teilmenge davon ein Hitting-Set ist. Ein Hitting-Set wird zum *Unique Minimum-Hitting-Set* (UMHS), wenn alle anderen Hitting-Sets eine höhere Kardinalität haben.

Der Hitting-Set Angriff (HS-Attack) [KP04] sammelt wiederholt Observationen und berechnet die Hitting-Sets auf alle gesammelten Observationen, bis er ein UMHS findet. Dieser UMHS identifiziert eindeutig die Freunde von Alice. Dieser Angriff nutzt den Umstand aus, dass die Sender und Empfänger in einer offenen Nutzergruppe sich fortlaufend ändern, während die Observationen immer einen Freund von Alice enthalten. Bei hinreichend vielen Observationen wird die Menge ${}_A\mathcal{H}$ aller Freunde von Alice daher zum UMHS in diesen Observationen[5]. Es wurde in [KAPR06] bewiesen, dass der HS-Attack die minimale Anzahl an Observationen benötigt, um die Freunde von Alice eindeutig zu identifizieren. Das Bestimmen des Unique Minimum-Hitting-Set ist ein NP-vollständiges Problem [GJ90].

2.2 Approximation der Anonymitätsschranke

Der Erwartungswert für die minimale Anzahl an Observationen um Alice zu deanonymisieren ist, wie in Abbildung 2 illustriert, abhängig von der Konfiguration des Mixes. Neben den Parametern u, b, m enthält diese die Wahrscheinlichkeitsfunktionen $P_A(a)$ und $P_N(r)$. Die Funktion $P_A(a)$ modelliert für jeden Freund a die Wahrscheinlichkeit, dass dieser von Alice kontaktiert wird. Für jeden möglichen Empfänger r beschreibt $P_N(r)$ hingegen die Wahrscheinlichkeit, dass r in einer Observation vorkommt, weil es von eines der anderen $(b-1)$ Sendern kontaktiert wird. Sei p der minimaler Wert von $P_A(a)$, für alle $a \in {}_A\mathcal{H}$ und P_N, der maximale Wert von $P_N(r)$ für alle möglichen Empfänger r. Sei ferner der Zusammenhang $P_N = 1 - (\frac{u-1}{u})^{b-1}$ gegeben[6].

Die Arbeit stellt folgende Approximation für den Erwartungswert der theoretischen Anonymitätsschranke im Bezug auf die Konfiguration des Mixes auf:

$$E(T_{2\times e}) \approx \left(\frac{1}{p}(\ln m + \gamma) + \frac{1}{p}\ln\ln m\right)\left(\frac{u-(m-1)}{u}\right)^{1-b} \quad , \tag{1}$$

wobei $\gamma \approx 0,57721$ die Euler-Mascheroni Konstante ist. Dabei ist u für gewöhnlich wesentlich größer als m, so dass der letzte Faktor in (1) durch $\frac{1}{1-(b-1)(\frac{m-1}{u})}$ approximiert werden kann, so dass $E(T_{2\times e})$ polynomiell ist. Daraus wird ersichtlich, dass der Chaum

[5] Es gibt theoretische Ausnahmefälle, in der dieses nicht erfüllt ist. Z.B. wenn alle Sender immer den gleichen Empfänger kontaktieren.

[6] Die Herleitung kann der Dissertation entnommen werden.

Mix die Anonymität eines Senders nur für eine polynomielle Anzahl an Kommunikationen schützt. Änderungen an der Konfiguration des Mixes würden nur eine polynominelle Wirkung auf den Erwartungswert der theoretischen Anonymitätsschranke haben.

Evaluation Abbildung 3 veranschaulicht die Anonymitätsschranke für verschiedene Konfigurationen des Mixes. Es vergleicht die Approximation ($E(T_{2\times e})$) mit der Erwarteten Anzahl an simulierten Observationen, die der HS-Attack für die Deanonymisierung von Alice benötigt (HS). Die Observationen werden wie in Abbildung 2 zufällige generiert. Dabei haben wir vereinfachend eine gleichförmige Wahrscheinlichkeitsfunktion $P_N(r) = 1 - (\frac{u-1}{u})^{b-1}$ gewählt[7]. Alice kontaktiert ihre Freunde nach der Zipf(m, α) Verteilung $P_A(i) = \frac{i^{-\alpha}}{\sum_{l=1}^{m} l^{-\alpha}}$, für $a_i \in {}_A\mathcal{H}$, $i = \{1, \ldots, m\}$.[8] Die Zipf-Verteilung ähnelt für $\alpha \approx 1$ der Verteilung des E-Mail Verkehrs eines Nutzers [BCF+99, AH02]. Die Anonymitätschranken in Abbildung 3 bestätigen, dass der Chaum Mix die Anonymität nur für eine relativ kleine Anzahl von Runden gewährleistet. Im rechten unteren Bild steigt die Anzahl der Beobachtungen exponentiell mit α. Das ist kein Widerspruch zum Kapitel 2.2, da der minimaler Wert p der Funktion $P_A(i)$ mit wachsendem α exponentiell abnimmt.

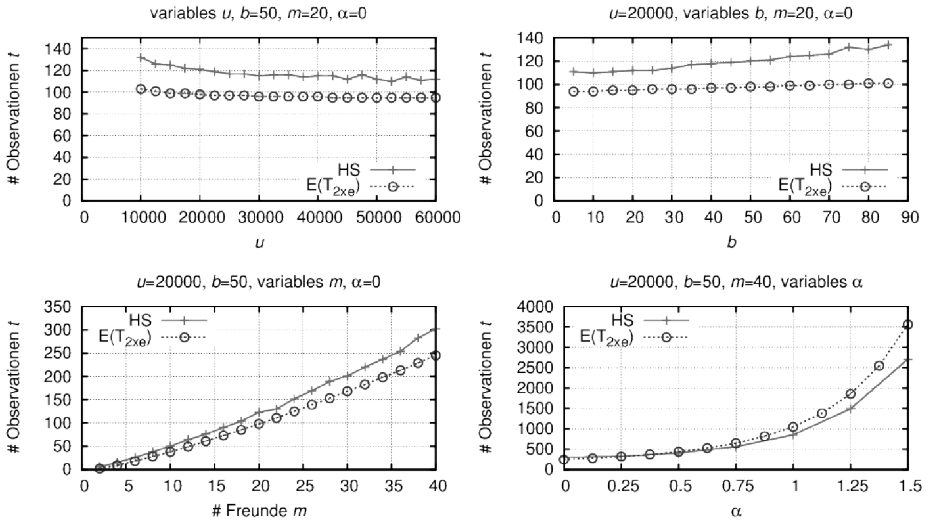


Abbildung 3: Empirischer und theoretischer Erwartungswert der Anonymitätsschranke: Empirisch durch Simulation mit HS-attack (HS) und theoretisch durch Approximation ($E(T_{2\times e})$).

[7] Es können auch komplexere Verteilungen für $P_N(r)$ betrachtet werden. Allerdings wäre die mathematische Analyse unnötig umständlicher.

[8] Da Alice durch ihr Verhalten einen direkten Einfluss auf ihre Anonymität hat, wird die Wirkung ihrer Kommunikation detailliert analysiert.

3 Rechenaufwand der Deanonymisierung

Der HS-Attack und die damit ermittelte Anonymitätsschranke wurde als eine rein theoretische Analyse betrachtet. Um ein UMHS zu identifizieren, wird ein Algorithmus verwendet, der alle möglichen Hitting-Sets der Größe m in den gesammelten Observationen berechnet. Wenn genügend viele Observationen vorhanden sind, liefert dieser Algorithmus genau ein Hitting-Set, der zugleich ein UMHS ist, vergl. [KP04]. Da es anfänglich[9] $\binom{u}{m}$ Hitting-Sets gibt, ist die Komplexität des Algorithmus $O(\binom{u}{m})$, was für realistische Gruppengrößen inpraktikabel ist. Diese Dissertation stellt einen neuen Algorithmus vor, der für viele realistische Konfigurationen des Mixes das UMHS effizient identifizieren kann.

3.1 ExactHS Algorithmus

In dieser Arbeit haben wir den *ExactHS* Algorithmus entworfen, der für eine gegebene Menge an Observationen nur die Minimal-Hitting-Sets bis zu einer gegebenen Kardinalität (z.B. m) berechnet. Da jedes Hitting-Set, das nicht minimal ist, eine Obermenge eines Minimal-Hitting-Sets ist, führt der ursprünglicher Algorithmus des HS-Attack durch die Betrachtung aller Hitting-Sets der größe m zu einem erheblich höheren Rechenaufwand. Wir beweisen in dieser Dissertation, dass es maximal b^m Minimal-Hitting-Sets gibt und die Worst-Case Laufzeit des ExactHS proportional dazu ist, während dessen Speicherkomplexität nur linear ist. Zum Beispiel ist bei einer Mixkonfiguration mit $u = 400$ ($u = 10000$) Empfängern, einer Anonymitätsmenge der Größe $b = 10$ ($b = 50$) bei einer Anzahl von $m = 10$ ($m = 20$) Freunden, die Laufzeitkomplexität des ursprünglichen Algorithmus 10^{19} (10^{61}), während sie beim ExactHS 10^{10} (10^{34}) ist. Die Verwendung des ExactHS, um ein UMHS zu identifizieren liefert bezüglich der minimalen Anzahl an erforderlichen Observationen dasselbe Ergebnis, wie der ursprüngliche Algorithmus des HS-Attack. Der HS-Attack unter der Verwendung des ExactHS zur Bestimmung der Minimum-Hitting-Sets stellt einen effizienteren Entwurf des HS-Attack dar. Daher beziehen wir uns in dem restlichen Text nur noch auf die effizientere Version, wenn wir den Begriff HS-Attack verwenden.

3.2 Mittlere Komplexität des ExactHS

Der ExactHS ist insbesondere praktisch verwendbar, weil seine mittlere Laufzeitkomplexität in vielen Fällen signifikant geringer als seine Worst-Case Komplexität ist. Die Dissertation beweist folgende mathematische Zusammenhänge für die mittlere Laufzeitkomplexität des ExactHS:

- Wenn die Wahrscheinlichkeit, dass ein Empfänger von irgendeinem Sender in einer Runde kontaktiert wird, der nicht Alice ist, höchstens $\frac{1}{m^2}$ ist (d.h. $P_N \leq \frac{1}{m^2}$), dann ist die mittlere Laufzeitkomplexität linear.

[9] Wenn noch keine Observation gesammelt wurde.

- Wenn $P_N \leq \frac{1}{m}$ ist, und Alice ihre Freunde ähnlich einer Zipf(m, α) Verteilung kontaktiert (d.h. $P_A(i) = \frac{i^{-\alpha}}{\sum_{l=1}^{m} l^{-\alpha}}$, für $a_i \in {}_A\mathcal{H}$, $i = \{1, \ldots, m\}$), dann konvergiert die mittlere Laufzeitkomplexität mit steigendem α gegen ein Polynom. Tendenziell führt eine ungleichförmige Verteilung der Nachrichten von Alice an ihre Freunde zu einer Verringerung der mittleren Laufzeitkomplexität.

Es wurde in der Literatur empirisch bestätigt, dass der E-Mail Verkehr und der Datenverkehr im Internet durch eine Zipf-Verteilung modelliert werden kann [BCF+99, AH02]. Daher ist der verbesserte HS-Attack insbesondere bei vielen realistischen Verteilungen effizient durchführbar. Zum Beispiel liegt die Worst-Case Komplexität von ExactHS für $b = 50, m = 40$ bei $O(50^{40}) \approx O(10^{68})$. Die in dieser Dissertation mathematisch und simulativ[10] ermittelte Komplexität führt hingegen zu den signifikant geringeren Werten in Tabelle 1. Es ist auch erkennbar, dass mit zunehmendem α die Komplexität weiter sinkt.

α	Theoretische Anzahl	Empirische Anzahl
0.0	1.8×10^7	1.7×10^6
0.5	1.0×10^5	1.5×10^5
1.0	8.7×10^3	2×10^4
1.5	2.2×10^3	1.7×10^3

Tabelle 1: Evaluierter Mengen zur Identifizierung des UMHS durch ExactHS: Für Mixparameter $u = 20000, b = 50, m = 40$ und Zipf(m, α) Verteilung der Nachrichten von Alice und $P_N < \frac{1}{m}$.

4 Erweiterte Analyse mit fehlerhaften Observationen

Bei der Verwendung des HS-Attack wird angenommen, dass die Observationen des Angreifers fehlerfrei sind. Die Dissertation zeigt, dass der HS-Attack adaptiert werden kann, um die Freunde von Alice mit einer hohen Wahrscheinlichkeit zu identifizieren, wenn die Fehlerrate der Observationen eine bestimmte Schranke unterschreitet. Eine Observation ist *fehlerhaft*, wenn sie entgegen der Erwartung des Angreifers keinen Freund von Alice enthält. Fehlerhafte Observationen können durch das Senden von Scheinnachrichten, oder unvollständige Beobachtungen der Anonymitätsmengen entstehen. Zum anderen können diese bei Mix Varianten auftreten, die Nachrichten indeterministisch weiterleiten, wie z.B. Pool-Mixe [DDM03]. Bei Pool-Mixen (wie Mixmaster [DDM03]) besteht eine Wahrscheinlichkeit p_{er} dass die Weiterleitung einer Nachricht auf eine spätere Runde aufgeschoben wird. Ein Angreifer, der die Empfängermenge nur in der Runde des Pool-Mixes beobachtet, in der Alice eine Nachricht sendet, wird daher mit der Wahrscheinlichkeit p_{er} eine fehlerhafte Observation erhalten.

[10] Dafür wurden entsprechend der Konfiguration des Mixes zufällige Observationen erzeugt und der HS-Attack auf diese angewendet, bis alle Alices Freunde identifiziert wurden. Diese Simulation wurde mehrmals wiederholt, bis 95% der Ergebnisse sich um maximal 5% vom empirischen Mittelwert unterschieden.

Die Dissertation zeigt den folgenden Zusammenhang: Wenn die Wahrscheinlichkeit einer fehlerhaften Observation die Ungleichung

$$p_{er} < \frac{1}{(\frac{1}{p}\ln\ln m)\left(\frac{u-(m-1)}{u}\right)^{1-b}+1} \tag{2}$$

erfüllt, dann konvergiert die Wahrscheinlichkeit, dass Alices Freunde durch den HS-Attack identifiziert werden kann, mit steigender Anzahl an Observationen gegen 1. In (2) is p der minimale Wert der Wahrscheinlichkeitsfunktion $P_A(a)$, für $a \in {}_A\mathcal{H}$.

Die Verwendung von Pool-Mixen schützt somit nicht gegen den erweiterten HS-Attack, wenn die Fehlerwahrscheinlichkeit von Observationen (2) erfüllt. Höhere Werte für p_{er} führen jedoch zu stärkeren Verzögerungen der Pakete. Allerdings könnte, selbst wenn (2) nicht erfüllt wird, die möglichen Hypothesen für die Freunde von Alice durch den HS-Attack stark einschränkt werden.

5 Zusammenfassung

Diese Arbeit [Pha13] beweist, dass der Chaum Mix einen Sender im Mittel nur für eine polynomielle Anzahl an Kommunikationen schützen kann. Durch eine Verbesserung der Effizienz des ursprünglichen HS-Attack und der Analyse der mittleren Laufzeit des Angriffs können wir zeigen, dass die Deanonymisierung eines Senders (d.h. die Identifizierung seiner Empfänger) in vielen realistischen Fällen sogar effizient möglich ist. Aus diesen Ergebnissen folgern wir, dass der Chaum Mix (der deterministisch Nachrichten weiterleitet), keinen starken Schutz gegenüber dem HS-Attack bietet.

Wir haben den HS-Attack daher erweitert, um auch Mix Varianten betrachten zu können, die indeterministisch Nachrichten weiterleiten, wie dem Pool-Mix. Dieser Indeterminismus führt zu fehlerhaften Observationen. Wir haben gezeigt, dass der HS-Attack die Freunde von Alice mit einer hohen Wahrscheinlichkeit identifizieren kann, wenn die Fehlerrate der Observationen einen bestimmten Grenzwert unterschreitet. Wenn dieser Grenzwert berücksichtig wird, können indeterministische Mix Varianten einen besseren Schutz vor dem HS-Attack bieten, als der Chaum Mix. Allerdings führt der Indeterminismus zu zusätzlichen Verzögerungen des Datenverkehrs und einer Aufweichung des Angreifermodells, im Vergleich zum Chaum Mix. Die Verzögerung erschwert insbesondere die Erkennung von aktiven Angriffen, wie z.B. das Blocken von $(b-1)$ Nachrichten vor dem Mix[11] [Cha81, SDS03], da Verzögerungen konzeptbedingt auch im Normalfall auftreten. Im Gegensatz dazu ermöglicht die deterministische Arbeitsweise des Chaum Mixes die Erkennung aktiver Angriffe durch die CUVE [KP06] Anforderungen.

Die vorliegende Arbeit zeigt die Schwächen bestehender Mix Varianten auf und motiviert daher die Forschung, nach neuen Techniken zu suchen, die eine Balance zwischen den genannten Vorteilen und Nachteilen herstellen.

[11] Wenn nur noch eine Nachricht den Mix passiert, dann ist dessen Sender und Empfänger nicht mehr anonym.

Literatur

[AH02] Lada A. Adamic und Bernardo A. Huberman. Zipf's Law and the Internet. *Glottometrics*, 3:143 – 150, 2002.

[BCF+99] Lee Breslau, Pei Cao, Li Fan, Graham Phillips und Scott Shenker. Web Caching and Zipf-like Distributions: Evidence and Implications. In *Proceedings of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM '99*, Jgg. 1, Seiten 126 – 134. IEEE, 1999.

[Cha81] David L. Chaum. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. *Communications of the ACM*, 24(2):84 – 88, 1981.

[DD08] George Danezis und Claudia Diaz. A Survey of Anonymous Communication Channels. Bericht MSR-TR-2008-35, Microsoft Research, 2008.

[DDM03] George Danezis, Roger Dingledine und Nick Mathewson. Mixminion: Design of a Type III Anonymous Remailer Protocol. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, Seiten 2 – 15. IEEE, 2003.

[EY09] Matthew Edman und Bülent Yener. On Anonymity in an Electronic Society: A Survey of Anonymous Communication Systems. *ACM Computing Surveys*, 42(1):1 – 35, 2009.

[GJ90] Michael R. Garey und David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, 1990.

[KAPR06] Dogan Kesdogan, Dakshi Agrawal, Vinh Pham und Dieter Rauterbach. Fundamental Limits on the Anonymity Provided by the Mix Technique. In *Proceedings of the 2006 IEEE Symposium on Security and Privacy*, Seiten 86 – 99. IEEE, 2006.

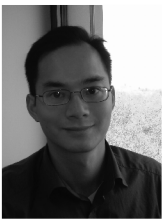
[KP04] Dogan Kesdogan und Lexi Pimenidis. The Hitting Set Attack on Anonymity Protocols. In *Information Hiding*, Jgg. 3200 of *LNCS*, Seiten 326 – 339. Springer, 2004.

[KP06] Dogan Kesdogan und Charles Palmer. Technical Challenges of Network Anonymity. *Computer Communications*, 29(3):306 – 324, 2006.

[Pha13] Dang Vinh Pham. *Towards Practical and Fundamental Limits of Anonymity Protection*. Dissertation, University of Regensburg, November 2013.

[SDS03] Andrei Serjantov, Roger Dingledine und Paul F. Syverson. From a Trickle to a Flood: Active Attacks on Several Mix Types. In *Information Hiding*, Jgg. 2578 of *LNCS*, Seiten 36 – 52. Springer, 2003.

[Sha49] Claude Shannon. Communication Theory of Secrecy Systems. *Bell System Technical Journal*, 28:656 – 715, 1949.



Vinh Pham wurde 1981 in Saigon (Vietnam) geboren. Er erhielt 2006 das Diplom für Informatik an der Rheinisch-Westfälischen Technischen Hochschule Aachen (RWTH-Aachen). Ergebnisse der Diplomarbeit führten zu der Publikation "Fundamental limits on the anonymity provided by the MIX technique" bei der IEEE Security&Privacy 2006. Nach seinem Abschluss war er wissenschaftlicher Mitarbeiter im RWTH UMIC Exzellenz Cluster. Das Forschungsinteresse für den Schwerpunkt Security and Privacy führte ihn 2008 an die Universität Siegen und dann Regensburg. Seine Promotion schloss er 2013 in Regensburg ab, wo er bis heute seine Forschungsrichtung in der Lehre und in Projekten weiterverfolgt.

Analyse von eingebetteten Echtzeitsystemen zur Laufzeit*

Thomas Reinbacher - thomas.reinbacher@gmail.com
Institut für Technische Informatik, Technische Universität Wien

Abstract: Die in dieser Kurzfassung beschriebene Dissertation stellt einen auf Hardware basierenden Ansatz für die automatisierte Analyse eingebetteter Systeme vor. Die Analyse erfolgt in Echtzeit während der operativen Phase dieser Systeme und ermöglicht es, Bugs zuverlässig zu erkennen und eine umfangreiche Fehlerdiagnose durchzuführen. Dieser Ansatz wurde erfolgreich am NASA Luftfahrzeug Swift erprobt.

Effiziente moderne Entwicklungstools erlauben es, hoch komplexe eingebettete Systeme zu entwickeln, welche zentrale Aufgaben in Komponenten der Automobilindustrie, der Luft- und Raumfahrt, etc. übernehmen. Die Verifikation dieser Systeme übersteigt dann jedoch oft die Möglichkeiten der verfügbarer Methoden [AO08]. Die Gefahr, dass Bugs während der Testphase unentdeckt bleiben, wird folglich größer [JGK+11]. Ein Beispiel für diese Gattung von Systemen ist das unbemannte Luftfahrzeug Swift [IEW10], welches am NASA Ames Research Center entwickelt wird und für irdische, wissenschaftliche Missionen [Pan12] eingesetzt wird. Abbildung 1 zeigt eine vereinfachte Darstellung, inklusive Sensoren, Aktuatoren, Flugcomputer und Kommunikationsinterface.

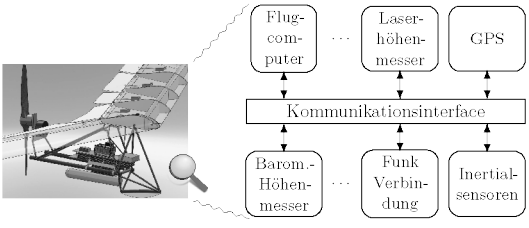


Abbildung 1: Aufbau des unbemannten Luftfahrzeuges Swift der NASA (vereinfacht).

Mit dem häufigeren Einsatz von unbemannten Luftfahrzeugen, Systemen ohne erfahrene Piloten als letzte Entscheidungsinstanz, in nationalen Lufträumen und somit dicht besiedelten Gebieten der EU [Eur12] und der USA [Dep05] muss die Frage gestellt werden, wie mit Fehlfunktionen dieser Luftfahrzeuge während des Fluges umgegangen wird. Der Verlust des Luftfahrzeugs sowie eine potentielle Gefährung von Personen am Boden sind unter allen Umständen zu verhindern. Angesichts der erwähnten Zunahme der allgemeinen Systemkomplexität beschränkt sich diese Problematik nicht nur auf den speziellen Fall der unbemannten Luftfahrzeuge, sondern stellt die Informatik vor die generelle Herausforderung, neue Ansätze zu entwickeln um mit Bugs, welche in der operativen Phase eines Systems auftreten, geeignet umzugehen. Die hier beschriebene Dissertation [Rei13] stellt einen solchen Ansatz für die automatisierte Analyse eingebetteter Systeme vor. Die Analyse erfolgt in Echtzeit während der operativen Phase dieser Systeme und ermöglicht es, Bugs zuverlässig zu erkennen und eine umfangreiche Fehlerdiagnose durchzuführen. Dieser Ansatz wurde am NASA Luftfahrzeug Swift erprobt.

*Englischer Titel der Dissertation: "Analysis of Embedded Real-Time Systems at Runtime"

Highlight 1: Anforderungsanalyse

Die Anforderungen der angestrebten Analysekomponente ergeben sich einerseits aus den Beschränkungen bei Speicher- und Rechenkapazität, Echtzeitanforderungen, usw. des zu analysierenden eingebetteten Systems [MHA+95, Men07] und andererseits aus Gesichtspunkten einer möglichst großflächigen Akzeptanz in der Industrie [JGK+11]:

UNOBTRUSIVENESS: Die Analysekomponente darf die wesentlichen Eigenschaften des zu analysierenden Systems nicht verändern. Dazu gehören die Funktionalität, Zertifizierbarkeit, temporale Eigenschaften, und sonstige Toleranzen wie Größe, Gewicht, Leistungsaufnahme, und die Telemetriebandbreite. Die Analysekomponente muss außerhalb des zu analysierenden Systems operieren und somit eine in sich geschlossene Implementierung ermöglichen.

RESPONSIVENESS: Die Analysekomponente muss das zu untersuchende System fortlaufend und regelmäßig überprüfen. Abweichungen von der Spezifikation müssen innerhalb einer engen und im Vorfeld bekannten Zeitspanne abgeschlossen sein. Nur dies ermöglicht die Einleitung von effektiven Gegenmaßnahmen um im Fehlerfall Beschädigungen am System und dessen Umwelt abzuwenden. Ein Beispiel ist die Einleitung einer koordinierten Notlandung sobald ein Ausfall des Flugcomputers des NASA Swift festgestellt wird.

REALIZABILITY: Die Analysekomponente muss nach dem plug-and-play Prinzip funktionieren und generische Schnittstellen bereitstellen um in möglichst vielen Systemen integriert werden zu können. Die unterstützte Spezifikationssprache muss einfach in einen bestehenden Arbeitsablauf integriert werden können aber auch ausdrucksstark sein, um Echtzeitanforderungen zu formulieren. Die Analysekomponente muss flexibel und ohne aufwendigen Kompilierungsprozess auf Änderungen in der Spezifikation reagieren können.

Highlight 2: Architektur zur automatisierten Analyse

Bestehende Ansätze für die Überprüfung von Abläufen in einem System bedingen meist die Instrumentierung der Hard- und/oder Software des zu prüfenden Systems und einen ressourcenintensiven Host Computer zur Ausführung des Korrektheitschecks. In der Praxis ist eine Instrumentierung jedoch schwer zu bewerkstelligen, da eingebettete Systeme oft aus einer Vielzahl von Komponenten unterschiedlichen Ursprungs bestehen. Manche sind zudem bereits zertifiziert oder stehen nur als Blackbox zur Verfügung und können daher nicht modifiziert werden. Bestehende Ansätze stehen somit im Widerspruch zu den Anforderungen bezüglich UNOBTRUSIVENESS, RESPONSIVENESS, und REALIZABILITY. Diese Anforderungen können am besten durch eine Auslegung der Analysekomponente als Hardwareeinheit bedient werden. In Anlehnung an die vorgestellten Anforderungen wird die entwickelte Analysekomponente als rtR2U2 bezeichnet (engl. für real-time, Realizable, Responsive, Unobtrusive Unit). Abbildung 2 zeigt eine überblicksmäßige Darstellung. Anstelle einer Instrumentierung baut rtR2U2 auf das passive Abgreifen aller relevanten Informationen an bereits vorhandenen Kommunikationsinterfaces auf (z.B.: industrielle Bussysteme wie CAN, FlexRay, PCI). Dies erlaubt das Nachrüsten von rtR2U2 in bestehenden Systemen und die einfache Integration in neue Designs. Durch die beabsichtigte

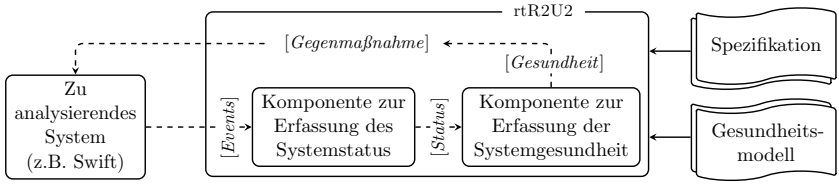


Abbildung 2: rtR2U2: Eine Analysekomponente für eingebettete Systeme.

Hardwareimplementierung von rtR2U2 muss auf Designansätze verzichtet werden die oft das Rückgrat von SW-Implementierungen darstellen: Schleifen und Rekursion (werden von CAD Tool ausgerollt und resultieren in duplizierten Hardwareinstanzen), dynamischer Speicher (Speicherplatz muss für Hardwaredesigns statisch festgelegt werden), sowie das Durchsuchen von Speicherbereichen (konstante Laufzeit kann nicht garantiert werden). Wie in Abbildung 2 dargestellt, arbeitet rtR2U2 zwei Schritten:

(i) Erfassung des Systemstatus mittels neuartigen Verifikationsalgorithmen.

Definition 1 (Systemstatus)**.** Ein Vektor $\mathfrak{s} = (\mathfrak{s}_1, \ldots, \mathfrak{s}_n)$ mit je einem Eintrag für alle n Elemente der Spezifikation. Ein Eintrag beinhaltet das Resultat der Überprüfung des zugehörigen Elements der Spezifikation. ■

Um der Anforderung REALIZABILITY gerecht zu werden, müssen formale Spezifikationen in verschiedenen Ausprägungen von temporaler Logik unterstützt werden. Diese erlauben es, temporale und logische Anforderungen an das zu analysierende System formal zu definieren. Das Herzstück dieser Komponente ist eine neuartige Familie an hoch-parallelen Algorithmen, welche den Systemstatus im Hinblick auf die vorliegende temporale Spezifikation kontinuierlich zur Laufzeit feststellen kann, siehe **Highlight 3**.

(ii) Erfassung der Systemgesundheit mittels einer neuartigen Hardwarearchitektur zur Auswertung von Bayes'schen Netzen.

Definition 2 (Systemgesundheit)**.** Ein Vektor $\mathfrak{g} = (\mathfrak{g}_1, \ldots, \mathfrak{g}_m)$ mit je einem Eintrag für die m Subsysteme des analysierten Systems. Jeder Eintrag ist eine bedingte Wahrscheinlichkeit, welche für einen Systemstatus $\mathfrak{s}$ den Gesundheitszustand eines Subsystems darstellt. ■

Diese Komponente verarbeitet die Resultate der vorgelagerten Komponente zur Erfassung des Systemstatus mit Hilfe der Theorie der Bayes'schen Statistik. Die Komponente kann Gesundheitsmodelle auswerten, welche als Bayes'sche Netze spezifiziert wurden. Das Bayes'sche Netz wird hierfür einmalig in einen Graphen übersetzt, um dann von einer hoch-parallelen Hardwareeinheit zur Laufzeit ausgewertet zu werden, siehe **Highlight 4**.

Highlight 3: Algorithmen zur Erfassung des Systemstatus

rtR2U2 verwendet eine Echtzeituhr als Zeitbasis. Mit jedem Tick $n \in \mathbb{N}_0$ dieser Uhr werden die relevanten Signale (Events in Abbildung 2) des zu analysierenden Systems abgetastet und digitalisiert. Die temporale Abfolge dieser Signale wird als Execution e bezeichnet

und ist eine unendliche Sequenz von Zuständen. Abbildung 3 zeigt die Resultate der Überprüfung des Zustandsprädikats $(\text{pitch} \geq 5°)$ anhand der Execution e. $e^n \vDash (\text{pitch} \geq 5°)$ ist immer dann erfüllt, wenn der aktuell gemessene Pitchwinkel größer oder gleich 5° ist, z.B.: für $n = 13$ gilt $e^{13} \vDash (\text{pitch} \geq 5°)$, für $n = 14$ jedoch $e^{14} \nvDash (\text{pitch} \geq 5°)$.

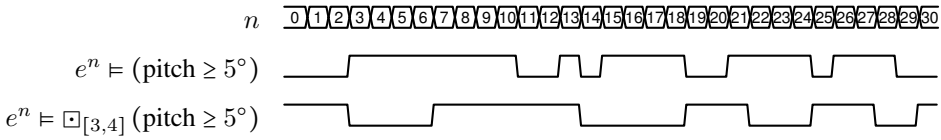


Abbildung 3: Zeitlicher Verlauf der Resultate der Überprüfung von zwei Spezifikationen.

Neben diesen einfachen Zustandsprädikaten, erlaubt rtR2U2 auch Spezifikationen in einer Vielzahl von temporalen Logiken [Pnu77] (ptLTL, ptMTL, LTL und MTL) um Echtzeiteigenschaften wie etwa harte Deadlines oder Flight Rules des Systems zu spezifizieren. Ein Beispiel ist die Eigenschaft $e^n \vDash \boxdot_{[3,4]} (\text{pitch} \geq 5°)$ die nur dann wahr ist, wenn pitch $\geq 5°$ in einem Intervall, welches in der Vergangenheit liegt und durch $J = [3, 4]$ relativ zum aktuellen Zeitstempel n beschrieben wird, wahr ist. Die Semantik dieses Operators der Logik ptMTL ist rekursiv auf einer Execution wie folgt definiert: $e^n \vDash \boxdot_J \varphi$ iff $\forall i (0 \leq i \leq n) : (\neg(n - i \in J) \vee e^i \vDash \varphi)$. Anhand dieser Semantik kann die Spezifikation $\boxdot_{[3,4]} (\text{pitch} \geq 5°)$ anhand der Execution e aus dem Beispiel in Abbildung 3 ausgewertet werden. Es lässt sich zeigen, dass $e^{21} \vDash \boxdot_{[3,4]} (\text{pitch} \geq 5°)$ (weil $e^{17} \vDash (\text{pitch} \geq 5°)$ und $e^{18} \vDash (\text{pitch} \geq 5°)$) aber $e^{22} \nvDash \boxdot_{[3,4]} (\text{pitch} \geq 5°)$. Für rtR2U2 wurden effektive Algorithmen entwickelt um solche Echtzeiteigenschaften unter den Gesichtspunkten der UNOBTRUSIVENESS, RESPONSIVENESS, und REALIZABILITY zur Laufzeit auszuwerten. Einer dieser insgesamt 13 Algorithmen wird nun kurz vorgestellt.

Algorithmus zur Auswertung des Operators $\boxdot_J \varphi$. Der Algorithmus zur Auswertung von $\boxdot_J \varphi$ in Alg. 1 benötigt eine Liste von 2-Tupel von Zeitstempeln, welche wir mit $l_{\boxdot_J \varphi}$ bezeichnen. Die Zeitstempel werden von der Echtzeituhr abgeleitet. Wir definieren das Prädikat $\textbf{valid}^{\boxdot}(T, n, J)$ mit $T \in (\mathbb{N}_0 \cup \{\infty\})^2$ wie folgt

$$\textbf{valid}^{\boxdot}(T, n, J) \equiv (T.\tau_s \leq \max(0, n - \max(J))) \wedge (T.\tau_e \geq n - \min(J)),$$

und das Prädikat $\textbf{feasible}(T, n, J)$ als

$$\textbf{feasible}(T, n, J) \equiv (T.\tau_e - T.\tau_s \geq \max(J) - \min(J)) \vee (T.\tau_s = 0 \wedge T.\tau_e \geq n - \min(J)).$$

Beispiel 1 (Algorithmus für $\boxdot_J \varphi$)**.** Wir betrachten die zeitbehaftete Spezifikation $\varphi := \boxdot_{[3,4]} (\text{pitch} \geq 5°)$ und die Execution e in obigem Beispiel. Zum Zeitpunkt $n = 0$ ist $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5°}$ leer, die Abfrage in Zeile 5 ist somit nicht erfüllt. Weil $\min(J) = 3$, liefert der Algorithmus **true** in Zeile 11 zurück. Zum Zeitpunkt $n = 3$ tritt eine ⎍⎾ Transition des Prädikats pitch $\geq 5°$ auf, und das Element $(3, \infty)$ wird zu $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5°}$ hinzugefügt (Zeile 3). Somit wird das Prädikat $\textbf{valid}^1$ in Zeile 11 wie folgt ausgewertet

$$\textbf{valid}^{\boxdot}(l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5°}[1], 3, [3, 4]) \quad = \quad (3 \leq 3 - 4) \wedge (\infty \geq 3 - 3) \quad = \quad \textbf{false},$$

[1] Wir verwenden die Notation $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5°}[i]$ um das i^{te} Element in $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5°}$ zu referenzieren.

Alg. 1 Algorithmus zur Auswertung des Operators $\boxdot_J \varphi$. Anfangsbedingung $l_{\boxdot_J \varphi} = ()$.

1: At each time $n \in \mathbb{N}_0$:
2: **if** $\_\lceil$ transition of φ occurs **then**
3: add (n, ∞) to $l_{\boxdot_J \varphi}$
4: **end if**
5: **if** $\rceil\_$ transition of φ occurs **and** $l_{\boxdot_J \varphi}$ is non-empty **then**
6: remove tail element (τ_s, ∞) from $l_{\boxdot_J \varphi}$
7: **if feasible**$((\tau_s, n-1), n, J)$ **then**
8: add $(\tau_s, n-1)$ to $l_{\boxdot_J \varphi}$
9: **end if**
10: **end if**
11: **return** $\bigvee_{k=1}^{|l_{\boxdot_J \varphi}|}$ **valid**$^{\boxdot}(l_{\boxdot_J \varphi}[k], n, J)$ in case $n \geq \min(J)$ and **true** otherwise

und wir erhalten $e^3 \not\models \psi$. Zum Zeitpunkt $n = 4$ wird Zeile 11 wie folgt ausgewertet

$$\textbf{valid}^{\boxdot}(l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}[1], 4, [3,4]) \quad = \quad (3 \leq 4-4) \wedge (\infty \geq 4-3) \quad = \quad \textbf{false},$$

und wir erhalten ebenfalls $e^4 \not\models \varphi$. Dieses Resultat wiederholt sich zu den Zeitpunkten $n \in [5,6]$. Zum Zeitpunkt $n = 7$ wird Zeile 11 wie folgt ausgewertet

$$\textbf{valid}^{\boxdot}(l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}[1], 7, [3,4]) \quad = \quad (3 \leq 7-4) \wedge (\infty \geq 7-3) \quad = \quad \textbf{true},$$

und wir erhalten somit $e^7 \models \varphi$. Durch eine ähnliche Argumentation erhalten wir $e^n \models \varphi$ für $n \in [8,10]$. Zum Zeitpunkt $n = 11$ tritt eine $\rceil\_$ Transition des Prädikats pitch $\geq 5^\circ$ auf und der Algorithmus ersetzt das Element $(3, \infty)$ in $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}$ durch $(3, 10)$ in Zeile 8. Für die Zeitpunkte $n \in [11,12]$ ist die Ausgabe des Algorithmus $e^n \models \varphi$. Zum Zeitpunkt $n = 13$ tritt eine $\_\lceil$ Transition des Prädikats pitch $\geq 5^\circ$ auf und der Algorithmus fügt das Element $(13, \infty)$ in $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}$ ein. Das Prädikat **valid**$^{\boxdot}$ ist erfüllt, daher gilt $e^{13} \models \varphi$. Zum Zeitpunkt $n = 14$ tritt eine $\rceil\_$ Transition des Prädikats pitch $\geq 5^\circ$ auf. Weil $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}$ nicht leer ist, wird das Prädikat **feasible** in Zeile 7 wie folgt ausgewertet

$$\textbf{feasible}((13,13), 14, [3,4]) \quad = \quad (13-13 \geq 4-3) \vee (13 = 0 \wedge 13 \geq 14-3) \quad = \quad \textbf{false}.$$

Das Element $(13,13)$ wird somit nicht in die $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}$ eingefügt. Das Prädikat **valid**$^{\boxdot}$ wird in Zeile 11 ausgewertet zu

$$\textbf{valid}^{\boxdot}(l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}[1], 14, [3,4]) \quad = \quad (3 \leq 14-4) \wedge (10 \geq 14-3) \quad = \quad \textbf{false},$$

und wir erhalten $e^{14} \not\models \varphi$. Zum Zeitpunkt $n = 15$ tritt eine $\_\lceil$ Transition des Prädikats pitch $\geq 5^\circ$ auf und das Element $(15, \infty)$ wird in Zeile 3 zu $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}$ hinzugefügt. Durch eine ähnliche Argumentation ergibt sich $e^n \not\models \varphi$ für die Zeitpunkte $n \in [16,18]$. Zum Zeitpunkt $n = 19$ tritt eine $\rceil\_$ Transition des Prädikats pitch $\geq 5^\circ$ auf. Das Element $(15,18)$ erfüllt das Prädikat **feasible** in Zeile 7 und somit wird $(15,18)$ zu $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}$ hinzugefügt. In Zeile 11 müssen nun zwei Elemente in $l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}$, nämlich $(3,10)$ und $(15,18)$, überprüft werden. Somit erhalten wir $e^{19} \models \psi$, weil

$$\begin{aligned}
\textbf{valid}^{\boxdot}(l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}[1], 19, [3,4]) \quad &= \quad (3 \leq 19-4) \wedge (10 \geq 19-3) \quad = \quad \textbf{false}, \\
\textbf{valid}^{\boxdot}(l_{\boxdot_{[3,4]}\,\text{pitch}\geq 5^\circ}[2], 19, [3,4]) \quad &= \quad (15 \leq 19-4) \wedge (18 \geq 19-3) \quad = \quad \textbf{true}.
\end{aligned}$$

Korrektheitsbeweise, Komplexität, und Abbildung in Hardware. In der Dissertation werden insgesamt 13 Algorithmen zur Auswertung aller Operatoren der unterstützten temporalen Logiken vorgestellt. Damit können sowohl Eigenschaften überprüft werden die in der Vergangenheit (relativ zum aktuellen Zeitpunkt) aufgetreten sind aber auch Eigenschaften welche zu einem späteren Zeitpunkt, also in der Zukunft, erwartet werden. Für alle diese Algorithmen wird der Korrektheitsbeweis angetreten, eine Untersuchung der Zeit- und Speicherkomplexität angestellt, sowie eine effiziente Abbildung in Hardware besprochen. Der oben vorgestellte Algorithmus wird bspw. so optimiert, dass in Zeile 11 nur ein Element aus $l_{\boxdot_J \varphi}$ überprüft werden muss, wodurch sich konstante Laufzeit ergibt.

Highlight 4: Bestimmung der Systemgesundheit

Die Komponente zur Erfassung der Systemgesundheit von rtR2U2 verwendet den Systemstatus (Def. 1) um Gesundheitsmodelle auszuwerten. Die Ergebnisse dieser Komponente sind bedingte Wahrscheinlichkeiten aus denen sich Gegenmaßnahmen ableiten lassen. Abbildung 4 (unten) zeigt die entwickelte Hardwarearchitektur zur massiv-parallelen Auswertung von Gesundheitsmodellen unter Berücksichtigung der Anforderungen an rtR2U2.

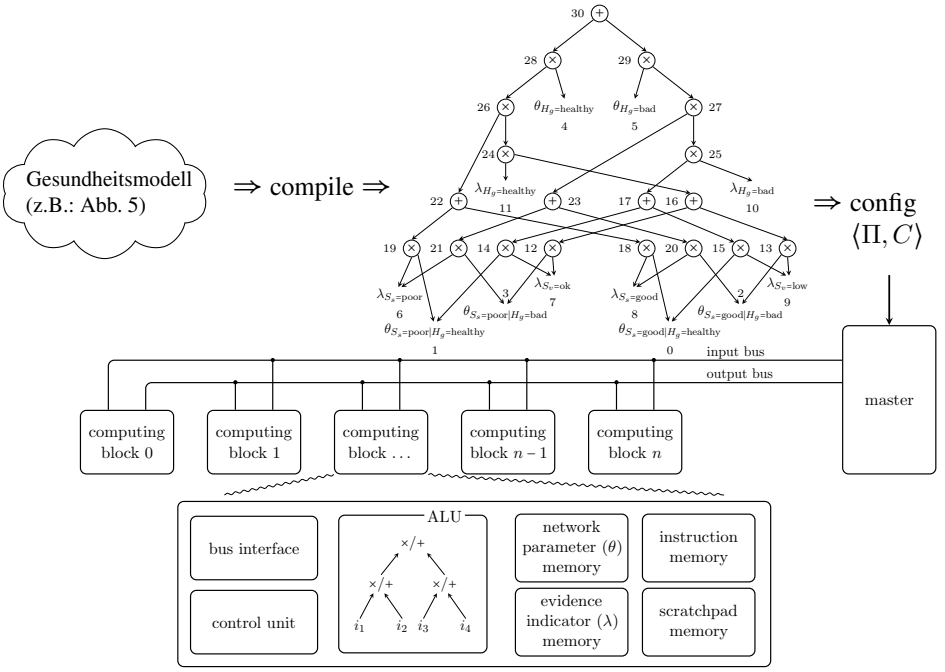


Abbildung 4: Kompiliervorgang eines Gesundheitsmodells als Bayes'sches Netz in einen Arithmetic Circuit (oben). Hoch parallele Hardwarearchitektur zur Auswertung von Gesundheitsmodellen während der operativen Phase des zu analysierenden Systems (unten).

Die Hardwarearchitektur ist darauf ausgelegt eine Graphenrepräsentation von Bayes'schen Netzen, sog. Arithmetic Circuits [Dar09], auszuwerten. Diese Graphenrepräsentation wird durch einen Kompiliervorgang aus einem beliebigen Gesundheitsmodell generiert. Aus diesem Graphen wird dann eine Instruktionsliste Π für die n parallelen Exekutionseinheiten (Computing Blocks) der Architektur generiert. Die Vorteile dieses programmierbaren Ansatzes sind: Die Hardwarearchitektur wird nur einmal auf die Zielplattform (FPGA, ASIC) synthetisiert; die Anzahl der parallelen Exekutionseinheiten ist dabei flexibel. Änderungen an Gesundheitsmodellen bedingen keine Synthese der rtR2U2 Hardware sondern können durch eine neue Instruktionsliste Π in die rtR2U2 Hardware eingespielt werden.

Highlight 5: Erprobung an einem Luftfahrzeug der NASA

Die industrielle Anwendbarkeit von rtR2U2 wird anhand der Analyse realer Flugdaten des unbemannten Luftfahrzeuges Swift (Abbildung 1) der NASA nachgewiesen. In dieser Kurzfassung wird exemplarisch eine vereinfachte temporale Spezifikation erstellt um zeitbehaftete Eigenschaften zu überprüfen, sowie ein Gesundheitsmodell entwickelt, welches die Subsysteme der Höhenmessung abdeckt. In dieser Erprobung wird nachgewiesen, dass rtR2U2, in Echtzeit und automatisch, einen Fehler in der Höhenmessung erkennt und korrekterweise eine Fehlfunktion des Laserhöhenmessers als Ursache identifiziert.

Beispiele zur temporalen Spezifikation. Für die Analyse der Flugdaten wurden temporale Spezifikationen in verschiedenen Kategorien aufgestellt, hier werden zwei genannt:

(i) Physikalische Zusammenhänge beschreiben Abhängigkeiten von Sensorgrößen, welche aus verschiedenen Subsystemen abgegriffen werden. Ein Beispiel dafür sind die Sensorgrößen Vertikalbeschleunigung und Höhe: Eine Vertikalbeschleunigung bedingt eine Änderung der Rate in der gemessenen Höhe. Dieser physikalische Zusammenhang wird durch die temporale Spezifikation φ_{R_2} erfasst.

Signale	s_vertVelocity | Vertikalbeschleunigung gemessen von Inertialsensor s_baroAltitude | Flughöhe gemessen von barometr. Höhenmesser
Spezifikation	$\varphi_{R_2} := \boxdot\,(\text{s\_vertVelocity} \geq 0.5\frac{m}{s} \rightarrow \textbf{rate}(\text{s\_baroAltitude}) \geq 0.2\frac{m}{s})$

(ii) Flight Rules sind durch nationale oder internationale Institutionen (z.B.: part 91 of the Federal Aviation Regulations in the USA [Fed13]) oder durch missionsspezifische Anforderung bestimmt. Ein Beispiel dafür ist die folgende Flugregel: Nach dem Start des Luftfahrzeuges muss ein Höhe von 600 ft nach spätestens 600 Zeiteinheiten erreicht werden. Diese Regel ist durch die temporale Spezifikation φ_{F_1} erfasst.

Signale	s_cmd | Von Bodenstation an Swift gesendeter Steuerbefehl s_laserAltitude | Flughöhe über Grund des Laserhöhenmessers
Spezifikation	$\varphi_{F_1} := \boxdot\,(\text{s\_cmd} = \text{takeoff} \rightarrow \blacklozenge_{600}\,(\text{s\_laserAltitude} \geq 600\text{ ft}))$

Ein Beispiel für Gesundheitsmodelle. Das Swift bedient sich verschiedener Subsysteme zur Bestimmung der aktuelle Flughöhe. Ein Laserhöhenmesser misst die Distanz des Luftfahrzeuges über Grund und ein Barometer gibt Auskunft über die Höhe über dem Meeresspiegel. Für den Fall, dass eines dieser Subsysteme ausfällt, oder falsche/wider-

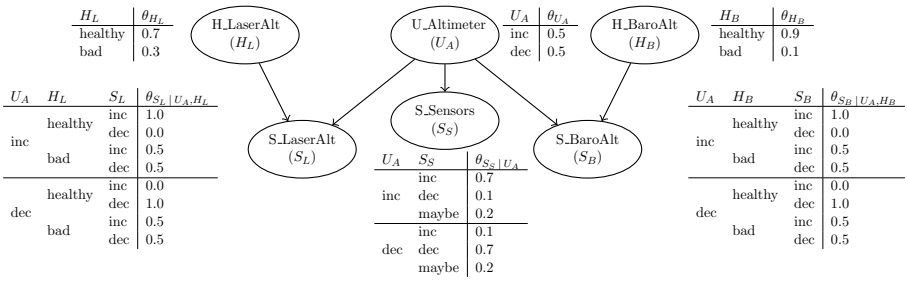


Abbildung 5: Gesundheitsmodell für die Höhenmessungssysteme des NASA Swift.

sprüchliche Informationen liefert, ist es für den Flugcomputer nicht möglich, die aktuelle Höhe genau zu bestimmen. Dies stellt eine Bedrohung der Sicherheit des Swift aber auch für dessen Umgebung dar. Darauf kann mit Hilfe von rtR2U2 regagiert werden. rtR2U2 wird dazu mit einem Gesundheitsmodell, wie es in Abbildung 5 dargestellt ist, konfiguriert.

Automatische Erkennung und Diagnose eines Ausfalls des Laserhöhenmessers. Abbildung 6 in Appendix A zeigt das Ergebnis einer industriellen, VHDL Register Transfer Level Hardwaresimulation des rtR2U2. rtR2U2 wurde hierfür mit den vorgestellten temporalen Spezifikationen und Gesundheitsmodellen (siehe Abb. 5) geladen und greift Signale mit einer Abtastfrequenz von 100 MHz von dem Kommunikationsinterface (siehe Abb. 1) des Swift ab. Die linke Abbildung zeigt die Resultate der Komponente zur Bestimmung des Systemstatus über einen gesamten Flug. Die Signalverläufe zeigen interne Signale wie zum Beispiel die Echtzeituhr (s_rtc_clk) und analoge und digitalisierte Eingangssignale von den on-board Sensoren des Swift (Höhenmessung: s_baroAltitude und s_laserAltitude, Vertikalbeschleunigung: s_vertVelocity, Pitchwinkel: s_pitch, etc.). Weiters ist der zur Laufzeit berechnete Systemstatus (z.B.: $e^n \vDash \varphi_{F_1}$ ist das Ergebnis der Überprüfung der Flight Rule φ_{F_1}) ersichtlich. Die Eingangssignale zeigen, dass während des Fluges der Laserhöhenmesser ausgefallen ist (siehe Signal: s_laserAltitude). Die rechte Abbildung zeigt einen detaillierten Ausschnitt dieser Phase des Fluges und wie rtR2U2 auf diesen Fehler reagiert. Die Signale $\Pr(H_B = \text{healthy} \mid S_L, S_B, S_S)$ und $\Pr(H_L = \text{healthy} \mid S_L, S_B, S_S)$ werden anhand des Gesundheitsmodells berechnet und repräsentieren den Gesundheitszustand der Subsysteme Barometerhöhenmesser und Laserhöhenmesser. Aus diesen Signalen zeigt sich, dass rtR2U2 diese widersprüchlichen Höhendaten unmittelbar feststellt und korrekterweise dem Laserhöhenmesser einen schlechten Gesundheitszustand attestiert, wobei der Gesundheitszustand des Barometerhöhenmessers nahezu unverändert bleibt.

Zusammenfassung

Durch die zunehmende Systemkomplexität aktueller eingebetteter Systeme steigt die Gefahr, dass (SW/HW) Bugs trotz sorgfältiger Validierungs- und Verifikationsanstrengungen nicht entdeckt werden bevor diese Ihren operativen Betrieb aufnehmen. Falls solche Bugs dann, wider Erwarten, während der operativen Phase auftreten, hat dies oft fatale Konse-

quenzen – besonders bei autonom agierenden Systemen wie etwa in der Automobilindustrie oder der Luft- und Raumfahrt. Diese Dissertation entwickelt einen neuen Ansatz welcher Bugs während der operativen Phase des Systems in Echtzeit nicht nur zuverlässig erkennen, sondern auch deren Ursache diagnostizieren kann. Dadurch können gezielte Gegenmaßnahmen eingeleitet werden um einen drohenden Systemcrash zu verhindern. Dieser Ansatz wurde erfolgreich an einem autonomen Luftfahrzeug der NASA erprobt.

Literatur

[AO08] P. Ammann und J. Offutt. *Introduction to Software Testing*. Cambridge University Press, 1. Auflage, January 2008. ISBN: 0521880386.

[Dar09] A. Darwiche. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, 1st. Auflage, 2009. ISBN: 0521884381.

[Dep05] Department of Defense, Office of the Secretary of Defense. Unmanned Aircraft Systems Roadmap 2005-2030. 2005.

[Eur12] European Commission. Towards a European strategy for the development of civil applications of Remotely Piloted Aircraft Systems RPAS. 2012.

[Fed13] Federal Aviation Administration. Federal Aviation Regulation §91, 2013.

[IEW10] C. Ippolito, P. Espinosa und A. Weston. Swift UAS: An electric UAS research platform for green aviation at NASA Ames Research Center. In *CAFE EAS IV*. April 2010.

[JGK+11] S.B. Johnson, T. Gormley, S. Kessler, C. Mott, A. Patterson-Hine, K. Reichard und J. Philip Scandura. *System Health Management: with Aerospace Applications*. Aerospace Series. John Wiley & Sons, 2011.

[Men07] O. J. Mengshoel. Designing Resource-Bounded Reasoners using Bayesian Networks: System Health Monitoring and Diagnosis. In *DX*, Seiten 330–337, 2007.

[MHA+95] D.J. Musliner, J.A. Hendler, A.K. Agrawala, E.H. Durfee, J.K. Strosnider und C. J. Paul. The challenges of real-time AI. *Computer*, 28(1):58–66, 1995.

[Pan12] M. Pandrika. Problem Solving on the Fly. *NASA Ask Magazine*, 48:30–34, 2012.

[Pnu77] A. Pnueli. The temporal logic of programs. In *FOCS*, Seiten 46–57. IEEE, 1977.

[Rei13] T. Reinbacher. *Analysis of Embedded Real-Time Systems at Runtime*. Dissertation, Technische Universität Wien, Institut für Technische Informatik, 2013.

Thomas Reinbacher wurde 1984 im niederösterreichischen Hollabrunn geboren. Er studierte Elektronik und Embedded Systems an der Fachhochschule Technikum Wien und Informatikmanagement an der Technischen Universität Wien. Während seiner Studienzeit lebte er für zwei Jahre in China; dort war er in der Hardwareentwicklung für Siemens tätig und studierte an der ZheJiang Universität Mandarin. 2013 promovierte er sub auspiciis Praesidentis rei publicae (unter den Auspizien des Bundespräsidenten) am Institut für Technische Informatik an der Technischen Universität Wien. Im Zuge seiner Promotion kooperierte er als Gastwissenschaftler unter anderem mit dem Lehrstuhl Informatik 11 der RWTH Aachen sowie der Robust Software Engineering Group des NASA Ames Research Centers und veröffentlichte 21 peer-reviewed Beiträge auf Konferenzen und in drei Journals, drei seiner Arbeiten wurden mit einem Best-Paper Award ausgezeichnet.

A Appendix: Hardwaresimulation

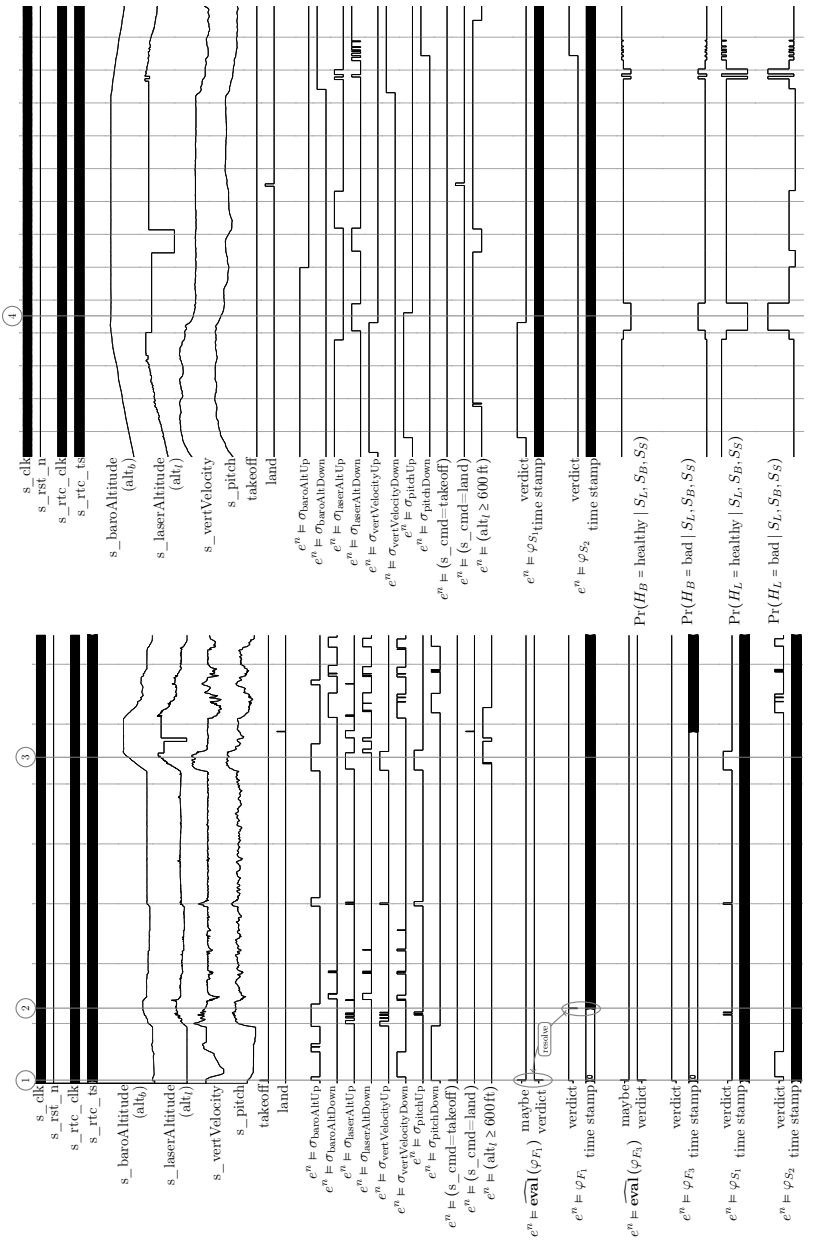


Abbildung 6: Signalverläufe einer low-level Hardwaresimulation des rtR2U2.

Energieautarker Betrieb drahtloser Sensorknoten mit regenerativen Energiequellen und Superkondensatoren*

Bernd-Christian Renner

Institut für Telematik
Technische Universität Hamburg-Harburg
christian.renner@tuhh.de

Abstract: Die Kombination aus Energie-Harvestern und Superkondensatoren schafft die Voraussetzung für den energieautarken, dauerhaften und ununterbrochenen Betrieb von drahtlosen Sensorknoten. Diese Dissertation schließt die Systemintegrationslücke zwischen Teilaspekten der Online-Energieerfassung hin zu einer umfassenden, ganzheitlichen Lösung. Sie entwickelt eine neue Methode zur Vorhersage des Energieertrags und analysiert die Güte bekannter Ansätze. Sie schlägt einen neuen Algorithmus zur Verbrauchsanpassung vor, der einen ununterbrochenen und gleichmäßigen Betrieb von Sensorknoten mit geringer Komplexität und geringem Berechnungsaufwand ermöglicht. Zur praktischen Aus- und Bewertung wird ein prototypischer Energie-Harvester mit einem Superkondensator für handelsübliche Sensorknoten entwickelt.

1 Einleitung

Die in den letzten Jahrzehnten erzielten Fortschritte in der Mikroelektronik haben die Voraussetzungen dafür geschaffen, Sensoren direkt mit einem Mikrocontroller und einer Funkschnittstelle in einer miniaturisierten Bauform zu integrieren. Kostengünstige, umfassende Sensorik erlaubt ein detailliertes, flächendeckendes Monitoring. Das Potential dieser Technik liegt in vielen Anwendungsbereichen wie z.B. Verkehrstelematik, Anlagensteuerung, Umweltüberwachung und Avionik. Das ökonomische Potential ist insbesondere für Deutschland hoch, da ein Großteil der Wertschöpfungskette im Inland angesiedelt ist. Nachdem in den vergangenen Jahren essentielle Fortschritte bei Routing-Verfahren, energiesparenden Algorithmen, Fehlertoleranz und Datenaggregation erreicht wurden, muss noch eine letzte große Hürde für den vollkommen autarken Einsatz drahtloser Sensornetze überwunden werden: eine Energieversorgung unabhängig von Batterien zur Vermeidung manueller Eingriffe. Diese Dissertation [Ren13] leistet einen wesentlichen Beitrag an der Schnittstelle zwischen Hard- und Software zum autarken Betrieb drahtloser Sensornetze.

Der klassische Ansatz, die Leistungsaufnahme eines Sensorknotens durch Energiesparmaßnahmen zu verringern, ist für sich genommen nicht geeignet, einen langfristigen Be-

*Englischer Titel der Dissertation: "Sustained Operation of Sensor Nodes with Energy Harvesters and Supercapacitors"

trieb zu erreichen. Zwar können optimierte Medienzugriffs- und Routing-Verfahren wie z.B. aus [LGDJ12, URT12] die Leistungsaufnahme um ein Vielfaches reduzieren, jedoch kann die erreichbare Betriebszeit i.d.R. lediglich von wenigen Tagen auf einige Monate heraufgesetzt werden. Für langfristige Anwendungen ist jedoch insbesondere aus ökonomischen und logistischen Gründen ein Batteriewechsel ausgeschlossen. Zur Beseitigung dieser Schwachstelle bietet sich die Versorgung von Sensornetzen durch regenerative Energiequellen (sog. *Energy Harvesting*), zumeist in Form von Solarenergie, an [PS05]. Der durchschnittlich erzielbare Energieertrag liegt zwar auf dem Verbrauchsniveau eines Sensorknotens unter Ausnutzung üblicher Energiesparmaßnahmen, allerdings ist er signifikanten zeitlichen Schwankungen und ortsabhängigen Einflüssen unterworfen.

Die grundlegende Herausforderung definiert sich daraus, den Verbrauch jedes Sensorknotens auf dessen individuellen Energieertrag abzustimmen, so dass einerseits ein ununterbrochener Betrieb gewährleistet und andererseits eine bestmögliche Ausnutzung des Ertrags erreicht wird. Diese gegensätzlichen Ziele ergeben sich aus dem Netzwerkaspekt, da Sensornetze i.A. Multihop-Netze sind—d.h. neben der Erfassung und Meldung eigener Daten müssen auch Daten anderer Sensorknoten weiterleitet werden. Über die Kommunikation sind die Energieverbräuche der einzelnen Sensorknoten gekoppelt, weil Sender und Empfänger mit aktivem Funkmodul aufeinander warten müssen [LGDJ12, URT12].

Wegen der a-priori unbekannten und wechselnden Funkkonnektivitäten [ZK07] und der durch o.g. Einflüsse schwer vorhersagbaren Energieerträge lässt sich eine derartige Abstimmung von Verbrauch und Ertrag, eine sog. *Energiebudgetierung*, nicht statisch und insbesondere auch nicht vor dem operativen Einsatz eines Sensornetzes bestimmen. Sie muss hingegen während der Laufzeit durchgeführt und dynamisch angepasst werden. Die Komplexität der Aufgabenstellung ist dadurch gegeben, dass es nicht ausreichend ist, den aktuellen Verbrauch an den aktuellen Ertrag anzugleichen. So kann sich beispielsweise ein Sensorknoten in Zeiten ohne jeglichen Energieertrag nicht vollständig abschalten, da er weiterhin Messungen durchführen und Datenpakete weiterleiten muss.

Daher ist die Verwendung eines Energiespeichers notwendig, um Zeiten geringer oder keiner Energieerträge zu überbrücken. Zu den wichtigsten Anforderungen an einen Energiespeicher zählen dessen Langzeithaltbarkeit, insbesondere die möglichen Ladezyklen, und die Genauigkeit bzgl. der Ladezustandsbestimmung, ein essentielles Kriterium zur automatischen und ausfallsicheren Verbrauchsanpassung. Außerdem sind Eigenschaften wie ein geringer Preis und eine hohe Umweltverträglichkeit im Sinne des Green Computing relevant. Besonders gut werden diese Eigenschaften von sog. *Superkondensatoren* erfüllt ([Sam]). Ihr größter Vorteil gegenüber wiederaufladbaren Batterien ist dabei die Möglichkeit, den Ladezustand direkt über die Spannung bestimmen zu können. Allerdings gibt es bis dato weder Langzeitstudien zum Einsatz von Superkondensatoren noch Machbarkeitsstudien zur ganzheitlichen Energieflussanalyse zur Verbrauchsanpassung. Wenig untersucht ist insbesondere die erzielbare Präzision vereinfachter, generischer Modelle zur Minimierung des manuellen Konfigurationsaufwands aus Kosten- und Effizienzgründen.

Existierende Verfahren zur Verbrauchsanpassung fußen zumeist auf vergleichsweise komplexen Ansätzen auf Basis von Integer Linearen Programmen (ILP). Deren Lösung ist auf der extrem ressourcenbeschränkten Sensorknotenhardware ausgeschlossen: Sensorknoten verfügen üblicherweise über wenige Kilobyte RAM, wenige hundert Kilobyte ROM und

eine Taktfrequenz von wenigen Megahertz. Die Komplexität derartiger Ansätze hinsichtlich ihrer Konfiguration verlangt dem Betreiber eines Sensornetzes außerdem ein Expertenwissen ab, das in der Praxis selten vorliegt. Einfachere Ansätze werden daher benötigt.

1.1 Wissenschaftlicher Beitrag der Dissertation

Im Rahmen der Dissertation werden Modelle, Methoden, Algorithmen und ein Experimentalsystem für Sensornetze, die mit regenerativen Energiequellen betrieben werden, entwickelt. Im Fokus steht hierbei die Praxistauglichkeit des Konzepts, um ein *energiegewahres* Verhalten und damit einen autarken, ausfallsicheren Betrieb von Sensornetzen zu realisieren. Im Einzelnen werden die folgenden Beiträge geleistet:

- Es wird ein Konzept für den autarken und autonomen Betrieb von typischen, mit regenerativen Energiequellen betriebenen Sensorknoten entwickelt. Das Konzept besteht aus einem umfassenden Energieflussmodell, um den Einfluss der Energiezu- und -abflüsse auf die Energiereserve beschreiben zu können. Es ermöglicht die Vorhersage der Entwicklung der Energiereserven sowie die anwendungsorientierte Verbrauchsanpassung und bildet die Grundlage der Dissertation.
- Die Praxistauglichkeit des Energieflussmodels wird durch diverse Experimente untersucht und mit einem in der Dissertation entwickelten Prototypen nachgewiesen. Dabei werden insbesondere diejenigen Parameter identifiziert, die den Energiefluss maßgeblich beeinflussen. Zur Verringerung der Komplexität und des Konfigurationsaufwands wird außerdem untersucht, inwieweit Vereinfachungen, eine generische Konfiguration sowie die automatische Kalibrierung zur Laufzeit die Genauigkeit des Modells beeinflussen. Ein zentrales Thema ist hierbei die Modellierung des Lade- und Entladeverhaltens der verwendeten Superkondensatoren.
- Es wird eine neue Methode zur Vorhersage des Energieertrags entwickelt und analysiert. Im Gegensatz zu bestehenden Lösungen zielt diese Methode auf die Erkennung des orts- und zeitabhängigen, quasi-periodischen Musters bei der Energiegewinnung ab, das insbesondere bei Solarenergie vorkommt. Die neue Methode wird mit realen Solardaten ausgewertet und mit den existierenden Methoden verglichen.
- Es wird die vorhersagebasierte Verbrauchsanpassung mit Energierichtlinien eingeführt. Basierend auf dem Energieflussmodell wird ein Algorithmus zur Vorhersage der Energiereserve-Entwicklung konzipiert. Über sog. Energierichtlinien wird der maximale, gleichmäßige und ausfallsichere Verbrauch bestimmt. Beim Entwurf des Algorithmus steht ein geringer Ressourcenbedarf bzgl. Speicher und Rechenleistung im Fokus, um einen Einsatz auf typischen Sensorknoten zu ermöglichen.
- Die entwickelten Methoden und Algorithmen werden schließlich für reale Sensorknoten implementiert und in einer Fallstudie evaluiert. Um ihren praktischen Nutzen zu zeigen, wird ein einfaches aber effektives Verfahren zur Verbrauchsanpassung eines energieeffizienten Routing-Verfahrens für ein Multihop-Sensornetz entwickelt und in einem mehrwöchigen Feldtest ausgewertet.

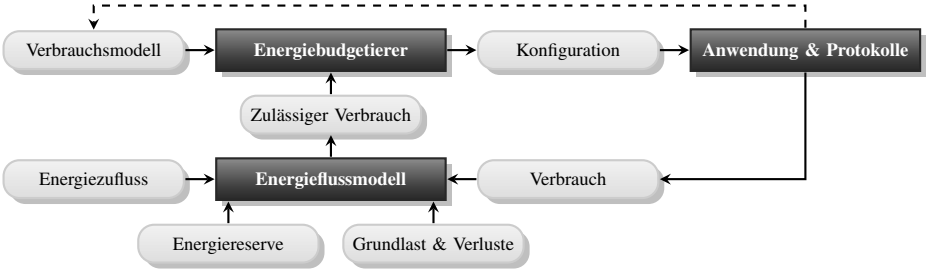


Abbildung 1: Funktionale Blöcke, Einflussfaktoren und Stellgrößen eines energiegewahren Systems.

2 Konzept

Im Folgenden wird das in der Dissertation erarbeitete Konzept zur Energiebudgetierung bzw. Verbrauchsanpassung erläutert. Abbildung 1 fasst die beschriebenen Zusammenhänge, Abhängigkeiten und Ansätze grafisch zusammen.

Zur erfolgreichen, d.h. ausfallsicheren und effizienten, Energiebudgetierung muss ein Sensorknoten zunächst seine aktuelle und zukünftige energetische Situation abschätzen und bewerten. Als Grundlage hierfür dient ein *Energieflussmodell*, das die verschiedenen Energieflusskomponenten umfasst und in Abschnitt 3 technisch erläutert wird. Der *Energiezufluss* entsteht durch die regenerative Energiequelle. Neben der Messung des aktuellen Energiezuflusses ist die Vorhersage zukünftiger Energiezuflüsse notwendig, um eine verlässliche Energiebudgetierung durchführen zu können. So kann beispielsweise der zulässige Verbrauch für Perioden geringer Zuflüsse besser geplant werden, so dass der Knoten eine hohe Funkverfügbarkeit für die Datenweiterleitung im Netzwerk bei gleichzeitig geringer Ausfallwahrscheinlichkeit erreichen kann. Die Abschätzung der aktuellen und zukünftigen Entwicklung der *Energiereserve* ermöglicht es dem Sensorknoten, eine Risikoanalyse für einen möglichen Ausfall durchzuführen sowie eine Ressourcenverschwendung in Form eines vollen Energiespeichers bei gleichzeitig hohem Energiezufluss und geringem Verbrauch zu erkennen. Bei der *Grundlast* handelt es sich um die nichtsteuerbaren Verbräuche des Sensorknotens. Hierunter fallen z.B. Verluste im Energiespeicher sowie statische Verbräuche der Ladeschaltung und der Verbrauch des Sensorknotens im Ruhezustand. Der tatsächliche bzw. erwartete *Verbrauch* der auf dem Sensorknoten ausgeführten Anwendung und Protokolle ist der letzte Baustein des Energieflussmodells.

Der *Energiebudgetierer* ist das zentrale Element zur Ermittlung und Einstellung des *maximal zulässigen Verbrauchs*. Ausgehend vom Energieflussmodell und von der Vorhersage zukünftiger Energiezuflüsse (siehe Abschnitt 4) wird der maximale Verbrauch für einen bestimmten Zeithorizont, beispielsweise einen Tag, online berechnet. Hierfür wird die Methode der Energierichtlinien verwendet. Hierunter versteht man eine Menge an Anforderungen an die Entwicklung der Energiereserve. Ein konkretes, in der Dissertation untersuchtes Beispiel hierfür ist die Anforderung, dass die Energiereserve im betrachteten Zeithorizont einen Schwellwert nicht unterschreiten darf (vgl. Abschnitt 5).

Schließlich muss der Budgetierer den ermittelten, maximal zulässigen Verbrauch in eine *Konfiguration* der *Anwendung und Protokolle* überführen, die in genau diesem Verbrauch resultiert. Zu diesem Zweck dient ein aus der Anwendung bzw. den Protokollen abgeleitetes *Verbrauchsmodell*, das den Verbrauch über eine Menge von Parametern steuerbar macht. Die in der Dissertation behandelte, konkrete Umsetzung für einen typischen Anwendungsfall von drahtlosen Sensornetzen wird in Abschnitt 6 genauer beleuchtet.

3 Energieflussmodell

Die einzelnen Komponenten des Energieflusses drahtloser Sensorknoten wurden in der Wissenschaft bereits untersucht. Methoden zur Online-Verbrauchsmessung wurden beispielsweise in [DÖTH07, HBNH11] vorgestellt. Die Eigenschaften von Superkondensatoren wurden in [WMKA11] studiert. Die jeweils verwendeten Modelle sind darauf ausgelegt, die Realität möglichst exakt, d.h. mit geringem Fehler, wiederzugeben. Hieraus ergibt sich eine hohe Komplexität sowie eine große Menge an Parametern. Für den Einsatz auf Sensorknoten sind diese Modelle damit jedoch kaum oder gar nicht geeignet. Während die hohe Komplexität einen hohen Rechenaufwand nach sich zieht, der selbst mit den Ressourcen moderner Sensorknoten nicht bewältigt werden kann, müssen die Parameter für jeden einzelnen Sensorknoten bestimmt werden; zu groß sind die fertigungsbedingten Exemplarstreuungen z.B. hinsichtlich der tatsächlichen Kapazität eines Superkondensators. Fertigungstoleranzen führen hier zu Abweichungen von bis zu 30% [Sam]. Gleichzeitig führen Alterungsprozesse zu einer Änderung der Parameter zur Laufzeit. Während eine manuelle Vermessung bzw. Konfiguration damit ausgeschlossen ist, erweist sich ein automatischer Ansatz in Anbetracht der Größe des Parameterraums als nicht praktikabel.

In der Dissertation wurde daher ein vereinfachtes Energieflussmodell entwickelt, um durch die Eingrenzung des Parameterraums und die Reduktion der Berechnungskomplexität ein für Sensorknoten berechenbares Modell zu erhalten. Hierzu wurde angenommen, dass

- der Energiezufluss mit einem einfachen Stromsensor gemessen werden kann,
- der Superkondensator sich annähernd verhält wie ein idealer Kondensator,
- der Energieverbrauch sich als Summe der Produkte von Zeit und Verbrauch pro Hardware-Zustand berechnen lässt und
- der benötigte Spannungswandler zur Einstellung einer festen Versorgungsspannung eine konstante und bekannte Effizienz hat.

Ferner wird angenommen, dass sich der Verbrauch der einzelnen Sensorknoten in wenige Zustände zerlegen lässt und sich der Verbrauch pro Zustand zwischen verschiedenen Sensorknoten nicht unterscheidet, so dass ein generisches Profil verwendet werden kann. Als einziger Parameter verbleibt damit die tatsächliche Kapazität des Superkondensators.

Dieses vereinfachte Modell wurde durch praktische Experimente analysiert. Zu diesem Zweck wurde ein Prototyp entwickelt, der in verschiedenen Konfigurationen und Szenarien zur Messdatenerhebung eingesetzt wurde. Es wurden Testläufe im Freien und un-

ter Laborbedingungen durchgeführt, verschiedene Superkondensatoren verwendet sowie mehrere Prototypexemplare und Sensorknoten eingesetzt. Die so erhaltenen Datensätze wurden verwendet, um den Modellfehler zu bestimmen. Es konnte gezeigt werden, dass das vereinfachte Modell eine gute qualitative Übereinstimmung mit dem tatsächlichen Energiefluss aufweist. Unter Verwendung einer in der Arbeit konzipierten Methode zur Kapazitätsbestimmung des Superkondensators lag der mittlere Fehler bei lediglich 5%.

4 Vorhersage des zukünftigen Energieertrags

Mit dem Aufkommen der ersten Energie-Harvester für Sensorknoten begann die Entwicklung von Vorhersageverfahren der Energieerträge [AARA10]. Im Fokus der Entwicklung stand die Prämisse geringer Komplexität und der Verzicht auf globale Information. Letzteres ist dem hohen Energieverbrauch der Netzwerkkommunikation geschuldet.

Die Verfahren haben sich dabei evolutionär entwickelt und basieren auf dem gleichen Konzept. Unter der Annahme quasi-periodischer Energieerträge—bei Solarenergie ist dies durch den täglichen Sonnenzyklus gedeckt—wird eine Periode in Zeitschlitze gleicher Länge unterteilt. Die Granularität reicht hierbei von einigen Stunden bis hin zu einigen Minuten. Für jede Periode und für jeden Zeitschlitz wird der mittlere Energieertrag zur Laufzeit aus diskreten Messwerten berechnet. Pro Zeitschlitz wird ein über die Perioden hinweg geglätteter Wert gespeichert, der im einfachsten Fall direkt zur Vorhersage des Mittelwertes im gleichen Zeitschlitz der folgenden Periode verwendet wird.

Derartige Verfahren sind hocheffizient, basieren jedoch auf der Annahme, dass der Energieertrag innerhalb der Zeitschlitze gleichmäßig ist. Bei langen Zeitschlitzen von beispielsweise einer Stunde ist dies nicht gegeben. Insbesondere durch lokale Gegebenheiten, wie z.B. Schattenwürfe, können die Erträge innerhalb eines Zeitschlitzes um ein Vielfaches voneinander abweichen. Dies führt zu einer Imbalance zwischen erwartetem (konstanten) und tatsächlichem (schwankenden) Energieertrag und damit zu einer schlechten Vorhersagequalität, die wiederum negativen Einfluss auf die durchgeführte Verbrauchsanpassung hat (siehe Abschnitt 5). In diesem Zusammenhang konnte in der Dissertation durch Auswertung verschiedener, realer Solarenergiedatensätze gezeigt werden, dass eine Erhöhung der Schlitzanzahl nur bis zu einer gewissen Grenze zu einer Verbesserung führt (bei Solarenergie ca. 12 bis 24 Schlitze). Normale Schwankungen innerhalb kurzer Zeitintervalle von wenigen Minuten führen zu Überanpassungseffekten bei einer zu hohen Schlitzanzahl.

Aus diesem Grund wurde ein neues Verfahren entwickelt, das die Länge der einzelnen Schlitze zur Laufzeit individuell anpasst, so dass die Fluktuation (in Form des quadratischen Fehlers) innerhalb der Schlitze minimiert wird. Zunächst wurde ein optimales Verfahren zur Berechnung der Schlitzlängen entwickelt, dessen Komplexität durch den Einsatz linearer Programmierung signifikant verringert werden konnte. Diese Lösung hat sich für den Einsatz auf Sensorknoten jedoch als immer noch zu rechenintensiv und energiehungrig erwiesen, so dass ein einfaches, adaptives Verfahren mit linearem Laufzeitverhalten und vernachlässigbarem Energieverbrauch entwickelt wurde. Durch extensive Simulationen konnte insbesondere gezeigt werden, dass das neue Verfahren bei gleicher

Vorhersagepräzision eine Halbierung der Schlitzanzahl erlaubt. Dies stellt eine Energieersparnis im Sinne der Verbrauchsanpassung dar, die i.d.R. einmal je Zeitschlitz ausgeführt wird. Hierüber hinaus konnte gezeigt werden, dass das neue Verfahren das tatsächliche Energiemuster deutlich besser abbildet, d.h. Intervalle vergleichsweise konstanter Energieerträge wurden zuverlässig erkannt. Die Analyse hat jedoch ebenfalls ergeben, dass durch ausschließlich lokales Wissen nicht in allen Fällen, d.h. insbesondere in veränderten Wettersituationen, eine gute Vorhersage erstellt werden kann. Der Einbezug globalen Wissens war jedoch nicht Bestandteil der Dissertation, weil hierfür zunächst adäquate und energieeffiziente Netzwerkverteilungsalgorithmen hätten entwickelt werden müssen.

5 Verbrauchsanpassung mit Energierichtlinien

Die Verbrauchsanpassung wurde im Umfeld drahtloser Sensornetze bereits von einigen Wissenschaftlern untersucht. Wesentliche Arbeiten unter Einbeziehung von Vorhersageverfahren für den zukünftigen Energieertrag finden sich in [MTBB10, KHZS07]. Bestehende Verfahren weisen jedoch mindestens einen der folgenden beiden Nachteile auf: (i) Sie basieren auf Integer Linearen Programmen (ILPs), sind also komplex, und erfordern damit einen rechenintensiven Lösungsalgorithmus; (ii) sie zielen auf eine rein lokale Optimierung ohne Berücksichtigung des Netzwerkaspekts ab, d.h. die Verfahren forcieren grundsätzlich einen hohen Verbrauch in Zeiten großer Energieerträge und einen geringen Verbrauch in Zeiten niedriger Energieerträge. In vielen Einsatzszenarien drahtloser Sensornetze sollte jedoch ein gleichmäßiger Verbrauch angestrebt werden, da ein wesentlicher Anteil des Verbrauchs auf die Netzwerkkommunikation entfällt. Zu diesem Schluss, der in der Dissertation näher beleuchtet wird, führen insbesondere zwei Überlegungen:

1. Eine zuverlässige, effiziente Funkkommunikation zwischen zwei Sensorknoten ist nicht gewährleistet, wenn einer der beiden Knoten in einem aggressiven Energiesparmodus ist und damit wenig kommunizieren kann, während der andere Knoten durch einen geringen Energiesparmodus viel mehr kommunizieren könnte. Dies trifft beispielsweise auf Sensorknoten zu, die sich auf unterschiedlichen Seiten (z.B. Osten und Westen) eines großen, hohen Baumes oder eines Hauses befinden.
2. Der Verbrauch eines einzelnen Knotens ist von der Kommunikation mit anderen Knoten abhängig. So basieren energieeffiziente Kommunikationsprotokolle wie z.B. [LGDJ12, URT12] darauf, dass der Sender aktiv (d.h. mit eingeschaltetem Funkmodul) auf den Empfänger wartet. Diese Wartezeiten beeinflussen den Verbrauch so massiv, dass sie bei der Protokollkonfiguration berücksichtigt werden müssen. Je weniger stabil sie sind, desto schwieriger lässt sich der eigene Verbrauch anpassen.

In der Dissertation wurde daher ein Algorithmus entwickelt, der einerseits eine geringe Rechenkomplexität besitzt und andererseits einen kontinuierlichen, stabilen Verbrauch gewährleistet. Grundlage des Algorithmus sind das Energieflussmodell und die Vorhersage der Energieerträge, wodurch die Entwicklung der Energiereserve (in Form der Spannung am Superkondensator) in Abhängigkeit des Verbrauchs vorhergesagt werden kann. Eine

effiziente Umsetzung des Algorithmus wurde durch die Einführung von Energieregeln erreicht. Bei Letzteren handelt es sich um Anforderungen an die Energiereserve im Vorhersagehorizont. So sollte beispielsweise eine kritische Mindestreserve nicht unterschritten werden, um einen Ausfall des Knotens zu verhindern. In der Dissertation wurden weitere Energieregeln vorgestellt und untersucht. Eine besondere Schwierigkeit bei dem gewählten Ansatz liegt darin, dass das Energieflussmodell eine nicht-lineare Differentialgleichung ohne (bekannte) explizite Lösung ist. Durch die Kombination des Newton-Verfahrens und der binären Suche konnte jedoch ein effizienter Lösungsalgorithmus entwickelt werden.

Der Algorithmus wurde in der Dissertation mittels extensiver Simulation unter Verwendung realer Solarmessdaten ausgewertet. Hierbei wurden die Ausfallsicherheit und die Effizienz hinsichtlich der maximal möglichen Solarausbeute untersucht. Es wurden verschiedene Energieregeln analysiert sowie unterschiedliche Vorhersagealgorithmen des Energieertrags verglichen. Dabei konnte ein quasi ununterbrochener Betrieb selbst bei Einsatz kleiner Superkondensatoren nachgewiesen werden. Außerdem wurde der Vorteil des Einsatzes von Vorhersagemethoden hinsichtlich der Solarausbeuteeffizienz gezeigt.

6 Anwendungsstudie

Im letzten Teil der Arbeit wurde eine Studie der entwickelten Methoden und Algorithmen im Kontext des prädominanten Anwendungsfalls drahtloser Sensornetze durchgeführt: die Datenerfassung. Hierzu wurden alle Algorithmen für reale Sensorknoten sowie das mit einem Kollegen gemeinschaftlich entwickelte, energieeffiziente Datensammelprotokoll ORiNoCo [URT12] implementiert. Schließlich wurde ein einfaches aber effizientes Verbrauchsmodell abgeleitet und eine Methode zur Konfiguration von ORiNoCo entworfen, um den vom Energiebudgetierer ermittelten, maximalen Verbrauch einzustellen.

Hierbei wurde ausgenutzt, dass ORiNoCo einen opportunistischen Routing-Ansatz verfolgt. Anstelle fester Netzwerkverbindungen, z.B. in Form eines Routing-Baumes, werden Datenpakete an den ersten antwortenden Knoten in Kommunikationsreichweite weitergeleitet, der sich näher an der Senke befindet als der Sender. Abbildung 2 zeigt zur Illustration ein Sensornetz mit dem Ziel der Datensammlung. Der dargestellte Routing-Baum bewirkt, dass der verbleibende Anteil des Energiebudgets von Knoten D bereits sehr gering ist, da er alle Daten aus seinem Teilnetz (Knoten E bis M) weiterleiten muss. Geschickter wäre eine Arbeitsteilung mit Knoten C. Die gewählte Verbrauchsanpassung erreicht in Kombination mit ORiNoCo, dass sich durch die rein lokale Anpassung der Knoten automatisch deren Funkverhalten (d.h. die Verfügbarkeit für den Nachrichtenempfang) regelt, ohne einen Overhead in Form weiterer Kontrolldatenpakete zu verursachen. Es wird ferner erreicht, dass sich sich die weitergeleitete Datenmenge automatisch auf die Knoten aufteilt, und zwar in Abhängigkeit ihres Energieertrags und ihrer Energiereserven.

Zur Auswertung wurde ein Feldtest mit 12 Sensorknoten, jeder mit einem Solar-Harvester-Prototyp betrieben, und einer Laufzeit von 30 Tagen durchgeführt. Durch die Positionierung der Sensorknoten auf unterschiedlichen Seiten des Universitätsgebäudes wurde sichergestellt, dass die Knoten zu unterschiedlichen Zeiten mit Solarenergie versorgt wur-

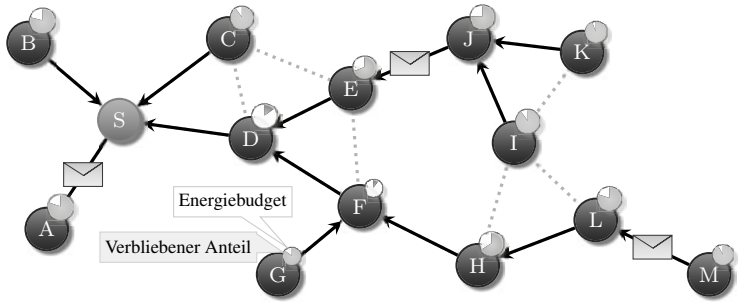


Abbildung 2: Drahtloses Sensornetz mit einer Senke S. Alle Knoten verfügen über unterschiedliche Energiebudgets. Die Abbildung zeigt potentielle (gepunktete Linien) und durch ein hypothetisches, baumbasiertes Routing-Verfahren tatsächlich genutzte (Pfeile) Funkverbindungen.

den. Die Knoten wurden außerdem mit Superkondensatoren verschiedener Kapazitäten bestückt, um den Einfluss inhomogener Energiereservenverteilungen auf die Funktionalität der Verbrauchsanpassung zu untersuchen. Das Netzwerk hatte eine Tiefe von 3 Hops.

Es konnte gezeigt werden, dass sich das Datenvolumen von Knoten mit gleichem Unterbaum im Netzwerk automatisch an deren energetische Situation (d.h. die Kapazität des Superkondensators und den durchschnittlichen Energieertrag) anpasst. Ferner wurde die Praxistauglichkeit des Verbrauchsmodells nachgewiesen, indem der eingestellte und der tatsächliche Verbrauch protokolliert und miteinander verglichen wurden. Gleichzeitig wurde ein quasi ununterbrochener, energieautarker Betrieb beobachtet.

7 Fazit

Die im Rahmen dieser Dissertation entwickelten Modelle, Methoden und Algorithmen bilden eine gebrauchsfertige Lösung zum energieautarken und ausfallfreien Betrieb drahtloser Sensorknoten, die sich über diverse Anwendungsmöglichkeiten von Sensornetzen, insbesondere zur Datenerfassung, erstreckt. Die Ergebnisse und Methoden sind übertragbar auf ähnliche Hardware und Energie-Harvester mit quasi-periodischen Energieerträgen. Die in der Dissertation erreichte Verflechtung von Hardware und Software gewährleistet geringe Herstellungs-, Installations- und Wartungskosten. Die geringe Komplexität der entwickelten Lösung zur autonomen Verbrauchsanpassung und die überschaubare Menge an Parametern erleichtern den praktischen Einsatz von Multihop-Netzen. Dies ist ein entscheidender Faktor zur Etablierung drahtloser Sensornetze als universal einsetzbares und leicht handhabbares Monitoring-Instrument.

Danksagung. Der Autor möchte den Koautoren Volker Turau, Christoph Weyer, Jürgen Jessen, Marcus Venzke, Stefan Unterschütz und Florian Meier danken. Die Forschungsarbeit wurde von der Deutschen Forschungsgemeinschaft (Projekt TU 221/4-1) unterstützt.

Literatur

[AARA10] M. Ali, B. Al-Hashimi, J. Recas Piorno und D. Atienza. Evaluation and Design Exploration of Solar Harvested-Energy Prediction Algorithm. In *Proc. Conf. on Design, Automation and Test in Europe*, DATE, 2010.

[DÖTH07] A. Dunkels, F. Österlind, N. Tsiftes und Z. He. Software-based On-line Energy Estimation for Sensor Nodes. In *Proc. Intl. Wkshp. on Embedded Networked Sensors*, Emnets, 2007.

[HBNH11] P. Hurni, T. Braun, B. Nyffenegger und A. Hergenröder. On the Accuracy of Software-based Energy Estimation Techniques. In *Proc. Europ. Conf. on Wireless Sensor Networks*, EWSN, 2011.

[KHZS07] A. Kansal, J. Hsu, S. Zahedi und M. Srivastava. Power Management in Energy Harvesting Sensor Networks. *Trans. on Embedded Computing Systems (TECS)*, 2007.

[LGDJ12] O. Landsiedel, E. Ghadimi, S. Duquennoy und M. Johansson. Low Power, Low Delay: Opportunistic Routing meets Duty Cycling. In *Proc. 11th ACM/IEEE Conf. on Information Processing in Sensor Networks*, IPSN, 2012.

[MTBB10] C. Moser, L. Thiele, D. Brunelli und L. Benini. Adaptive Power Management for Environmentally Powered Systems. *Trans. on Computers (TC)*, 59(4), 2010.

[PS05] J. Paradiso und T. Starner. Energy Scavenging for Mobile and Wireless Electronics. *Pervasive Computing*, 4(1), 2005.

[Ren13] C. Renner. *Sustained Operation of Sensor Nodes with Energy Harvesters and Supercapacitors*. Dissertation, Hamburg Univ. of Technology, Hamburg, Germany, 2013.

[Sam] Samwha Electric Co., Ltd. Green Cap: Electric Double Layer Capacitor. http://samwha.co.kr/SW_Catalogue/ecatalog.asp?Dir=60. Last access: 2014/01/02.

[URT12] S. Unterschütz, C. Renner und V. Turau. Opportunistic, Receiver-Initiated Data-Collection Protocol. In *Proc. Europ. Conf. on Wireless Sensor Networks*, EWSN, 2012.

[WMKA11] A. Weddell, G. Merrett, T. Kazmierski und B. Al-Hashimi. Accurate Supercapacitor Modeling for Energy-Harvesting Wireless Sensor Nodes. *Trans. on Circuits and Systems II (TCAS-II): Express Briefs*, 58, 2011.

[ZK07] M. Zuniga und B. Krishnamachari. An Analysis of Unreliability and Asymmetry in Low-Power Wireless Links. *Trans. on Sensor Networks (TOSN)*, 3, 2007.

Bernd-Christian Renner wurde am 2. September 1982 in Winsen/Luhe geboren. Von Oktober 2003 bis Juni 2008 studierte er Ingenieur-Informatik an der TU Hamburg-Harburg (TUHH) und der ETH Zürich. Im Februar 2006 wurde er Stipendiat der Studienstiftung des Deutschen Volkes. Von Juli 2008 bis September 2012 arbeitete er als wissenschaftlicher Mitarbeiter an der TUHH, wo er neben der Forschung eine Vielzahl studentischer Arbeiten betreute. Seine Promotion schloss er im Frühjahr 2013 ab. Seine Arbeiten wurden in zahlreichen internationalen Journalen und Tagungsbänden veröffentlicht und er ist Mitglied in Programmkomitees internationaler Konferenzen. Seit Oktober 2013 forscht und lehrt er an der Universität zu Lübeck und strebt dort die Habilitation an. Seine Forschungsinteressen umfassen u.a. Unterwasser-Umweltmonitoring mit mobilen Über- und Unterwasserfahrzeugen sowie Kommunikation und Energiemanagement in drahtlosen Sensornetzen.

Nicht-asymptotische Leistungsbewertung und abtastbasierte Parameterschätzung für Kommunikationsnetzwerke mit langzeitkorreliertem Datenverkehr*

Amr Rizk
Institut für Kommunikationstechnik
Leibniz Universität Hannover
amr.rizk@ikt.uni-hannover.de

Abstract: Die Entdeckung der statistischen Eigenschaften der Selbstähnlichkeit und Langzeitkorrelation im Internetverkehr hat die Annahmen und somit die Anwendbarkeit traditioneller Modelle bzw. Leistungsbewertungsmethoden in Kommunikationsnetzen wie dem Internet in Frage gestellt. Diese Arbeit liefert obere Schranken für Leistungsmetriken in Netzwerken mit langzeitkorreliertem Datenverkehr basierend auf dem stochastischen Netzwerkkalkül. Ein grundlegendes Resultat dieser Arbeit beziffert den Zuwachs von Ende-zu-Ende Latenzen für Netzwerkpfade mit langzeitkorreliertem Verkehr. Außerdem werden Methoden zur abtastbasierten Schätzung der statistischen Eigenschaften des Datenverkehrs vorgestellt. Praktische Messergebnisse untermauern die analytischen Resultate zur Anwendbarkeit und Genauigkeit der vorgeschlagenen Schätzer.

1 Einleitung und verwandte Arbeiten

Die Eigenschaften der Selbstähnlichkeit und Langzeitkorrelation im Internetdatenverkehr können schwer auf traditionelle Verkehrsmodelle bzw. Leistungsbewertungsmethoden für Kommunikationsnetzwerke abgebildet werden [LTWW94, FGHW99]. Approximationen und asymptotische Resultate aus der Literatur weisen auf einen starken Einfluss der Langzeitkorrelation auf die Dienstgüte hin [Nor94, FMN00, DO95]. Langzeitkorrelationen "Long Range Dependence (LRD)" werden durch den sogenannten Hurstparameter $H \in (0.5, 1)$ charakterisiert. Ein etabliertes, stochastisches Modell für den selbstähnlichen LRD Internetdatenverkehr ist die Superposition aus einem linearen Prozess mit der Rate λ und einer fraktalen Brownschen Bewegung "fractional Brownian motion (fBm)" mit einem Varianzparameter σ^2 und dem Hurstparameter $H \in (0.5, 1)$ [Nor94]. Der entsprechende, stationäre Inkrementprozess $Y(t)$ besitzt eine abklingende Autokovarianzfunktion[1], so dass $c_Y(\tau) \simeq H(2H-1)\sigma_Y^2\tau^{2H-2}$ für $\tau \to \infty$ gilt [Ber94]. Dies steht im Kontrast zur traditionellen Annahme der exponentiell abklingenden Autokovarianz für z.B. Markov oder ARMA Verkehrsmodelle. Das Entstehen von Selbstähnlichkeit und LRD im

*Englischer Titel der Dissertation: "Non-asymptotic Performance Evaluation and Sampling-based Parameter Estimation for Communication Networks with Long Memory Traffic"

[1]Definiert für Prozess $Y(t)$ und Abstand τ als $c_Y(\tau) := \mathsf{E}[Y(t)Y(t+\tau)] - \mathsf{E}[Y(t)]\mathsf{E}[Y(t+\tau)]$.

Datenverkehr wurde sowohl analytisch untersucht als auch immer wieder in Internetverkehrsmitschnitten festgestellt und nachemuliert [GMR09, LGD+10].

Praktische Studien, Approximationen und asymptotische Resultate deuten darauf hin, dass sich Warteschlangensysteme im Zusammenhang mit LRD Verkehr fundamental anders verhalten als in Verbindung mit gedächtnislosem oder Markov Verkehr. Für fBm Verkehr mit den Parameter $\lambda, \sigma^2, H \in (0.5, 1)$ beziffert ein prominentes Resultat die Wahrscheinlichkeit, dass der Pufferfüllstand B an einem Bediensystem mit konstanter Kapazität C eine Grenze b überschreitet mit

$$\mathsf{P}[B > b] \approx \exp\left(-\frac{1}{2\sigma^2}\left(\frac{C-\lambda}{H}\right)^{2H}\left(\frac{b}{1-H}\right)^{2-2H}\right) := \varepsilon_a. \tag{1}$$

Im Vergleich ist gedächtnisloser bzw. Markov Verkehr für den exponentiellen Tail-Abfall der Pufferfüllstandverteilung bekannt. Das Resultat (1) wurde als Approximation in [Nor94] und als asymptotische Grenze für $b \to \infty$ in [DO95] hergeleitet.

In den letzten Jahren hat sich das stochastische Netzwerkkalkül als mathematisches Rahmenwerk für die Leistungsbewertung von Kommunikationsnetzen entwickelt [Cha00, BLP06, CBL06]. Dieses basiert auf der separaten Beschreibung von dem Datenverkehr einerseits und dem von Netzwerkkomponenten bereitgestellten Dienst andererseits. Diese Betrachtung führt zu einer hohen Anwendbarkeit des Netzwerkkalküls, da somit unterschiedliche Verkehrs- und Dienstmodelle kombiniert werden können. Weiterhin ermöglichen Kompositionsresultate die Abstraktion von Netzwerkpfaden als ein zusammengefasstes Äquivalentsystem und somit die Ausweitung der Analyse auf Ende-zu-Ende Pfade. In dieser Arbeit wird das stochastische Netzwerkkalkül benutzt, um nicht-asymptotische, obere Schranken für Dienstgütemetriken für den LRD Verkehr zu bestimmen. Es werden Szenarien der Resourcenverteilung (Scheduling) und der Ende-zu-Ende Performanz analysiert. Die hier zusammengefassten Ergebnisse sind in [RF12, RBF13] erschienen.

Das Netzwerkkalkül beschreibt die Beziehung zwischen den kumulativen Datenankünften A und Abgängen D an einem Warteschlangensystem wie in Abb. 1. Die Ankünfte und Abgänge in einem Zeitintervall $[s, t)$ lauten $A(s, t)$ bzw. $D(s, t)$ für $0 \leq s \leq t$. $A(0, t)$ wird als $A(t)$ abgekürzt. Angenommen werden hier verlustfreie Systeme und ein fluid Datenmodell. Zeitfunktionen sind nicht-negativ, nicht-fallend, linksstetig angenommen und gehen durch den Ursprung, z.B. $0 \leq A(s) \leq A(t), \forall s \leq t$ und $A(0) = 0$. Bediensysteme sind "work-conserving" und haben eine Kapazität C. Angenommen werden Topologien ohne zyklische Abhängigkeiten wie z.B. in Abb. 1. Statistische, punktweise Einhüllende sind Schranken für die stationären Ankünfte im Sinne von $\mathsf{P}\left[A(s, t) > E(t - s) + b\right] \leq \varepsilon_p(b)$ [YS93]. Ankünfte der Verkehrsklasse "Exponentially Bounded Burstiness (EBB)" besitzen eine lineare Einhüllende $E(t) = \rho t$ mit $\varepsilon_p(b) = \phi e^{-\theta b}$ und $\phi, \theta > 0$ wie z.B. gedächtnisloser bzw. Markov Verkehr. Das Ziel der Analyse sind Leistungsschranken, die als Dienstgütekriterien gelten wie z.B. $P\left[B > b\right] \leq \varepsilon$. Diese kann mithilfe der Lindley Gleichung umformuliert werden als $\mathsf{P}[B > b] = \mathsf{P}\left[\sup_{s\in[0,t]}\{A(s, t) - C(t - s)\} > b\right]$. Die Approximation in (1) zieht das Supremum aus dem Wahrscheinlichkeitsausdruck heraus und verwendet die punktweise Einhüllende der Ankünfte. Dies führt jedoch zu einer unerwünschten unteren Schranke für $\mathsf{P}[B > b]$. Ein Ansatz zur Bestimmung einer oberen Schranke für diesen Ausdruck basiert auf der *pfadweise* Einhüllenden $E(t) \leq b + Ct$, so

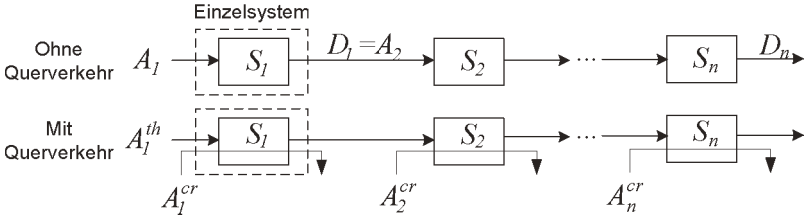


Abbildung 1: Kumulative Ankünfte A bzw. Abgänge D in einem Netwerk von Warteschlangensystemen. Ankünfte und Abgänge sind über die Dienstkurve S verknüpft. Durchgehender Verkehr A^{th} teilt sich Ressourcen an jedem System mit Querverkehr A^{cr}.

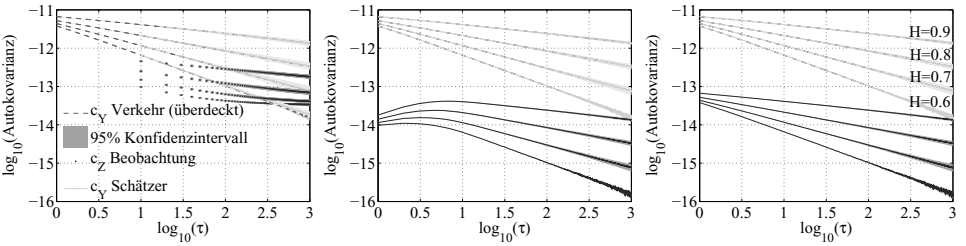
dass $\mathsf{P}\left[\sup_{s\in[0,t]}\{A(s,t)-C(t-s)\}>b\right] \leq \mathsf{P}\left[\sup_{s\in[0,t]}\{A(s,t)-E(t-s)\}>0\right]$ gilt. Das Pendant zur Einhüllenden ist die statistische Dienstkurve $\mathsf{P}\left[D(t) < \inf_{s\in[0,t]}\{A(s)+[S(t-s)-b]_+\}\right] \leq \varepsilon_d(b)$ mit $[x]_+ = \max\{x,0\}$ [CBL06]. Zur Herleitung von Leistungsschranken werden Einhüllende und Dienstkurven wie folgt kombiniert: Für Ankünfte mit einer Einhüllenden $\mathsf{P}\left[\sup_{s\in[0,t]}\{A(s,t)-E(t-s)\}>b\right] \leq \varepsilon_s(b)$ zu einem System mit der obigen Dienstkurve gilt folgende Schranke für den Pufferfüllstand $\mathsf{P}\left[B>\sup_{s\geq 0}\{E(s)-S(s)\}+b\right] \leq \varepsilon(b)$, mit $\varepsilon(b) = \inf_{x\in[0,b]}\{\varepsilon_s(x) + \varepsilon_d(b-x)\} := \varepsilon_s \otimes \varepsilon_d(b)$. Die Operation $\otimes$ ist bekannt als Min-Plus Faltung [Cha00]. Abbildung 1 zeigt den Fall von durchgehendem Verkehr $A^{th}(t)$ an Systemen mit Querverkehr $A^{cr}(t)$ mit pfadweise Einhüllenden $E^{cr}(t)$ und Verletzungswahrscheinlichkeiten $\varepsilon_s(b)$. Ohne Scheduling Annahmen liefert $S^{lo}(t) = [Ct - E^{cr}(t)]_+$ mit $\varepsilon_s(b)$ eine gültige Dienstkurve für den verfügbaren Dienst für den durchgehenden Verkehr $A^{th}(t)$.

Eine stochastische Netzwerkdienstkurve für eine Reihenschaltung (wie in Abb. 1) von n Systemen jeweils mit obiger Dienstkurve ist in [CBL06] hergeleitet als $S^{net}(t) = S_1 \otimes S_2^{-\varrho} \otimes \cdots \otimes S_n^{-(n-1)\varrho}(t)$ mit $S_i^{-\varrho} = S(t) - \varrho t$, den freien Ratenparameter $\varrho > 0$ und der Verletzungswahrscheinlichkeit $\varepsilon^{net}(b) = \varepsilon_1^{\varrho} \otimes \varepsilon_2^{\varrho} \otimes \cdots \varepsilon_{n-1}^{\varrho} \otimes \varepsilon_n(b)$. Die Netzwerkdienstkurve basiert auf einer sogenannten pfadweisen Dienstkurve, die wie folgt definiert ist: $\mathsf{P}\left[\sup_{t\in[0,u]}\{\inf_{s\in[0,t]}\{A(s)+[S(t-s)-\varrho(u-t)-b]_+\}-D(t)\}>0\right] \leq \varepsilon^{\varrho}(b)$ mit $\varepsilon^{\varrho}(b) = \frac{1}{\varrho}\int_b^{\infty}\varepsilon_d(x)dx$ und $0 \leq s \leq t \leq u$. Die Netzwerkdienstkurve bildet ein Äquivalentsystem und erlaubt somit die Herleitung von Ende-zu-Ende Leistungsschranken für Netzwerkpfade.

2 Problemstellung

Der erhebliche Einfluss der statistischen Eigenschaften von langzeitkorreliertem Datenverkehr auf die Dienstgüte begründet die Erforschung von Methoden zur Schätzung dieser Eigenschaften. Traditionelle Methoden basieren auf einer vollständigen Aufnahme des Datenverkehrs. Abschnitt 3 zielt auf die Konstruktion von statistischen Schätzmethoden für den netzweiten Einsatz ohne administrativen Zugriff auf den Datenverkehr.

Die Approximation (1) deutet darauf hin, dass LRD einen wesentlichen Einfluss auf die Netzwerkperformanz besitzt. Eine punktweise Einhüllende für fBm Verkehr mit $H \in$



(a) Periodisches Abtasten (b) Gamma Abtasten (c) Geometrisches Abtasten

Abbildung 2: Autokovarianzinferenz aus abgetasteten LRD Prozessen für verschiedene Hurstparameter H. Die Zwischenabtastzeiten sind (a) konstant, (b) Gamma bzw. (c) geometrisch verteilt mit $\mu_X = 0.1$. Die Autokovarianz "c_Z Beobachtung" ist in (a) und (b) verzerrt und in (c) nur verschoben. Die geschätzte Autokovarianz "c_Y Schätzer" stimmt mit der Autokovarianz des Verkehrs "c_Y traffic" überein.

$(0.5, 1)$ existiert in der Literatur [FMN00]. Standardmethoden zur Konstruktion von pfadweise Einhüllenden verletzen die Nebenbedingungen der Herleitung von Leistungsschranken für fBm Verkehr [Riz13]. Die Herleitung von nicht-asymptotischen, oberen Leistungsschranken für LRD Verkehr für Einzelsysteme und Netzwerkpfade war für eine gewisse Zeit ein offenes Problem, welches durch einen neuen Ansatz in den Abschnitten 4 und 5 bewältigt wird.

3 Abtastbasierte Schätzung statistischer Eigenschaften des langzeitkorrelierten Verkehrs

Im Folgenden wird ein Modell zur Schätzung von LRD Verkehrseigenschaften mittels Abtastung präsentiert. Ein fundamentales Abtastresultat aus der Warteschlangentheorie beweist, dass der Anteil der Poisson Ankünfte, die ein Warteschlangensystem in einem bestimmten Zustand erfahren, im Durchschnitt gleich dem Anteil der Zeit ist, die das System in diesem Zustand verbringt. Dieses Resultat ist bekannt als PASTA, "Poisson Arrivals see Time Averages" [Wol82]. Eine Verallgemeinerung dieses Resultats ist bekannt als NIMASTA "Non-intrusive Mixing Arrivals See Time Averages" [BMVB09] und bildet die Grundlage für erwartungstreue, abtastbasierte Schätzer von beliebigen Momenten von Zufallsvariablen.

Das Ziel dieser Arbeit ist eine stochastische Abtastung zur erwartungstreuen Inferenz der Kovarianz des LRD Verkehrs. Das Abtastmodell ist definiert auf Basis von drei stationären, zeitdiskreten Zufallsprozessen, nämlich $Y(t), X(t)$ und $Z(t)$ für $t \in \mathbb{N}_0$. Der LRD Verkehrsinkrementprozess $Y(t)$ ist charakterisiert durch die Autokovarianzfunktion aus Abschnitt 1. Der Abtastprozess $X(t)$ ist ein Kronecker Delta Prozess, wobei jedes Kronecker Delta eine erfolgte Abtastung beschreibt. Der beobachtete Prozess $Z(t)$ ist gegeben als $Z(t) = X(t)Y(t)$, wobei $X(t)$ und $Y(t)$ unabhängig sind. Die Zeit zwischen zwei Abtastwerten ist unabhängig und identisch verteilt und charakterisiert den Abtastprozess. Die Intensität $\mu_{(\cdot)}$ von Prozess $(\cdot)$ ist gegeben durch den Erwartungswert $\mathsf{E}[(\cdot)]$. Der Zusam-

menhang zwischen der Autokovarianz $c_Z(\tau)$ des beobachteten Prozesses $Z(t)$ und der unbekannten Autokovarianz $c_Y(\tau)$ des Verkehrsprozesses $Y(t)$ ist gegeben als [Riz13]

$$c_Z(\tau) = \left(c_X(\tau) + \mu_X^2\right) c_Y(\tau) + c_X(\tau)\mu_Y^2. \quad (2)$$

Aus (2) is zu erkennen, dass die Wahl des Abtastprozesses einen Einfluss auf die beobachtete Autokovarianz hat. Dieser Einfluss ist unter bestimmten Bedingung reversibel, so dass die Inferenz von $c_Y(\tau)$ möglich ist.

Beispiele verschiedener Abtastprozesse mit entsprechenden Rekonstruktionsformeln für die Autokovarianz des Verkehrsprozesses $c_Y(\tau)$ sind in [RBF13] gegeben. Abbildung 2 zeigt beispielhaft die erfolgreiche Rekonstruktion der Autokovarianz aus den Beobachtungen verschiedener Abtastprozesse. Das geometrische Abtasten führt zu einer Skalierung der Autokovarianzfunktion des Verkehrs, so dass z.B. der Hurst Parameter H direkt aus der Steigung von $c_Z(\tau)$ in Abb. 2c geschätzt werden kann.

Die Untersuchung des Schätzers der Autokovarianzfunktion für eine endliche Anzahl an Beobachtungen T ergibt, dass der Schätzer asymptotisch erwartungstreu ist. Der Bias des Schätzers nimmt für steigendes T mit $(T-\tau)^{2H-2}$ ab. Abtastprozesse, welche die schwach-mixing Eigenschaft besitzen, liefern asymtotisch erwartungstreue Schätzer für die Autokovarianz des LRD Datenverkehrs.

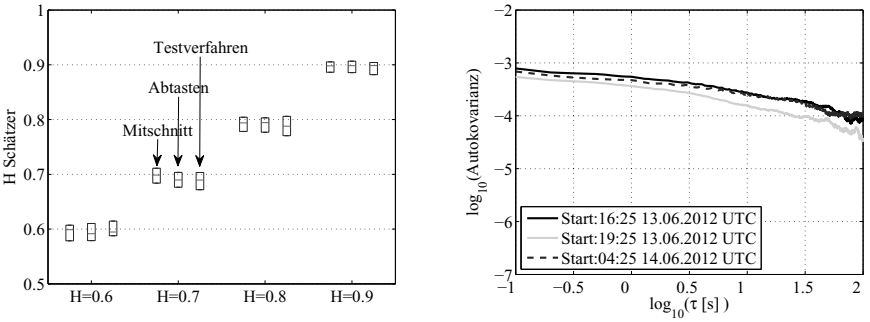
Eine Implementierung des geometrischen Abtastens wird in kontrollierter Umgebung getestet und später in Internetmessungen eingesetzt [Riz13]. Das Testverfahren basiert auf Ende-zu-Ende Latenzen injizierter Testpakete, um aus der Interaktion mit dem Datenverkehr auf die dominante Verkehrsautokovarianz, d.h. das größte H, entlang des Netzwerkpfades zu schließen. Dabei besteht kein direkter Zugang zum Verkehr. Ein Vergleich der Hurstparameterschätzung sowohl bei direktem Zugang zum Datenverkehr als auch beim Testverfahren ist auf Abb. 3a zu sehen. Hierfür wurde Datenverkehr mit verschiedenem Hurstparameter synthetisiert und in einer kontrollierten Testumgebung FILab [IL] getestet. Abbildung 3a zeigt die erfolgreiche Schätzung des Hurstparameters sowohl bei direktem Abtasten als auch unter Verwendung des Testverfahrens. Abbildung 3b zeigt den typischen LRD Verlauf der Datenverkehrsautokovarianz bei einer Momentaufnahme auf einem Internetpfad zu einem PlanetLab Rechner an einer US Universität.

4 Warteschlangensystemanalyse mit langzeitkorreliertem Verkehr

In diesem Abschnitt werden obere, nicht-asymptotische Dienstgüteschranken für LRD Ankunftsverkehr an einem Warteschlangensystem hergeleitet. Der LRD Verkehr ist modelliert durch einen fBm Prozess mit einem Hurstparameter $H \in (0.5, 1)$, eine mittlere Rate $\lambda > 0$ und einem Varianzparameter $\sigma^2 > 0$. Es wird ein zeitdiskretes Modell angenommen. Im Folgenden wird eine pfadweise Einhüllende für den fBm Verkehr hergeleitet. Man betrachte die punktweise Einhüllende $E(t) = \lambda t + \sqrt{-2\log \varepsilon_p(t)}\sigma t^H$ mit zeitabhängiger Verletzungswahrscheinlichkeit, die durch die Einführung der freien Parameter $\beta \in (0, 1-H)$ and $\eta \in (0,1)$ als $\varepsilon_p(t) = \eta^{t^{2\beta}}$ ausgedrückt werden kann.

Theorem 4.1 (FBM Pfadweise Einhüllende)**.** *Gegeben sei fBm Verkehr charakterisiert durch das Tripel λ, σ^2, und $H \in (0.5, 1)$. Eine pfadweise Einhüllende ist gegeben als*

$$E(t) = \lambda t + \sqrt{-2\log \eta}\sigma t^{H+\beta}$$



(a) FILab Testbed (b) Planetlab1.cis.upenn.edu

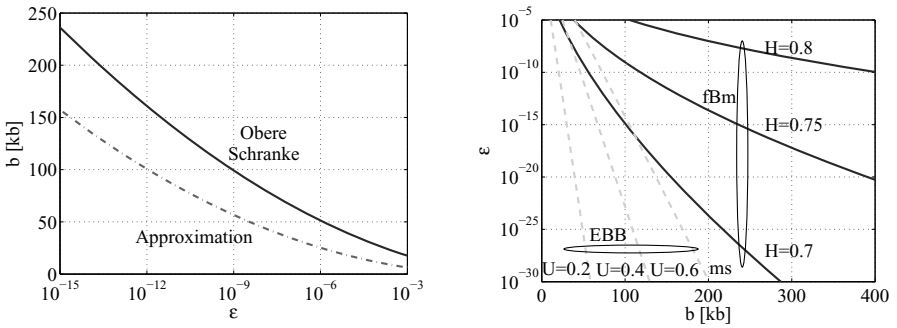
Abbildung 3: (a) Hurstparameterschätzungen bei direktem Zugang zum Datenverkehr als auch über das Testverfahren. (b) Beispiel einer Autokovarianzschätzung aus einer Ende-zu-Ende Internetmessung.

mit der Verletzungswahrscheinlichkeit $\varepsilon_s = \frac{\Gamma(\frac{1}{2\beta})}{2\beta(-\log\eta)^{\frac{1}{2\beta}}}$ *und den freien Parameter* $\beta \in (0, 1-H)$ *und* $\eta \in (0,1)$.

Der Beweis von Theorem 4.1 ist gegeben in [Riz13]. Eine intuitive Skizze des Beweises ist wie folgt: Es werden Chernoffs Theorem und Booles Ungleichung benutzt. Die punktweise Einhüllende wird durch den Parameter β relaxiert, so dass die punktweise Verletzungswahrscheinlichkeit $\varepsilon_p(t)$ mit t abklingt. Ein Lemma in [Riz13] liefert die obere Schranke für die Verletzungswahrscheinlichkeit. Die Grenzen der freien Parameter erlauben die Herleitung von Leistungsschranken für Pufferfüllstand oder Latenz. Eine Parametrisierung in geschlossener Form ist gegeben in [Riz13].

Eine obere Schranke für den Pufferfüllstand an einem Bediensystem mit konstanter Kapazität C kann gefunden werden als $\mathsf{P}[B > b] \leq \varepsilon_s$ mit ε_s und β aus Theorem 4.1. Die entsprechende Parametrisierung lautet $\eta = \exp\left(-\frac{1}{2\sigma^2}\left(\frac{C-\lambda}{H+\beta}\right)^{2(H+\beta)}\left(\frac{b}{1-(H+\beta)}\right)^{2-2(H+\beta)}\right)$. Entsprechend ist die Latenz beschränkt durch $\mathsf{P}[W > b/C] \leq \varepsilon_s$. Abbildung 4a vergleicht die Pufferfüllstandschranke zur Approximation (1) für ein Beispielszenario mit den Parameter $C = 1$ Gbps, $\lambda = 0.5$ Gbps, $\sigma = \lambda/2$, und $H = 0.75$ und eine Zeitschlitzlänge von 10 μs. Beide Verletzungswahrscheinlichkeiten besitzen einen Weibull-Tail. In [Riz13] wird bewiesen, dass der Tail-Abfall beider Formulierung log-asymptotisch gleich ist.

Die Formulierung der Pufferfüllstandschranke nach der Optimierung von β liefert den Ausdruck $\tilde{\varepsilon}_s = \frac{b}{C-\lambda}\,\varepsilon_a^{1+\log\chi}\sqrt{2\pi(-\log\varepsilon_a)}$ mit $\chi \in (\frac{1}{4}, 1)$. Diese rigorose, obere Schranke lässt ähnlich wie Approximation (1) aus [Nor94] folgendes ableiten: Die mittlere verfügbare Bandbreite $C - \lambda$ hat einen stärkeren Einfluss auf die Verletzungswahrscheinlichkeit als die Puffergröße b bei LRD Verkehr. Für $H = 0.5$, d.h. EBB Verkehr, ist der Einfluss beider Faktoren gleich. Dieses Resultat untermauert die Bedeutung von verfügbarer Bandbreite und unterstützt moderne Aufforderungen zu kleinen Routerpuffer. Abbildung 4b vergleicht den Tail-Abfall für EBB und LRD Verkehr.



(a) Pufferfüllstandschranke vs. Approximation. (b) Tail-Verhalten der Pufferfüllstandschranken.

Abbildung 4: (a) Numerisches Beispiel für die Pufferfüllstandschranke. (b) Ineffizienz von großen Puffer bei LRD Datenverkehr: Der Weibull-Tail entspricht $\log \varepsilon \simeq -K_1 b^{2-2H}$ im Vergleich zu $\log \varepsilon \simeq -K_2 b$ für den EBB Verkehr mit den Konstanten K_1, K_2. Die Änderung der Bursthaftigkeit U des Markov EBB Verkehrs ändert die Steigung.

Für Einzelsysteme mit LRD Querverkehr, wie in Abb. 1, kann eine verfügbare Dienstkurve für den durchgehenden Verkehr als $S(t) = (C - r)t$ gefunden werden, wobei der LRD Querverkehr eine affine Einhüllende $E(t) = rt$ besitzt. Die affine Einhüllende kann geometrisch bestimmt werden als obere Schranke für die Einhüllende in Theorem 4.1 [Riz13]. Die Verletzungswahrscheinlichkeit der Dienstkurve entspricht $\varepsilon_s(b)$ der affinen Einhüllenden. Mit Hilfe dieser Dienstkurve kann im nächsten Abschnitt die Analyse auf gesamte Pfade mit LRD Querverkehr ausgeweitet werden.

5 Netzwerkpfadanalyse mit langzeitkorreliertem Datenverkehr

Im Folgenden werden obere Dienstgüteschranken für Netzwerkpfade mit LRD Verkehr wie in Abb. 1 vorgestellt. Hierfür wird eine Netzwerkdienstkurve durch die Faltung der Dienstkurven der einzelnen Systeme hergeleitet. Die Netzwerkdienstkurve erlaubt die Herleitung von oberen Ende-zu-Ende Leistungsschranken z.B. für die Latenz.

Zur Bestimmung der Netzwerkdienstkurve wird eine pfadweise verfügbare Dienstkurve aufbauend auf Abschnitt 4 hergeleitet. Für ein System mit Kapazität C und LRD Querverkehr mit der affinen Einhüllenden $E(t) = rt$ ist der Ausdruck $S(t) = (C - r)t$ eine gültige, *pfadweise* verfügbare Dienstkurve mit Verletzungswahrscheinlichkeit $\varepsilon^{\varrho}(b) = \frac{\Gamma(\frac{1}{2\beta})}{2\varrho\vartheta^{\frac{1}{2\beta}}(1-(H+2\beta))} b^{-\frac{1-(H+2\beta)}{\beta}}$ mit den freien Parameter $\varrho \in (0, C - r)$ und $\beta \in \left(0, \frac{1-H}{2}\right)$ und mit $\vartheta = \frac{1}{2\sigma^2}((r-\lambda)/(H+\beta))^{2(H+\beta)}(1/(1-(H+\beta)))^{2-2(H+\beta)}$.

Im Folgenden werden n homogene Systeme angenommen. Weitere Szenarien sind in [Riz13] beschrieben. Der durchgehende Verkehr besitzt eine affine Einhüllende mit der Rate r^{th}. Jedes System besitzt eine Kapazität C und der LRD Querverkehr wird durch das Tripel λ, σ^2 und $H^{cr} \in (0.5, 1)$ beschrieben. Die Abkürzungen th and cr werden für den durchgehenden bzw. für den Querverkehr benutzt. An jedem System $i \in [1, n]$ ist eine verfügbare Dienstkurve $S_i(t) = (C - r^{cr})t$ zu finden, wobei r^{cr} die Rate der affi-

nen Einhüllenden des Querverkehrs wiedergibt. Eine Netzwerkdienstkurve im Sinne von Abschnitt 1 ist gegeben als $S^{net}(t) = (C - r^{cr} - \Delta)t$ mit konstantem $\Delta := (n-1)\varrho$ und mit $\Delta \in (0, C - r^{cr})$ für die Stabilität. Wie in Abschnitt 1 beschrieben, ergibt sich die Verletzungswahrscheinlichkeit für $S^{net}(t)$ aus der Min-Plus Faltung als $\varepsilon^{net}(b) = \frac{\Gamma(\frac{1}{2\beta})}{2\vartheta^{\frac{1}{2\beta}}} \inf_x \left\{ \frac{(n-1)^2 (\frac{b-x}{n-1})^{-\frac{1-(H+2\beta)}{\beta}}}{\Delta(1-(H+2\beta))} + \frac{x^{-\frac{1-(H+\beta)}{\beta}}}{\beta} \right\}$, mit $x \in (0, b)$ und $\beta \in \left(0, \frac{1-H}{2}\right)$.

Die Netzwerkdienstkurve erlaubt die Herleitung von Ende-zu-Ende Leistungsschranken wie in Abschnitt 1 dargestellt. Zum Beispiel ist eine obere Latenzschranke für durchgehenden Verkehr mit konstanter Rate r^{th} gegeben als $\mathsf{P}\left[W > b/(C - r^{cr} - \Delta)\right] \leq \varepsilon^{net}(b)$ mit der Stabilitätsbedingung $r^{th} + r^{cr} + \Delta \leq C$. Eine direkte Erweiterung für durchgehenden Verkehr mit der affinen Einhüllenden mit der Rate r^{th} und der Verletzungswahrscheinlichkeit ε^{th} ist gegeben als $\mathsf{P}[W > b/(C - r^{cr} - \Delta)] \leq \varepsilon^{net} \otimes \varepsilon^{th}(b)$.

Abbildung 5a zeigt ein numerisches Beispiel für Ende-zu-Ende Latenzschranken für durchgehenden Verkehr mit konstanter Rate von 250 Mbps und LRD Querverkehr mit $\lambda = 250$ Mbps, $\sigma = \lambda/2$ und unterschiedlichen Hurstparameter. Die Kapazität C beträgt 1 Gbps und die Verletzungswahrscheinlichkeit ist festgelegt auf $\varepsilon = 10^{-12}$. Die freien Parameter der Einhüllenden werden numerisch optimiert. Man erkennt, dass die End-zu-Ende Latenzschranken superlinear in Abhängigkeit von der Pfadlänge n wachsen.

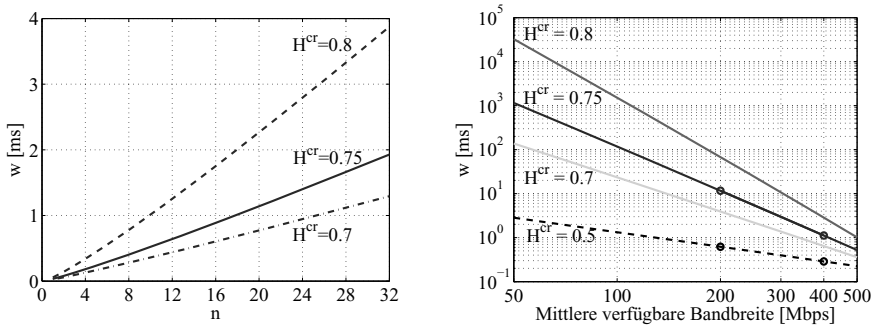
Das nachfolgende Theorem beschreibt den Zuwachs von Ende-zu-Ende Latenzen für Netzwerkpfade mit n Systemen in Folge und LRD Querverkehr.

Theorem 5.1 (Zuwachs von Ende-zu-Ende Dienstgüteschranken)**.** *Gegeben seien n homogene Systeme jeweils mit LRD Querverkehr mit Hurstparameter $H \in (0.5, 1)$. Ende-zu-Ende Dienstgüteschranken für den durchgehenden Verkehr skalieren für gleichbleibende Verletzungswahrscheinlichkeit mit $\mathcal{O}\left(n\,(\log n)^{\frac{1}{2-2H}}\right)$.*

Der Beweis zu Theorem 5.1 ist gegeben in [Riz13]. Folgend ist eine Skizze des Beweises: Der Zuwachs ist durch den Tail-Abfall der Verletzungswahrscheinlichkeiten der einzelnen, verfügbaren Dienstkurven des Pfades charakterisiert. Diese werden für die Netzwerkdienstkurve mittels der Min-Plus Faltung verknüpft. Der Einfluss des Hurstparameters H des Querverkehrs ist moderat aufgrund der polylogarithmischen Komponente in Theorem 5.1. Für $H = 0.5$ gibt Theorem 5.1 das aus der Literatur bekannte Zuwachsresultat für EBB Verkehr wieder [CBL06]. Abbildung 5b zeigt die Bedeutung der mittleren verfügbaren Bandbreite für Ende-zu-Ende Latenzschranken im Zusammenhang mit LRD Querverkehr. Für eine festgelegte Verletzungswahrscheinlichkeit führt das Halbieren der verfügbaren Bandbreite zur Verdoppelung der Latenzschranke w für $H = 0.5$. Dieser Zusammenhang verschlechtert sich wesentlich für $H \in (0.5, 1)$. Dieses Beispiel untermauert die Bedeutung von Überkapazitäten in Rechnernetzen.

6 Schlüsselbeiträge

Im Rahmen dieser Arbeit wurde eine abtastbasierte Schätzmethode für die Langzeitkorrelationen des Datenverkehrs vorgestellt. Die Rückgewinnung der statistischen Eigenschaften des Verkehrs aus den zufälligen Beobachtungen wurde analytisch bewiesen. Der Einfluss einer begrenzten Anzahl an Beobachtungen auf die Schätzgenauigkeit wurde quan-



(a) (b)

Abbildung 5: (a) Ende-zu-Ende Latenz für LRD Querverkehr wächst superlinear mit der Pfadlänge n an. (b) Der starke Einfluss der mittleren, verfügbaren Bandbreite auf die Dienstgüte im Zusammenhang mit LRD Querverkehr.

tifiziert. Weiterhin wurden asymptotisch erwartungstreue Schätzer für die Langzeitkorrelationen des Verkehrs vorgestellt. Diese Arbeit ergänzt traditionelle Schätzmethoden aus der Literatur durch ein Verfahren, welches Testpakete in das Netzwerk injiziert, auf zufälligem Abtasten aufbaut und keine administrative Berechtigung an Router erfordert. Im Gegensatz zu traditionellen Methoden, welche die statistischen Eigenschaften an einem einzelnen Router wiedergeben, ermöglicht das vorgestellte Verfahren die dominanten Eigenschaften entlang eines untersuchten Pfades zu schätzen. Das präsentierte Verfahren wird in einer kontrollierten Umgebung verifiziert und es werden beispielhafte Ergebnisse einer Langzeit-Internetstudie vorgestellt.

Im Rahmen des stochastischen Netzwerkkalküls wurden Leistungsschranken für Ende-zu-Ende Latenzen in Netzwerken mit langzeitkorreliertem Datenverkehr hergeleitet. Ein grundlegendes Resultat dieser Arbeit beziffert den Zuwachs von Ende-zu-Ende Latenzen für Netzwerkpfade mit n Systemen in Folge auf $\mathcal{O}\left(n(\log n)^{\frac{1}{(2-2H)}}\right)$. Vergleichbare Resultate aus der Literatur sind z.B. $\Theta(n)$ unter statistischer Unabhängigkeit und $\Theta(n \log n)$ für EBB Verkehr. Die Ergebnisse dieser Arbeit quantifizieren den Einfluss der Langzeitkorrelationen auf die Netzwerkperformanz. Die in geschlossener Form hergeleiteten Resultate haben eine grundlegende Auswirkung auf die Dimensionierung und den Betrieb von Kommunikationsnetzwerken.

Literatur

[Ber94] J. Beran. *Statistics for Long-Memory Processes*. Chapman & Hall/CRC, Oktober 1994.

[BLP06] A. Burchard, J. Liebeherr und S. D. Patek. A Min-Plus Calculus for End-to-End Statistical Service Guarantees. *IEEE Trans. Inform. Theory*, 52(9):4105–4114, September 2006.

[BMVB09] F. Baccelli, S. Machiraju, D. Veitch und J. Bolot. The Role of PASTA in Network Measurement. *IEEE/ACM Trans. Networking*, 17(4):1340–1353, August 2009.

[CBL06] F. Ciucu, A. Burchard und J. Liebeherr. Scaling Properties of Statistical End-to-end Bounds in the Network Calculus. *IEEE/ACM Trans. Networking*, 14(6):2300–2312, Juni 2006.

[Cha00] C.-S. Chang. *Performance Guarantees in Communication Networks*. Springer, 2000.

[DO95] N. G. Duffield und N. O'Connell. Large Deviations and Overflow Probabilities for the General Single-Server Queue, with Applications. *Math. Proc. Camb. Phil. Soc.*, 118(2):363–375, September 1995.

[FGHW99] A. Feldmann, A. C. Gilbert, P. Huang und W. Willinger. Dynamics of IP traffic: A Study of the Role of Variability and the Impact of Control. *SIGCOMM Comput. Commun. Rev.*, 29(4):301–313, August 1999.

[FMN00] N. Fonseca, G. Mayor und C. Neto. On the Equivalent Bandwidth of Self-similar Sources. *ACM Trans. Model. Comput. Simul.*, 10(2):104–124, April 2000.

[GMR09] H. Gupta, A. Mahanti und V. Ribeiro. Revisting Coexistance of Poissonity and Self-similarity in Internet Traffic. In *Proc. of IEEE/ACM MASCOTS*, September 2009.

[IL] Future Internet Lab. http://www.filab.uni-hannover.de/.

[LGD+10] P. Loiseau, P. Goncalves, G. Dewaele, P. Borgnat, P. Abry und P. Primet. Investigating Self-Similarity and Heavy-Tailed Distributions on a Large-Scale Experimental Facility. *IEEE/ACM Trans. Networking*, 18(4):1261–1274, August 2010.

[LTWW94] W. Leland, M. Taqqu, W. Willinger und D. Wilson. On the Self-similar Nature of Ethernet Traffic (Extended Version). *IEEE/ACM Trans. Networking*, 2(1):1–15, Februar 1994.

[Nor94] I. Norros. A Storage Model with Self-similar Input. *Queueing Systems*, 16(3):387–396, September 1994.

[RBF13] A. Rizk, Z. Bozakov und M. Fidler. Estimating Traffic Correlations from Sampling and Active Network Probing. In *Proc. of IFIP Networking*, Mai 2013.

[RF12] A. Rizk und M. Fidler. Non-asymptotic End-to-end Performance Bounds for Networks with Long Range Dependent FBM Cross Traffic. *Computer Networks*, 56(1):127–141, 2012.

[Riz13] Amr Rizk. *Non-asymptotic Performance Evaluation and Sampling-based Parameter Estimation for Communication Networks with Long Memory Traffic*. Dissertation, Leibniz Universität Hannover, 2013.

[Wol82] R. Wolff. Poisson Arrivals See Time Averages. *Operations Research*, 30(2):223–231, Marz 1982.

[YS93] O. Yaron und M. Sidi. Performance and Stability of Communication Networks via Robust Exponential Bounds. *IEEE/ACM Trans. Networking*, 1(3):372–385, Juni 1993.

Amr Rizk schloss im Jahr 2008 das Wirtschaftsingenieur-Studium an der Technischen Universität Darmstadt ab. Im Jahr 2013 wurde er an der Fakultät für Elektrotechnik und Informatik der Leibniz Universität Hannover zum Dr.-Ing. (mit Auszeichnung) promoviert. Seine Forschungsschwerpunkte liegen in den Bereichen der Leistungsbewertung von Rechnernetzen, der stochastischen Modellierung von Kommunikationssystemen und der Nachrichtenverkehrstheorie.

Ein Rahmenwerk für die qualitative Analyse der Paarprogrammierung

Stephan Salinger

Institut für Informatik
Freie Universitat Berlin
stephan.salinger@fu-berlin.de

Abstract: *Hintergrund:* Der Praktik der Paarprogrammierung (PP) werden verschiedene Vor- und Nachteile zugeschrieben. Der weitaus überwiegende Teil der bisherigen Untersuchungen der PP kommt allerdings bez. der Eigenschaften der PP zu keinem einheitlichen Bild. Dies liegt vor allem daran, dass in diesen meist quantitativen Studien der eigentliche PP-Prozess als Black Box betrachtet wird. *Ziel:* Mit der vorliegenden Arbeit wird angestrebt, den ersten entscheidenden Schritt zu einem holistischen Verständnis des PP-Prozesses zu erbringen und somit langfristig dabei zu helfen, die Praktik in der Industrie zielführend einsetzen zu können. *Methode:* Qualitative Analyse von Videoaufzeichnungen industrieller PP-Sitzungen mit Hilfe der Grounded Theory Methodology (GTM). *Ergebnis:* Es wird sowohl eine Forschungsmethode (Erweiterung der GTM) sowie ein von Regeln begleitetes Kategoriensystem zur Analyse des Prozesses der PP bereitgestellt. Beide Teile zusammen erlauben es, in weiteren, aufeinander aufbauenden Studien spezielle Aspekte der PP zu analysieren und so sukzessive zu einem Gesamtverständnis der PP zu kommen. Die hier vorgestellten Untersuchungen wurden 2013 im Rahmen einer gleichnamigen Dissertation veröffentlicht [Sal13].

1 Einführung

Bei der Praktik *Paarprogrammierung* (PP) teilen sich zwei Softwareentwickler einen Computer und somit in der Regel auch eine Tastatur und eine Maus, um zeitgleich zusammen Artefakte (im allgemeinen Programmkode) zu bearbeiten. Die Praktik kommt häufig, aber keineswegs ausschließlich, im Rahmen agiler Softwareentwicklungsprozesse zum Einsatz. Ihr werden unterschiedliche Vor- und Nachteile zugeschrieben. Befürworter der PP betonen folgende Eigenschaften (siehe z.B. [HDAS09]): 1. *Erhöhte Geschwindigkeit:* Programmkode kann schneller entwickelt und zum Einsatz gebracht werden; 2. *Verbesserte Qualität:* Der entwickelte Programmkode enthält weniger Defekte; 3. *Verbessertes Design:* Der entwickelte Programmkode ist besser entworfen und leichter lesbar; 4. *Konzentrierteres Arbeiten:* Paarprogrammierer arbeiten konzentrierter als Soloprogrammierer; 5. *Erhöhte „Truck Number":* Jeder Teil des Kodes ist zumindest zwei Programmierern bekannt und kann somit zumindest von zwei Personen ohne längere Einarbeitungszeit korrigiert oder weiterentwickelt werden; 6. *Verbesserter Wissenstransfer:* Die Partner lernen voneinander in unterschiedlichen Bereichen (Arbeitsstil, Design, Kodierpraktiken, Details von Programmiersprachen und Bibliotheken, Anforderungen, Anwendungsdomänen etc.);

7. *Größeres Vertrauen:* Paarprogrammierer haben ein höheres Vertrauen in ihre Arbeitsergebnisse; 8. *Höhere Zufriedenheit:* Paarprogrammierer haben mehr Spaß an ihrer Arbeit. Kritiker befürchten hingegen zumindest die beiden folgenden Effekte: 1. *Höhere Kosten:* Wenn ein Paar nicht doppelt so schnell wie ein einzelner Entwickler arbeitet, ist PP teurer als die klassische Form der Programmierung mit nur einem Entwickler pro Aufgabe (*Soloprogrammierung* (SP)); 2. *Starke Aversionen:* Einige Entwickler lehnen eine derartige Form der Zusammenarbeit prinzipiell oder in Bezug auf bestimmte Partner ab. PP mindert deren Produktivität und Motivation.

Die Annahmen über die Eigenschaften sowie das in den letzten Jahren allgemein hohe Interesse an agilen Entwicklungsmethoden haben dazu geführt, dass die PP zu einem beachteten Forschungsthema innerhalb des Software Engineerings geworden ist. Sie wurde in einer Vielzahl von Studien zumeist quantitativ analysiert. Der eigentliche Prozess wurde hierbei in der Regel als Black Box betrachtet, also nicht näher untersucht. Legt man die Ergebnisse dieser Studien nebeneinander, so ergibt sich zumeist ein heterogenes Bild: Viele der propagierten Eigenschaften werden unterschiedlich bewertet. So benötigten beispielsweise die Soloprogrammierer bei Nosek [Nos98] im Durchschnitt (nicht signifikante) 41% länger als die Paarprogrammierer, während bei Arisholm et al. [AGDS07] die Paare „nur“ 8% schneller waren. In zwei Studien zeigte sich sogar ein negativer Effekt der PP auf die Entwicklungsgeschwindigkeit, also eine Verlangsamung.[1] Darüber hinaus liefern die bisherigen Resultate weder (verifizierte) Erklärungsmodelle für die beobachteten Effekte noch für die Diskrepanzen zwischen den Ergebnissen unterschiedlicher Studien. Auch wurde bisher nicht versucht, eine allgemeine Theorie der PP zu erarbeiten oder Grundlagen für eine solche zu legen und somit die PP als zu erlernende Methode im Rahmen von Softwareentwicklungsprozessen zu etablieren.

2 Zielsetzung

Betrachtet man die bisherigen Untersuchungen zur PP, kann festgehalten werden, dass der weitaus überwiegende Teil der Arbeiten zwar Zahlen liefert, deren Zustandekommen aber nicht hinreichend erklären und somit kaum einen Beitrag zur Optimierung des Einsatzes der Praktik liefern kann. Die hier beschriebene empirische Untersuchung leistet einen ersten Schritt, um diese Lücke zu schließen. Den Ausgangspunkt bildete die Einsicht, dass ein Verständnis der Wirkungsweise der PP nur erlangt werden kann, wenn in einem ersten Schritt die interne Prozessstruktur, der sogenannte *Mikroprozess* [Ost06] der PP, analysiert wird. In einer Pilotstudie zeigte sich schnell, dass der Versuch, alle relevanten Mikroprozesseigenschaften auf einmal bzw. zusammen zu analysieren aufgrund der Vielzahl der potentiellen relevanten Aspekte zum Scheitern verurteilt ist (siehe hierzu auch [SPP08]). Das Ziel der hier beschriebenen Arbeit konnte deshalb nicht darin bestehen, eine möglichst vollständige Theorie des Prozesses der PP bereitzustellen, sondern musste erst einmal ins Auge fassen, die Grundlagen für zukünftige Erforschungen des Mikroprozesses der PP zu schaffen – und zwar derart, dass sich deren Ergebnisse direkt konsolidieren lassen und

[1] Siehe die Metaanalyse von Hannay et al. [HDAS09] zu den Faktoren Entwicklungsgeschwindigkeit, Entwicklungsaufwand/Kosten und Qualität.

dann auf längere Sicht zu einer umfassenden Theorie über die PP führen. Da hierbei nur in sehr geringem Maß auf vorhandene wissenschaftliche Erkenntnisse über den Paarprogrammierungsprozess an und für sich und somit auch nicht auf Erfahrungen in Hinsicht auf diesbezüglich geeignete Forschungsmethoden zurückgegriffen werden konnte, wurden zwei Kernziele festgelegt[2]:

1. Entwicklung einer speziell auf die Analyse des Paarprogrammierungsprozesses zugeschnittenen qualitativen Untersuchungsmethode.

2. Entwicklung eines Kategoriensystems samt Anwendungsregeln – später zusammen *Basisschicht* (BS) genannt – mit dem die den Prozess der PP formenden grundlegenden Aktivitäten der einzelnen Paarmitglieder, die sogenannten *Basisaktivitäten* der PP, beschrieben bzw. konzeptualisiert werden können. Mit der BS sollte ein Werkzeug zur Verfügung gestellt werden, das es ermöglicht, in nachgelagerten, partiell aufeinander aufbauenden Untersuchungen spezielle Aspekte des Paarprogrammierungsprozesses (z.B. den Wissenstransfer oder den Lösungsprozess) im Detail beleuchten zu können. Insbesondere sollte die BS weitere Untersuchungen dadurch erleichtern, dass bei den jeweils notwendig werdenden Konzeptionalisierungen nicht bei Null, sprich mit einer leeren Konzeptmenge, begonnen werden muss, sondern stattdessen die Möglichkeit besteht, das Kategoriensystem – später *Basiskonzeptmenge* (BKM) genannt – oder seine Elemente auszudifferenzieren, zu verallgemeinern, zu erweitern, zu beschneiden, zu ergänzen oder zum Zweck der Verbesserung der theoretischen Sensibilität [SC90] einzusetzen. Die BKM war also von vorn herein nicht als reines Kodierschema angelegt.

Entscheidend für die Herleitung war die Festlegung, dass die BS/BKM nicht unter Zuhilfenahme von allgemeinen Theorien oder anderen Kodierschemata postuliert, sondern direkt aus Sitzungsdaten extrahiert werden muss. Hierdurch sollte sichergestellt werden, dass mittels der BS/BKM nachfolgend tatsächlich auf die PP passende Theorien entwickelt werden können (siehe hierzu auch die Diskussion „Emergence vs. Forcing“ in [Kel05]). Eine Kernaufgabe der Extraktion bestand also darin, empirisch begründet zu charakterisieren, was unter einer Basisaktivität in Hinsicht auf Inhalt und Granularität verstanden werden soll. Die Entwicklung der Forschungsmethode und die Herleitung der BS gingen zumindest zu Beginn der Untersuchungen Hand in Hand.

3 Ergebnisse

Im Weiteren werden die Ergebnisse bez. der beiden oben aufgeführten Ziele erörtert. Hierbei mag es verwunderlich erscheinen, dass einer der beiden Ergebnisblöcke mit dem Begriff *Untersuchungsmethode* überschrieben ist. Hier ist zu beachten, dass es sich keineswegs nur um die Beschreibung der im Rahmen der hier vorgestellten Arbeit verwende-

[2] Es existiert zwar eine Reihe auf die Untersuchung von Programmierarbeit bezogener Kodierschemata (z.B. [ML99]), diese weisen aber jeweils zumindest eine der folgenden Eigenschaften auf und sind somit im Rahmen der angestrebten Untersuchungen nur begrenzt geeignet: i) Keine Ausrichtung auf die Analyse dialogischer Kommunikation oder die PP an und für sich. ii) Rückgriff auf Schemata aus anderen Bereichen, deren Eignung oder Angemessenheit für das hier angestrebte Ziel nicht per se offensichtlich ist. iii) Ausrichtung auf qualitativ-quantitative Studien. iv) Eingeschränkte Ausrichtung auf Teilaspekte der PP.

ten Methode handelt, sondern um die Erörterung eines Verfahrens für zukünftige, sich ergänzende bzw. aufeinander aufbauende Untersuchungen der PP.

3.1 Untersuchungsmethode

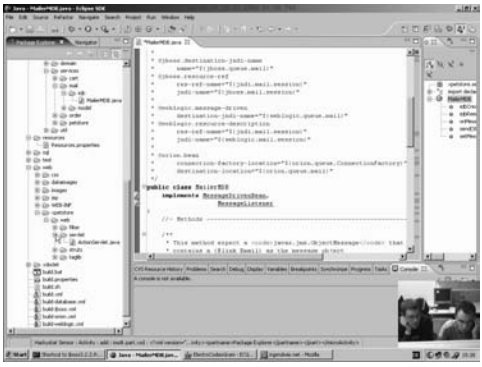


Abbildung 1: Aufbau der analysierten Videos aus drei miteinander synchronisierten Teilen: 1. Desktop des Rechners, an dem das Paar arbeitet, 2. Webcam-Video der Entwickler (unten links) und 3. während der Sitzung aufgezeichneter Ton.

Den Kern der entwickelten Analysemethode bildet die qualitative Untersuchung von Audio-/Videoaufzeichnungen von Paarprogrammierungssitzungen (siehe Abbildung 1) mittels der *Grounded Theory Methodology* (GTM) nach Strauss und Corbin [SC90][3]. Mit der Wahl der GTM, die im Kern ein exploratives Vorgehen vorsieht, wurde der Tatsache Rechnung getragen, dass zu Beginn kaum Erkenntnisse vorlagen, die sich direkt auf den Mikroprozess der PP beziehen. Der GTM wurden – nachdem erste Analyseversuche zeigten, dass die Gefahr besteht, in den komplexen Videodaten verloren zu gehen – vier neu entwickelte Praktiken zur Seite gestellt. Erste Versionen der Praktiken wurden 2007/2008 veröffentlicht [SPP08].

1. Blickwinkel auf die Daten: Durch die Verwendung eines sogenannten Blickwinkels lässt sich die bei einer GTM-Untersuchung notwendigerweise offene Forschungsfrage strukturiert formulieren. Er hilft somit dabei, während der Untersuchung interessierende besser von weniger interessierenden Phänomenen unterscheiden zu können. Über den Blickwinkel auf die Daten werden drei Aspekte festgelegt: Die Ausrichtung des Interesses, die Art der überhaupt zu berücksichtigenden Phänomene sowie der Ergebnistyp der Untersuchung (bei dem es sich nicht immer notwendigerweise um eine Theorie handeln muss). Der Blickwinkel muss explizit als etwas Vorläufiges angesehen und im Laufe einer Untersuchung regelmäßig hinterfragt werden.

Der zu Beginn der Herleitung der BS festgelegte Blickwinkel [Sal13, S. 122 ff] besagte, dass die den Prozess der PP formenden grundlegenden Aktivitäten der einzelnen Paarmitglieder zu ermitteln und zu konzeptualisieren sind, um eine extensionale Beschreibung eines auf die PP bezogenen Aktivitätsbegriffs zu erhalten. Hierbei sollten nur direkt beobachtbare Phänomene, d.h. im Wesentlichen Aspekte von tatsächlichem Verhalten berücksichtigt werden. Das Ergebnis sollte aus einer Menge von Konzepten (*Basiskonzepten*) bestehen, die weder mittels Eigenschaften ausdifferenziert noch mit Relationen versehen sein muss.

[3] Im Rahmen der Dissertation wurde hierbei die QDA-Software ATLAS.ti (http://www.atlasti.com) eingesetzt, um ohne Umweg über Transkriptionen direkt auf Videos kodieren zu können.

2. Syntaktische Regeln für Konzeptnamen: Die GTM macht keinerlei Vorgaben oder Anmerkungen dahingehend, wie Konzepte zu benennen sind. Dieser Freiheitsgrad führte im Rahmen der ersten Analyseversuche dazu, dass unterschiedlichste Formen von Konzeptnamen verwendet wurden, die sich darüber hinaus auf schwierig auseinanderzuhaltenden Granularitäts- und Abstraktionsniveaus bewegten. Diese Diversität machte es zunehmend schwierig, sich bei einzelnen Kodierungen an alle potentiell geeigneten Konzepte zu erinnern oder Konzepte bzw. Gruppen von Konzepten miteinander zu vergleichen. Hieraus wurde der Schluss gezogen, dass es – zumindest in der ersten Phase der Analyse – hilfreich sein kann, mit einem Namensschema zu arbeiten. Auf Basis der in den ersten Untersuchungen gesammelten Erfahrung wurde ein solches für die BKM festgelegt. Es besteht im Kern aus einem Verb, das die Aktivität beschreibt, der ein Entwickler gerade nachgeht, und einem Objekt, auf das sich die Aktivität bezieht (in der Form *verb_object*).

3. Modell der Analysemethoden und -objekte: In der ersten Phase der Untersuchung zeigte sich, dass drei mit der Analysemethode und dem Analysewerkzeug zusammenhängende Aspekte Probleme bereiten: i) Der dem GTM-Prozess inhärente fließende Übergang zwischen Phänomenwelt und konzeptueller Welt, ii) fehlende Leitlinien, mit denen Ideen aus der GTM systematisch zur Lösung von Kodierproblemen eingesetzt werden können sowie iii) unterschiedliche Begriffsbildungen in der GTM und der verwendeten Analysesoftware (hier ATLAS.ti). Um diesen Problemen zu begegnen, wurde das sogenannte Modell der Analysemethoden und -objekte entwickelt. Dieses Modell zeigt in Form eines UML-Klassendiagramms, welche Objekte in welcher Form auf welchem Abstraktionsniveau betrachtet werden sollten [Sal13, S. 112 ff]. Es kann jederzeit im Analyseprozess herangezogen werden, um Fragen wie „Worüber rede ich gerade?“ oder „Welche Form von Erkenntnis strebe ich gerade in welcher Weise an?“ zu beantworten.

4. Kodieren im Paar: Als neue Praktik wurde das Kodieren im Paar eingeführt. Sie fußt auf der Empfehlung von Glaser und Strauss, „die theoretischen Begriffe mit einem oder mehreren Teamkollegen zu diskutieren“, um „Verfehltes klarzustellen sowie Gesichtspunkte hinzuzufügen, auf die [die Kollegen] während der eigenen Kodierung und Datenerhebung gestoßen sind“ [GS05, S. 114], dehnt den Ratschlag aber auf alle Teile einer GTM-Analyse aus: Alle Kodierarbeit sollte zumindest zu Beginn einer Untersuchung von zwei Personen zusammen – und im Falle der Verwendung von ATLAS.ti oder einer anderen QDA-Software auch an einem Rechner – also analog der PP durchgeführt werden.

Die Verwendung der Praktiken führte dazu, dass eine Vielzahl der Probleme, die die ersten Analyseversuche behindert hatten, verschwanden oder abgemildert wurden [SPP08].

3.2 Die Basisschicht

Das Hauptergebnis der Arbeit bildet die *Basisschicht* (BS), welche aus einer Menge von Konzepten – der *Basiskonzeptmenge* (BKM) – und aus Regeln zur Anwendung dieser Konzepte besteht (ca. 300 Seiten in [Sal13]). Weiterführende Untersuchungen sollten die Regeln allerdings keineswegs stur anwenden, sondern vor dem Hintergrund neuer, sich aus der Analyse weiterer Daten ergebenden Erkenntnisse anpassen.

Tabelle 1: Nicht zusammenhängende Beispiele für die Verwendung des Konzeptes *propose_strategy*.

Äußerung	Kontext/Kommentar	Typ
„Also erst mal würd' ich den `EmailSpy` komplett außen vor lassen. Und erst mal nur den‿den `XPetstore` umstellen."	Der Entwickler teilt das weitere Vorgehen in zwei Schritte auf: Zuerst sollen die Klassen der Applikation `XPetstore` vollständig umgestellt, danach an der neuen Funktionalität `EmailSpy` gearbeitet werden.	OWP
„Weißt Du, was wir jetzt machen? Unseren vorhandenen `EmailSpy` benutzen wir zum Debuggen."	Der Sprecher schlägt eine Strategie vor, wie vorgenommene Kodeänderungen im weiteren Verlauf getestet werden können. Es soll ein Programm verwendet werden, welches vom Paar im Vorfeld der Sitzung entwickelt worden war. Dieses spricht genau die Funktionalitäten an, die momentan vom Paar editiert werden. Das Paar erspart sich hierdurch kompliziertere Testaufrufe.	DPR
„Und dann müssen wir nur noch dieses Ding überall da einbauen wo (˜sich Freunde ändern)."	Der Sprecher ergänzt einen Vorschlag des Partners um einen weiteren Arbeitsschritt. Mit der adverbialen Bestimmung „nur noch" stellt er heraus, dass man auf diese Weise zum Ziel gelangt.	EXS

Um zu einer stabilen Form[4] der BS zu kommen, wurden sieben Paarprogrammierungssitzungen mit einer Gesamtlänge von ca. 14 Stunden untersucht. Bis auf eine Sitzung, die im Rahmen einer Hauptstudiumsveranstaltung an der FU Berlin stattfand, handelte es sich bei allen analysierten Sitzungen um solche mit professionellen, PP-erfahrenen Entwicklern. Sie wurden in drei verschiedenen Firmen aufgezeichnet. Es wurde kein Einfluss auf die Auswahl der zu bearbeitenden Programmieraufgabe genommen. Diese stammten vielmehr direkt aus dem jeweiligen Arbeitsalltag (Java und PHP). Die aufgezeichneten Sitzungen hatten Längen zwischen gut einer und knapp drei Stunden. Drei der Sitzungen wurden soweit im Detail untersucht, dass weitgehend alle Äußerungen und Handlungen kodiert wurden (letztendlich 3008 Kodeinstanzen an 2464 Videoausschnitten). Die Anwendung der erweiterten GTM führte dazu, dass zwei Gruppen von Konzepten identifiziert wurden:

1. HHI (human-human interaction): Die Klasse HHI umfasst alle Konzepte, mit denen Interaktionen zwischen den Paarmitgliedern, also aufeinander bezogene Handlungen, beschrieben werden können. Da verbale Kommunikation als zentrale Interaktion identifiziert wurde, wurde festgelegt, dass die Klasse nur solche Elemente enthalten soll, die der Konzeptualisierung verbaler Äußerungen dienen. Nachfolgende Untersuchungen sollten aber darüber nachdenken, das Anwendungsgebiet der Klasse HHI auszudehnen.

2. HCI/HEI (human-computer interaction/human-environment interaction): Die Menge HCI umfasst alle Konzepte, die Entwickleraktivitäten adressieren, die sich auf den Computer bzw. auf Programme, die auf ihm laufen, beziehen, während die Klasse HEI alle Konzepte bündelt, die sich auf Aktivitäten beziehen, die in Hinsicht auf die sonstige Arbeitsumgebung geschehen.

Im Wesentlichen wurden 69 Konzepte in den Daten gegründet, von denen die HHI-Konzepte den Kern der BKM bilden. Sie verteilen sich auf vier Hauptklassen:

Produktorientierte Konzepte werden im Zusammenhang mit Vorschlägen (und Entgeg-

[4] Eine erste Version der BS wurde 2008 auf dem Workshop der Psychology of Programming Interest Group (PPIG) präsentiert [SP08].

nungen auf solche) verwendet, die die Gestalt des zu entwickelnden Programms betreffen, also die Inhalte, die Struktur und die Anordnung der Programmartefakte.

Prozessorientierte Konzepte referenzieren Vorschläge (und Entgegnungen auf solche), die sich auf die Gestaltung des Arbeitsprozesses beziehen. Z.B. adressiert die Klasse *strategy* Äußerungen zu einem längerfristig ausgerichteten, planvollen und in der Regel aus mehreren komplexeren Einzelschritten bestehenden Handeln.

Universelle Konzepte adressieren Äußerungen, mit denen Wissen angefordert, transferiert und beurteilt wird sowie Wissenslücken offenbart oder Vermutungen bzw. Hypothesen aufgestellt und bewertet werden. Sie heißen universell, da sie sowohl im Produkt- wie auch im Prozesskontext verwendet werden können.

Fassadenkonzepte bilden eine Teilmenge der universellen Konzepte und liefern einen vereinfachten, auf einen bestimmten Aspekt fokussierten Blick auf in der Regel komplexere Phänomene, z.B. auf Äußerungen oder Sequenzen von Äußerungen bez. momentan ablaufender HCI- oder HEI-Aktivitäten. Hierdurch stellen sie unter anderem einen Kontext zur Verfügung, in dem diese komplexeren Phänomene aufgesplittet und durch spezifischere Konzepte beschrieben werden können.

In Abbildung 2 sind die HHI-Konzepte in Übersicht dargestellt. Sie wurden derart konzipiert, dass bei ihrer Anwendung die folgenden Ziele verfolgt werden können: 1. Erfassen der primären Intention des Sprechenden. 2. Sichtbarmachen von sprachlichen Bezügen, beispielsweise von Frage-Antwort-Paaren oder Entscheidungsfindungsprozessen. Im Rahmen der Herleitung der BS musste deshalb herausgearbeitet werden, was unter der primären Intention einzelner Äußerungen verstanden werden soll. Anschließende Vergleiche mit Konzepten aus der Linguistik bzw. der linguistischen Gesprächsanalyse zeigten, dass das Intentionsverständnis der BS eng mit Konzepten aus der Sprechakttheorie [Sea69], z.B. dem primären und sekundären illokutionären Akt, verwandt ist.

Im Zusammenhang mit der Beschreibung der einzelnen Unterklassen – z.B. der durch das Objekt *strategy* gebildeten – wurden die sogenannten *primär charakterisierenden Eigenschaften* eingeführt. Grob gesprochen handelt es sich bei diesen um Eigenschaften, deren Kenntnis unverzichtbar dafür ist, dass die Konzepte der Klasse im intendierten, aus den Daten ermittelten Sinn verstanden und verwendet werden können. Im Falle von *strategy* gehört beispielsweise die Information darüber dazu, dass es drei, sich in vieler Hinsicht stark unterscheidende, grundsätzliche Typen von Strategien gibt, die unter dem Begriff *strategy* subsummiert werden: 1. *OWP (organising work packages)*: Teile der vorliegenden Aufgabe werden in einzelne konkrete Arbeitsschritte zerlegt; 2. *DPR (determining procedure rules)*: Es werden Leitlinien vorgeschlagen, die zumeist in einer kontinuierlichen Art und Weise helfen sollen, nachfolgende Arbeiten durchzuführen oder zu strukturieren; 3. *EXS (expansioning step)*: Taktischen Äußerungen (vom Typ *step* oder *design*) werden zu einer Strategie ausgebaut. Man beachte hierzu auch die Beispiele in Tabelle 1.

Neben den primär charakterisierenden Eigenschaften wird auch auf spezielle Eigenschaften verbaler Kommunikation eingegangen. Hiermit wird der Tatsache Rechnung getragen, dass es mittels verbaler Äußerungen vielerlei Möglichkeiten gibt, um ein und denselben Aspekt zu artikulieren, dass also gleichartige Phänomene in gänzlich unterschiedlicher Gestalt auftreten können. Außerdem wird diskutiert, wie damit umzugehen ist, dass kon-

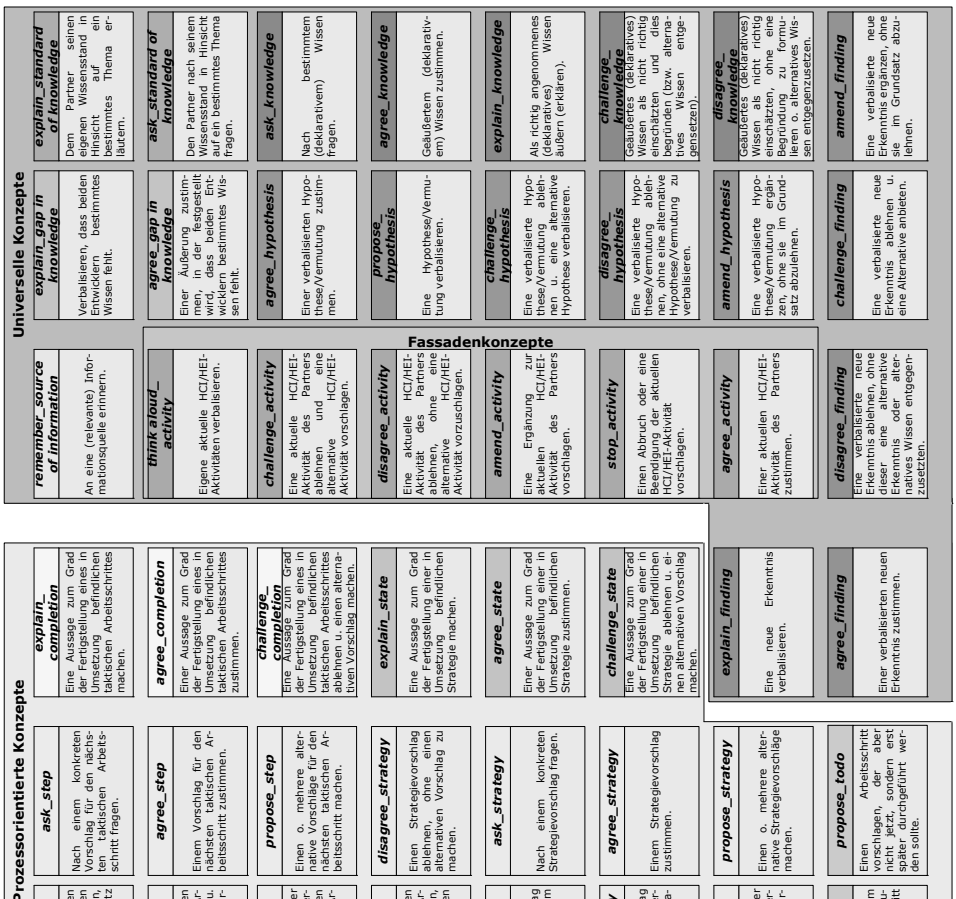


Abbildung 2: Die HHI-Konzepte geordnet nach Haupt- und Unterklassen. Einen Sonderfall bildet die Klasse *activity*. Hier beziehen sich die reaktiven Verben (z.B. *challenge*) nicht auf Äußerungen zu Äußerungen, sondern auf Äußerungen zu HCI/HEI-Aktivitäten.

textfreie oder situationsenthobene Äußerungen „im Grunde“ pragmatisch mehrdeutig sind [LNP04], der Kontext aber oft nur begrenzt eruiert werden kann. Nicht zuletzt wird besonderen Wert auf die Abgrenzung von Konzepten zueinander gelegt, und das, obwohl in der Arbeit der Ansicht gefolgt wird, dass „Skepsis einem Denken gegenüber angebracht [ist], das ‚die Welt im Rahmen eindeutig unterschiedener Kategorien‘ zu begreifen versucht“ [Muc07]. Dass trotzdem angestrebt wird, zu klaren Abgrenzungen zu kommen, geschieht aus der Überlegung heraus, dass nur so deutlich gemacht werden kann, wo dies schwierig ist und in nachfolgenden Untersuchungen genauer hingesehen werden muss.

4 Zusammenfassung und Ausblick

Mit der BS (und der Untersuchungsmethode) steht ein interdisziplinär ausgerichtetes Rahmenwerk zur Verfügung, mit dem direkt in die qualitative Analyse spezifischer PP-Aspekte eingestiegen werden kann. Es ist dahingehend konzipiert worden, die Ergebnisse unterschiedlicher PP-Studien leicht konsolidieren zu können. Die umfassende Beschreibung der BS stellt hierzu neben den eigentlichen Konzepterläuterungen/Beispielen auch eine große Anzahl von ggf. gerichtet zu modifizierenden Kodierregeln zur Verfügung [Sal13]. Um das Rahmenwerk einem größeren Kreis als Werkzeug zugänglich zu machen, wurde im Dezember 2013 ein englischsprachiges Buch zur praktischen Handhabung der BS veröffentlicht [SP13]. Ihre Feuerprobe hat die BS bereits hinter sich gebracht, z.B. beim erfolgreichen Einsatz in Untersuchungen zu den Themen Aktivitätsbeeinflussung [Zie12], Rollen (Revidierung des bisher angenommenen Driver/Observer-Rollenmodells) [SZP13] und Wissenstransfer (von Zieris und Prechelt an der Freien Universität Berlin; Veröffentlichung noch ausstehend). Sie bildet somit den Ausgangspunkt, um zu einem kontextsensitiven, prozess- und zielorientierten Verständnis und Einsatz der PP zu kommen. Nebenbei demonstriert die Arbeit, wie auch im Software Engineering gewinnbringend auf die in anderen Disziplinen häufig verwendeten qualitativen Methoden zurückgegriffen werden kann.

Literatur

[AGDS07] Erik Arisholm, Hans Gallis, Tore Dybå und Dag I.K. Sjøberg. Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise. *IEEE Transactions on Software Engineering*, 33(2):65–86, 2007.

[GS05] Barney G. Glaser und Anselm L. Strauss. *Grounded Theory: Strategien qualitativer Forschung*. Hans Huber, Bern, 2.. Auflage, 2005. Original erschienen 1967: The Discovery of Grounded Theory: Strategies for Qualitative Research.

[HDAS09] Jo E. Hannay, Tore Dybå, Erik Arisholm und Dag I.K. Sjøberg. The effectiveness of pair programming: A meta-analysis. *Information and Software Technology*, 51(7):1110–1122, 2009.

[Kel05] Udo Kelle. “Emergence” vs. “Forcing” of Empirical Data? A Crucial Problem of “Grounded Theory” Reconsidered. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, 6(2), 2005.

[LNP04] Angelika Linke, Markus Nussbaumer und Paul R. Portmann. *Studienbuch Linguistik*. Niemeyer, Tübingen, 5. Auflage, 2004.

[ML99] Anneliese von Mayrhauser und Stephen Lang. A Coding Scheme to Support Systematic Analysis of Software Comprehension. *IEEE Transactions on Software Engineering*, 25(4):526–540, 1999.

[Muc07] Petra Muckel. Die Entwicklung von Kategorien mit der Methode der Grounded Theory. *Historical Social Research, Supplement*, (19):211–231, 2007.

[Nos98] John T. Nosek. The case for collaborative programming. *Communications of the ACM*, 41(3):105–108, 1998.

[Ost06] Leon J. Osterweil. Unifying Microprocess and Macroprocess Research. In *Unifying the Software Process Spectrum*, Jgg. 3840 of *Lecture Notes in Computer Science*, Seiten 68–74. Springer Berlin / Heidelberg, 2006.

[Sal13] Stephan Salinger. *Ein Rahmenwerk für die qualitative Analyse der Paarprogrammierung*. Dissertation, Freie Universität Berlin, 2013.

[SC90] Anselm L. Strauss und Juliet Corbin. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*. Sage Publications, Inc., London, 1990.

[Sea69] John Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.

[SP08] Stephan Salinger und Lutz Prechelt. What happens during Pair Programming? In *Proceedings of the 20th Annual Workshop of the Psychology of Programming Interest Group (PPIG '08)*, Lancaster, England, September 2008.

[SP13] Stephan Salinger und Lutz Prechelt. *Understanding Pair Programming: The Base Layer*. Books on Demand, 2013.

[SPP08] Stephan Salinger, Laura Plonka und Lutz Prechelt. A Coding Scheme Development Methodology Using Grounded Theory for Qualitative Analysis of Pair Programming. *Human Technology: An Interdisciplinary Journal on Humans in ICT Environments*, 4:9–25, 2008.

[SZP13] Stephan Salinger, Franz Zieris und Lutz Prechelt. Liberating Pair Programming Research from the Oppressive Driver/Observer Regime. In *Proceedings of the 2013 International Conference on Software Engineering*, ICSE '13, Seiten 1201–1204, 2013.

[Zie12] Franz Zieris. Qualitative Untersuchungen von Beeinflussungen der Aktivität des Partners bei der Paarprogrammierung. Masterarbeit, Freie Universität Berlin, 2012.

Dr. Stephan Salinger wurde 1965 in Berlin geboren. Er studierte Mathematik und Informatik an der Freien Universität Berlin. Nachfolgend war er in verschiedenen Softwareeentwicklungs- und Consultingunternehmen als Softwareentwickler, Projektmanager und Entwicklungsleiter tätig, vornehmlich im Bereich Content Management. Als Mitglied der AG Software Engineering forscht und lehrt er seit 2003 am Lehrstuhl von Prof. Dr. Lutz Prechelt an der Freien Universität Berlin im Bereich der agilen (verteilten) Softwareprozesse. Er promovierte im Jahr 2013.

Trace-basiertes Testen von nebenläufigen reaktiven Systemen*

Roland Schatz
roland@rschatz.org

Abstract: Speicherprogrammierbare Steuerungsprogramme sind Programme, die zur Steuerung von Maschinen eingesetzt werden. SPS-Programme bestehen aus einem Regelungs- und einem reaktiven Steuerungsteil. Während die Analyse von Regelungen ein etabliertes Forschungsgebiet ist, wurde der Analyse des reaktiven Verhaltens von SPS-Programmen bisher wenig Beachtung geschenkt. Gleichzeitig ist aber die Analyse von reaktiven Systemen eine komplexe und anspruchsvolle Aufgabe.

Diese Arbeit präsentiert einen Formalismus zur Spezifikation und Analyse des reaktiven Verhaltens von SPS-Programmen. Aufbauend auf ein Verfahren zur Aufzeichnung und deterministischen Wiedergabe von SPS-Programmen wird eine auf Hybriden Automaten basierende Spezifikationssprache vorgestellt, mit der sowohl Testen als auch die Analyse von Aufzeichnungen ermöglicht wird.

1 Einleitung

Speicherprogrammierbare Steuerungsprogramme (kurz: SPS-Programme) sind Programme zur Steuerung von Maschinen wie zum Beispiel Fertigungsanlagen oder Roboter. Sie interagieren mit der physikalischen Welt, indem sie laufend Sensoreingaben lesen und Aktuatorausgaben schreiben. SPS-Programme sind lang laufende Anwendungen die komplexes zeitliches Verhalten aufweisen.

SPS-Programme bestehen aus zwei Ebenen: Einerseits implementieren sie Regelkreise, die die grundlegenden Aktivitäten des angeschlossenen Geräts regeln. Darüber hinaus gibt es noch eine Entscheidungsebene, die Zustandsübergänge steuert, die einzelnen Regler-Bausteine ein- und ausschaltet und das Gesamtverhalten des Systems koordiniert. Solche Systeme haben komplexes reaktives Verhalten.

Reaktive Systeme sind diskrete Systeme, die laufend mit ihrer Umgebung interagieren und auf externe und interne Stimuli reagieren [HP85]. Im speziellen Fall von SPS-Programmen beschreibt das reaktive Verhalten, wie das Steuerungsprogramm auf bestimmte Eingabewerte von den Sensoren reagiert (z.B. das Abschalten eines Motors an der Endposition). Im Vergleich dazu besteht der Regler nur aus einer mathematischen Formel die die gewünschte Geschwindigkeit des Motors regelt.

Während die Analyse von Reglern ein gut erforschtes Gebiet mit etablierten Anwendungen in der Praxis ist, wurde der Analyse des reaktiven Verhaltens von SPS-Programmen

*Englischer Titel der Dissertation: "Trace-based Testing of Multi-threaded Reactive Systems"

bisher wenig Aufmerksamkeit in Praxis oder Forschung geschenkt. Auf der anderen Seite ist es weitgehend anerkannt, dass Entwurf, Entwicklung und Analyse von reaktiven Systemen eine komplexe Herausforderung darstellt. Ziel dieser Arbeit ist es, Beiträge zum einfacheren Verständnis und zur Analyse des reaktiven Verhaltens von SPS-Programmen zu leisten.

1.1 Vorgehen

Diese Arbeit präsentiert einen Ansatz zur Analyse von SPS-Programmen, zur Verbesserung des Programmverständnisses und zum Aufspüren von Defekten.

In Kapitel 2 wird ein Verfahren zum Aufzeichnen und deterministischen Wiederabspielen von SPS-Programmen präsentiert. Dieses Verfahren bildet die Grundlage der vorliegenden Arbeit und liefert die Daten, auf deren Basis in den weiteren Kapiteln Programmanalysen durchgeführt werden.

In Kapitel 3 wird dann ein Formalismus zur Spezifikation des Verhaltens von reaktiven Systemen vorgestellt. Basierend auf dem Formalismus der *Hybriden Automaten* [Hen96] kann damit gewünschtes oder fehlerhaftes Verhalten des Systems spezifiziert werden. Weiters werden Algorithmen präsentiert, mit denen automatisch Spezifikationsverletzungen gefunden werden können.

Ein besonderes Problem der Analyse von SPS-Systemen ist, dass sich das Verhalten aus einer Interaktion zwischen dem Programmcode und dem damit verbundenen physikalischen System ergibt. Um diese Interaktion zu analysieren, wird in Kapitel 4 ein Verfahren zur Synthese eines abstraktes Modells eines SPS-Systems aufgestellt. Dieses Verhaltensmodell lässt sich als SMT-Formel codieren. So kann dieses Modell mit Model Checkern weiter verarbeitet werden, um zum Beispiel automatische Beweise über das Programmverhalten zu führen.

2 Replay Debugging

In diesem Kapitel wird ein Verfahren zur Aufzeichnung und zum deterministischen Wiederabspielen von SPS-Programmen vorgestellt. Dieses Verfahren wurde als Basis für die Analyse von SPS-Programmen entwickelt. Es liefert die Daten, die den in den weiteren Kapiteln vorgestellten Analyseverfahren zugrunde liegen.

Eine grundlegende Herausforderung in der Analyse von SPS-Programmen ist die Echtzeitfähigkeit. In der Literatur existieren verschiedene Ansätze zum deterministischen Wiederabspielen (z.B. [DKC+02, JO07, KDC05]). Die meisten dieser Ansätze sind entweder gar nicht für Echtzeit-Anwendungen geeignet, oder besitzen einen deutlich höheren Overhead als das hier vorgestellte Verfahren.

Abbildung 1 zeigt den generellen Ansatz des Verfahrens zum Aufzeichnen und Wiederabspielen von SPS-Programmen (siehe auch [PSWM11] und [Wir13]).

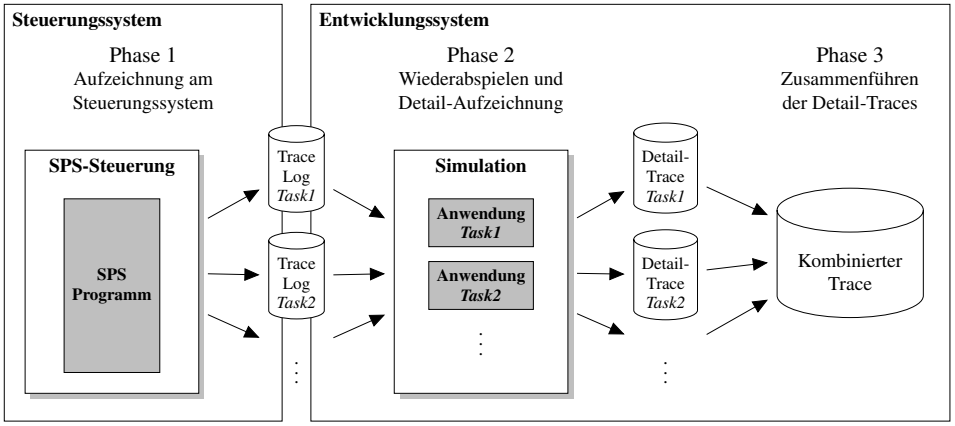


Abbildung 1: Mehrphasiges Verfahren zum deterministischen Wiederabspielen

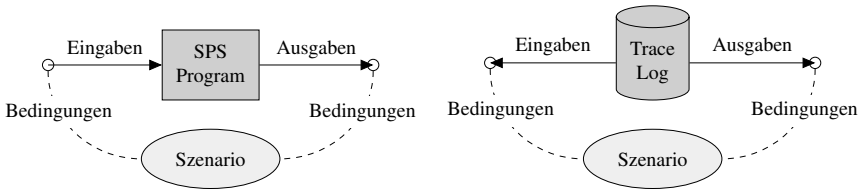
In der ersten Phase wird nur die Information aufgezeichnet, die zum Wiederabspielen eines einzelnen Tasks notwendig ist. Diese Aufzeichnung erfolgt für jeden nebenläufigen Task separat. In der zweiten Phase werden die Aufzeichnungen der ersten Phase verwendet, um jeden Task des SPS-Programms isoliert am Entwicklungssystem abzuspielen. Dabei werden zusätzliche Informationen über den Programmlauf aufgezeichnet. Dann werden diese Informationen zu einem kombinierten Trace-Log zusammengefasst, das auch Informationen über das Scheduling enthält.

Ein Vorteil dieses mehrphasigen Ansatzes ist, dass nur die erste Phase in einer Umgebung mit Echtzeitanforderungen laufen muss. In den späteren Phasen gibt es keine Ressourceneinschränkungen, es können beliebige Daten zur späteren Analyse gesammelt werden. Die folgenden Kapitel beschreiben Verfahren, die die so gesammelten Daten nutzen.

3 Modellbasiertes Testen

In Kapitel 2 wunde ein Verfahren zum deterministischen Wiederabspielen von Programmen vorgestellt. Im Gegensatz zu herkömmlichen Debugging-Techniken hat das deterministische Wiederabspielen den Vorteil, dass das Verhalten der Anwendung exakt reproduziert werden kann. Es löst alle Probleme mit der Reproduzierbarkeit von sporadischen Fehlern, allerdings ergibt sich eine neue Herausforderung: Wenn ein Entwickler die Aufzeichnung einer langlaufenden Anwendung bekommt, ist es sehr schwierig, einen bestimmten Fehler in der Aufzeichnung zu finden.

Dieses Kapitel präsentiert ein Verfahren, das modellbasiertes Testen eines reaktiven Programms erlaubt. Basieren auf Hybriden Automaten wird ein Formalismus entwickelt, der die Spezifikation von Ein-/Ausgabeverhalten eines reaktiven Systems erlaubt. Der Neuheitswert dieses Spezifikationsformalismus ist, dass die selbe Spezifikation sowohl zum Testen von Implementierungen verwendet werden kann, als auch zum Lokalisieren von



(a) Bedingungen auf E/A-Verhalten (b) Analyse von Trace basierend auf Szenario

Abbildung 2: Szenario-Spezifikation

Fehlern in einem aufgezeichneten Ereignisstrom. Der Entwickler kann ein Szenario spezifizieren, und auf Grund dieser Spezifikation eine Implementierung testen. Genauso können basierend auf der selben Spezifikation alle Stellen in einer Aufzeichnung gefunden werden, wo dieses Szenario auftritt.

3.1 Szenario-Spezifikation

Um Tests vollautomatisch ausführen zu können, ist ein Mechanismus nötig, mit dem das erwartete Verhalten spezifiziert werden kann. Bei SPS-Programmen bedeutet das, dass der Entwickler erwartetes oder ungültiges Verhalten des Systems in Bezug auf die Sensoren und Aktuatoren über die Zeit spezifizieren muss.

Durch deterministisches Wiederabspielen entfällt die Notwendigkeit, zur Reproduktion von Fehlern minimale Testfälle zu schreiben, jeder Fehler kann durch Wiederabspielen reproduziert werden. Allerdings muss der Fehler in einer langen Aufzeichnung lokalisiert werden. Um das zu automatisieren ist wieder ein Mechanismus zur Spezifikation des erwarteten Verhaltens notwendig. Ein Algorithmus kann dann die Aufzeichnung nach einer Stelle durchsuchen, an der das spezifizierte Szenario auftritt.

Abbildung 2 zeigt den generellen Ansatz. Szenarios werden als Bedingungen auf dem Ein-/Ausgabeverhalten spezifiziert. Ein Szenario definiert eine Menge von korrekten bzw. fehlerhaften Verhalten. Diese Szenario-Spezifikationen können zum Testen einer Implementierung verwendet werden: Ein Test-Treiber generiert gültige Eingaben aus der Szenario-Spezifikation, und überprüft ob die Ausgaben des Programms die Bedingungen aus der Spezifikation erfüllen (Abbildung 2a).

Die Aufzeichnungsanalyse funktioniert ähnlich: Die Ein- und Ausgabewerte werden aus dem Trace-Log gelesen, und es wird überprüft ob sie den Bedingungen in der Szenario-Spezifikation entsprechen (Abbildung 2b).

Das erwartete Verhalten eines reaktiven Programms und der physikalischen Umgebung kann mit dem Formalismus der *Hybriden Automaten* [Hen96] spezifiziert werden. Ein hybrider Automat modelliert das diskrete Verhalten eines hybriden Systems mit Zustandsübergängen in einem endlichen Automaten. Das stetige Verhalten des Systems wird mit Differentialgleichungen modelliert.

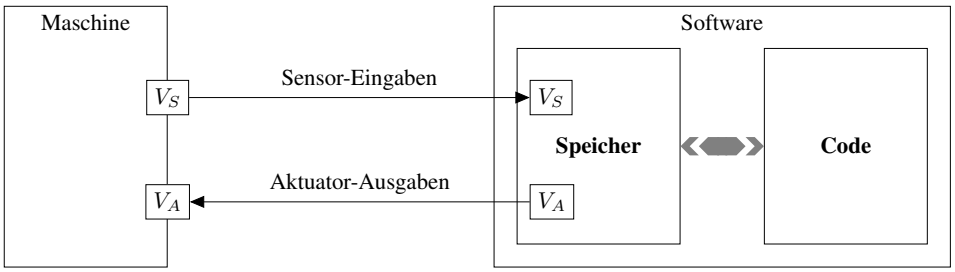


Abbildung 3: SPS-Programmausführung

Definition 1. *Ein Hybrider Automat ist ein Tupel* $H = (V, S, s_0, \text{init}, I, F, T)$.

V ist eine endliche Menge von Variablen. S ist eine endliche Menge von Zuständen, $s_0 \in S$ ist der Anfangszustand. init *ist die Anfangsbelegung der Variablen. $I(s)$ sind Invarianten über die Variablen, und $F(s)$ gibt Differentialgleichungen die die Entwicklung der Variablenwerte beschreiben. T ist die Übergangsrelation.*

Ein wichtiger Vorteil des gemeinsamen Spezifikationsformalismus für Trace-Analyse und Testen ist, dass nachdem eine Szenario-Spezifikation zum Auffinden eines Fehlers verwendet wurde, die selbe Spezifikation mit geringem Aufwand in einen Regressions-Test umgewandelt werden kann, der sicher stellt dass der selbe Fehler nicht später wieder auftritt.

3.2 Testausführung

Mit hybriden Automaten kann ein testbares Szenario spezifiziert werden, mit dem SPS-Programme vollautomatisch getestet werden können. Für eine rigorose Definition der Voraussetzungen an die Spezifikation, sowie Beweise der Korrektheit der Algorithmen, wird auf die Langfassung der Arbeit verwiesen [Sch13].

Abbildung 3 zeigt den typischen Aufbau eines Steuerungssystems: Die gesteuerte Maschine hat Sensoren (V_S) und Aktuatoren (V_A). Diese werden in den Speicher der Software abgebildet. Aus Sicht des Test-Treibers ist die Software-Box undurchsichtig, keine bestimmte Implementierung wird vorausgesetzt.

Die Implementierung eines Steuerungssystems wird im Test-Algorithmus durch einen Moore-Automaten abstrahiert. Der Test-Treiber simuliert die Maschine, indem der hybride Automat eines Testfalls parallel zum Moore-Automaten der Implementierung geschalten wird.

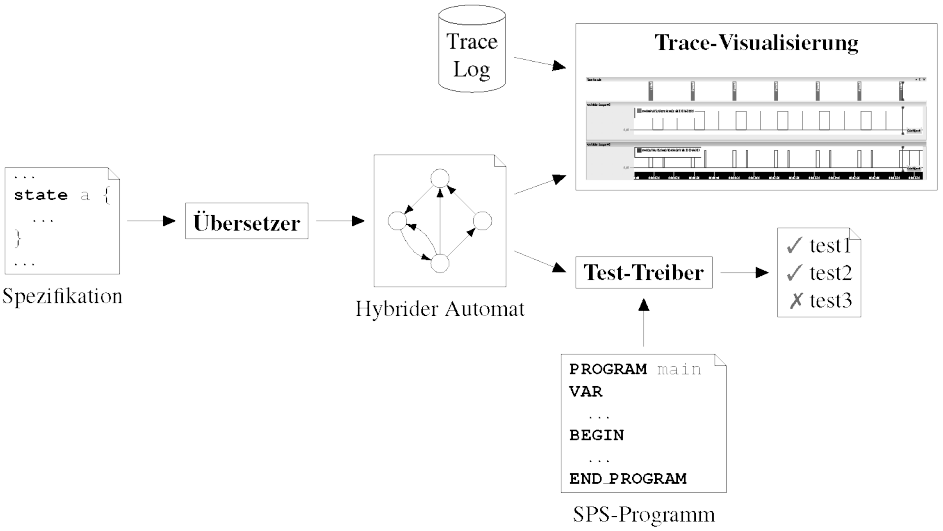


Abbildung 4: Test- und Trace-Analyse-Verfahren

3.3 Trace-Analyse

Mit dem selben Spezifikationsformalismus ist es auch möglich, eine Offline-Analyse eines aufgezeichneten Programmlaufs durchzuführen. Die Aufzeichnung von Programmläufen wurde in Kapitel 2 beschrieben (siehe auch [PSWM11]).

Der Algorithmus zur Trace-Analyse funktioniert ähnlich wie der Test-Algorithmus. Wieder wird der hybride Automat simuliert, anstatt allerdings die Werte von einem laufenden Programm zu lesen, werden alle Variablenwerte aus dem Trace-Log entnommen. So kann der Analysealgorithmus alle Sektionen im Trace-Log finden, die zu einer Szenariospezifikation passen, und auch korrektes von fehlerhaftem Verhalten unterscheiden.

Abbildung 4 gibt einen Überblick über das Test- und Trace-Analyse-Verfahren. Der Entwickler schreibt eine Spezifikation in einer Spezifikationssprache. Diese Spezifikation wird in einen hybriden Automaten übersetzt, der als Eingabe für den Test-Treiber zum modellbasierten Testen, oder zur Analyse eines Trace-Logs verwendet werden kann. Um die Spezifikation von Szenarios zu vereinfachen, wurde auch eine graphische Spezifikationssprache entwickelt.

4 Spezifikationssynthese

In Kapitel 2 wurde ein Verfahren präsentiert, um Programmläufe aufzuzeichnen und wiederabzuspielen. In Kapitel 3 wurde ein Verfahren präsentiert, mit dem in diesen Aufzeichnungen Stellen gesucht werden können, an denen Spezifikationen erfüllt oder verletzt wer-

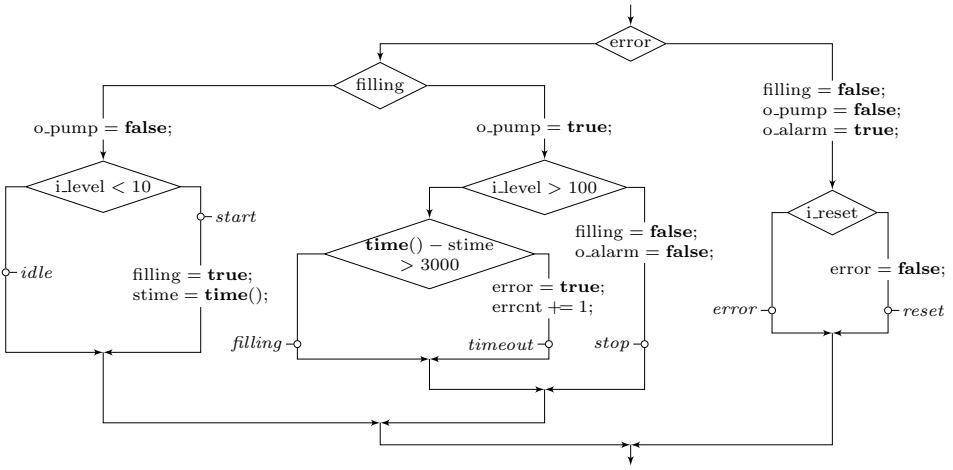


Abbildung 5: Programmcode einer Füllstandsregelung

den. In diesem Kapitel wird nun ein Verfahren vorgestellt, mit dem ein abstraktes Modell des Ein-/Ausgabeverhaltens eines SPS-Programms synthetisiert werden kann. Zuerst wird mittels symbolischer Ausführung des Programmcodes ein Automatenmodell gebaut. Mit Hilfe von Techniken aus dem Übersetzerbau wird daraus ein auf temporaler Logik basiertes Modell überführt. Dieses Modell kann als SMT-Formeln codiert werden, um so Model-Checking-Techniken darauf anzuwenden.

Der Neuheitswert des hier präsentierten Verfahrens ist, dass dieses Modell aus einer Kombination aus statischer Analyse des Quelltexts und dynamischer Information aus Programmaufzeichnungen erstellt. Dadurch werden die Vorteile von statischer und dynamischer Analyse kombiniert.

4.1 Reaktives Verhalten

Ein Steuerungssystem als Ganzes weist typischerweise komplexes Verhalten über die Zeit auf. Während das Verhalten der Software nur vom Quellcode bestimmt ist, ist das Verhalten des physikalischen Systems oft unbekannt und nur schwer vorhersagbar. Das Gesamtverhalten des Systems ist bestimmt von der komplexen Interaktion zwischen der Dynamik des physikalischen Geräts und der Software, und dieses komplexe Zusammenspiel kann nicht durch eine Analyse der Software alleine erfasst werden.

Um dieses Problem zu lösen ist es Zielführend, die formale Analyse des Programmcodes mit Beobachtungen der Reaktionen des physikalischen Geräts zu kombinieren. Die Beobachtungen können mit dem Aufzeichnungs- und Wiederabspiel-Verfahren von Kapitel 2 erzeugt werden.

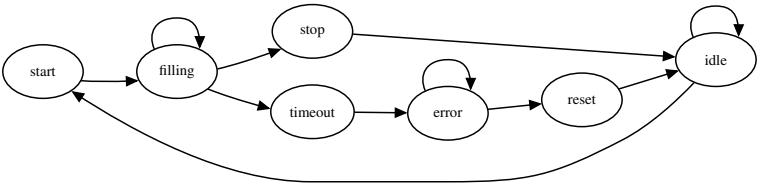


Abbildung 6: Transitionsgraph der Füllstandsregelung

Abbildung 5 zeigt den Programmcode einer einfachen Füllstandsregelung: Eine Pumpe muss so gesteuert werden, dass der Füllstand immer zwischen zwei Grenzwerten gehalten wird.

Von diesem Programm kann ein Transitionsmodell des Kontrollflusses erzeugt werden, indem eine Aufzeichnung eines Programmlaufs wiederabgespielt wird, wobei in jedem Zweig des Programms eine Trace-Anweisung eingefügt wird. Abbildung 6 zeigt den so gewonnenen Transitionsgraphen der Füllstandsregelung.

Mit Hilfe symbolischer Ausführung der einzelnen Kontrollflusspfade des Programms kann dieser Transitionsgraph nun mit Informationen aus dem Quelltext des Programms ergänzt werden, um ein vollständiges Modell des reaktiven Verhaltens des SPS-Programms zu bauen.

Definition 2. *Ein* Reaktives Programm *ist ein Tupel* $(V_i, V_e, C, \rightsquigarrow, P, U)$.

V_i und V_e sind die internen bzw. externen Variablen. C ist eine Menge von Kontrollflusspfaden, $\rightsquigarrow \subset C \times C$ ist die Transitionsrelation.

$P(c)$ ist eine Formel, die die Verzweigungsbedingungen des Kontrollflusspfads $c \in C$ beschreibt, und $U(c, v_i)$ gibt eine Update-Formel für die interne Variable v_i im Kontrollflusspfad c.

4.2 Verhaltensmodell

Das im letzten Abschnitt konstruierte Modell des Programmverhaltens enthält Formeln, die das Verhalten des Programms vollständig beschreiben. In diesem Abschnitt wird dieses Modell benutzt, um Wissen über das extern sichtbare Ein- und Ausgabeverhalten eines SPS-Programms herzuleiten. Dieses Wissen wird als past-time temporale Ausdrücke über die Sensor- und Aktuator-Werte formuliert:

$$\mathcal{T} \quad ::= \quad v_e \mid \text{‘}\mathbf{h}\text{’ ‘(’ } c \text{ ‘)’} \mid \text{‘}\mathbf{Y}\text{’ } \mathcal{T} \mid \text{‘}\mathbf{D}\text{’ } \mathcal{T}$$

$\mathbf{Y}$ ist der „vorher“-Operator, der auf einen Wert im vorhergegangenen Zeitschritt verweist. $\mathbf{h}(c)$ gibt an, dass im aktuellen Zeitschritt der Pfad c ausgeführt wurde.

Zu diesen Operatoren kommt noch $\mathbf{D}$, der „Dominator“-Operator: Er verweist auf Werte, die bei der letzten Ausführung des Dominator-Pfades aufgetreten sind:

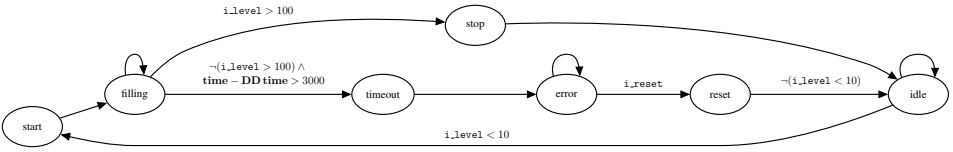


Abbildung 7: Übergangsbedingungen

Definition 3. *Ein Pfad c* dominiert *einen Pfad c' (kurz: $c \gg c'$), wenn jeder gültige Programmlauf $c_0 \rightsquigarrow c_1 \rightsquigarrow \cdots \rightsquigarrow c'$ den Pfad c enthält [CFR+91].*

Der Vorteil dieses Operators ist, dass der Dominator einen Referenzpunkt in der Vergangenheit bietet, der immer existiert, und damit Wissen generiert werden kann, das unbedingt gilt. Mit Hilfe eines aus dem Übersetzerbau stammenden Algorithmus [CFR+91] kann nun das abstrakte Verhaltensmodell in ein temporales Prädikat G umgewandelt werden, das gültige Programmläufe charakterisiert.

Das Prädikat G kann in eine Form gebracht werden, die von einem SMT-Solver gelesen werden kann. Der SMT-Solver kann dann dazu verwendet werden, *Übergangsbedingungen* abzuleiten. Diese Bedingungen beschreiben den Unterschied zwischen zwei Zuständen im Transitionsgraph anhand der Änderung der Eingabewerte.

Abbildung 7 zeigt den Transitionsgraphen des Füllstandsregler-Beispiels mit den automatisch generierten Übergangsbedingungen. Dieses Beispiel illustriert einige Vorteile dieses Verfahrens zum Programmverständnis. Zum Beispiel ist der Übergang $\mathit{filling} \rightsquigarrow \mathit{stop}$ mit `i_level` > 100 beschriftet. *filling* ist also ein Wartezustand, in dem darauf gewartet wird, dass der Füllstand 100 erreicht.

5 Zusammenfassung

Diese Arbeit beschreibt Methoden zum Testen und zur Analyse des reaktiven Verhaltens von SPS-Programmen. Die Hauptbeiträge dieser Arbeit sind: (1) Ein Verfahren wird vorgestellt, das die Aufzeichnung und das deterministische Wiederabspielen von SPS-Programmen ermöglicht. (2) Es wird ein Spezifikationsformalismus vorgestellt, der basierend auf hybriden Automaten das modellbasierte Testen von SPS-Programmen ermöglicht. (3) Es wird ein Verfahren präsentiert, das mit einer Kombination aus Model-Checking-Techniken und Techniken aus dem Übersetzerbau vollautomatisch ein Modell des Ein-/Ausgabeverhaltens von SPS-Programmen in Form von temporaler Logik synthetisiert.

Die in dieser Arbeit vorgestellten Verfahren wurden anhand von Beispielen aus der Literatur und anhand einer Fallstudie mit einem realistischen Steuerungsprogramm unseres Industriepartners evaluiert.

Literatur

[CFR+91] Ron Cytron, Jeanne Ferrante, Barry K. Rosen, Mark N. Wegman und F. Kenneth Zadeck. Efficiently Computing Static Single Assignment Form and the Control Dependence Graph. *ACM Trans. Program. Lang. Syst.*, 13(4), 1991.

[DKC+02] George W. Dunlap, Samuel T. King, Sukru Cinar, Murtaza A. Basrai und Peter M. Chen. ReVirt: Enabling Intrusion Analysis Through Virtual-Machine Logging and Replay. In *OSDI*, 2002.

[Hen96] Thomas A. Henzinger. The Theory of Hybrid Automata. In *LICS*, 1996.

[HP85] David Harel und Amir Pnueli. Logics and models of concurrent systems. Kapitel On the development of reactive systems. Springer-Verlag New York, Inc., New York, NY, USA, 1985.

[JO07] Shrinivas Joshi und Alessandro Orso. SCARPE: A Technique and Tool for Selective Capture and Replay of Program Executions. In *ICSM*, 2007.

[KDC05] Samuel T. King, George W. Dunlap und Peter M. Chen. Debugging Operating Systems with Time-Traveling Virtual Machines. In *USENIX Annual Technical Conference, General Track*, 2005.

[PSWM11] Herbert Prähofer, Roland Schatz, Christian Wirth und Hanspeter Mössenböck. A Comprehensive Solution for Deterministic Replay Debugging of SoftPLC Applications. *IEEE Trans. Industrial Informatics*, 7(4), 2011.

[Sch13] Roland Schatz. *Trace-based Testing of Multi-threaded Reactive Systems*. Dissertation, Johannes Kepler University, Linz, Austria, 7 2013.

[Wir13] Christian Wirth. *Dynamic Analysis of Multi-threaded Real-time PLC Applications using Record and Replay*. Dissertation, Johannes Kepler University, Linz, Austria, 6 2013.

Roland Schatz wurde geboren am 2. Oktober 1982. Nach Volks- und Hauptschule in Naarn absolvierte er die HTL für EDV und Organisation in Leonding, wo er 2002 die Matura mit Auszeichnung bestand. Anschließend absolvierte er in Linz ein Bachelor- und Diplomstudium der Informatik. In dieser Zeit nahm er an internationalen Programmierwettbewerben teil, und erreichte 2007 im Welt-Finale des ACM International Collegiate Programming Contest in Tokio den 26. Platz. Ab 2006 arbeitete er als Forscher im Christian Doppler Labor für Automated Software Engineering, wo er auch 2013 seine Dissertation im Projekt „Testing and Diagnosis of Programmable Logic Controller Programs“ verfasste. Am 2. Juni 2014 wurde im der Doktortitel unter den Auspizien des Bundespräsidenten verliehen. Seit 2013 arbeitet er als Forscher bei Oracle Labs Austria im Projekt „Graal – a next generation compiler for the HotSpot VM“.

Quantifizierung und Schutz von Location Privacy*

Reza Shokri
reza.shokri@inf.ethz.ch

Abstract: Diese Arbeit beschäftigt sich mit dem zeitgemäßen Bedarf nach Schutz der Privatsphäre in der Zeit von "Big-Data". Hierfür identifizieren wir die zwei folgenden, fundamentalen Probleme für computational privacy: (i) Konsistente Quantifizierung von Privatsphäre in verschiedenen System und (ii) optimaler Schutz der Privatsphäre durch Obfuscation-Mechanismen. Wir beschreiben das Problem der Quantifizierung der Privatsphäre als die Berechnung des Schätzfehlers in einem Bayes'schen Inferenzproblem, in dem ein Angreifer seine Beobachtungen, sein Hintergrundwissen und Seitenkanalinformationen kombiniert um eine Schätzung der sensiblen Informationen des Nutzers zu bekommen. Dies gibt uns die Möglichkeit Privatsphäre von Nutzern in verschiedenen Systemen zu evaluieren und die Effektivität verschiedener Datenschutzmechanismen zu vergleichen. Mit Hilfe unseres Quantifizierungsframeworks können wir quantitativ zeigen, dass verschiedene populäre Metriken wie k-Anonymität und Entropie die Privatsphäre nicht korrekt messen. Wir formulieren das offene Problem der Optimierung der Privatsphäre eines Nutzers unter Berücksichtigung der Nützlichkeit der Daten als ein interaktives Optimierungsproblem, in dem sowohl Nutzer als auch Angreifer ihre jeweiligen Ziele erreichen wollen, die sich naturgemäß widersprechen. Wir lösen diese im Widerspruch stehenden Optimierungskriterien, indem wir unser Problem als ein Bayes'sches Stackelberg-Spiel modellieren. Wir zeigen außerdem, dass dieses Spiel durch lineare Programmierung gelöst werden kann. Zudem wenden wir unsere Methodologie auf die Quantifizierung und den Schutz von location privacy (der Privatsphäre im Bezug auf Ortsdaten) in ortsbasierten Diensten an. Wir stellen außerdem ein Open-Source-Tool zur Verfügung – Location-Privacy and Mobility Meter (LPM). Dieses erlaubt Forschern das Erlernen und Analysieren von menschlichen Mobilitätsmodellen sowie die Evaluierung und den Vergleich verschiedener location-privacy-Schutzmechanismen.

1 Einleitung

Die Entwicklungen in Informations- und Kommunikationstechnologien sind tiefgreifend und verändern potentiell das Leben der Menschen. Die meisten Menschen besitzen heutzutage Smartphones mit hoher Rechenleistung und vielfältigen Kommunikationsfähigkeiten. Diese Geräte können effizient verschiedene Applikationen parallel ausführen, einen nicht zu vernachlässigenden Teil an (persönlichen) Benutzerdaten speichern, verschiedene Sensoren und Aktuatoren verarbeiten und über verschiedene kabellose Medien kommunizieren. Außerdem sind sie üblicherweise mit hochgenauen Lokalisierungsfähigkeiten ausgestattet. So enthalten sie z.B. GPS-Empfängern oder können durch Triangulation von Basisstationen oder WLAN-Zugriffspunkten ihr Position bestimmen. Mobile Anwendungen nutzen diese Fähigkeiten um ihren Nutzern ortsbasierte Dienste zur Verfügung zu stellen.

*Englischer Titel der Dissertation: "Quantifying and Protecting Location Privacy"

Diese stets zunehmende Nutzung persönlicher Kommunikationsgeräte und mobiler Anwendungen haben hohe Kosten in Bezug auf die Privatsphäre der Nutzer, auch wenn sie zum Komfort und der Bequemlichkeit derselben beitragen. Die Interaktion mit ortsbasierten Diensten (location based services (LBS)) produziert eine beinahe unlöschbare Spur der Aufenthalts- und Bewegungsdaten des Nutzers. Hinzu kommt noch die kontextuelle Information, die mit diesen Spuren verbunden werden kann, wie persönliche Angewohnheiten, Interessen, Aktivitäten und Beziehungen. Konsequenterweise gibt die Offenlegung dieser privaten Informationen gegenüber Dritten (z.b. Diensteanbietern) diesen eine gewisse Macht über die Nutzer und erlaubt verschiedenstartigen Missbrauch ihrer persönlichen Daten.

Individuen haben das Recht und sollten die Möglichkeiten haben die Menge ihrer privaten (ortsbasierten) Informationen zu kontrollieren die Dritten zugänglich gemacht werden. Im Kontext von ortsbasierten Diensten werden in der gängigen Literatur diverse Mechanismen hierzu vorgeschlagen, bspw. Obfuscation und Benutzeranonymisierung. Allerdings modellieren existierende *Designmethoden* für Datenschutz von Ortsdaten nicht die (Privatsphäre- und Servicequalitäts-) Ansprüche von Nutzer unter Berücksichtigung des Wissens und der Ziele eines Angreifers. Schutzmechanismen werden daher ad-hoc designt und unabhängig vom Angreifermodell. Von daher gibt es immer eine Differenz zwischen Ziel und Ergebnis dieser Schutzmechanismen. Zudem bleibt die *Evaluierung* von Datenschutzmechanismen und ihr Vergleich problematisch, da systematische Methoden zur Quantifizierung fehlen. Im Besonderen gibt es Lücken bei der Modellierung des Angreifers mit dem Resultat möglicher Fehleinschätzungen der zu erwartenden Privatsphäre des Nutzers. Das Fehlen eines generischen analytischen Frameworks zur Spezifizierung von Schutzmechanismen und zur Evaluierung von location privacy ist daher offensichtlich. Die Abwesenheit eines solchen Frameworks macht daher das Design von effektiven Schutzmechanismen und den objektiven Vergleich zwischen ihnen unmöglich.

Diese Arbeit benennt diese Probleme und stellt Lösungen zur Verfügung für eine systematische Quantifizierung von location privacy und ihrer Schutzmechanismen. Hierzu konstruieren wir ein analytisches Framework für location privacy. Wir formalisieren das Mobilitätsmodell von Nutzern, ihre Zugriffsmuster auf ortsbasierte Dienste und ihre Privatsphäre- und Dienstqualitätsanforderungen. Wir modellieren außerdem verschiedene location privacy Schutzmechanismen als probabilistische Funktionen, die die (Orts- und Identitäts-)Daten von Nutzer verschleiern bevor sie mit den relevanten Diensten geteilt werden. Um die Privatsphäre eines Nutzers quantifizieren zu können, definieren wir Inferenzmechanismen, die die Informationsweitergabe an Dritte messen. Diese kombinieren verschiedene Teilinformationen von Nutzern und schätzen (durch probabilistische Methoden) die privaten Informationen einer Nutzerin (bspw. ihren Aufenthaltsort zu einer gegebenen Zeit). Daher schlagen wir den zu erwartenden Schätzfehler des Angreifers als die (zu diskutierende) korrekte *Metrik* für die location privacy vor.

In unserem Inferenz-Framework formalisieren wir das *vorherige Wissen* des Angreifers über Nutzer, seine *Beobachtungen* (bei Zugriffen von Nutzern auf die LBS) und seine *Inferenzziele* (bspw. die Wiedererkennung oder Lokalisierung von Nutzern). Wie nehmen dabei an, dass der Angreifer ein Mobilitätsprofil für jeden Nutzer erstellt, das er für seine Inferenzangriffe nutzt. Wir nutzen hierzu statistische Methoden um diese Profile aus partiel-

len Traces zu erstellen. Weiterhin modellieren wir die *Inferenzangriffe* als Schätzungen der tatsächlichen Aufenthaltsorte von Nutzern unter Berücksichtigung ihrer Profile und ihrer LBS-Zugriffe (aus der Sicht des Angreifers). Wir verwenden größtenteils Bayes'sche Netze zur Schätzung. Genauer gesagt nutzen wir bekannte Inferenzalgorithmen für Hidden-Markov-Modelle um Deanonymisierungs-, Lokalisierungs- und Trackingattacken zu entwerfen. Um mehr Ziele eines Angreifers abzudecken präsentieren wir außerdem einen Algorithmus für generische Ortsinferenzangriffe, basierend Monte-Carlo-Algorithmen für Markov-Ketten.

Des Weiteren stellen wir eine Software zur Verfügung: *Location-Privacy and Mobility Meter* (LPM).[1] Sie wurde auf der Grundlage unseres formalen Frameworks zur Evaluation der Effektivität verschiedener location-privacy-Schutzmechanismen und Quantifizierungsmechanismen für die location privacy von Nutzern entwickelt. Beispielsweise nutzen wir LPM um die Wirksamkeit von existierenden Obfuscationmechanismen für Ortsdaten und Anonymisierungsmechanismen mit echten Ortstraces zu testen. Wir zeigen, dass die Privatsphäre des Nutzers, so wie sie durch die populären Metriken k-Anonymität und Entropie bestimmt wird, nicht mit dem Erfolg eines Angreifers (die privaten Informationen des Nutzers zu bekommen) korreliert. Diese Metriken sind also nicht angemessen als Metriken bzw. Indikatoren für die Privatsphäre. Unsere Ergebnisse bestätigen außerdem, dass Anonymisierung allein ein schlechter location-privacy-Schutzmechanismus ist. Weiterhin zeigen unsere Ergebnisse, dass die Widerstandsfähigkeit eines Schutzmechanismus sich stark verändert abhängig von den verschiedenen Inferenzangriffen. Es ist daher notwendig, dass Datenschutzmechanismen vor dem Hintergrund konkreter Angriffsszenarien designt werden.

Auf der Grundlage dieser Daten designen wir optimale Obfuscation-Techniken für Ortsdaten die zugeschnitten sind auf Ortsbestimmungsangriffe. Ein Nutzer braucht einen *Datenschutzmechanismus* der seine location privacy maximiert. Dies steht im Widerspruch zu den Zielen eines Angreifers der einen Inferenzangriff mit minimalem Schätzfehler designt. Wir modellieren daher eine spieltheoretische Methodologie, die die verschiedenen Ziele von Nutzer und Angreifer gleichzeitig darstellt. Genauer gesagt modellieren wir das Problem als Stackelberg-Spiel und lösen es mit Hilfe von linearer Programmierung. Im Optimierungsproblem schränken Nutzer den Datenschutzmechanismus ein in Bezug auf ihre Anforderungen an die Servicequalität. Dies gibt uns die Möglichkeit den optimalen Punkt auf der Tradeoff-Kurve zwischen Privatsphäre und Servicequalität zu finden, der gleichzeitig die Bedürfnisse des Nutzers nach Privatsphäre und Servicequalität berücksichtigt. Unsere Ergebnisse deuten an, dass die Erwartung von Inferenzangriffen und die Berücksichtigung des Wissens des Angreifers zu Designs von effektiveren Schutzmechanismen führen.

Diese Arbeit ist ein Schritt hin zu systematischer Modellierung, Analyse und Design von location-privacy-verbessernden Technologien. Wir sind überzeugt, dass unsere analytische Herangehensweise genutzt werden kann um die Privatsphäre auch in Szenarien und Bereichen, die nicht Teil dieser Arbeit sind, zu quantifizieren und schützen.

[1] Die Software und Dokumentation sind verfügbar unter: `http://icapeople.epfl.ch/rshokri/lpm`. Der Sourcecode wurde bereits auf über 60 verschiedenen Netzwerken herunter geladen.

2 Quantifizierung der Location Privacy

Unser Ziel ist die Verbesserung der Quantifizierung der Leistung von location privacy erhaltenden Mechanismen (location privacy preserving mechanisms (LPPM)). Dies ist ein wichtiges Thema, da (i) Menschen notorisch schlechte Schätzer für Risiken im Allgemeinen und Risiken der Privatsphäre im Besonderen sind, (ii) es der einzige aussagekräftige Vergleich zwischen verschiedenen LPPM ist und (iii) die aktuelle Forschungsliteratur in diesem Bereich noch nicht weit genug fortgeschritten ist.

In bestimmten Bereichen wurden bereits Beiträge geliefert zur Quantifizierung von Privatsphäre, bspw. für Datenbanken [Dwo06], für Anonymitätsprotokolle [CPP08] oder für Anonymisierungsnetzwerke [SD02, Tro11]. Im Bereich der location privacy allerdings, trotz der Beiträge vieler anderer Disziplinen (wie Datenbanken, mobile Netzwerke und Ubiquitous Computing [Kru09]) zum Schutz der location privacy, ist das Fehlen eines vereinheitlichen und generischen Frameworks für die Spezifizierung von Schutzmechanismen und für die Evaluierung der location privacy offensichtlich. Dies hat zu einer Defokussierung der Beiträge und dadurch zu einer gewissen Verwirrung darüber geführt, welche Mechanismen effizienter sind. Im Allgemeinen wird der Angreifer oft nicht angemessen modelliert und formalisiert und ein gutes Modell für dessen möglichen Inferenzangriffe fehlt. Dies kann zu falschen Schätzungen der location privacy mobiler Nutzer führen. In den wenigen Fällen in denen die Verletzbarkeit der Privatsphäre diskutiert wird, wird die location privacy als die Anonymität der Nutzer evaluiert [ZB11]. Daher wird die Menge an weitergegebenen Ortsinformationen durch das Teilen von Ortsdaten nicht korrekt quantifiziert. Weiterhin können die vorgeschlagenen Evaluationen nicht auf generische LPPM verallgemeinert werden.

Wir schlagen daher ein generisches, theoretisches Framework zur Quantifizierung der location privacy von LBS-Nutzern vor. Unsere Arbeit enthält folgende Beiträge:

- Wir stellen ein generisches Model zur Verfügung, dass die möglichen Angriffe auf private Ortsinformationen von mobilen Nutzern abbildet. Im Besonderen definieren wir rigoros *Deanonymisierung* (die Wiedererkennung von pseudonymisierten location traces), *Lokalisierung* (das Auffinden des Aufenthaltsorts eines Nutzers zu einer spezifischen Zeit) und *Tracking*-Angriffe auf anonymisierte Traces als statistisches Inferenzproblem. Wir konstruieren außerdem einen generischen Angriff, der noch mehr ortsbezogene Informationen über solche Nutzer aufzeigt, die ihre (pseudonymisierten und verschleierten) Aufenthaltsorte mit LBS teilen.

- Wir benutzen bekannte statistische Methoden zur Evaluierung der Leistungsfähigkeit solcher Inferenzangriffe. Genauer gesagt nutzen wir Bayes'sche Inferenztechniken – insbesondere solche, die für verstecke Markov-Modelle geeignet sind. Wir bestimmen die privaten Informationen eines Nutzer auch unter den Umständen, dass ihre Zugriffe auf das LBS nur pseudonymisiert und verschleiert erfolgen. Wir formalisieren den Erfolg des Angreifers und klarifizieren, erklären und rechtfertigen die korrekte Metrik zur Quantifizierung der location privacy: Der *erwartete Schätzfehlers (expected estimation error)* des Angreifers.

- Wir stellen ein Programm zur Verfügung: Das *Location-Privacy and Mobility Meter*

(LPM) wurde auf der Basis unseres formalen Frameworks entwickelt und ist für die Evaluierung der Effektivität von verschiedenen location privacy erhaltenden Mechanismen designt worden. Wir benutzen LPM zur Evaluierung verschiedener LPPM und zum Vergleich ihrer Effektivität under vergleichbaren Bedingungen.

- Wir zeigen die Unangemessenheit einiger existierender Metriken, namentlich Entropie und k-Anonymität, zur Quantifizierung von location privacy.

Ein Deanonymisierungsangriff findet die wahrscheinlichste Zuordnung von Benutzern $\{u_1, u_2, ..., u_N\}$ zu den beobachteten (pseudonymisierten und verschleierten) Traces $\{\mathbf{o}_{\tilde{u}_1}, \mathbf{o}_{\tilde{u}_2}, ..., \mathbf{o}_{\tilde{u}_N}\}$. Hierbei ist zu beachten, dass es nicht korrekt ist einfach jedem Nutzer den Trace zuzuweisen, den er am wahrscheinlichsten erschaffen hat, da hierbei mehrere Nutzer dem gleichen Trace zugewiesen werden könnten. Die wahrscheinlichste Zuweisung ist diejenige, die *multivariate* Wahrscheinlichkeit der Zuweisung aller Nutzer zu den beobachteten Traces (Pseudonymen) maximiert. Mathematisch betrachtet:

$$\boldsymbol{\sigma}^* = \arg\max_{\sigma} \mathbb{P}\mathrm{r}\{\boldsymbol{\Sigma} = \boldsymbol{\sigma} \mid \boldsymbol{O} = \mathbf{o}\}, \tag{1}$$

wobei σ eine Zuweisung von Nutzern zu Pseudonymen ist und $\mathbf{o}$ die Folge von beobachteten, verschleierten Aufenthaltsorten. Wir nutzen Bayes'sche Inferenz zur Deanonymisierung. Um die wahrscheinlichste Zuweisung $\boldsymbol{\sigma}^*$ zu bekommen, müssen wir die folgende Wahrscheinlichkeit maximieren:

$$\mathbb{P}\mathrm{r}\{\boldsymbol{\Sigma} = \boldsymbol{\sigma} \mid \boldsymbol{O} = \mathbf{o}\} = \frac{\mathbb{P}\mathrm{r}\{\mathbf{o} \mid \boldsymbol{\sigma}\} \cdot \mathbb{P}\mathrm{r}\{\boldsymbol{\sigma}\}}{\mathbb{P}\mathrm{r}\{\mathbf{o}\}} = \prod_{\tilde{u}} \mathbb{P}\mathrm{r}\{\mathbf{o}_{\tilde{u}} \mid \boldsymbol{\sigma}\} \cdot \underbrace{\frac{\mathbb{P}\mathrm{r}\{\boldsymbol{\sigma}\} \equiv \frac{1}{N!}}{\mathbb{P}\mathrm{r}\{\mathbf{o}\}}}_{\text{constant}}. \tag{2}$$

Daraus folgt $\boldsymbol{\sigma}^* = \arg\max_{\boldsymbol{\sigma}} \prod_{\tilde{u}} \mathbb{P}\mathrm{r}\{\mathbf{o}_{\tilde{u}} \mid \boldsymbol{\sigma}\}$.

Wir berechnen die Wahrscheinlichkeit $\mathbb{P}\mathrm{r}\{\mathbf{o}_{\tilde{u}} \mid \boldsymbol{\sigma}\}$ für jedes Paar beobachteter Traces $\mathbf{o}_{\tilde{u}}$ und Nutzer u mit dem *Forward-Backward*-Algorithmus. Dies ist ein Inferenzalgorithmus für Hidden Markov Modelle. Er berechnet die Posterior-Mode-Schätzungen für alle unbekannten Statusvariablen einer Folge von Beobachtungen. Nach der Berechnung der Wahrscheinlichkeit aller Paare von beobachteten Traces und Nutzern im bipartiten Graphen vervollständigen wir den Deanonymisierungsangriff durch die Lösung des *Zuordnungsproblems* des maximalen Gewichts im Graph (Maximum Weight Assignment (MWA)) mit Hilfe der ungarischen Methode.

Unter Berücksichtigung der Ergebnissen des Deanonymisierungsangriffs ist die *Lokalisierung* der Angriff in welchem der Angreifer die Wahrscheinlichkeitsverteilung über Regionen berechnet, in denen ein bestimmter Nutzer sich zu einer bestimmten Zeit aufgehalten hat. Diese Wahrscheinlichkeit wird aus den beobachteten Traces von Nutzern berechnet. Mathematisch ausgedrückt berechnet der Angreifer $\mathbb{P}\mathrm{r}\{\boldsymbol{A}_u^t = r \mid \mathbf{o}\}$ für Nutzer u zur Zeit t für alle Regionen $r \in \mathcal{R}$, wobei $\boldsymbol{A}_u^t$ der tatsächliche Aufenthaltsort von Nutzer u zur Zeit t ist.

Gegeben die wahrscheinlichste Zuordnung von Nutzern zu den beobachteten Traces $\boldsymbol{\sigma}^*$, die wir in (1) durch den Deanonymisierungsangriff berechnet haben, berechnen wir nun

die Wahrscheinlichkeitsverteilung des Aufenthaltsorts eines Nutzers zu einer gegebenen Zeit als $\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}, \boldsymbol{\Sigma} = \boldsymbol{\sigma}^*\}$, was wiederum als $\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\}$ berechnet werden kann. Wir benutzen dann Bayes zur Berechnung der Ortswahrscheinlichkeiten.

$$\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\} = \frac{\Pr\{\boldsymbol{A}_u^t = r, \mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}\}}{\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}\}}, \tag{3}$$

wobei die Wahrscheinlichkeit auf der rechten Seite leicht mit Hilfe des Forward-Backward Algorithmus berechnet werden kann..

Die forward- α und backward-Variablen β seien

$$\alpha_r^{u,\tilde{u}}(t) = \Pr\{\boldsymbol{A}_u^t = r, \mathbf{o}_{\tilde{u}}^{1:t} \,|\, \sigma(u) = \tilde{u}\}, \beta_r^{u,\tilde{u}}(t) = \Pr\{\mathbf{o}_{\tilde{u}}^{t+1:T} \,|\, \boldsymbol{A}_u^t = r, \sigma(u) = \tilde{u}\}$$

Die Variable $\alpha_r^{u,\tilde{u}}(t)$ enthält die Beobachtungen bis zum Zeitpunkt t und Region r zum Zeitpunkt t, und $\beta_r^{u,\tilde{u}}(t)$ steht für den Rest des beobachteten Traces mit gegebenem Aufenthaltsort des Nutzers zur Zeit t als Region r. Diese beiden Variablen können in polynomieller Zeit durch einen iterativen Algorithmus berechnet werden

$$\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\} = \frac{\alpha_r^{u,\tilde{u}}(t) \cdot \beta_r^{u,\tilde{u}}(t)}{\Pr\{\mathbf{o}_{\tilde{u}} \,|\, \sigma^*(u) = \tilde{u}\}}. \tag{4}$$

Schlussendlich können wir mit $\Pr\{\boldsymbol{A}_u^t = r \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\}$ die location privacy eines Nutzers u zur Zeit t quantifizieren als den erwarteten Schätzfehler des Angreifers:

$$\sum_{\hat{r}} \Pr\{\boldsymbol{A}_u^t = \hat{r} \,|\, \mathbf{o}_{\tilde{u}}, \sigma^*(u) = \tilde{u}\} \cdot d_p(\hat{r}, r) \tag{5}$$

wobei $d_p(\hat{r}, r)$ der Abstand zwischen dem tatsächlichen Aufenthaltsort eines Nutzers r und der Schätzung des Angreifers $\hat{r}$ ist.

3 Schutz der Location Privacy

Die Aufdeckung der Aufenthaltsorte von Nutzern gegenüber LBS gibt diesen Informationen über Aspekte des Privatlebens, die im ersten Moment nicht offensichtlich sind, aber die aus den aufgedeckten Ortsinformationen abgeleitet werden können [STLBH11, STD+11, FSH11, GP09]. Ein großer Teil der Forschung hat sich damit beschäftigt, location privacy Schutzmechanismen (LPPMs) zu entwickeln, die es Nutzern erlauben, LBS zu nutzen und dabei die Menge an offenbarten, empfindlichen Informationen zu limitieren [FSH09, MRC09]. Diese Schutzmechanismen basieren auf dem Verbergen oder Stören der wahren Aufenthaltsorte eines Nutzers, oder sogar in dem Senden von falschen Aufenthaltsorten an die LBS, um die Unsicherheit des Angreifers über den tatsächlichen Aufenthaltsorts eines Nutzers zu erhöhen. Allerdings ignoriert die Evaluation dieser Methoden normalerweise die Tatsache, dass der Angreifer bereits Informationen über die Zugriffsmuster des Nutzers auf das LBS sowie über die Algorithmen haben kann, die im LPPM implementiert sind. Wie wir im Quantifizierungskapitel zeigen, erlauben derartige Informationen dem Angreifer die Reduzierung seiner Unsicherheit in Bezug auf den tatsächlichen

Aufenthaltsort eines Nutzers. Daher überschätzen bisherige Evaluationen oftmals die location privacy, die durch diese Schutzsysteme gewährleistet wird.

Wir legen unseren Fokus auf eine große Bandbreite an LBS und standortteilende Dienste, in denen der Nutzer seinen Aufenthaltsort *sporadisch* mitteilt, z.b. mittels Applikationen zum Check-In und Location-Tagging oder mittels Applikationen um nahegelegene Points-Of-Interests, lokale Veranstaltungen oder Freunde zu finden. Wir gehen davon aus, dass ein Angreifer daran interessiert ist den Aufenthaltsort eines Nutzers zum Zeitpunkt des LBS-Querys aufzudecken (d.h. ein Angreifer versucht einen Lokalisierungsangriff). Des Weiteren betrachten wir besonders *Benutzer-zentrische* LPPM, in denen die Entscheidungen zum Schutz der Privatsphäre (d.h. Verstecken, Stören oder Fälschen des Aufenthaltsorts) lokal von jedem Nutzer getroffen werden. Diese LPPM sind leicht in mobilen Geräten zu implementieren, die zum Zugriff auf LBS genutzt werden. Wir erwähnen hierbei besonders, dass die Prinzipien unserer Schutzmechanismen auch auf LBS anwendbar sind, in denen Nutzer konstant (statt nur sporadisch) ihren Aufenthaltsort mitteilen und in denen das Ziel eines Angreifers ist, Nutzern dauerhaft durch Raum und Zeit zu folgen.

Hierfür präsentieren wir ein analytisches Framework, das es Systemdesignern erlaubt die optimalen LPPM gegen einen strategisch agierenden Angreifer zu finden, der dadurch, dass er die Zugriffsmusters jedes Nutzers und die zugrundeliegenden Obfuscations Algorithmen kennt, in der Lage ist, einen optimalen Angriff zu ihrer Lokalisierung durchzuführen. Die Herausforderung hierbei ist es einen optimalen Schutzmechanismus zu designen, wenn die Inferenzattacke, je nach designtem Mechanismus, dem Designer unbekannt ist. Anstatt Annahmen über den Inferenzalgorithmus eines Angreifers zu treffen (d.h. dessen Möglichkeiten einzuschränken), finden wir den optimalen Angriff zeitgleich zum Schutzmechanismus. Außerdem begrenzt unsere Methodik den Suchraum auf die LPPM, die Aufenthaltsorte in einer solchen Art verschleiern, dass die Qualität des LBS nicht unter eine gewisse, vom Nutzer festgelegte Grenze sinkt. D.h. die vom Nutzer benötigte Servicequalität ist garantiert. Wir nehmen außerdem an, dass der Angreifer von dieser nutzerspezifizierten Beschränkung weiß. Nach unserem besten Wissen ist dies das erste analytische Framework, das es Entwicklern erlaubt methodisch das Wissen eines Angreifers in das Design von nutzerzentrischen Privatssphäreschutzmechanismen zu integrieren.

Wir formalisieren das Problem des Auffindens eines optimalen LPPM mit der Antizipation eines optimalen Interferenzangriffs als eine Instanz eines Nullsummen-Stackelberg-Spiels. In diesem Spiel interagieren ein Stackelbergführer und ein Stackelbergfolger strategisch, wobei der Gewinn des einen der Verlust des anderen ist. Der Stackelbergführer entscheidet seine Strategie im Wissen darum, dass er von seinem Stackelbergfolger beobachtet wird, der seine Entscheidungen auf seinen Beobachtungen basierend optimieren wird. In unserem Szenario ist der Nutzer der Stackelbergführer und der Angreifer der Stackelbergfolger. Das Spiel modelliert also, dass der Angreifer die Entscheidungen des Nutzers zu Schutzmechanismen kennt und dieses Wissen zur Verbesserung der Effektivität seines Angriffs nutzen wird. Wir erweitern die klassische Formulierung des Stackelberg-Spiels mit einer weiteren Einschränkung um die Servicequalität für den Nutzer zu sichern. Dies erlaubt uns den optimalen Punkt auf der Tradeoff-Kurve zwischen Privatsphäre und Servicequalität zu finden, der sowohl die Privatsphäreansprüche als auch die Servicequalitätsansprüche des

Nutzers erfüllt. Das Problem der Optimierung der location privacy ist folgender:

Gegeben sei

1. ein maximaler Verlust in Servicequalität $Q_{loss}^{\max}$ gegeben durch den Nutzer als Grenzwert für $Q_{loss}(.)$, berechnet durch die Qualitätsfunktion $d_q(.)$, und
2. das a priori-Wissen des Angreifers über des Profil des Nutzers $\boldsymbol{\psi}$,

dann ist das Problem des Auffindens der LPPM-Obfuscation-Funktion $f(.)$, die die location privacy des Nutzers maximiert (definiert als den Inferenz- bzw. Schätzfehler des Angreifers). Die Lösung muss berücksichtigen dass ein Angreifer

1. die Ausgaben $\tilde{r}$ des LPPM sieht, und
2. den internen Algorithmus $f(.)$ des LPPM kennt.

Der Angreifer implementiert also einen *optimalen* Angriff $h(.)$, der den wahren Aufenthaltsort des Nutzers bestimmt mit der geringsten Störung, bestimmt durch $d_p(.)$.

Das Problem der Bestimmung eines LPPM, das optimale location privacy unter dem Wissen des Angreifers gibt, ist eine Instanz des Nullsummen-Stackelberg-Spiels. In diesem spielt der Stackelbergführer, in unserem Fall der Nutzer, zuerst, in dem er ein LPPM wählt und sich zu diesem durch die Anwendung auf seinen tatsächlichen Standort bekennt. Der Stackelbergfolger, in unserem Fall der Angreifer, spielt danach in dem er den Standort des Nutzers schätzt unter Zuhilfenahme des Wissens um das LPPM, welches der Nutzer ausgewählt hat. Es ist ein Bayes-Spiel, da der Angreifer nur unvollständige Informationen über den tatsächlichen Aufenthaltsort des Nutzers hat, und daher mit seiner Annahme über ebendiesen spielt. Es ist also eine Instanz eines Nullsummenspiels, da der Gewinn (oder Verlust) an Nutzen des Angreifers sich genau ausgleicht mit dem Verlust (oder Gewinn) des Nutzers. Wir können nun also das Spiel an unser Problem anpassen:

Schritt 0 Die "Natur" wählt einen Standort $r \in \mathcal{R}$ für den Nutzer, um auf das LBS zuzugreifen, unter Berücksichtigung der Wahrscheinlichkeitsverteilung $\boldsymbol{\psi}$. D.h. Standort r wird mit Wahrscheinlichkeit $\psi(r)$ ausgewählt.

Schritt 1 Bei gegebenem r wendet der Nutzer das LPPM $f(\tilde{r}|r)$ an, um den Pseudostandort $\tilde{r} \in \widetilde{\mathcal{R}}$ zu bekommen, unter der Voraussetzung, dass $f(.)$ den Anforderungen an die Servicequalität genügt.

Schritt 2 Mit der Beobachtung $\tilde{r}$ bestimmt der Angreifer einen geschätzten Aufenthaltsort $\hat{r} \sim h(\hat{r}|\tilde{r}), \hat{r} \in \mathcal{R}$. Der Angreifer kennt die Wahrscheinlichkeitsverteilung $f(\tilde{r}|r)$, die das LPPM nutzt. Er kennt außerdem das Profil des Nutzers$\boldsymbol{\psi}$, aber nicht den wahren Aufenthaltsort r.

Abschließender Schritt Der Angreifer zahlt einen Betrag $d_p(\hat{r}, r)$ an den Nutzer. Dieser Betrag entspricht dem Fehler des Angreifers (oder äquivalent der location privacy, die der Nutzer gewonnen hat).

Die obige Beschreibung ist gemeinsames Wissen für Angreifer und Nutzer. Beide wollen ihre Auszahlung maximieren, d.h. der Angreifer versucht den Betrag zu minimieren, der er zahlen muss, während der Nutzer versucht diesen zu maximieren. Wir zeigen, dass dieses Spiel durch das folgende lineare Programm [STT+12] gelöst werden kann. Wähle $f(\tilde{r}|r), x_{\tilde{r}}, \forall r, \tilde{r}$, so dass $\sum_{\tilde{r}} x_{\tilde{r}}$ maximiert wird, unter der Voraussetzung, dass

$$x_{\tilde{r}} \leq \sum_{r} \psi(r) \cdot f(\tilde{r}|r) \cdot d_p(\hat{r}, r), \forall \hat{r}, \tilde{r} \tag{6}$$

$$\sum_{r} \psi(r) \sum_{\tilde{r}} f(\tilde{r}|r) \cdot d_q(\tilde{r}, r) \leq Q_{loss}^{\max} \tag{7}$$

Die Ungleichungen (6) sind eine Folge linearer Beschränkungen, um eine maximale Privatsphäre für den Nutzer zu garantieren. Eine Folge für jeden Wert $\tilde{r}$. Die Ungleichung (7) spiegelt die minimale Servicequalität wieder.

4 Fazit

In dieser Arbeit haben wir unseren Fokus darauf gelegt, wie man die location privacy von mobilen Nutzern quantifizieren kann im Kontext ortsbasierter Dienste (location-based services (LBS)). Die Informationen, die Nutzer mit Diensteanbietern in verschiedenen LBS teilen, kann zur Aufdeckung ihrer persönlichen Daten führen. Aus diesem Grund werden Privatsphäre-erhaltende Mechanismen genutzt um die Menge an Informationen, die ein Nutzer gegenüber Dritten (z.B. Diensteanbietern) preisgibt, möglichst gering zu halten. Wir haben hierzu ein analytisches Framework entwickelt um die location privacy von Nutzern zu evaluieren. Unser Schwerpunkt liegt hierbei auf dem Modell des Angreifers. Wir haben das Problem der Quantifizierung der location privacy formalisiert als ein Schätzproblem: (i) der Angreifer hat ein gewisses Vorwissen über Benutzer, (ii) er beobachtet ihre Zugriffe auf ein LBS, die seine (möglicherweise anonymisierten und verschleierten) Standortinformationen enthalten, und (iii) er versucht die privaten Daten der Nutzer zu schätzen, die vor ihm versteckt sind. Wir haben eine Methode mit Hilfe von Hidden Markov Modellen entwickelt, um die Identität von Nutzern festzustellen und ihre Bewegungen nachvollziehen zu können. Wir haben diese Methode in einer Open-Source-Software implementiert (location-privacy and mobility meter (LPM)). Die Ergebnisse aus der Nutzung LPMs auf realen Traces lassen vermuten, dass die location privacy der Nutzer nicht nur vom gewählten Schutzmechanismus abhängt, sondern auch von dem Wissen des Angreifers und seinen Zielen und Inferenzangriffen. Wir haben dann das Problem des Designs location privacy erhaltender Mechanismen als Stackelberg-Spiel zwischen Nutzer und Angreifer formalisiert. Da der Angreifer nicht die tatsächliche Position des Nutzers kennt, aber dennoch eine probabilistische Schätzung dieser hat, ist das Spiel Bayes'sch. Durch die Lösung des Spiels finden wir eine optimale Strategie für die Nutzer gegen einen Gegner, der versucht, seinen Schätzfehler zu minimieren (der der Privatsphäre des Nutzers entspricht). Unsere Ergebnisse legen nahe, dass die Strategie des Nutzers, der einen Angreifer antizipiert, einen höheren Grad an Privatsphäre für den Nutzer ermöglichen, verglichen mit einfachen Verschleierungsmechanismen.

Diese Arbeit ist ein großer Schritt hin zu einem besseren Verständnis und Analyse von Gefahren für die location privacy von Nutzern [Sho13].

Literatur

[BS03] A. Beresford, F. Stajano. Location Privacy in Pervasive Computing. In *IEEE Pervasive Computing*, 2003.

[CG09] R. Chow, P. Golle. Faking contextual data for fun, profit, and privacy. In *WPES*, 2009.

[CPP08] K. Chatzikokolakis, C. Palamidessi, P. Panangaden. Anonymity protocols as noisy channels. *Information and Computation*, 2008.

[Dwo06] C. Dwork. Differential privacy. *Automata, languages and programming*, 2006.

[FSH09] J. Freudiger, R. Shokri, J.-P. Hubaux. On the Optimal Placement of Mix Zones. In *PETS*, 2009.

[FSH11] J. Freudiger, R. Shokri, J.-P. Hubaux. Evaluating the Privacy Risk of Location-Based Services. In *FC*, 2011.

[GP09] P. Golle, K. Partridge. On the Anonymity of Home/Work Location Pairs. In *Pervasive*, 2009.

[Kru09] J. Krumm. A survey of computational location privacy. *Personal Ubiquitous Comput.* 2009.

[MRC09] J. Meyerowitz, R. Roy Choudhury. Hiding stars with fireworks: location privacy through camouflage. In ACM *MobiCom* 2009.

[MYYR10] C. Ma, D. Yau, N. Yip, N. Rao. Privacy vulnerability of published anonymous mobility traces. In ACM *MobiCom* 2010.

[SD02] A. Serjantov, G. Danezis. Towards an Information Theoretic Metric for Anonymity. In *PET* 2002.

[Sho13] Reza Shokri. *Quantifying and Protecting Location Privacy*. Dissertation, EPFL, 2013.

[STD+11] R. Shokri, G. Theodorakopoulos, G. Danezis, J.-P. Hubaux, J.-Y. Le Boudec. Quantifying location privacy: the case of sporadic location exposure. In *PETS* 2011.

[STLBH11] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, J.-P. Hubaux. Quantifying Location Privacy. In IEEE *S&P*, 2011.

[STT+12] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, J.-Y. Le Boudec. Protecting Location Privacy: Optimal Strategy against Localization Attacks. In ACM *CCS* 2012.

[Tro11] C. Troncoso. *Design and analysis methods for privacy technologies*. Dissertation, Katholieke Universiteit Leuven, 2011.

[ZB11] H. Zang, J. Bolot. Anonymization of location data does not work: a large-scale measurement study. In ACM *MobiCom*, 2011.

Reza Shokri ist seit Oktober 2013 Post-Doktorand am Institut für Informationssicherheit, Departement für Informatik, ETH Zürich. Zuvor war er wissenschaftlicher Mitarbeiter an der "School of Computer and Communication Sciences" der EPFL Lausanne, wo er im März 2013 seine Dissertation abschloss. Seine Forschung beschäftigt sich mit der Entwicklung von Lösungen zum Schutz personenbezogener Daten und der Quantitativen Analyse von Datenschutz in verschiedenen Anwendungen, unter anderem in standortbezogenen Diensten, Empfehlungssystemen, und Webdiensten. Seine Arbeit zur Messung des Datenschutzes in standortbezogenen Diensten wurde beim jährlichen "Award for Outstanding Research in Privacy Enhancing Technologies (PET Award) 2012" mit dem zweiten Platz ausgezeichnet. Weitere Informationen auf seiner Webseite http://www.shokri.org.

Formale Spezifikationsebene*

Mathias Soeken

AG Rechnerarchitektur
Universität Bremen
msoeken@informatik.uni-bremen.de

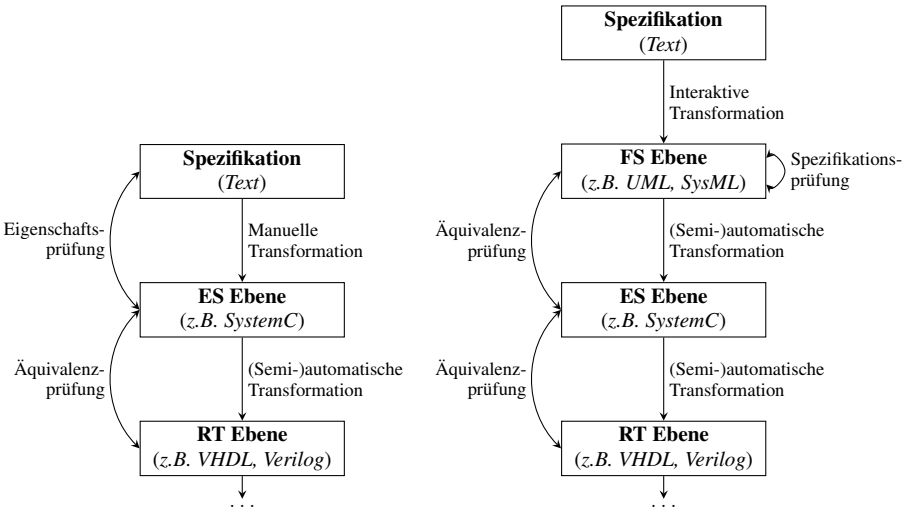
Abstract: Während die hardwarenahen Abstraktionsebenen im Entwurf von komplexen sicherheitskritischen Systemen mittlerweile gut verstanden sind, stellt insbesondere die korrekte Transformation auf höheren Ebenen (wie beispielsweise die Überführung einer textuellen Spezifikation in eine Implementierung) eine große Herausforderung da. Dies lässt sich insbesondere mit der großen konzeptuellen Diskrepanz dieser beiden Ebene erklären. Als Lösungsvorschlag wurde in der hier zusammengefassten Dissertation eine neue Abstraktionsebene zwischen Spezifikation und Implementierung eingeführt, die (a) eine semiautomatische Übersetzung der informellen Spezifikation in ein formales Modell erlaubt und (b) Verifikationsmethoden auf dem resultierenden Modell ermöglicht. Dadurch wird nicht nur die Qualität des Entwurfs verbessert; auch Fehler lassen sich so früh im Entwurf entdecken, was zu Zeitersparnissen führt.

1 Einführung

Der Entwurf von eingebetteten Systemen, die heutzutage aus bis zu Milliarden von Einzelkomponenten bestehen, ist eine der komplexesten Aufgaben mit denen sich Wissenschaftler und Ingenieure auseinandersetzen. Während es vor 40 Jahren noch möglich war solche Systeme Gatter für Gatter auf dem Reißbrett zu entwerfen, ist dieses Vorgehen heute praktisch wegen der stetig steigenden Komplexität nicht mehr umsetzbar. Um den Entwurf in den Griff zu bekommen, haben Wissenschaftler innerhalb der letzten Jahrzehnte einen ausgeklügelten und gründlich durchdachten Entwurfsablauf entwickelt, welcher aus einer Vielzahl von unterschiedlichen Abstraktionsebenen besteht, die das zu entwerfende System sukzessive verfeinern. Dieser Ablauf beginnt bei einer überwiegend textuellen Spezifikationen und mündet schließlich in einer ausgeflachten Transistornetzliste mit präzisen Layoutinformationen.

Der Entwurfsablauf wie er heutzutage üblicherweise Anwendung findet ist dargestellt in Abbildung 1(a). Dessen Startpunkt und somit auch die abstrakteste Beschreibung ist eine informelle natürlichsprachliche Spezifikation. Um jedoch auch nur die einfachste automatische Methode anwenden zu können, muss eine formale Beschreibung vorliegen, die maschinenlesbar ist. Aus diesem Grunde wird ausgehend von der textuellen Spezifikation

*Englischer Titel der Dissertation: “Formal Specification Level”. Die Arbeit wurde betreut von Prof. Dr. Rolf Drechsler, Leiter der Arbeitsgruppe Rechnerarchitektur an der Universität Bremen.



(a) Traditioneller Entwurfsablauf (b) Neuer Entwurfsablauf

Abbildung 1: Gegenüberstellung der Entwurfsabläufe

eine initiale Implementierung in der elektronischen Systemebene (engl.: *Electronic System Level*, kurz ESL, [BM09]) generiert. Auf dieser Ebene stehen softwarenahe Programmiersprachen wie SystemC zur Verfügung. SystemC ist eine C++-Klassenbibliothek für ereignisgesteuerte Simulation nebenläufiger Prozesse. Die Systemebene erlaubt die Ausführung und Simulation des zu entwerfenden Systems, benötigt jedoch keine konkreten Details über die Umsetzung als Hardware. Ausgehend von dieser Beschreibung wird das System sukzessive verfeinert, was zu immer detailreicheren Beschreibungen auf Register-Transferebene (engl.: *Register Transfer Level*, kurz RTL), Gatterebene und physikalischer Ebene führt. Am Ende dieses Prozesses kann das resultierende System zur Fabrikation gesendet werden.

Da eingebettete Systeme oft in sicherheitskritischen Bereichen wie Avionik-, Automobil- und Medizinanwendungen zum Einsatz kommen, ist insbesondere ein *korrekter* Entwurf von höchster Bedeutung. Um die Korrektheit sicherzustellen, werden üblicherweise die Transformationen von einer Abstraktionsebene auf die nächst präzisere auf Äquivalenz verglichen. Da jedoch kein formales Modell für die Spezifikation vorliegt, kann die textuelle Spezifikation nicht automatisch mit dem Modell auf der Systemebene verglichen werden. Die Lücke zwischen diesen beiden Ebenen ist die größte im Entwurfsablauf. Dies wird insbesondere dadurch ersichtlich, dass die Spezifikation *informell* beschreibt *was* entworfen werden soll, die Systemebene hingegen *formal* beschreibt *wie* das System entworfen werden soll. Da zusätzlich zu dieser konzeptuellen Diskrepanz die Spezifikation manuell in ein Modell der Systemebene übersetzt werden muss, ist dieser erste Schritt im Entwurf besonders anfällig für Fehler.

Zur Überprüfung, ob die Intentionen aus der Spezifikation auf Systemebene umgesetzt werden, wird heutzutage Eigenschaftsprüfung als Alternative zum Äquivalenzvergleich

eingesetzt. Dazu werden formale temporallogische Eigenschaften aus der Spezifikation extrahiert welche mithilfe von *Model Checking* [CGP99] mit dem Modell auf Systemebene abgeglichen werden. Darauf aufsetzend existieren Techniken, welche die Vollständigkeit der formalen Abdeckung sicherstellen [GD10]. Das größte Hindernis bleibt jedoch weiterhin bestehen: die Spezifikation ist in natürlicher Sprache gegeben und eine formale Beschreibung muss manuell von ihr abgeleitet werden. Motiviert durch dieses Problem haben Wissenschaftler kürzlich damit begonnen, Techniken zu entwickeln, welche die Lücke zwischen Spezifikations- und Systemeebene schließen [Har12].

Aufbauend auf diesen Arbeiten wird in der Dissertation die formale Spezifikationsebene (engl.: *Formal Specification Level*, kurz: FSL, [DSW12a][1]) als neue Abstraktionsebene eingeführt wie es in Abbildung 1(b) illustriert ist. Die Beschreibungsmittel in dieser Ebene basieren auf Modellierungssprachen mit denen *formal* beschrieben wird *was* entworfen worden soll. Somit verbindet die FSL die textuelle Spezifikationsebene mit der ESL in idealer Weise. Modellierungssprachen wie die *Unified Modeling Language* (UML, [RJB99]), die *System Modeling Language* (SysML, [Wei07]) kombiniert mit Ausdrücken der *Object Constraint Language* (OCL, [WK99]) erlauben eine genaue Beschreibung der Syntax und Semantik für die Modelle auf der FSL. Durch ihre Abstraktheit erlauben diese Modelle eine geeignete Beschreibung der Spezifikationen und ermöglichen zusätzlich die Anwendung automatischer Analysen durch ihre formale Natur. Mit Einführung dieser neuen Abstraktionsebene liefert die Dissertation folgende Beiträge:

- Da die FSL näher zur textuellen Spezifikation ist — auf beiden Ebenen wird beschrieben, was entworfen werden soll — wird eine semiautomatische Übersetzung von informellen Spezifikationselementen in formale Modelle ermöglicht. Techniken aus den Bereichen der maschinellen Textverarbeitung [JM00], Informationsextraktion [CL96] und Wissensrepräsentation [Mil95] werden zu diesem Zweck verwendet. Schon mit einfachen grammatikalische Analysen können beispielsweise die Komponenten des Systems (repräsentiert durch Nomen), ihre Funktionalität (repräsentiert durch Verben) sowie ihre Eigenschaften (repräsentiert durch Adjektive) extrahiert werden. Anstatt auf einen automatischen Ansatz anzustreben, werden in der Dissertation interaktive Methoden entwickelt, mit denen der Entwickler zusammen mit dem Computer die informelle Spezifikation in formale Modelle übersetzt. Dadurch wird die Qualität der Transformation signifikant verbessert.

- Durch neue Verifikationsmethoden auf Modellebene können kritische Fehler in der Spezifikation früher entdeckt werden; sogar noch bevor eine Implementierung geschrieben ist. Diese Fehler sind meist von konzeptueller Art und eine späte Entdeckung impliziert oft aufwändige Korrekturen, die sich über viele Ebenen ausstrecken. Die Verifikationsmethoden sind bezüglich Skalierbarkeit optimiert und betrachten sowohl strukturelle als auch verhaltensbasierte Eigenschaften des Modells.

In dieser Zusammenfassung soll insbesondere auf den zweiten Punkt eingegangen werden. Zunächst wird im nächsten Abschnitt ein detaillierter Blick auf die neue Abstraktionsebene geworfen.

[1] Die Verfahren und Ergebnisse aus der Dissertation werden derzeit für ein Buch aufgearbeitet [SD14].

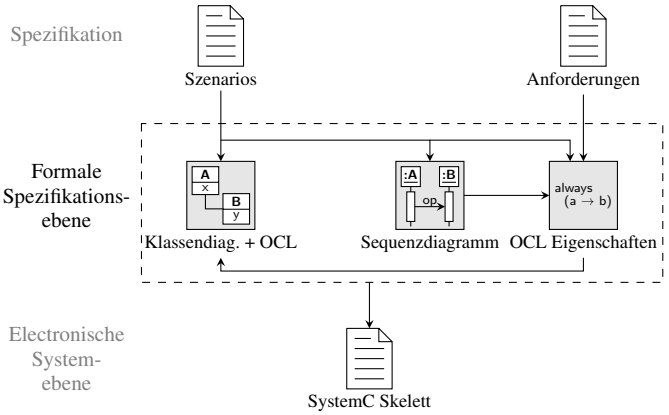


Abbildung 2: Überblick über die formale Spezifikationsebene

2 Die Formale Spezifikationsebene

Abbildung 2 zeigt einen detaillierten Überblick über die vorgeschlagenen Erweiterungen des vorgegenwärtigen Entwurfablaufes. Die Hauptziele sind eine semiautomatische Übersetzung der textuellen Spezifikation in eine Beschreibung auf der elektronischen Systemebene (z.B. SystemC) sowie die Möglichkeit zur Anwendung von Methoden für die Korrektheitsprüfung des formalen Modells selbst. Wir unterscheiden auf der textuellen Spezifikationsebene zwischen Szenarien und Anforderungen als Beschreibungsmittel. Anforderungen können sowohl funktionale als auch nicht-funktionale Eigenschaften beschreiben. In einer Automobilanwendungen können beispielsweise “Die Fenster des Autos können automatisch bedient werden.” und “Der Airbag löst innerhalb von höchstens 30 ms aus.” Anforderungen sein. Anforderungen werden verwendet, um formale Eigenschaften aus ihnen zu generieren, die automatisch auf dem Modell der Systemebene geprüft werden können. Für eine Testfallgeneration bieten sich Szenarien an, da diese konkrete Anwendungsfälle beschreiben. Ein Beispiel für ein Szenario ist “Nachdem die Zündung geschaltet worden ist, startet der Motor.” Szenarien können einfach in Testfälle übersetzt werden, sind jedoch nicht zur vollständigen Verifikation geeignet, da nicht alle möglichen Fälle berücksichtigt werden.

Mit der FSL können bei gegebenen Szenarien und Anforderungen in natürlicher Sprache eine initiale SystemC-Implementierung semiautomatisch generiert werden. Zusätzlich kann die Spezifikation auf konzeptuelle Korrektheit automatisch geprüft werden. Dazu werden innerhalb der FSL zwei Schritte vollzogen, die im Folgenden erläutert werden.

Im ersten Schritt werden Szenarien und Anforderungen in formale Modelle übersetzt. Techniken der maschinellen Textverarbeitung werden angewendet, um die folgenden Teilaufgaben dieses ersten Schrittes zu lösen:

- **Extraktion der Struktur:** Mithilfe der grammatikalischen Analyse von Sätzen innerhalb der textuellen Spezifikation können Basiskomponenten des Systems extra-

hiert werden. Aus den extrahierten Informationen können beispielsweise Klassendiagramme erstellt werden, die eine erste formale Beschreibung der Struktur bieten.

- **Extraktion des Verhaltens:** Auf ähnliche Weise lassen sich auch Ausführungssequenzen extrahieren und in Form von Sequenzdiagrammen darstellen.
- **Extraktion von formalen Eigenschaften:** Nachdem die Struktur extrahiert worden ist, können basierend auf dem formalen Modell auch natürlichsprachliche Anforderungen in formale Eigenschaften, beispielsweise in Form von OCL Ausdrücken, extrahiert werden.

Als Resultat liefert der erste Schritt eine formale Beschreibung des Systems in Form von Modellen. Konkret sind es in diesem Fall Klassen- und Sequenzdiagramme, die um OCL Ausdrücke angereichert sind.

Im zweiten Schritt werden diese formalen Beschreibungen verwendet, um initiale automatische Analysen auszuführen, welche die Spezifikation auf eine konzeptuelle Korrektheit überprüfen:

- **Verifikation struktureller Aspekte.** Bei großen Spezifikationen ist es nicht unwahrscheinlich, dass widersprüchliche Anforderungen enthalten sind. Dies bedeutet, dass es keine gültige Implementierung der Spezifikation geben kann. Bei zu vielen Anforderungen und großen Modellen kann diese Aufgabe nicht mehr manuell bewältigt werden.
- **Verifikation des Verhaltens.** Wenn zusätzlich zu den Anforderungen auch die Operationen des Modells, welche in Form von Vor- und Nachbedingungen spezifiziert sind, betrachtet werden, lassen sich Erreichbarkeitsanalysen automatisch durchführen. Dies beinhaltet beispielsweise Verfahren, ob ein nichterlaubter Zustand erreicht werden kann oder ob es möglich ist in einen Zustand zu gelangen, der nicht mehr verlassen werden kann.

Da diese Verifikationstechniken bereits angewendet werden können, bevor eine konkrete Implementierung des Systems vorliegt, können kritische Fehler in der Spezifikation bereits in einer sehr frühen Phase des Entwurfs entdeckt und behoben werden.

3 Abbildung von Szenarien in die FSL

Zur semiautomatischen Extraktion von Informationen aus der informellen Spezifikation mithilfe von maschineller Textverarbeitung wurden in der Dissertation verschiedene Ansätze vorgestellt, die in diesem Abschnitt kurz zusammengefasst sind. In [SWD12a] ist ein Ansatz vorgestellt worden, der den Entwickler bei der Übersetzung von natürlichsprachlichen Szenarien in formale Modelle unterstützt. Mittels syntaktischer Analyse werden dabei Klassen, Attribute, Operationen und Assoziation extrahiert. Dabei wird die natürliche Sprache nicht eingeschränkt. Mögliche Mehrdeutigkeiten werden vom Computer erkannt

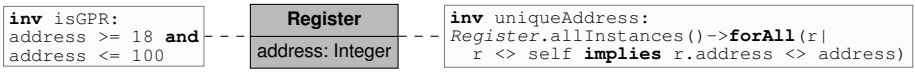


Abbildung 3: Klassendiagramm

und in einem Dialog mit dem Entwickler aufgelöst [DSW12b]. Dies führt insbesondere zu einer klareren Spezifikation, denn Texte, die durch einen Computer missverstanden werden können, werden wahrscheinlich auch von anderen Entwicklern falsch interpretiert. Das Extraktionsverfahren wurde auch in Form einer integrierten Entwicklungsumgebung implementiert [KSKD13].

4 Verifikation von formalen Modellen

Durch den ersten Schritt in der FSL wurde ein formales Modell aus der Spezifikation extrahiert. Im zweiten Schritt soll das Modell auf konzeptuelle Konsistenz überprüft werden, da es bei vielen Anforderungen möglich ist, dass sie widersprüchlich sind. Zu diesem Zweck sind in der Dissertation verschiedene Ansätze entwickelt worden [SWK+10, SWD11b, SWD11a, WSD12], von denen in diesem Abschnitt einige vorgestellt werden.

4.1 Verifikation struktureller Aspekte

Falls die formale Repräsentation des Designs Widersprüche enthält, kann keine valide Implementierung existieren, die der Spezifikation genügt. Verifikationsmethoden auf der FSL erlauben diese Fehler zu erkennen, noch bevor eine Zeile Quellcode geschrieben worden ist.

Strukturelle Aspekte werden insbesondere durch Invarianten in OCL beschrieben, die mögliche Instanzen eines Klassendiagramms einschränken. Zu restriktive Invarianten können zu einer Inkonsistenz des Modells führen.

Beispiel 1. *Abbildung 3 zeigt ein einfaches Klassendiagramm, das ein Register mit einer Adresse modelliert. Zusätzliche Invarianten fordern, dass in einem Objektdiagramm jedes Register eine Adresse zwischen 18 und 100 hat und dass je zwei Register paarweise unterschiedliche Adressen haben. Das Klassendiagramm ist genau dann konsistent, wenn es höchstens 83 Registerinstanzen im Objektdiagramm gibt, da ansonsten nicht ausreichend Adressen zugewiesen werden können.*

Verifikationsmethoden versuchen die Konsistenz eines Klassendiagramms durch das Auffinden einer validen Instanz zu bestätigen. Der in der Dissertation entwickelte Ansatz *ocl2smt* [SWK+10] ist insbesondere in Hinsicht auf Skalierbarkeit optimiert worden. Dazu wird das Konsistenzproblem als Instanz des Erfüllbarkeitsproblems (SAT) formuliert und mit entsprechenden Beweisern automatisch gelöst [BHvMW09]. Im Folgenden wird ein Experiment erläutert, dass die signifikante Performanzsteigerung durch die Transfor-

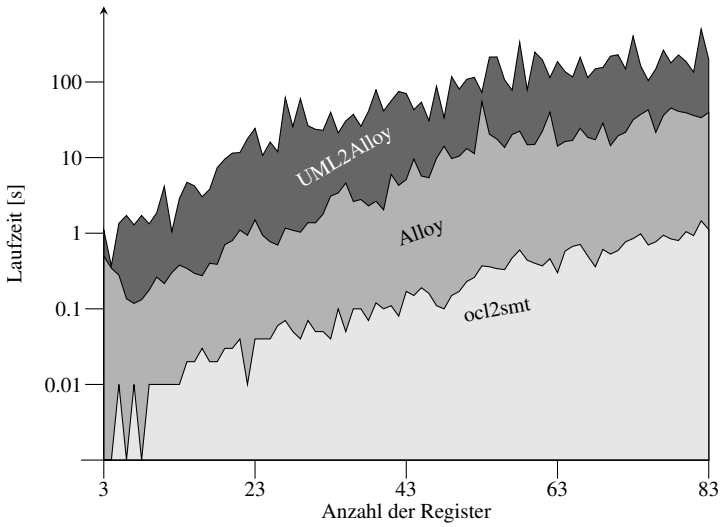


Abbildung 4: Ergebnis des Experiments zum Performanzvergleich

mation in eine SAT Instanz illustriert.

Ein naiver Ansatz wie beispielsweise in [GKH09] illustriert löst das Problem mit ausschöpfender Suche. Der Suchraum für dieses Problem ist allein durch die Anzahl der Instanzen im Objektdiagramm und dem einzigen Attribute *address* charakterisiert. Bei drei Registern benötigt dieser enumerative Ansatz 2,952 Sekunden, bei vier Registern 112 Sekunden. Da alle betrachteten Objektdiagramme in geordneter Reihenfolge validiert werden, lässt sich eine Simulationsgeschwindigkeit von ca. 150.000 Objektdiagrammen pro Sekunde abschätzen. Dadurch lässt sich extrapolieren, dass bereits für ein Objektdiagramm mit sieben Registern mehr als 3,5 Jahre benötigt werden, um ein valides zu finden.

Es gibt andere Ansätze wie z.B. *Alloy* [Jac06], die das Problem auch mithilfe einer Transformation in eine SAT Instanz lösen, jedoch nicht Klassendiagramme mit OCL Invarianten als Eingabesprache verwenden. Abhilfe schafft hier der Ansatz *UML2Alloy* [ABGR07], der Klassendiagramme mit OCL Ausdrücken in ein für Alloy verständliches Format überführt. Im Experiment wurden *Alloy*, *UML2Alloy* und der in der Dissertation entwickelte Ansatz *ocl2smt* verwendet, um für das Klassendiagramm in Abbildung 3 konsistente Objektdiagramme zu finden. Dazu wurde die Anzahl der Register im Objektdiagramm bis zu einer Maximalanzahl von 83 sukzessive erhöht. Zudem wurde als SAT Beweiser MiniSAT [ES03] verwendet, da dieser auch in *Alloy* zum Einsatz kommt.

Die Ergebnisse des Experiments sind in Abbildung 4 illustriert. Alle Ansätze finden Objektdiagramme in weniger als 10 Minuten (die Laufzeit ist auf einer logarithmischen Skala aufgetragen). *UML2Alloy* benötigt am meisten Zeit. *Alloy* ist signifikant schneller, wenn das Problem direkt in der Sprache von *Alloy* formuliert und nicht automatisch übersetzt wird. Der Unterschied in der Laufzeit wächst exponentiell mit Anzahl der Register. Diese Performanzsteigerung erhält man ein weiteres Mal, wenn man das Problem mit *ocl2smt* direkt in eine SAT Instanz formuliert. Um eine möglichst gute Laufzeit zu erhalten, müs-

sen also Transformationsschritte vermieden werden. Auch wenn hier nur ein Beispiel von kleiner Größe betrachtet worden ist, ist der Einfluss der verwendeten Verifikationsmethode dennoch signifikant. Erstens ist der Effekt viel stärker bei größeren Modellen und zweitens kann es sich bei dem Beispiel um ein Teilmodell eines komplexeren Modells handeln.

Die Skalierbarkeit von *ocl2smt* erlaubte unter anderem die Anwendbarkeit größerer Instanzen bei der Testfallgenerierung für Modelltransformationen [GS13]. Außerdem wurde in [WSD12] der erste Ansatz vorgestellt, der Ursachen für inkonsistente Klassendiagramme ermittelt.

4.2 Verifikation des Verhaltens

Auch ein konzeptuelles Verhalten kann bereits auf Basis der Modelle auf der FSL modelliert werden. Dazu werden Vor- und Nachbedingungen verwendet, die an Operationen annotiert werden, ohne eine explizite Implementierung zu fordern. Eine Vorbedingung beschreibt in welchen Zuständen eine Operation aufgerufen werden kann und eine Nachbedingung beschreibt wie sich der Zustand durch einen Operationsaufruf ändert.

Auf Basis dieser durch Vor- und Nachbedingungen annotierten Modelle wurde in der Dissertation erstmals ein Verfahren entwickelt [SWD11b], dass durch *Bounded Model Checking* (BMC, [BCC$^+$03]) inspiriert ist. Durch die Vor- und Nachbedingungen kann eine Ausführungssemantik modelliert werden, die sich auch in einer SAT Instanz formulieren lässt. Dadurch lassen sich dann Verifikationsaufgaben formulieren wie beispielsweise, ob alle Operationen ausgeführt werden können oder ob es einen Deadlock gibt, d.h. ein Zustand, in dem keine Operation aufgerufen werden kann. Da auf Modellebene keine Implementierung der Operationen vorliegt, lässt sich eine viel höhere Skalierbarkeit im Vergleich zur Verifikation auf der ESL erreichen. Das Verfahren wurde auch angewendet um globale OCL Invarianten semiautomatisch in lokale Vor- und Nachbedingungen zu transformieren [SWD12b].

5 Zusammenfassung

In der Dissertation ist eine neue Abstraktionsebene — die formale Spezifikationsebene (FSL) — für den Entwurf von komplexen Systemen vorgeschlagen und detailliert ausgearbeitet worden, welche die informelle und vorwiegend textuelle Spezifikation mit der initialen Implementierung auf der Systemebene verbindet. Während die Spezifikation informell beschreibt, was entworfen werden soll und die Implementierung formal beschreibt, wie es entworfen werden soll, bietet die FSL hier eine ideale Verbindung, da sie formal beschreibt was entworfen werden soll. Somit ist es erstmals möglich Verifikationsmethoden bereits vor der Implementierungsphase automatisch auszuführen, was zu erheblichen Zeiteinsparungen führt. Außerdem wurden Verfahren vorgestellt, die eine interaktive und qualitätsorientierte Übersetzung der informellen Spezifikation in ein formales Model auf Basis von maschineller Textverarbeitung ermöglichen. Dies trägt zusätzlich zur Qualität des zu entwerfenden Systems bei. Ausgewählte Verfahren der Dissertation wurden in die-

ser Zusammenfassung kurz vorgestellt.

Danksagung

Die Arbeit wurde unterstützt durch die DFG im Rahmen des Reinhart Koselleck Projektes "Entwicklung eines durchgängigen Verifikationsablaufes für den ESL Entwurf" (DR 287/23-1) und durch das BMBF im Projekt "SPECifIC – Qualitätsgetriebener Entwurf durch Formale Spezifikation und Funktionales Änderungsmanagement" (01IW13001).

Literatur

[ABGR07] Kyriakos Anastasakis, Behzad Bordbar, Geri Georg und Indrakshi Ray. UML2Alloy: A Challenging Model Transformation. In *Int'l Conference on Model Driven Engineering Languages and Systems*, Seiten 436–450, 2007.

[BCC+03] Armin Biere, Alessandro Cimatti, Edmund M. Clarke, Ofer Strichman und Yunshan Zhu. Bounded model checking. *Advances in Computers*, 58:117–148, 2003.

[BHvMW09] Armin Biere, Marijn Heule, Hans van Maaren und Toby Walsh, Hrsg. *Handbook of Satisfiability*, Jgg. 185 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2009.

[BM09] Brian Bailey und Grant Martin. *ESL Models and Their Application: Electronic System Level Design and Verification in Practice*. Springer, 2009.

[CGP99] Edmund M. Clarke, Jr., Orna Grumberg und Doron A. Peled. *Model Checking*. MIT Press, 1999.

[CL96] James R. Cowie und Wendy G. Lehnert. Information Extraction. *Commun. ACM*, 39(1):80–91, 1996.

[DSW12a] Rolf Drechsler, Mathias Soeken und Robert Wille. Formal Specification Level: Towards verification-driven design based on natural language processing. In *Forum on Specification and Design Languages*, Seiten 53–58, 2012.

[DSW12b] Rolf Drechsler, Mathias Soeken und Robert Wille. Towards dialog systems for assisted natural language processing in the design of embedded systems. In *Int'l Design & Test Symposium*, 2012.

[ES03] Niklas Eén und Niklas Sörensson. An Extensible SAT-solver. In *Theory and Applications of Satisfiability Testing*, Seiten 502–518, 2003.

[GD10] Daniel Große und Rolf Drechsler. *Quality-Driven SystemC Design*. Springer, 2010.

[GKH09] Martin Gogolla, Mirco Kuhlmann und Lars Hamann. Consistency, Independence and Consequences in UML and OCL Models. In *Tests and Proofs*, Seiten 90–104, 2009.

[GS13] Esther Guerra und Mathias Soeken. Specification-Driven Model Transformation Testing. *Software and System Modeling*, 2013. accepted.

[Har12] Ian G. Harris. Extracting design information from natural language specifications. In *Design Automation Conference*, Seiten 1256–1257, 2012.

[Jac06] Daniel Jackson. *Software Abstractions: Logic, Language, and Analysis*. MIT Press, 2006.

[JM00] Daniel Jurafsky und James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, 2000.

[KSKD13] Oliver Keszocze, Mathias Soeken, Eugen Kuksa und Rolf Drechsler. lips: An IDE for Model Driven Engineering Based on Natural Language Processing. In *Workshop on Natural Language Analysis in Software Engineering*, Seiten 31–38, 2013.

[Mil95] George A. Miller. WordNet: A Lexical Database for English. *Commun. ACM*, 38(11):39–41, 1995.

[RJB99] James E. Rumbaugh, Ivar Jacobson und Grady Booch. *The unified modeling language reference manual*. Addison-Wesley-Longman, 1999.

[SD14] Mathias Soeken und Rolf Drechsler. *Formal Specification Level*. Springer, 2014. in preparation.

[SWD11a] Mathias Soeken, Robert Wille und Rolf Drechsler. Encoding OCL Data Types for SAT-Based Verification of UML/OCL Models. In *Tests and Proofs*, Seiten 152–170, 2011.

[SWD11b] Mathias Soeken, Robert Wille und Rolf Drechsler. Verifying Dynamic Aspects of UML Models. In *Design, Automation and Test in Europe*, Seiten 1077–1082, 2011.

[SWD12a] Mathias Soeken, Robert Wille und Rolf Drechsler. Assisted Behavior Driven Development Using Natural Language Processing. In *Int'l. Conference on Objects, Models, Components, Patterns*, Seiten 269–287, 2012.

[SWD12b] Mathias Soeken, Robert Wille und Rolf Drechsler. Eliminating Invariants in UML/OCL Models. In *Design, Automation and Test in Europe*, Seiten 1142–1145, 2012.

[SWK+10] Mathias Soeken, Robert Wille, Mirco Kuhlmann, Martin Gogolla und Rolf Drechsler. Verifying UML/OCL models using Boolean satisfiability. In *Design, Automation and Test in Europe*, Seiten 1341–1344, 2010.

[Wei07] Tim Weilkiens. *Systems engineering with SysML / UML - modeling, analysis, design*. Morgan Kaufmann, 2007.

[WK99] Jos Warmer und Anneke Kleppe. *The Object Constraint Language: Precise Modeling with UML*. Addison-Wesley Longman, 1999.

[WSD12] Robert Wille, Mathias Soeken und Rolf Drechsler. Debugging of Inconsistent UML/OCL Models. In *Design, Automation and Test in Europe*, Seiten 1078–1083, 2012.

Mathias Soeken ist PostDoc an der Universität Bremen in der Arbeitsgruppe Rechnerarchitektur und Wissenschaftler bei der DFKI GmbH in der Gruppe Cyber-Physical Systems. Seine Forschungsinteressen sind formale Verifikation, maschinelle Textverarbeitung und zukünftige Computertechnologien.

Hierarchische Modelle für das visuelle Erkennen und Lernen von Objekten, Szenen und Aktivitäten*

Jens Spehr

Institut für Robotik und Prozessinformatik
Technische Universität Braunschweig
jensspehr@gmx.de

Abstract: In zahlreichen Computer Vision Anwendungen müssen Objekte in einzelnen Bildern oder Bildsequenzen erlernt und erkannt werden. Viele dieser Objekte sind hierarchisch aufgebaut. In dieser Arbeit werden neue probabilistische hierarchische Modelle vorgestellt, die es ermöglichen mehrere Objekte verschiedener Kategorien, Skalierungen, Rotationen und aus verschiedenen Blickrichtungen effizient zu repräsentieren. Eine Idee ist hierbei, Ähnlichkeiten unter Objekten, Objektteilen oder auch Aktionen und Bewegungen zu nutzen, um redundante Informationen und Mehrfachberechnungen zu vermeiden. Das einheitliche Framework ermöglicht die Anwendung des vorgestellten Modells für die klassische Objekterkennung, die Erkennung von menschlichen Posen, die Aktivitätenerkennung und die Umfeldwahrnehmung von intelligenten Fahrzeugen. Eine detaillierte Ausführung dieser Arbeit ist in [Spe13] zu finden.

1 Einleitung

Das visuelle Erkennen und Lernen von Objekten, Szenen und Aktivitäten in bzw. aus Bildern oder Bildsequenzen ist eine sehr relevante, aber gleichzeitig auch sehr schwierige Aufgabe, die Anwendungen in vielen Bereichen wie Robotik, Industrie, Unterhaltungselektronik, Luft- und Raumfahrt, Transportsysteme oder Ambient Assisted Living hat. In industriellen Szenarien kann die Standard Computer Vision Aufgabe des Erfassens und Erkennens von Objekten aus Bilddaten meist mit einem einfachen Schwellwertverfahren und geeigneten Formdeskriptoren gelöst werden. In der realen Welt müssen hingegen mehrere Ansichten, verschiedene Artikulationen und Skalen beachtet und außerdem verschiedene Erscheinungsmerkmale für eine robuste Erkennung kombiniert werden. Darüber hinaus sollte die Repräsentation eine Invarianz gegenüber der typischen Bildverarbeitungsprobleme wie Beleuchtungsänderungen, Hintergrundstörungen oder Verdeckungen bieten. Die erkannten Objekte können zum Beispiel für die inhaltsbasierte Bildsuche oder in einer Augmented-Reality-Anwendung verwendet werden, bei der die Ansicht einer realen Umgebung durch computergenerierte Grafiken ergänzt wird. Andere derzeit sehr interessante

*Englischer Titel der Dissertation: „On Hierarchical Models for Visual Recognition and Learning of Objects, Scenes, and Activities“

Beispiele sind Ambient Assisted Living Anwendungen. Sie sind besonders im Hinblick auf die Alterung der Gesellschaft, die die Unterstützung der älteren Menschen in ihrer häuslichen Umgebung mehr und mehr notwendig macht, interessant. Auf der einen Seite steigt die Zahl der älteren Menschen, auf der anderen Seite nimmt die Zahl der Menschen, die Hilfe anbieten können, ab. Durch den damit verbundenen Rückgang an Pflegekräften werden freie Plätze in Altenheimen knapp. Ein vielversprechender Weg, dieses Dilemma zu lösen, ist es, älteren Menschen zu ermöglichen, so lange wie möglich in ihrer gewohnten häuslichen Umgebung zu bleiben, wobei gleichzeitig die Gesundheit der Person mit Hilfe eines Überwachungssystems sichergestellt wird. Solch ein Überwachungssystem extrahiert Schätzungen der Vitalparameter wie Gehgeschwindigkeit und erkennt kritische Werte und Veränderungen. Visuelle Sensoren wie Kameras haben den Vorteil, unaufdringlich und unsichtbar für den Benutzer zu sein, denn sie benötigen keine explizite Interaktion zwischen Benutzer und System. Solche visuellen Sensorsysteme können verwendet werden, um plötzlich eintretende kritische Ereignisse wie Stürze zu erfassen. Hierfür benötigt der visuelle Sensor jedoch ein effizientes Bildverarbeitungsframework, das Merkmale wie Körpergröße oder Körperachse aus den Bilddaten extrahiert. Darüber hinaus können vision-basierte Ansätze auch zur Sturzprävention verwendet werden. Durch eine Ganganalyse können Merkmale wie Schrittweite, Schritthöhe, Geschwindigkeit und Geschwindigkeitsvarianz, Gangharmonie, Ausgleichsbewegung und Körperschwankung extrahiert und zur Einschätzung, wie kritisch die aktuellen Gangparameter sind, verwendet werden. Eine noch größere Herausforderung ist die Repräsentation von menschlichen Verhaltensmustern und die Erkennung von Anomalien, die häufig auf gesundheitliche Probleme hindeuten.

Ein weiteres vielversprechendes Anwendungsgebiet ist das autonome Fahren eines intelligenten Fahrzeugs, welches eine robuste sensorbasierte Wahrnehmung der Umwelt erfordert. Durch 3D-Rekonstruktionstechniken wie 'structure from motion' stellt eine Kamera einen sehr leistungsfähigen 3D-Sensor dar. Allerdings ist die Interpretation der gesammelten Daten immer noch sehr anspruchsvoll und erfordert aufwendige Ansätze, die ein 'high-level' Verständnis der Szene bezüglich der Erkennung der Straße, anderen Fahrzeugen, Fußgängern, Parkplätzen und so weiter ermöglichen.

In dieser Arbeit werden hierarchische Modelle vorgestellt, die ideal für das Lösen der zuvor beschriebenen Herausforderungen geeignet sind. Objekte können in Teile zerlegt werden, diese Teile in visuelle Grundelemente und schließlich die Grundelemente in lokale Gradienten und Farbwerte. Statt Elemente verschiedener Ansichten oder Konfigurationen unabhängig voneinander zu modellieren, können Elemente untereinander geteilt werden, um so die Gesamtkomplexität der Modelle und die Anzahl der Teile zu reduzieren. Dies führt zu besseren Generalisierungseigenschaften und zu einer effizienten Darstellung, die den Rechenaufwand während des Lern- und Erkennungsprozesses reduziert. Es ermöglicht auch die vorgeschlagenen hierarchischen Modelle in realen Anwendungen in Echtzeit anzuwenden. Durch das Teilen von Informationen werden redundante Informationen vermieden und dadurch unnötige Mehrfachberechnungen reduziert. Dieses Teilen von Informationen ist auch besonders für Gelenkstrukturen, wie dem menschlichen Körper, geeignet, da hier der Grad des Teilens für verschiedene Konfigurationen sehr hoch ist. Ähnliche Ansätze zur hierarchischen Zerlegung und dem Teilen von Informationen können auch für Aktivitäten verwenden werden. Diese sind in Aktionen zerlegbar und Aktionen wie-

derum in Aktionsprimitive. Auch Verkehrsszenen lassen sich in Objekte wie Autos und Fußgänger zerlegen und diese wiederum in einfache Grundelemente.

2 Überblick über die Beiträge dieser Arbeit

Die wichtigsten Beiträge der neuen vorgeschlagenen hierarchischen Methoden und Modelle sind:

- Wir stellen eine hierarchische Repräsentation von blickpunktabhängigen Instanzen vor und bilden die Skalierung und Rotation der Instanzen direkt in der Hierarchie ab. Unsere Hierarchie ist weniger eingeschränkt als andere Darstellungen in der Literatur, d.h. unsere Hierarchien haben bis zu 20 Ebenen und flexible Abhängigkeiten zwischen den Schichten. Diese Flexibilität erlaubt es, die Wiederverwendbarkeit von Teilen zu maximieren und führt damit zu einer kompakten und effizienten Darstellung. Abweichend von früheren hierarchischen Modellen, in denen die höchste Ebene explizit die Objektebene darstellt, wird in der vorgestellten Repräsentation die Hierarchieebene direkt mit der Größe und Komplexität eines Objekts gekoppelt. So wird ein komplexes Objekt mit Textur auf einer höheren Ebene dargestellt als ein einfaches Objekte ohne Textur.

- Wir kombinieren die hierarchische Repräsentation mit einer „Grob-zu-Fein"-Suche mittels einer Ähnlichkeitshierarchie. Obwohl die Idee der „Grob-zu-Fein"-Suche bekannt ist, wird sie in dieser Arbeit erstmals direkt in die Hierarchie auf verschiedenen Abstraktionsebenen integriert und mit einer 'Scale-Space' Repräsentation kombiniert. Wie wir sehen werden, erlaubt diese Kombination das effiziente Generieren und Verifizieren von Teil- und Objekthypothesen. Diese werden in der kompositorischen Hierarchie auf einer groben Auflösungsstufe generiert, in einer „Grob-zu-Fein"-Ähnlichkeitshierarchie verfeinert, und schließlich in der kompositorischen Hierarchie auf feiner Auflösungsstufe bewertet.

- Wir schlagen eine unbeaufsichtigte 'top-down' Lernmethode vor, die die Wiederverwendbarkeit von Teilen maximiert und das Lernen von effizienten Hierarchien sowohl offline als auch online ermöglicht.

- Die Ansätze werden anhand verschiedener Anwendungen demonstriert: Objekterkennung, menschliche Posenschätzung, Aktivitätenerkennung und Szeneninterpretation für intelligente Fahrzeuge.

- Zu weiteren Beiträgen zählen die entwickelten Inferenztechniken: Kombination von 'bottom-up' Message Passing zur Hypothesengenerierung mit 'top-down' Message Passing zur Hypothesenverifikation, Ausführen des 'bottom-up' Message Passings in mehreren sequentiellen 'Sweeps' und beobachtbare 'high-level' Knoten mittels Importance Sampling.

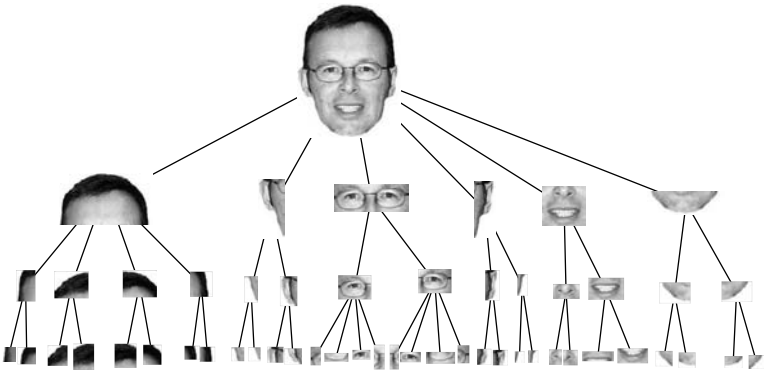


Abbildung 1: Einfaches Beispiel für eine hierarchische Dekomposition. Das Gesicht wird in seine Bestandteile zerlegt, diese Teile werden weiter zerlegt in kleinere Teile und so weiter. Die Kanten definieren die räumlichen Beziehungen zwischen den Elementen.

3 Ein einfaches Beispiel

Wir beginnen mit der Darstellung des Kompositionskonzeptes unter Verwendung der einfachen Hierarchie in Abb. 1. Das Gesicht wird in seine Bestandteile zerlegt, wie die Stirn, die Seiten mit den Ohren, die Augen, das Kinn, die Nase und den Mund. Diese Teile können wieder in kleinere Teile wie Haaransatz, ein einzelnes Auge oder ein Teil des Kinns zerlegt werden. Und schließlich werden diese kleinen Teile in visuelle Primitive wie Kanten oder Ecken zerlegt. Die kompositorische Hierarchie besteht aus zwei Komponenten: (1) die 'low-level' Merkmale und (2) die räumlichen Beziehungen zwischen den Merkmalen, Teilen und Primitiven. Die 'low-level' Features stellen die Beobachtungen dar und sind direkt aus dem Eingangsbild gewonnen. Alle höheren Ebenen sind nicht direkt beobachtbar, so dass Informationen von den unteren Ebenen über die räumlichen Relationen zu den höheren gesendet werden müssen. Diese räumlichen Relationen bestimmen die relativen Positionen zwischen dem Objekt und seinen Teilen.
Da ein Objekt auf verschiedene Weisen zerlegt werden kann, kann es auch durch verschiedene Hierarchien repräsentiert werden. Dies ist eine wichtige Eigenschaft, die dazu genutzt werden kann, unter allen Hierarchien diejenige auszuwählen, die besonders aus rechnerischen Sicht attraktiv ist. Die Idee hierbei ist redundante Berechnungen zu reduzieren, indem Informationen über ähnliche Teile wie z.B. das linke und rechte Auge nur einmal berechnet werden.

4 Hierarchische Graphische Modelle

Das Ziel der vorgeschlagenen hierarchischen graphischen Modelle ist es, verschiedene Instanzen unterschiedlicher Objektklassen in Bildern, Bildsequenzen oder andere Szenenrepräsentationen zu erkennen. Der Begriff „Objekt“ wird in diesem Zusammenhang als

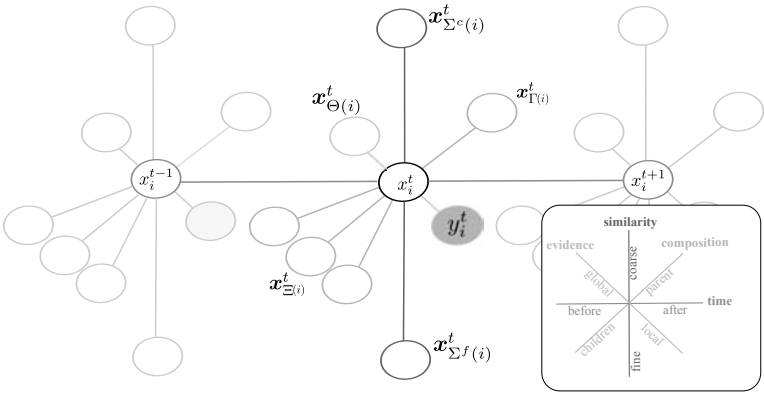


Abbildung 2: Informationsquellen von Knoten x_i (Evidenzen, Komposition, Ähnlichkeit und Zeit).

ein allgemeiner Begriff für visuelle Objekte, visuelle Teile, visuelle Merkmale, visuelle Grundtypen, aber auch Aktivitäten, Aktionen oder Bewegungsprimitive verwendet. Im Folgenden werden zwei verschiedene Arten von Hierarchien betrachtet, deren Unterscheidung für das Verstehen, der in dieser Arbeit vorgeschlagenen Ansätze sehr wichtig ist: kompositorische Hierarchien und Ähnlichkeitshierarchien. In kompositorischen Hierarchien wird die Zerlegung eines Objektes durch Knoten und Kanten repräsentiert, wobei Kanten räumliche Beziehung zwischen den Eltern und den Kindern beschreiben. Auf diese Weise können komplexe 'high-level' Knoten rekursiv in einfache 'low-level' Merkmale zerlegt werden. Zu solchen Hierarchien gehört auch die in Abb. 1 dargestellte Hierarchie. Ähnlichkeitshierarchien beschreiben hingegen Ähnlichkeiten zwischen Objekten und zwischen Teilen. In dieser Arbeit werden sie verwendet, um eine „Grob-zu-Fein"-Suche durchzuführen. In der Realität sind alle Messungen mit Unsicherheiten behaftet. Die Knoten der Hierarchie sind daher mit Zufallsvariablen assoziiert, die die Unsicherheiten in Form von Wahrscheinlichkeitsdichtefunktionen beschreiben. Die ungerichtete Graphenrepräsentation entspricht einem 'Markov Random Field'.

Es werden nun die Informationsquellen diskutieren, auf die ein Knoten in der Hierarchie zugreifen kann: Wir unterscheiden vier verschiedene Arten von Informationsquellen: evidenzbasierte, kompositorische, ähnlichkeitsbasierte und zeitliche Informationsquellen (dargestellt in Abb.2 durch verschiedene Farben). Der Knoten i kann seine a-posteriori Marginalverteilung $b_i(x_i) = p(x_i|\boldsymbol{y})$ durch das Kombinieren der Informationen seiner Nachbarn wie folgt berechnen:

$$b_i(x_i) \propto p(x_i|\boldsymbol{y}_{obs})p(x_i|\boldsymbol{y}_{comp})p(x_i|\boldsymbol{y}_{sim})p(x_i|\boldsymbol{y}_{temp}) \tag{1}$$

Über die Verteilung $p(x_i|\boldsymbol{y}_{obs})$ kommen die Sensorinformationen in das graphische Modell. Hierbei wird zwischen lokaler und globaler Evidenz unterschieden. Lokale Evidenzen beschreiben lokale Merkmale des Eingangsbildes oder -sequenz, die mit Merkmalsextraktionsverfahren wie beispielsweise Kantendetektoren aus den Bilddaten bestimmt werden. Globale Evidenzen lassen sich aus den Eingangsdaten als Ganzes ableiten und beschreiben so globale Merkmale wie z.B. Farbhistogramme des gesamten Bildes. Informationen, wie

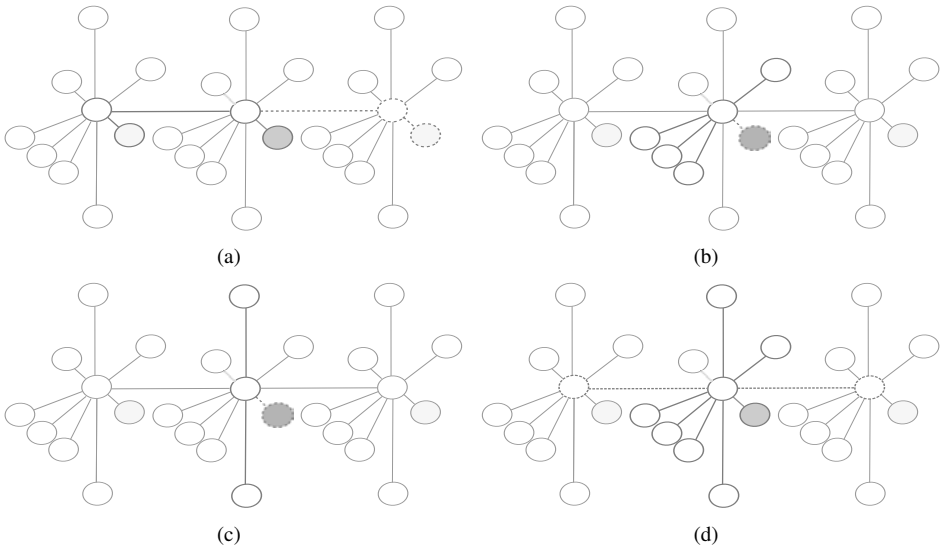


Abbildung 3: Beispiele für Informationsquellen (rot) in den folgenden Ansätzen: a) Zeitliches Tracking (z.B. Kalman- oder Partikelfilter). b) Kompositorische Hierarchie [SP05, ZLH+08, FL07]. c) Ähnlichkeitshierarchie (z.B. [Gav98]). d) Kombinierte kompositorische und Ähnlichkeitshierarchie (diese Arbeit).

sie im Beispiel aus Abschnitt 3 beschrieben wurden, werden durch $p(x_i|\boldsymbol{y}_{comp})$ berücksichtigt. Hierbei werden auf der einen Seite Informationen berücksichtigt, die von den Kinderknoten 'bottom-up' an x_i gesendet werden. Auf der anderen Seite werden Informationen vom Elternteil 'top-down' an x_i gesendet. Die Verteilung $p(x_i|\boldsymbol{y}_{sim})$ berücksichtigt Ähnlichkeit zwischen den Knoten. Hierbei wird zwischen Ähnlichkeiten zu Knoten auf einer groben und einer feinen Auflösung unterschieden. Genau diese Ähnlichkeitskanten sind es, aus denen die Ähnlichkeitshierarchien zusammengesetzt sind. Sie spielen eine entscheidende Rolle beim „Grob-zu-Fein" Generieren von Objekthypothesen. Bisher wurde die Bestimmung von Objekthypothesen lediglich zu einem festen Zeitpunkt betrachtet. Oft werden Sensorinformationen jedoch über die Zeit gesammelt, so dass es neben den Sensorinformationen aus dem aktuellen Zeitschritt auch Informationen aus der Vergangenheit und offline auch aus der Zukunft gibt. Diese Informationen werden durch $p(x_i|\boldsymbol{y}_{temp})$ berücksichtigt.

Abb. 3 zeigt einige Beispiele für Ansätze aus der Literatur und ihren verwendeten Informationsquellen. Ansätze, die im wesentlichen auf zeitlichen Informationen beruhen, sind in Abb. 3 a) dargestellt, wobei in rot die verwendeten Kanten hervorgehoben wurden. Bekannte Beispiele dieser Ansätze sind der Kalman- oder Partikelfilter. Abb. 3 b) zeigt die Informationsquellen von kompositorischen Ansätzen, die Objekte in kleinere Teile zerlegen. Ähnlichkeitshierarchien verwenden die in Abb. 3 c) dargestellten Kanten. Die in dieser Arbeit vorgestellten Ansätze greifen auf alle zur Verfügung stehenden Informationen, wie in Abb. 3 d) dargestellt, zu.

Stehen x_i alle Informationsquellen zur Verfügung, stellt sich die schwierige Frage, wie das

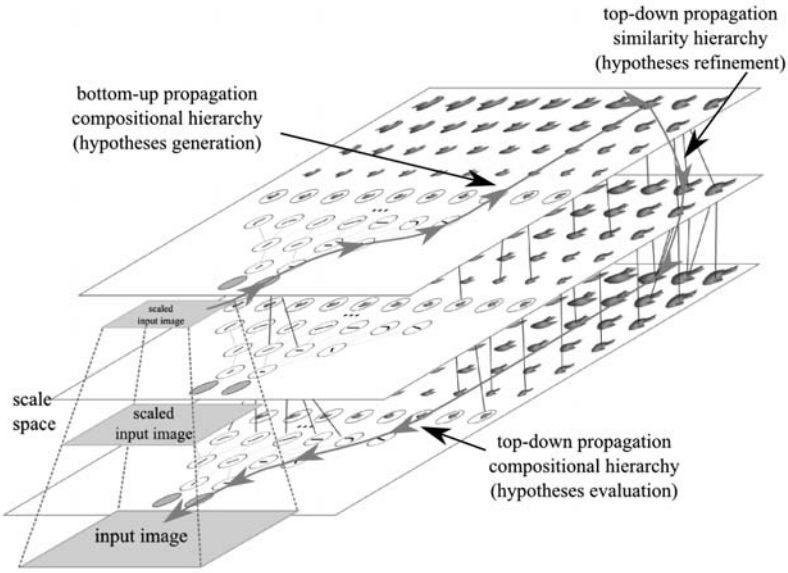
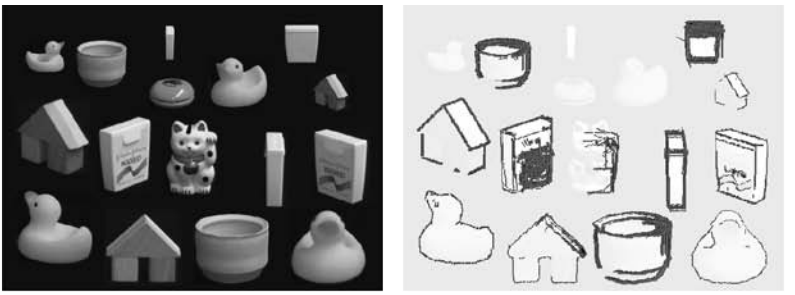


Abbildung 4: Das Versenden von Nachrichten in einer kompositorischen Hierarchie und Ähnlichkeitshierarchie für das effiziente Lösen des Inferenproblems.

Inferenzproblem zu lösen ist. In dieser Arbeit wird hierfür Belief Propagation verwendet. Da in unserem Fall die Zufallsvariablen kontinuierlich und multimodal sind, wird nichtparametrisches Belief Propagation angewendet. Dieser Ansatz ist ähnlich dem bekannten Partikelfilter, bei dem die Wahrscheinlichkeitsdichtefunktion durch eine Menge von Partikeln repräsentiert wird. In dieser Arbeit werden verschiedene Ansätze vorgestellt, mit denen sich das Inferenzproblem effizient und in Echtzeit lösen lässt. Abb. 4 gibt ein Beispiel für das Lösen des Inferenzproblem durch intelligentes Nachrichtenversenden. Um Objekte im Eingangsbild erkennen zu können, wird das Bild in Form einer Bildpyramide an die Kombination aus kompositorischer Hierarchie und Ähnlichkeitshierarchie angelegt. Die Bildpyramide enthält neben dem Eingangsbild in Originalauflösung auch verschiedene skalierte Versionen. Die grundlegende Idee des effizienten Nachrichtenversendens ist (1) Objekthypothesen auf einer groben Auflösungsstufe durch das 'bottom-up' Senden von Nachrichten in der kompositorischen Hierarchie zu generieren, (2) diese anschließend in der Ähnlichkeitshierarchie von der groben Auflösung hin zur detaillierten zu verfeinern und (3) schließlich in der kompositorischen Hierarchie auf der feinen Auflösung 'top-down' zu evaluieren (siehe roten Pfad in Abb. 4).

Objekterkennung: Die Erkennung von Objekten aus Bildern ist ein klassisches Problem aus dem Bereich Computer Vision. Eine besondere Herausforderung ist hierbei verschiedene Objekte aus verschiedenen Blickrichtungen und Skalierungen zu erkennen. Die Ansätze aus dieser Arbeit lösen dieses Problem effizient, indem Ähnlichkeiten unter den Objekten ausgenutzt werden und so eine effiziente Lösung des Inferenzproblems ermöglicht wird. Abb. 5 zeigt ein qualitatives Beispielergebnis.



(a) (b)

Abbildung 5: Beispielergebnis für die rotations- und skalierungsunabhängige Erkennung von verschiedenen Objekten: (a) Eingangsbild (b) Erkannte Objekte in rot/blau hervorgehoben.

Posenerkennung: Ähnlich der Erkennung von Objekten können die vorgeschlagenen Ansätze auch zur menschlichen Posenschätzung verwendet werden. Dieses Problem stellt eine noch größere Herausforderung dar, da die Gelenke des menschlichen Körpers zusätzliche Freiheitsgrade darstellen, die mitgeschätzt werden müssen. Glücklicherweise ähneln sich die Erscheinungsbilder des menschlichen Körpers für verschiedene Konfigurationen sehr stark, so dass viele Informationen geteilt werden können. Abb. 6 zeigt Beispiele für erkannte menschliche Posen, die für eine Ganganalyse verwendet wurden. Bemerkenswert ist hierbei, dass der menschliche Körper sehr robust unter verschiedenen Skalierungen erkannt werden kann, dadurch dass die Skalierung des Modells direkt in der Repräsentation abgebildet ist.

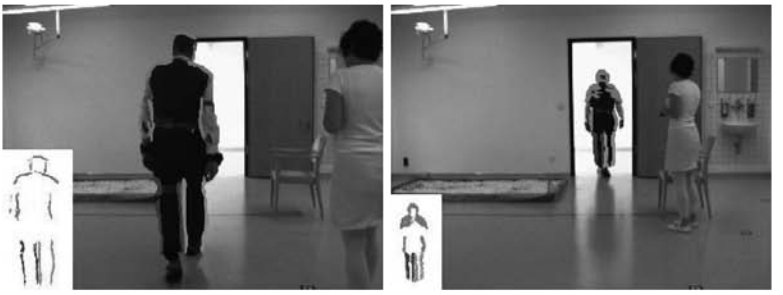


Abbildung 6: Beispiele für die Erkennung der menschlichen Pose: Die Pose ist farblich hervorgehoben in rot (Kopf), blau (linkes Bein) und grün (rechtes Bein).

Verhaltensanalyse: Werden die kompositorischen Hierarchien und ihre Kanten nicht nur räumlich sondern auch raum-zeitlich betrachtet, können die Ansätze auch für die Erkennung von Aktivitäten und somit zur Verhaltensanalyse verwendet werden. Entscheidend ist hierbei, dass neben den räumlichen Beziehungen auch zeitliche Abhängigkeiten modelliert werden. Abb. 7 zeigt ein Beispiel. Die Aktivität 'Kaffee trinken' wird durch eine Menge an Bewegungsvektoren beschrieben (links in rot hervorgehoben). Diese Bewegungen wurden aus der Bildsequenz mithilfe von optischen Fluss berechnet. Die Hierarchie

(rechts) beschreibt nun die Zerlegung der Aktivität 'Kaffee trinken' in einzelne Aktionen wie 'Greifen der Tasse' oder 'Führen der Tasse zum Mund' und schließlich die Zerlegung der Aktionen in einzelne Bewegungsprimitive.

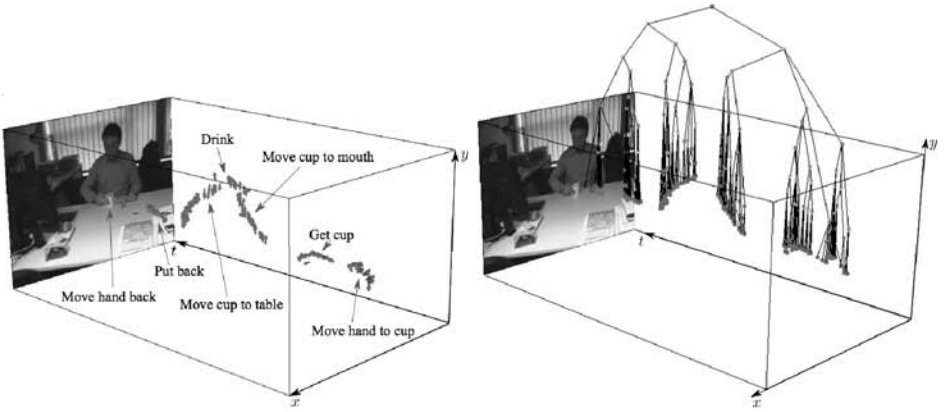


Abbildung 7: Beispiel für die Aktivitätenerkennung: Erkannte Bewegungsvektoren im Raum-Zeit Volumen (links), Zerlegung der gesamten Aktivität in einzelne Aktionen.

Szenenverstehen für intelligente Fahrzeuge: Die vorgeschlagenen Ansätze wurden auch zur Interpretation des Fahrzeugumfelds in Echtzeit verwendend. Basierend auf einfachen Kamerasensoren, die unter den Seitenspiegeln angebracht wurden, wurden die Ansätze verwendet, um effizient Parklücken zu erkennen. Die Idee war hierbei, die Parkplatzszene hierarchisch in einzelne Parklücken zu zerlegen. Die Parklücken wurden wiederum in Markierungen und ggf. einem geparkten Fahrzeug zerlegt. Die Zerlegung wurde über geometrische Grundkörper weitergeführt bis schließlich einfache Belegungswahrscheinlichkeiten einer kartenbasierten Umfeldrepräsentation erreicht wurden. Abb. 8 (links) zeigt den Versuchsträger Paul auf der Hannover Messe 2008. Ergebnisse sind rechts dargestellt, wobei oben die Eingangsdaten und unten die Ergebnisse gezeigt sind.

5 Fazit

In dieser Arbeit wurde eine neue hierarchische Repräsentation vorgestellt, die für ein breites Spektrum von Anwendungen geeignet ist. Unsere Ansätze nutzen drei grundlegende Konzepte: kompositorische Hierarchien, „Grob-zu-Fein"-Ähnlichkeitshierarchien, und das 'Sharing' von Teilen, um eine robuste und effiziente Darstellung von Multi-Objekten, Multi-Skalen-und Multi-Ansichten zu erhalten. Die in dieser Arbeit untersuchten Anwendungen spiegeln nur einen kleinen Teil der möglichen Einsatzgebiete wieder. Andere Anwendungsgebiete sehen wir überall dort, wo Modelle hierarchisch in Teile zerlegt werden können. Die vorgeschlagenen Verfahren sind besonders geeignet, wenn das Modell eine große Menge von verschiedenen Instanzen, Posen oder Konfigurationen in Echtzeit

Abbildung 8: Ergebnisse für das Szenenverstehen intelligenter Fahrzeuge: Versuchtsträger Paul (links), Eingangsdaten als Draufsicht (rechts oben), Beispielergebnisse (rechts unten).

repräsentieren muss. Für all diese Anwendungen liefert diese Arbeit wichtige Grundkonzepte, effiziente Methoden sowie praktische 'How-to'-Beispiele.

Literatur

[FL07] S. Fidler und A. Leonardis. Towards Scalable Representations of Object Categories: Learning a Hierarchy of Parts. Minnesota, USA, June 2007.

[Gav98] D M. Gavrila. Multi-Feature Hierarchical Template Matching Using Distance Transforms. In *Proceedings of the 14th International Conf. on Pattern Recognition-Volume 1*, Seiten 439–, Washington, DC, USA, 1998.

[SP05] F. Scalzo und J. H. Piater. Statistical Learning of Visual Feature Hierarchies. Jgg. 3, Seite 44, 2005.

[Spe13] Jens Spehr. *On Hierarchical Models for Visual Recognition and Learning of Objects, Scenes, and Activities*. Dissertation, Technische Universität Braunschweig, 2013.

[ZLH+08] Long Zhu, Chenxi Lin, Haoda Huang, Yuanhao Chen und Alan L. Yuille. Unsupervised Structure Learning: Hierarchical Recursive Composition, Suspicious Coincidence and Competitive Exclusion. Jgg. 5303 of *Lecture Notes in Computer Science*, Seiten 759–773, 2008.

Jens Spehr studierte Elektrotechnik an der TU Braunschweig. Nach dem Abschluss des Studiums 2006 war er als wissenschaftlicher Mitarbeiter am Institut für Robotik und Prozessinformatik der TU Braunschweig beschäftigt, an dem er 2013 promovierte. Momentan arbeitet er bei der Konzernforschung der Volkswagen AG, wo er verantwortlich ist für die Umfeldwahrnehmung von automatisch fahrenden Fahrzeugen. Zu seinen Forschungsinteressen zählen hierarchische Modelle im Bereich Computer Vision aber auch grundlegende Konzepte im Bereich des maschinellen Lernens und der künstliche Intelligenz.

Dynamisches Resourcen Scheduling auf Grafik-Prozessoren*

Markus Steinberger
steinberger@icg.tugraz.at

Abstract: Das Ziel dieser Arbeit ist die Rechenleistung moderner Grafik Prozessoren einer breiteren Klasse von Algorithmen zur Verfügung zu stellen. Dazu wird ein neues Ausführungsmodell präsentiert, welches die effiziente Abarbeitung von Algorithmen mit inhomogenem, zeitlich veränderlichem Parallelismus ermöglicht. Ergebnis unserer Forschung ist nicht nur der zurzeit schnellste Warteschlangenalgorithmus für Grafikprozessoren, der effizienteste Speicherverwalter für massiv parallele Architekturen und der momentan einzige autonome Scheduler auf Grafik Prozessoren mit Unterstützung verschiedener Granularitäten von Parallelismus, sondern auch eine Weiterentwicklung des momentanen Standes der Technik in den Bereichen Bildsynthese, Visualisierung und geometrischer Algorithmen.

1 Einleitung

Innerhalb der letzten Jahre hat ein Paradigmenwechsel im Bereich der Computertechnologie begonnen. Da die Chipproduktion an die physikalische Grenzen gestoßen ist, ist eine Erhöhung der Prozessortaktraten nicht mehr gewinnbringend. Der einzige Ausweg, um die weiterhin steigende Nachfrage nach immer mehr Rechenleistung bedienen zu können, ist Parallelisierung. Der Beginn der parallelen Verarbeitung wurde mit der Produktion der ersten Zweikern-Prozessoren am Beginn des 21. Jahrhunderts eingeleitet. Während diese ersten Mehrkern-Prozessoren ihre Rechenleistung einfach damit erbrachten zwei Anwendungen gleichzeitig auszuführen, sind solch einfache Strategien heutzutage nicht mehr ausreichend, um die volle Rechenleistung moderner paralleler Chips voll auszureizen. Jegliche Algorithmen müssen von Grund auf darauf ausgerichtet sein, gleichzeitig von einer Vielzahl an Prozessoren abgearbeitet zu werden.

Der aktuell leistungsfähigste parallele Chip ist die Graphics Processing Unit (GPU, dt. Grafikprozessor). Sie spiegelt den Höhepunkt der Entwicklung im Bereich energieeffizienter, hoch-paralleler Rechenleistung wider. Auf einer aktuellen GPU finden sich 3000 individuelle Rechenkerne, deren effiziente Nutzung für eine Vielzahl an Forschungsgebieten essentiell geworden ist. Simulationen in diversen Gebieten wie Genetik, Molekularbiologie, Weltraumforschung, Archäologie, Neurowissenschaften oder Medizin sind Paradebeispiele für den Einsatz von Grafikprozessoren. Und obwohl die Nutzung der GPU für diese Gebiete von essentieller Wichtigkeit ist, gestaltet sich ihr Einsatz in vielen Bereichen als schwierig. Eine Portierung bestehender Software auf die GPU erreicht meist nur einen

*Englischer Titel der Dissertation: “Dynamic Resource Scheduling on Graphics Processors”

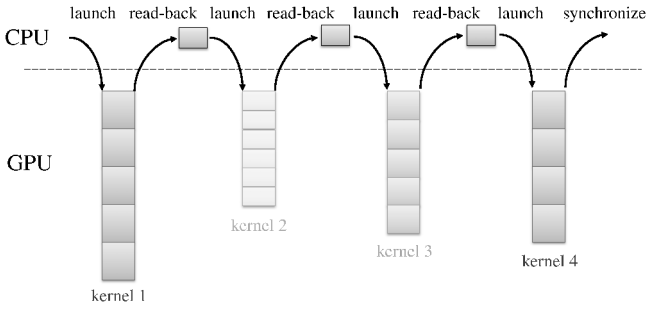


Abbildung 1: Im traditionellen Kernel-basierten Programmiermodell wird jeder Algorithmus in Kernel unterteilt. Kernel werden von der CPU aus kontrolliert, was zu einem ständigen Hin und Her zwischen CPU und GPU führt.

Bruchteil der Leistung. Die Gründe für diese Problematik finden sich im Programmier- und Ausführungsmodell von Grafikprozessoren.

Die vorliegende Arbeit beschäftigt sich mit dieser Problematik und zeigt wie neue dynamische Schedulingstrategien, welche speziell für die Ausführung auf massiv parallelen Systemen entworfen wurden, fähig sind eine unflexible GPU in eine parallele Rechenmaschine zu verwandeln, die sich an hochdynamische Algorithmen anpasst und somit die wahre Rechenleistung von Grafikchips für eine Vielzahl an Applikationen freilegt. Durch die Fortschritte, die in dieser Arbeit diskutiert werden, ist es nicht nur möglich eine neue Klasse von Algorithmen auf der GPU auszuführen, sondern auch traditionelle GPU-Algorithmen effizienter zu gestalten. Die Details der Arbeit finden sich in der Langschrift der Dissertation [Ste13], sowie in ausgewählten Publikationen in Fachzeitschriften [SKKS12, SKK+12, SKK+14a, SKK+14b].

1.1 Einschränkungen des aktuellen Ausführungsmodells

Um die Problematik hinter der Ausführung vor Algorithmen auf der GPU zu analysieren, ist es essentiell die Unterschiede zwischen der Central Processing Unit (CPU, dt. Hauptprozessor) und der Architektur eines Grafikchips zu beleuchten. Während es üblich ist Routinen zu schreiben, die jeweils einen Thread (Ausführungsstrang) auf der CPU darstellen, wird eine Prozedur auf der GPU – auch Kernel genannt – von tausenden Threads gleichzeitig ausgeführt. Während all diese Threads beginnen denselben Code auszuführen, steht es ihnen frei verschiedene Ausführungspfade zu wählen. Jedoch muss der Programmierer dabei die zusätzlichen Einschränkungen der Grafikchiphardware in Betracht ziehen. Eine GPU besteht aus einer Vielzahl an Multiprozessoren, welche Threads in kleinen Gruppen – in der Größe von z. B. 32 Threads – gemeinsam ausführen. Dieses Prinzip nennt sich Single Instruction Multiple Data (SIMD). Eine effiziennte Ausführung ist nur dann gewährleistet, wenn alle Threads innerhalb einer SIMD-Gruppe die gleichen Ausführungspfade wählen.

Expliziter Parallelismus Die erste Einschränkung des traditionellen GPU-Ausführungsmodells ist, dass ausreichend Parallelismus in jedem Abschnitt bzw. Kernel des Algorithmus zur Verfügung stehen muss. Die Anzahl an Threads, die für einen Kernel gestartet werden, muss fixiert werden bevor der Kernel ausgeführt wird. Eine dynamische Anpassung des Parallelismus ist während der Ausführung nicht möglich. Aus der Perspektive der Hardware würde es genügen, wenn eine ausreichende Anzahl an Gruppen von kohärenten Threads zur Verfügung stünde. Das Ausführungsmodell diktiert jedoch, dass tausende Threads gleichzeitig gestartet werden müssen, um eine gute Leistung zu erzielen. Dies limitiert die Anzahl an Algorithmen, die auf Grafikchips zur Ausführung gebracht werden können.

Kontrollflussänderung Die zweite Einschränkung für die Benutzung von Grafikchips für allgemeine Programmierung ergibt sich aus der Tatsache, dass jegliche Ausführung vom Hauptprozessor aus kontrolliert wird, siehe Abbildung 1. Um die Ausführung eines Kernels zu starten, muss der Hauptprozessor ein Kommando an die GPU schicken. Das Starten eines Kernels erzeugt Kosten. Zwischen zwei Kernels kann sich die Auslastung der GPU wesentlich verschlechtern. Der Datentransfer zwischen Kernels ist nur über den Hauptspeicher des Grafikchips möglich, welcher um einen Faktor 1000 langsamer ist als der lokale Speicher. In einigen Fällen ist es sogar notwendig Resultate eines Kernels von der GPU zur CPU zu übertragen. Dieser Vorgang dauert sehr lange und kann zur kompletten Inaktivität der GPU führen.

Beeinflussung der Ausführung Die dritte Einschränkung ist die Tatsache, dass es unmöglich ist, die Ausführung von Kernels zu beeinflussen, nachdem diese gestartet wurden. Der auf Grafikchips verbaute Scheduler bearbeitet Kernels nach einem einfachen first-in-first-out (FIFO) System. Daher können arbeitsintensive Hintergrundaufgaben die GPU blockieren, während hoch priorisierte Vordergrundaufgaben unvorhersehbar verzögert werden. Da die GPU keine Prioritäten unterstützt, ist es nicht möglich Aufgaben mit verschiedenen Charakteristiken gleichzeitig auszuführen.

Dynamische Speicherverwaltung Die vierte Einschränkung ist durch die Abwesenheit einer effizienten dynamischen Speicherverwaltung gegeben. Dynamische Speicherverwaltung ist ein essentieller Bestandteil jedes Betriebssystems, und praktisch jedes Computerprogramm benötigt diese Fähigkeit, um auf Eingaben reagieren zu können. Die einzige dynamische Speicherverwaltung, die es bis jetzt auf Grafikprozessoren gibt, wird von NVIDIA im CUDA Toolkit [NVI13] bereitgestellt. Der zur Verfügung gestellte Speicherverwalter ist langsam, unzuverlässig und skaliert schlecht für eine große Anzahl an Threads.

Kenntnis von Zeit Die fünfte Einschränkung ist die Tatsache, dass die GPU keine Kenntnis von Zeit zulässt. Grafikprozessoren stellen weder eine gemeinsame Zeitquelle zwischen den einzelnen Kernen noch zwischen CPU und GPU zur Verfügung. Somit ist es unmöglich einen Algorithmus an die bereits verstrichene Zeit oder die noch zur Verfügung stehende Zeit anzupassen. Die Kenntnis von Zeit ist für viele Problemstellungen, im Speziellen für Echtzeitanwendungen, welche sich an Deadlines orientieren müssen, äußerst wichtig.

1.2 Ziele der Arbeit

Diese Arbeit bietet Lösungen für alle zuvor aufgelisteten Probleme des aktuellen GPU-Schedulingmodells. Die Ziele der Arbeit lassen sich in sechs Punkten zusammenfassen:

O1 Mit Hilfe neuartiger Techniken wird eine autonome Ausführung von dynamischen Algorithmen mit Höchstleistung auf Grafikprozessoren ermöglicht.

O2 Aufgaben mit variabler Parallelität lassen sich effizient in massiv parallelen Umgebungen wie z. B. auf Grafikprozessoren ausführen.

O3 Eine dynamische Speicherverwaltung, welche Anfragen von zehntausenden Threads gleichzeitig erfüllen kann, wird ermöglicht.

O4 Das Scheduling auf Grafikprozessoren wird kontrollierbar und lässt sich an dynamisch veränderliche Bedingungen anpassen.

O5 Ein autonomes Scheduling kann für die Grafikhardware suboptimale Ausführungscharakteristika erkennen und effiziente Konfigurationen herstellen.

O6 Ein Schedulingsystem, das **O1**-**O5** erfüllt, erweitert die Bandbreite an Algorithmen, die auf massiv paralleler Hardware ausführbar sind.

Die Zielvorgaben werden dann als erfüllt angesehen, wenn sie sowohl in fundierten Teiltests als auch praktischen Anwendungen signifikante Vorteile gegenüber dem traditionellen GPU-Ausführungsmodell zeigen und die Umsetzung von Algorithmen auf der GPU erlauben, welche zuvor nicht für eine Ausführung auf Grafikprozessoren geeignet schienen.

2 Scheduling Modell

Zur Umsetzung neuartiger Schedulingstrategien auf Grafikprozessoren ist es zunächst notwendig, das vorherrschende Programmiermodell (das sogenannte Flussverarbeitungsmodell) durch ein flexibleres Modell zu ersetzen. Im Unterschied zum Flussverarbeitungsmodell geht das von uns vorgestellte Modell – Softshell – nicht davon aus, dass alle Daten vor der Ausführung im Speicher vorliegen, sondern sieht Algorithmen vielmehr als dynamische, sich anpassende, datenabhängige Ausführungspfade. Um dieses Modell umzusetzen, definieren wir fünf Basisbausteine:

- **Arbeitselemente** beschreiben Daten, welche von einem Thread bearbeitet werden.
- **Arbeitsfragmente** beschreiben Daten, welche von einer kleinen Gruppe von Threads gemeinsam bearbeitet werden.
- **Arbeitspakete** beschreiben Daten, welche von größeren Threadgruppe bearbeitet werden. Zusätzlich muss es zwischen diesen Threads Kommunikationskanäle geben.
- **Prozeduren** beschreiben die Arbeitsschritte, die für ein Arbeitselement, eine Arbeitsfragment oder ein Arbeitspaket ausgeführt werden sollen.
- **Ereignisse** werden von Benutzereingaben ausgelöst und initiieren die Ausführung von Algorithmen, indem sie Arbeitselemente, -fragmente oder -pakete generieren.

Mit Hilfe dieser fünf einfachen Bausteine ist es nicht nur möglich jeden Algorithmus des Flussverarbeitungsmodells abzubilden, sondern eine wesentlich größere Klasse an parallelen Algorithmen zu beschreiben. Durch den Einsatz von Softshell wird jede noch so kleine Ansammlung von Parallelismus darstellbar und kann an ein Schedulingsystem weitergereicht werden.

3 GPU-Schedulingalgorithmen

Zur Umsetzung aller Ziele der Arbeit ist es notwendig eine Reihe von grundlegenden Algorithmen auf massiv parallelen Architekturen umzusetzen. Die für die Arbeit wichtigsten Erneuerungen lassen sich in vier Kategorien unterteilen: Warteschlangenalgorithmen, Arbeitspaketprioritäten, dynamische Speicherverwaltung und Ausführung von heterogenen Prozeduren.

Parallele Warteschlangen Den Grundstein vieler Schedulingsysteme bilden Warteschlangenalgorithmen. Eine Warteschlange erlaubt es Arbeitselemente, -fragmente, und -pakete zu sammeln, zu kombinieren und zur Ausführung zu bringen. Bisherige Warteschlangen für parallele Architekturen sind immer davon ausgegangen, dass sogenannte *nicht blockierende* Algorithmen die beste Leistung erzielen. In dieser Arbeit zeigen wir, dass Warteschlangenalgorithmen, die in die Kategorie von *blockierenden* Algorithmen fallen, ihren *nicht blockierenden* Gegenstücken weitaus überlegen sein können. Unsere Warteschlange ist dem bisherigen Stand der Technik weitaus überlegen und kann somit als die schnellste Warteschlange für massiv parallele Architekturen angesehen werden.

Dynamische Arbeitspaketprioritäten Einer der wichtigsten Punkte des vorgestellten Schedulingsystems sind Prioritäten. Mit Hilfe von feingranularen Prioritäten auf dem Level von individuellen Arbeitspaketen ist es möglich, eine Vielzahl an Schedulingstrategien umzusetzen. Leider ist es nicht möglich Prioritätswarteschlangen basierend auf sortierten Listen oder Halden zu verwenden, da die Synchronisationskosten auf Grafikprozessoren zu hoch sind. Deshalb präsentieren wir einen adaptiven, progressiven, nicht invasiven Warteschlangensortierungsansatz, der jegliche Synchronisation mit dem Einfüge- und Entfernungsvorgängen der Warteschlangen vermeiden kann.

Dynamische Speicherverwaltung auf massiv parallelen Prozessoren Während dynamische Speicherverwaltung auf Hauptprozessoren ein gut studiertes Gebiet darstellt, lässt sich diese Forschung nicht direkt auf Grafikprozessoren übertragen, da deren Architektur zu unterschiedlich ist. Um Speicheranfragen von zigtausenden von Threads effizient bearbeiten zu können, schlagen wir einen neuen dynamischen Speicherverwalter für massiv parallele Architekturen vor: ScatterAlloc. ScatterAlloc vermeidet Kollisionen und somit Synchronisationspunkte, indem es Anfragen über eine Hashfunktion an verschiedene Stellen im Speicher verteilt. Durch eine geschickte Wahl dieser Hashfunktion ist es nicht nur möglich

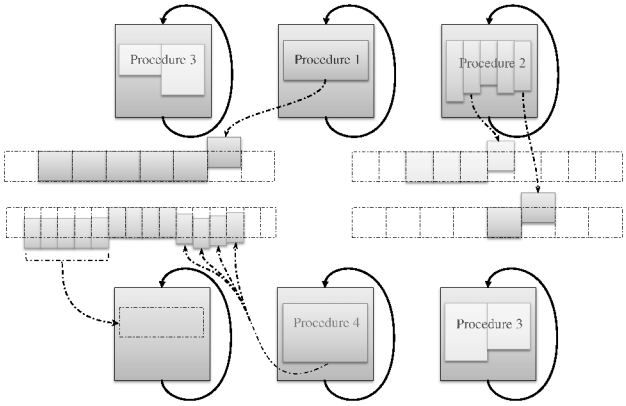


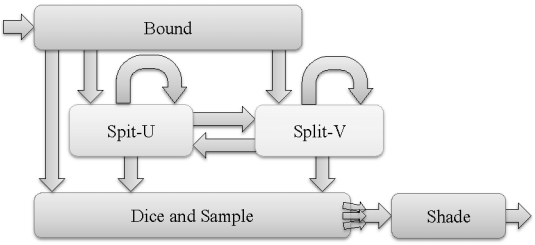
Abbildung 2: Unser Versatile Megakernel passt sich dynamisch an die ankommenden Arbeitspakete an und ermöglicht somit ein Maximum an aktiven Threads.

Anfragen zu verteilen, sondern auch Distanzen zwischen benachbarten Threads zu optimieren, um Speicherzugriffsmuster zu erzeugen, die auf Grafikhardware effizient abgearbeitet werden können. Durch dieses sorgfältige Design stellt ScatterAlloc den zurzeit effizientesten dynamischen Speicherverwalter für Grafikprozessoren dar und liegt sogar einige Größenordnungen vor den kommerziellen Implementierungen der Grafikkartenhersteller.

Ausführung von heterogenen Prozeduren Um die Rechenkraft von Grafikchips auch dynamischen Algorithmen mit verteiltem Parallelismus zur Verfügung stellen zu können, ist es essentiell, dass jegliche Teile eines Algorithmus mit höchster Rechenleistung ausgeführt werden können. Deshalb stellen wir einen Verarbeitungs- und Schedulingalgorithmus vor, welcher die zur Verfügung stehenden Ressourcen dynamisch verteilt und sich selbst an die Charakteristika der zur Ausführung eingereihten Arbeitspakte anpasst, siehe Abbildung 2. Wegen seiner Anpassungsfähigkeit nennen wir diesen Algorithmus *Versatile Megakernel*. Mit Hilfe eines einzigen Kernelstarts übernimmt der Versatile Megakernel die Kontrolle über die GPU und startet eine Maximalanzahl an Arbeiterthreads. Jede Gruppe von Arbeiterthreads lädt Arbeitseinheiten aus der globalen Warteschlange und stellt den ankommenden Arbeitseinheiten die benötigten Ressourcen zur Verfügung.

4 Anwendungsbeispiele

Die vorgestellten Schedulingalgorithmen zeigen nicht nur in synthetischen Tests hervorragende Leistungen, sondern auch erhöhte Leistungspotentiale in einer Vielzahl an Anwendungsbeispielen. Im folgenden Abschnitt zeigen wir eine Auswahl an Algorithmen, die von diesen neuen Schedulingalgorithmen profitieren bzw. sich erst mit Hilfe dieser Algorithmen auf Grafikprozessoren umsetzen lassen.



(a) Die Reyes Pipeline lässt sich mit fünf Prozeduren modellieren. Die Charakteristiken der Prozeduren unterscheiden sich stark mit 1 bis 256 aktiven Threads und 16 bis 63 benötigten Registern.

(b) Der Killeroo Datensatz besteht aus 11532 Eingabeprimitiven, die in Echtzeit in 99 Millionen Mikroflächen umgesetzt werden.

Echtzeit Reyes Pipeline Die Reyes Pipeline [CCC87] wird hauptsächlich in Filmproduktionen eingesetzt, um hochqualitative Bilder zu erzeugen. Eine Implementierung auf Grafikprozessoren gestaltet sich schwierig, da die Pipeline rekursiv, irregulär, und expandierend ist. Die grundlegende Idee hinter Reyes ist die Unterteilung der Szenengeometrie in eine Vielzahl an Mikroflächen. Für den gesamten Vorgang werden normalerweise vier bis sechs Abschnitte benötigt, welche sich, wie in Abbildung 3a dargestellt, in Softshell umsetzen lassen. Bisherige Ansätze versuchten die Pipeline in mehrere Kernel aufzuteilen [PO08, ZHR+09, TPO10]. Mit unseren Schedulingalgorithmen wird die gesamte Pipeline als eine Ausführungseinheit auf dem Grafikprozessor abgearbeitet. Für das Beispiel in Abbildung 3b werden aus 11532 Eingangsprimitiven mehr als eine Million Arbeitsbeschreibungen generiert und zur Ausführung gebracht. Temporär liegen dabei 99 Millionen Mikroflächen vor. Auf einem aktuellen Grafikprozessor (NVIDIA GTX Titan) benötigt diese Simulation dank der vorgestellten Schedulingalgorithmen nur 25ms.

Bildbasiertes Rendering mit Kamerasynchronisation Mit Hilfe des sogenannten Image-based Visual Hull (IBVH) Algorithmus lassen sich Tiefenbilder für beliebige virtuelle Kamerapositionen aus einer Reihe von segmentieren Eingabebildern erzeugen [MBR+00]. Jedoch ist der Algorithmus meist noch zu aufwendig, um eine komplette Szene in Echtzeit zu rekonstruieren. Es ist jedoch möglich, statische Teile einer Szene mit geringeren Frequenzen zu rekonstruieren und somit Rechenzeit einzusparen. Die zu rekonstruierenden Teile der Szene werden normalerweise über einen Schwellenwert definiert. Dieser Schwellenwert beeinflusst die Ausgangsqualität direkt, die Ausführungszeit jedoch nur indirekt, welche mit der Aufnahmegeschwindigkeit der Kameras synchronisiert werden muss. Mit Hilfe der vorgestellten Schedulingalgorithmen ist dies jedoch leicht realisierbar. Jedem Bereich wird anhand bestimmter Qualitätsmerkmale eine Priorität zugewiesen. Der Scheduler rekonstruiert jene Bereiche mit der höchsten Priorität vor den übrigen. Sollte der nächste Synchronisationspunkt näher rücken, lässt sich der IBVH Algorithmus durch eine einfache Wiederverwendung der älteren Daten austauschen. Somit ist es möglich, dynamisch Qualität gegen Rechenzeit zu tauschen und gleichzeitig die bestmögliche Bildqualität zu erhalten, siehe Abbildung 4.

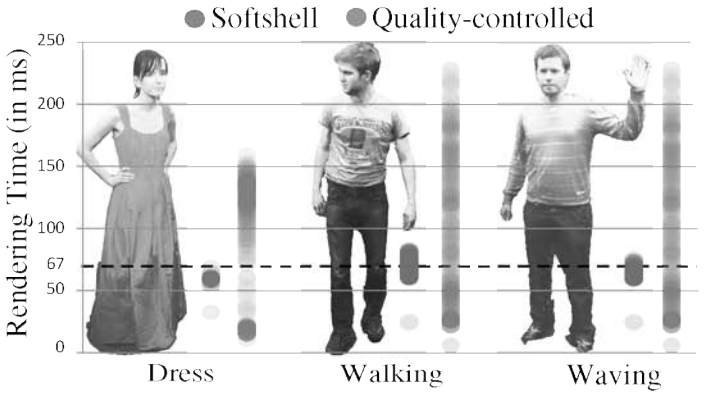


Abbildung 4: Image-based Visual Hull Rendering mit einer durch die Kameraaufzeichnungsrate vorgegebenen Ausführungszeit von $67ms$. Während mit einer qualitätsgesteuerten Anpassung des Algorithmus (rot) die Zielvorgabe (strichliert) nicht eingehalten werden kann, ist es mit dem Schedulingalgorithmus (blau) leicht möglich die Vorgaben zu erfüllen und gleichzeitig die bestmögliche Qualität zu erzielen.

Prozedurale Modellierung Mit Hilfe von prozeduraler Modellierung lassen sich komplette Welten basierend auf einem relativ einfachen Regelwerk erzeugen. Die Berechnung solcher Städte kann jedoch Stunden dauern. Da Designer einzelne Parameter der Städte in einem interaktiven Prozess anpassen müssen, führt die lange Rechendauer zu einem sehr ineffizienten Prozess. Da die für solch prozedurale Modellierung zum Einsatz kommenden Grammatiken sehr komplex sind, galt eine umfassende Abbildung dieser Regeln auf Grafikprozessoren bisher als unmöglich. Alle bisherigen Versuche beschnitten die Grammatiken stark, um Parallelität zu erzeugen und erzielten selbst für diese simplen Varianten nur geringe Vorteile im Vergleich zu CPU-Grammatiken. Mit Hilfe unserer Schedulingstrategien ist es möglich, die aktuell komplexeste Grammatik (CGA [MWH+06]) auf Grafikprozessoren abzubilden. Da die gesamte Grammatikableitung nun auf der Grafikhardware durchgeführt wird, kann die Ableitung mit der Darstellung verknüpft werden, um nur jene Teile einer Welt zu erzeugen, die auch dargestellt werden sollen. Durch diese Kombination generieren wir detaillierte Metropolen in Echtzeit, während sich ein Betrachter mit Überschallgeschwindigkeit durch die Stadt bewegen kann. Eine Beispielstadt ist in Abbildung 5 zu sehen. Mit traditionellen Techniken würde die Erzeugung dieser Stadt drei Stunden benötigen, während wir diese 20 mal innerhalb einer Sekunde neu generieren.

5 Schlussbemerkung

Das in dieser Arbeit vorgestellte GPU-Programmier- und Ausführungsmodell ist darauf ausgerichtet, verteilten und zeitveränderlichen Parallelismus nutzbar zu machen. Den Grundstein für dieses neuartige Ausführungsmodell bildet der zurzeit schnellste Warte-

Abbildung 5: Hochdetaillierte Städte mit komplexen Gebäuden können durch den Einsatz unserer Schedulingalgorithmen in Echtzeit auf einer GPU generiert werden. Die sichtbaren $28km^2$ dieser Stadt beinhalten $47\,000$ Gebäude, welche von 240 Millionen Regeln generiert werden. Mit Hilfe des vorgestellten Algorithmus generieren wir die Geometrie zur Darstellung 20 mal pro Sekunde. Somit ist es sogar möglich sich mit Überschallgeschwindigkeit durch die Stadt zu bewegen.

schlangenalgorithmus für Grafikprozessoren. Wir weisen jedem Thread eine individuelle, separat geschützte Warteschlangenposition zu, womit unser Algorithmus Anfragen in quasi konstanter Zeit beantworten kann – unabhängig von der Anzahl an Threads die gleichzeitig Zugriff suchen. Durch die Kombination mit unserem Versatilen Megakernel ist es nun möglich, Algorithmen mit diversen Ausführungscharakteristiken und unterschiedlichen Graden an Parallelismus effizient zur Ausführung zu bringen und die GPU-Auslastung auf ein Maximum zu erhöhen. Um Algorithmen die Möglichkeit zu geben sich an veränderliche Eingabeparameter anzupassen, haben wir den zurzeit schnellsten dynamischen Speicherverwalter für massiv parallele Architekturen entwickelt. Dieser Speicherverwalter reduziert die Anzahl an Zugriffsversuchen auf dieselbe Speicherstelle, indem er die Zugriffe von individuellen Threads über eine intelligente Hashfunktion verteilt. Um die Ausführung auf Grafikprozessoren beeinflussbar zu machen, führen wir feingranulare Prioritäten ein, mit deren Hilfe sich die Ausführungsreihenfolge von Arbeitspaketen sowie kompletten Algorithmen steuern lassen. Diese Prioritäten lassen zum ersten Mal Schedulingstrategien wie z. B. faires Scheduling, zeitanteilgesteuertes Scheduling, oder Earliest-Deadline-First Scheduling auf Grafikchips zu.

Da der Grad an paralleler Verarbeitung in Zukunft noch weiter ansteigen wird, ist es von essentieller Wichtigkeit genügend Parallelismus innerhalb von Algorithmen zur Verfügung zu stellen. Grundsätzlich gibt es drei Möglichkeiten zukünftige Generationen von Grafikchips mit ausreichend Parallelismus zu versorgen: Durch algorithmische Anpassungen kann ein größerer Anteil an Parallelismus freigelegt werden, mehrere Algorithmen können gleichzeitig ausgeführt werden, oder das Scheduling kann angepasst werden, sodass die Hardware die Parallelität, die in einem Algorithmus vorhanden ist, voll ausnutzen kann. In allen drei Fällen sind die in dieser Arbeit beschriebenen Ausführungs- und Programmiertechniken von großem Nutzen. Daher kann davon ausgegangen werden, dass diese Arbeit eine wichtige Rolle im Design zukünftiger GPU-Programmiersprachen, -Compiler und -Scheduler spielen wird.

Literatur

[CCC87] Robert L. Cook, Loren Carpenter, and Edwin Catmull. The Reyes image rendering architecture. *SIGGRAPH Comput. Graph.*, 21(4):95–102, August 1987.

[MBR+00] Wojciech Matusik, Chris Buehler, Ramesh Raskar, Steven J. Gortler, and Leonard McMillan. Image-based visual hulls. In *Proc. SIGGRAPH '00*, SIGGRAPH '00, pages 369–374. ACM, 2000.

[MWH+06] Pascal Müller, Peter Wonka, Simon Haegler, Andreas Ulmer, and Luc Van Gool. Procedural Modeling of Buildings. *ACM Trans. Graph.*, 25(3):614–623, 2006.

[NVI13] NVIDIA. *NVIDIA CUDA Compute Unified Device Architecture - Programming Guide*. NVIDIA, 2013.

[PO08] Anjul Patney and John D. Owens. Real-time Reyes-style adaptive surface subdivision. *ACM Trans. Graph.*, 27(5):143:1–143:8, December 2008.

[SKK+12] Markus Steinberger, Bernhard Kainz, Bernhard Kerbl, Stefan Hauswiesner, Michael Kenzel, and Dieter Schmalstieg. Softshell: Dynamic Scheduling on GPUs. *ACM Transactions on Graphics (TOG)*, 31(6):161, 2012.

[SKK+14a] Markus Steinberger, Michael Kenzel, Bernhard Kainz, JÃűrg MÃijller, Peter Wonka, and Dieter Schmalstieg. Parallel Generation of Architecture on the GPU. *Computer Graphics Forum*, 33(2):73–82, 2014.

[SKK+14b] Markus Steinberger, Michael Kenzel, Bernhard Kainz, Peter Wonka, and Dieter Schmalstieg. On-the-fly Generation and Rendering of Infinite Cities on the GPU. *Computer Graphics Forum*, 33(2):105–114, 2014.

[SKKS12] Markus Steinberger, Michael Kenzel, Bernhard Kainz, and Dieter Schmalstieg. ScatterAlloc: Massively parallel dynamic memory allocation for the GPU. In *Innovative Parallel Computing (InPar), 2012*, pages 1–10, 2012.

[Ste13] Markus Steinberger. *Dynamic Resource Scheduling on Graphic Processors*. PhD thesis, Graz University of Technology, 2013.

[TPO10] Stanley Tzeng, Anjul Patney, and John D. Owens. Task management for irregular-parallel workloads on the GPU. In *Proc. High Performance Graphics*, HPG '10, pages 29–37, 2010.

[ZHR+09] Kun Zhou, Qiming Hou, Zhong Ren, Minmin Gong, Xin Sun, and Baining Guo. RenderAnts: interactive Reyes rendering on GPUs. *ACM Trans. Graph.*, 28(5):155:1–155:11, December 2009.

Markus Steinberger ist Universitätsassistent in der Gruppe von Dieter Schmalstieg am Institut für Maschinelles Sehen und Darstellen an der Technischen Universität Graz in Österreich. Er hat seine Doktorarbeit in Graz im Jahr 2013 mit Auszeichnung abgeschlossen, was ihm die Ehre der Promotion unter den Auspizien des Bundespräsidenten einbrachte. Von 2013 bis 2014 arbeitete er in der Forschungsgruppe von Kari Pulli bei Nvidia in Kalifornien.

Seine Arbeiten wurden mit Best Paper Awards bei den Konferenzen NPAR'11 und Infovis'11, sowie Honorable Mention Awards bei CHI'11, Eurographics'14, sowie CHI'14 ausgezeichnet. Er ist in mehreren wissenschaftlichen Organisationen tätig, sitzt in Programmkomitees und ist Gutachter für top Journals und Konferenzen. Seine wissenschaftlichen Interessen liegen im Bereich GPU Scheduling, Scientific Visualization, Information Visualization, und Procedural Geometry.

Visual Analytics von Mustern in hochdimensionalen Daten*

Andrada Tatu

tatu@dbvis.inf.uni-konstanz.de

Abstract: Der tägliche technologische Fortschritt ermöglicht die Speicherung und Verarbeitung riesiger Datenmengen. Die Verwendung solcher Datenarchive bringt neue Herausforderungen an Analysetechniken mit sich. Die große Anzahl der Datenpunkte sowie der beschreibenden Attribute (Dimensionen) erschweren die Datenanalyse. Diese Arbeit widmet sich numerischen hochdimensionalen Daten, die durch den *Fluch der Dimensionalität* und der exponentiellen Anzahl von Attributkombinationen neue Methoden der Analyse erforderlich machen. Mithilfe visueller Analysemethoden werden hier Qualitätsmaße vorgeschlagen, um Visualisierungen zu bewerten, die wichtige Strukturen aufweisen. Ebenso werden automatisch-interaktive Verfahren vorgestellt, die die Suche nach interessanten Mustern unterstützen.

1 Einleitung

Bedingt durch den technologischen Fortschritt der letzten Jahrzehnte sind kommerzielle Applikationen heute in der Lage, riesige Datenmengen zu erzeugen, zu speichern und zu verarbeiten. Diese Entwicklung beeinflusst auch die Natur der erzeugten Daten, d.h. dass für jeden Dateneintrag unterschiedliche Aspekte in der gleichen Datenbank gespeichert werden. Oft sind die beschreibenden Attribute numerisch. Datensätze, die mehr als fünf solcher Attribute (Dimensionen) beinhalten, nenne ich hochdimensional.

Meine Dissertation [Tat13] behandelt die Fragestellung, wie interessante Muster (bedeutende Informationen) in hochdimensionalen Räumen gefunden werden können. Diese Aufgabenstellung ist durch das Problem des *Fluches der Dimensionalität* äußerst herausfordernd. Dieses Problem besagt, dass Daten im hochdimensionalen Raum spärlich vorkommen.

Herkömmliche Analysetechniken werden dadurch beeinträchtigt. Automatische Methoden müssen die Datenkomplexität nicht nur im Hinblick auf ihre Laufzeit, sondern auch auf ihre Berechnungsfunktionen (z.B. Distanzfunktionen) einbeziehen. Außerdem wird die visuelle Exploration dieser Daten durch die Zweidimensionalität der Darstellungen beeinträchtigt. Die Herausforderung des Informationsgewinnungsprozesses aus hochdimensionalen Daten besteht darin, die Dimensionalität der Daten geschickt zu reduzieren – ein Prozess der auch für die Visualisierung dieser Daten ausschlaggebend ist.

*Englischer Titel der Dissertation: “Visual Analytics of Patterns in High-Dimensional Data”

Eine Möglichkeit der Dimensionsreduzierung ist die Selektion von Dimensionen. Die Visualisierung dieser Daten zur Informationsgewinnung beinhaltet, dass eine geschickte Abbildung zwischen Datendimensionen und Visualisierungsmerkmalen gefunden wird. Somit können hochdimensionale Daten auf den 2D Raum projiziert werden und weitere Merkmale auf weitere visuelle Attribute wie Farbe, Position, Richtung, etc. abgebildet werden. So kann beispielsweise ein zehndimensionaler Datensatz in einem Scatterplot dargestellt werden, indem jeweils zwei Dimensionen auf der x- bzw. y-Achse abgebildet werden und die dritte Dimension die Farbe und die vierte die Größe der Punkte bestimmt (siehe Abbildung 1). Somit ergeben sich für diesen Datensatz für die Selektion von vier Attributen und für die Darstellung mit vier visuellen Parametern über 5000 mögliche Abbildungen[1].

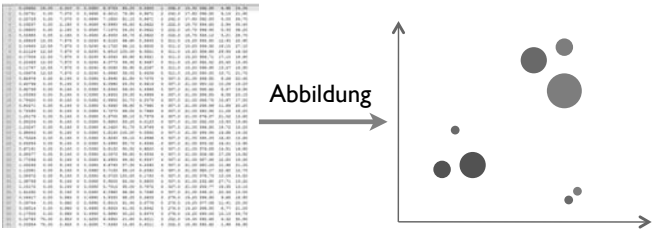


Abbildung 1: Ein 10 dimensionaler Datensatz hat über 5000 mögliche Abbildungen mit vier visuellen Parametern.

Eine manuelle Durchsuchung dieser Darstellungsräume ist nicht möglich. Meine Dissertation beruht auf dem Konzept, dass die Suche nach interessanten Mustern in hochdimensionalen Datenmengen mit einem kombinierten Ansatz von automatischen, visuellen und interaktiven Methoden durchgeführt werden kann.

2 Qualitätsmaße zum Bewerten von Mustern in hochdimensionalen Räumen

Die hohe Anzahl an möglichen Abbildungen von hochdimensionalen Datensätzen erfordert eine automatische Durchsuchung der Darstellungen. Die Ausprägung der Muster einer Visualisierung kann durch sogenannte Qualitätsmaße gemessen werden. Durch diese automatischen Funktionen kann die große Menge an hochdimensionalen Visualisierungen eingegrenzt und dem Benutzer eine ausgewählte Menge zur weiteren Untersuchung zur Verfügung gestellt werden. Ich schlage Qualitätsmaße für Scatterplots [CLN86] und Parallele Koordinaten [Ins85] vor – den am häufigsten verwendeten Visualisierungen für hochdimensionale Daten. Dabei werden nichtklassifizierte und klassifizierte Daten in Betracht gezogen. Diese Maße konzentrieren sich auf unterschiedliche Aufgaben, wie die Identifikation von Gruppen oder Korrelationen. Dabei werden unterschiedliche Zielfunktionen behandelt und die Ergebnisse abhängig davon sortiert. In Abbildung 2 werden die

[1] $\binom{10}{4} \cdot 4! = \frac{10!}{6!}$

Ergebnisse von drei solchen Zielfunktionen für parallele Koordinaten mit den jeweils zwei besten Ergebnissen links und der schlechtesten gefundenen Abbildung rechts dargestellt. Diese Methoden selektieren die besten vier Dimensionen, um diese Funktionen zu maximieren.

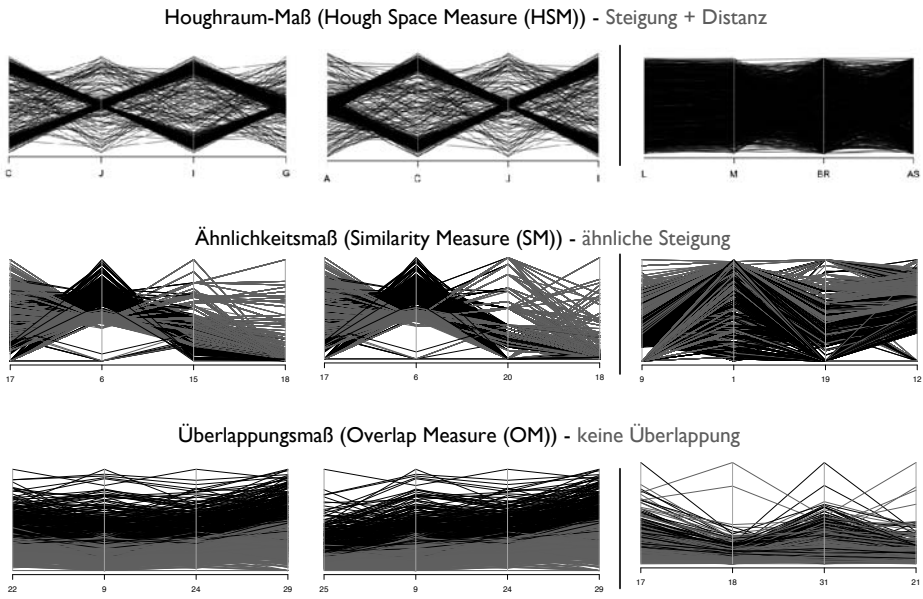


Abbildung 2: Qualitätsmaße für Parallele Koordinaten mit verschiedenen Zielfunktionen (blau) - jeweils die zwei Besten (links) und das schlechteste Ergebnis (rechts).

In Abbildung 3 werden die Ergebnisse der Qualitätsmaße für Scatterplots dargestellt. Dabei kann man auch erkennen, dass zur Identifikation von Gruppen unterschiedliche Kriterien für die automatische Suche eingesetzt werden können.

Um sicherzustellen, dass die automatischen Qualitätsmaße das menschliche Handeln widerspiegeln, werden diese Techniken (1) hinsichtlich ihrer Fähigkeit, Cluster in unterschiedlichen realen und synthetischen Datensätzen zu finden, und (2) hinsichtlich ihrer Korrelation mit der menschlichen Wahrnehmung untersucht. Der ausführlichen Diskussion dieser Resultate folgen Überlegungen für die zukünftige Forschung. An dieser Stelle möchte ich eine dieser Überlegungen ansprechen. Diese Studie untersucht, inwieweit die automatischen Qualitätsmaße die menschliche Wahrnehmung widerspiegeln, und zeigt deutliche Korrelationen der Maße mit dem menschlichen Handeln, wobei diese Wahrnehmung als Optimum angesehen wird. Diese Annahme erfordert weitere Untersuchungen, da automatische Methoden auch dafür eingesetzt werden können, Muster zu identifizieren, die vom menschlichen Auge schwer erfasst werden können, jedoch für die Datenanalyse von großer Bedeutung sein können.

Da dieses Thema in den letzten Jahren großes Interesse geweckt hat, sind viele verschiedene Qualitätsmaße für eine Reihe weiterer hochdimensionaler Visualisierungen entwickelt

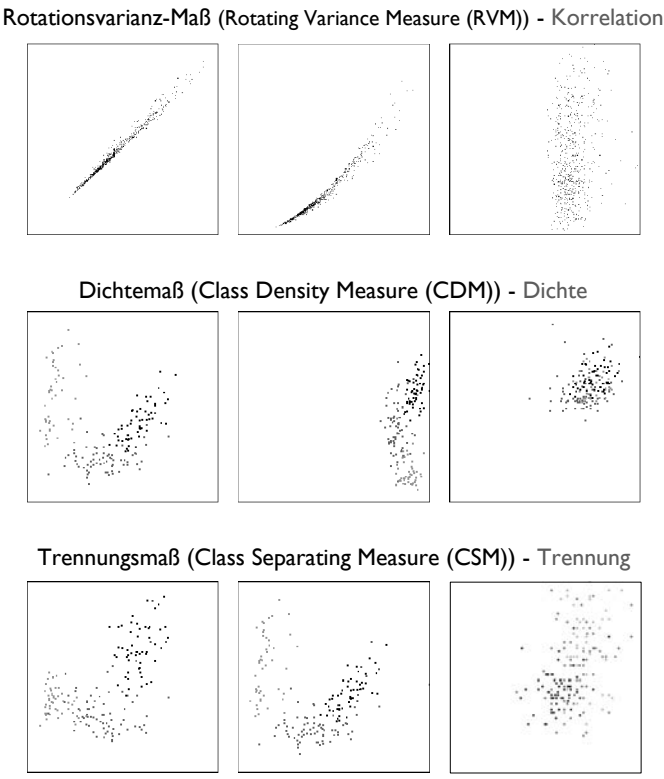


Abbildung 3: Qualitätsmaße für Scatterplots mit verschiedenen Zielfunktionen (blau) - jeweils die zwei Besten (links) und das schlechteste Ergebnis (rechts).

worden. Ich stelle in meiner Dissertation Vorschläge für deren Vernetzung und Weiterentwicklung vor. Hierfür wird eine Übersicht über die verschiedenen Ansätze erstellt, welcher eine Systematisierung zugrunde liegt, die aufgrund einer umfassenden Literaturauswertung zustande kam. Die Systematisierungsfaktoren sind in fünf Kategorien eingeteilt:

1. **Visualisierungsart** – z.B. Scatterplots, Parallele Koordinaten, Pixelvisualisierungen, Histogramme und andere.
2. **Was gemessen wird** – z.B. Korrelation, Gruppierung, Ausreißer, etc.
3. **Wo gemessen wird** – im Daten- oder Bildraum.
4. **Grund für die Messung** – z.B. die richtige Projektion zu finden, die richtige Anordnung der Koordinatenachsen, die richtige Abbildung von Datendimensionen auf visuellen Variablen, usw.
5. **Interaktionsmöglichkeit** – ob man mit den Maßen interagieren kann, z.B. durch die Auswahl von Parametern.

Das Hauptziel dieser Systematisierung ist, deskriptive Modelle zu bilden, um das Verständnis über dieses Gebiet zu erweitern, wie Qualitätsmaße heutzutage eingesetzt werden und wie sie eingesetzt werden könnten, um effektivere Visualisierungstechniken zu entwickeln.

3 Visuelle Suche von Mustern in Unterräumen

Nachdem sich der erste Teil meiner Arbeit der explorativen Aufgabe gewidmet hat, Muster in hochdimensionalen Räumen zu suchen und darzustellen, ist der Fokus im zweiten Teil auf die konkrete Aufgabe des Clustering gesetzt. Ein typisch angewandtes Paradigma für die automatische Mustersuche ist das Clustering, d.h. die Gruppierung von Punkten abhängig von ihrer Ähnlichkeit. Im hochdimensionalen Raum existieren manche Muster nur in verschiedenen Unterräumen des Datenraumes. Subspace Clustering Algorithmen [KKZ09] wurden entwickelt, um Unterräume zu finden, in denen Cluster existieren, die durch traditionelle Clustering Algorithmen nicht gefunden werden würden. Obwohl diese Algorithmen spärlich mit Daten besetzte, hochdimensionale Räume gut explorieren können, ist das Entwickeln von effektiven Visualisierungstechniken, um diese Clusteringresultate zu analysieren, nicht trivial. Zusätzlich zu der Clusterzugehörigkeit von Elementen müssen die relevanten Attributmengen eines Clusters und die Objekt- und Dimensionsüberlappungen von Subspaceclustern dargestellt werden (siehe Abbildung 4).

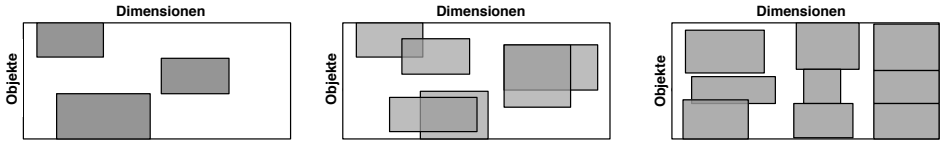


Abbildung 4: Mögliche Relationen von Clustern in hochdimensionalen Räumen, schematisch dargestellt. Jedes große Rechteck stellt eine Datentabelle dar und die blauen Rechtecke jeweils die Cluster in diesen Daten.

Auch wenn eine Reihe von Techniken für die Visualisierung von traditionellen Clustering Resultaten existiert (z.B. Scatterplots, Heatmaps, Dendrogramme, hierarchische Parallele Koordinaten), gibt es nur wenige Ansätze, um das Resultat von Subspace Clustering Algorithmen zu visualisieren. Außerdem wurden bisher erstaunlich wenige Ansätze vorgestellt, die eine visuelle Analyse der Subspace Clustering Ergebnisse unterstützen können, obwohl im Bereich der Subspace Clustering Algorithmen viel Forschung betrieben wurde. In meiner Dissertation stelle ich eine mögliche Visualisierungsumgebung für dieses Problem vor. Angemessene Visualisierungstechniken, die von speziellen Methoden zur Extraktion von Informationen unterstützt werden, würden nicht nur die Nachverfolgung der Clustering Ergebnisse ermöglichen, sondern auch Fachleuten dabei helfen, den Subspace Clustering Prozess so zu steuern, dass relevante Muster zum Vorschein kommen. Dieses Ziel vor Augen stelle ich in Abschnitt 3.1 ein Konzept vor, das Subspace Clustering Algorithmen mit interaktiven skalierbaren Visualisierungen kombiniert. Meine Ansätze widmen sich deshalb der Aufgabe der Visualisierung zum Vergleich von alternativen Clustergruppen, die durch Nutzerfeedback gesteuert werden.

3.1 Visuelle Analyse zum Auffinden von Alternativen Clustern

In den folgenden Abschnitten möchte ich einen interaktiven Ansatz für die Subspace-Analyse (Unterraum-Analyse) vorstellen. Ziel dieses Ansatzes ist es, verschiedene Aspekte und Muster in unterschiedlichen Subspaces (Unterräumen) hochdimensionaler Daten zu finden und deren Relationen zu verstehen.

Die Tatsache, dass interessante Muster in hochdimensionalen Räumen nur in Subspaces zu finden sind, erschwert die Suche nach diesen erheblich, da die Anzahl der Subspaces exponentiell zu der Anzahl der Dimensionen des Datenraumes ansteigt. Eine Suche in all diesen Subspaces ist somit zeitintensiv, daher praktisch nicht möglich. Ich schlage deshalb eine analytische Pipeline vor, die aus automatischen und visuellen Schritten aufgebaut ist und es ermöglicht, einzelne Sichten der Daten auszuwählen und zu analysieren.

Abbildung 5 verdeutlicht die Schritte dieser Pipeline. Die hochdimensionalen Daten werden mit einem Subspacesearch-Algorithmus durchlaufen, der vielversprechende Unterräume zur Mustersuche selektiert. Wegen der Redundanz der Ergebnisse werden Gruppierungsfunktionen basierend auf der Ähnlichkeit der Datentopologie angewendet, um ähnliche Muster zu gruppieren. Der Benutzer kann dann interaktiv steuern, wie viele Gruppen zur weiteren Analyse dargestellt werden sollen. Als repräsentativer Subspace einer Gruppe wird derjenige gewählt, der vom Subspacesearch-Algorithmus den höchsten Interssantheitswert hat. Die darin identifizierten Muster können mit Hilfe von interaktiven Werkzeugen markiert und verglichen werden. So können sowohl übereinstimmende als auch komplementäre, sogenannte alternative Muster identifiziert werden.

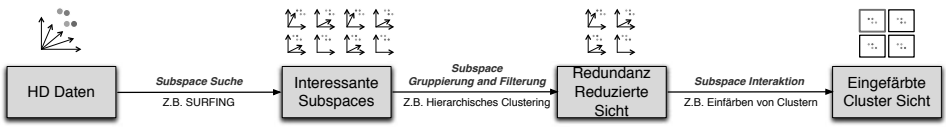


Abbildung 5: Analysepipeline zur visuellen Subspacesuche zur Identifikation von Alternativen Clustern.

Zur Veranschaulichung möchte ich auf die visuellen Schritte im Einzelnen eingehen und das Ergebnis in jedem Schritt der Pipeline erläutern. Dafür wird der zwölf-dimensionale Datensatz aus [FBT+10] verwendet. Im ersten Schritt werden mit Hilfe von einem Subspace-search-Algorithmus die Subspaces identifiziert, die potentiell interessante Muster aufweisen können. An dieser Stelle wurde SURFING [BPK+04] ausgesucht, da dies ein praktisch parameterloser State-of-the-Art Algorithmus ist, der die Subspaces als interessant bewertet, deren k-NN Distanzen nicht uniform verteilt sind [BPK+04]. Dies führt dazu, dass die Punkte im Raum nicht gleich verteilt sind, sondern sich unterschiedlich im Raum gruppieren. Jeder Subspace wird durch ein Scatterplot dargestellt, der die projizierten Daten zeigt (z.B. mit Hilfe von MDS) und einen Dimensionsglyph, der in den grauen Zellen die vorhandenen und in den weißen die nicht vorhandenen Dimensionen darstellt. Das Ergebnis des ersten Schrittes ist in Abbildung 6 zu sehen.

Ein erster Blick auf diese Abbildung lässt Redundanzen im Bezug auf die vorhandenen

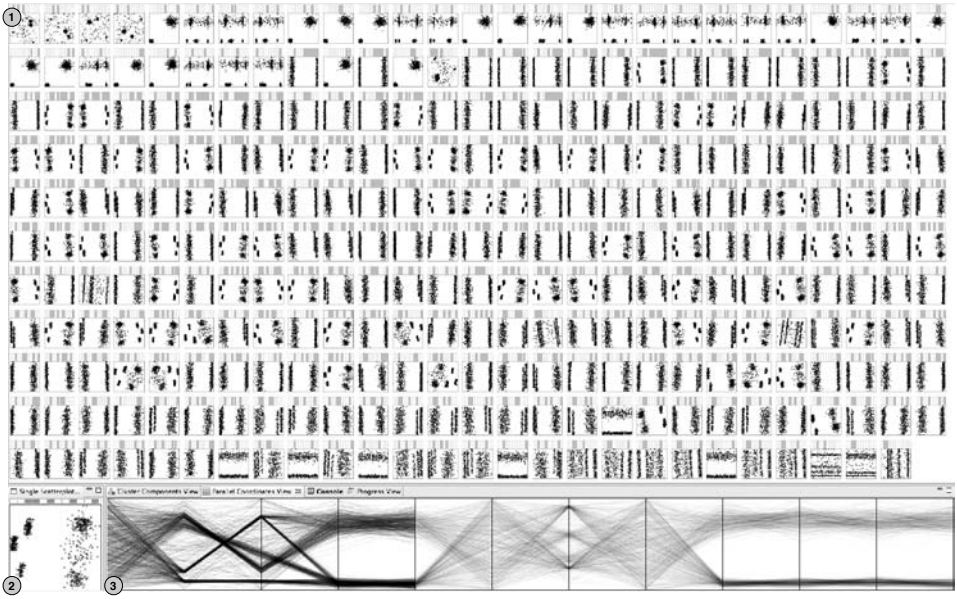


Abbildung 6: (1) Sicht der von SURFING als interessant gewerteten 296 Subspaces des 12D synthetischen Datensatzes aus [FBT+10], sortiert nach dem Interessantheitswert. Der selektierte Subspace in dieser Sicht wird in (2) in einer Einzelsicht dargestellt, die eine interaktive Selektion der Datenpunkte ermöglicht. In (3) ist eine Parallele Koordinaten Ansicht dargestellt, mit den Dimensionen dieses Subspace als hervorgehobene erste Achsen und die übrigen Datendimensionen am Ende zum Vergleich aufgereiht.

Muster erkennen. Deshalb werden Mechanismen benötigt, diese zu reduzieren und die vorhandenen Unterräume zu gruppieren. Ich schlage deshalb im zweiten Schritt Gruppierungsverfahren vor, die die Unterräume entweder basierend auf der Objektähnlichkeit oder basierend auf der Dimensionsüberlappung gruppieren. Details über diese Maße können meiner Dissertation [Tat13] entnommen werden.

Einen Unterraum alleine zu betrachten, ist in vielen Fällen nicht sehr aussagekräftig in Bezug auf die Beschaffenheit der Daten. Unterschiedliche Subspaces können übereinstimmende, komplementäre oder widersprechende Relationen zwischen Datenpunkten aufweisen. Über die Gruppierungsfunktionen können Unterräume verglichen werden und die darin enthaltenen Muster besser interpretiert werden. So identifiziere ich vier verschiedene Arten von Relationen (siehe Abbildung 7), in denen Unterräume zueinander stehen. So sind sie **redundant**, wenn sie sowohl ähnliche Dimensionen als auch topologische Merkmale aufweisen. Sie sind **übereinstimmend**, wenn sie die gleiche Topologie, jedoch unterschiedliche Unterräume aufweisen. Der Fall ähnlicher Dimensionen mit unterschiedlichen Topologien heißt **dominante Dimension**, da die unterschiedlichen Dimensionen das Muster bestimmen. Im letzteren Fall unterschiedlicher Topologien und Dimensionen weisen die Subspaces **komplementäre** Muster auf.

Nach der Gruppierung können im letzten Schritt der Analyse durch Interaktion die Daten

		Enthaltene Dimensionen	
		ähnlich	nicht ähnlich
Topologie der Daten	ähnlich	redundant	übereinstimmend
	nicht ähnlich	dominante Dimensionen	komplementär

Abbildung 7: Relationen von Unterräumen abhängig von ihrer Dimensionsüberlappung und Datentopologie, die durch die Gruppierungs- und Filterfunktionen entdeckt werden können.

in Bezug auf diese Muster untersucht werden. Die Ansichten in Abbildung 8 helfen zur Identifizierung der Relationen. Im vorherigen Beispiel sind nach der Gruppierung in Abbildung 8(1) sechs Gruppen zu sehen, jeweils mit einer Farbe umrandet. In Abbildung 8(5) sind die gruppenzugehörigen Unterräume der orangenen, grünen und lilanen Gruppe zu sehen. Rechts daneben sind die Subspaces abhängig von ihrer Dimensionsähnlichkeit positioniert. Mit Hilfe von Interaktion mit den verlinkten Ansichten und den unterschiedlichen Perspektiven auf die Daten können die bislang besprochenen Relationen identifiziert werden. Ein Beispiel dieser Relationen ist auch anhand eines zweiten Datensatzes einer Ernährungsdatenbank in Abbildung 9 zu sehen.

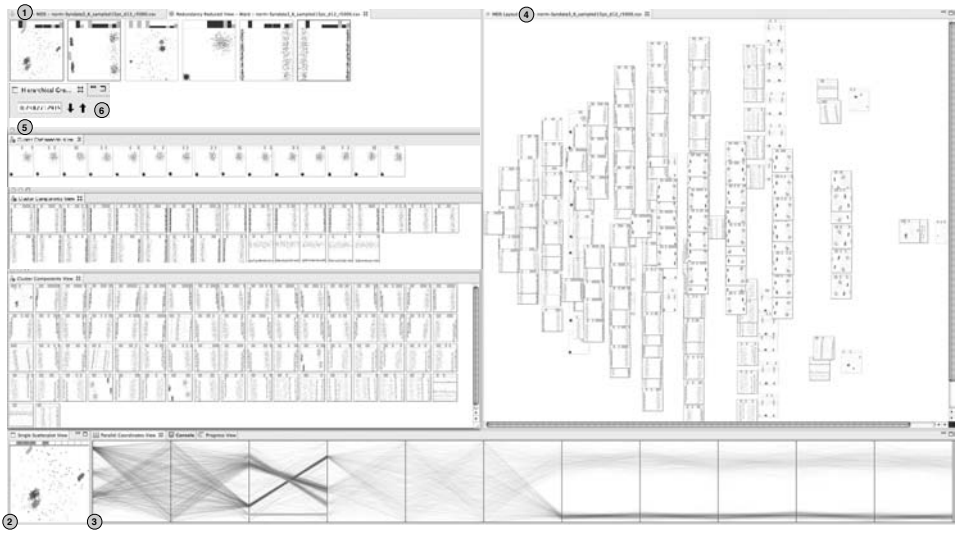


Abbildung 8: (1) Sechs Gruppen nach der Gruppierung basierend auf die Ähnlichkeit der Topologie der Daten. Die erste Gruppe in dieser Sicht wird in (2) in einer Einzelsicht dargestellt, die eine interaktive Selektion der Datenpunkte ermöglicht. In (3) ist eine Parallele Koordinaten Ansicht dargestellt, mit den Dimensionen dieses Subspace als hervorgehobenen ersten Achsen und den übrigen Datendimensionen am Ende zum Vergleich aufgereiht. In (4) werden die Subspaces mit Hilfe von MDS basierend auf die Ähnlichkeit ihrer Dimensionen im 2D Raum positioniert. In (5) sind die Komponenten von drei Gruppen aus (1) zu sehen. (6) stellt die Navigationsknöpfe für das Durchlaufen der Hierarchie dar.

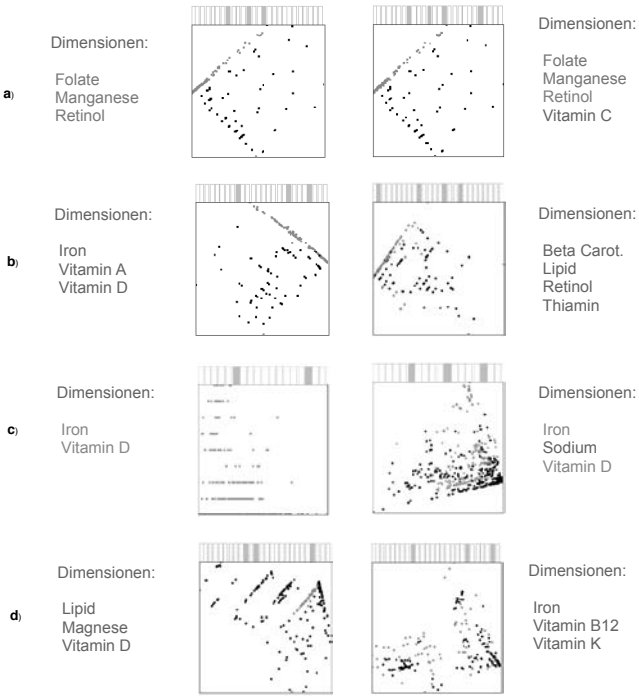


Abbildung 9: Relationen von Unterräumen abhängig von ihrer Dimensionsüberlappung und Datentopologie, die durch die Gruppierungs- und Filterfunktionen entdeckt werden können.

Der vorgestellte Ansatz ist für die Erforschung der Relationen in Subspaces ein möglicher Ausgangspunkt. Erweiterungen dieses Ansatzes können in Betracht gezogen werden. Im Bereich der Datenanalyse wären dies zum einen die Skalierbarkeit der Algorithmen mit anwachsenden Datenmengen sowie unterschiedliche Interessantheitsmaße für das Filtern der Unterräume im ersten Schritt, zum anderen die Sensitivität der verschiedenen Komponenten im Umgang mit Rauschen. Auf der Seite der Visualisierung werden Verbesserungen bei der Darstellung der Unterräume benötigt, um erstens deren visuelle Skalierbarkeit zu gewährleisten, um zweitens die Projektionsstabilität zu beeinflussen und um drittens möglicherweise eine neue Darstellung, die nicht auf Scatterplots beruht, vorzuschlagen. Ebenso könnten automatische Clusteringalgorithmen in den verschiedenen Unterräumen eingesetzt werden, um den manuellen Schritt zu ersetzen und so gleichzeitig eine Vielfalt von möglichen Gruppierungen zu vergleichen.

4 Schlussfolgerung

Die Mustersuche in hochdimensionalen Räumen ist heute ebenso komplex wie von großer Wichtigkeit. Die Verwendung von Visualisierungen, um numerische Daten darzustellen,

von automatischen Methoden, um interessante Elemente zu berechnen, und von Interaktion, um durch die Informationsräume zu navigieren, kann dazu führen, wichtige Muster zu finden. Durch die Komplexität der Daten ist ein Bereich allein nicht ausreichend, um diese zu analysieren, aber in der Kombination können diese sehr nützlich sein.

Literatur

[BPK+04] Christian Baumgartner, Claudia Plant, Karin Kailing, Hans-Peter Kriegel und Peer Kröger. Subspace Selection for Clustering High-Dimensional Data. In *Proceedings of the Fourth IEEE Conference on Data Mining (ICDM '04)*, Seiten 11–18. IEEE CS Press, 2004.

[CLN86] Daniel B. Carr, Richard J. Littlefield und Wesley L. Nichloson. Scatterplot Matrix Techniques for Large N. In *Proceedings of the Seventeenth Symposium on the Interface of Computer Sciences and Statistics on Computer Science and Statistics*, Seiten 297–306. Elsevier North-Holland, Inc., 1986.

[FBT+10] Bilkis J. Ferdosi, Hugo Buddelmeijer, Scott Trager, Michael H. F. Wilkinson und Jos B. T. M. Roerdink. Finding and Visualizing Relevant Subspaces for Clustering High-Dimensional Astronomical Data Using Connected Morphological Operators. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST '11)*, Seiten 35–42. IEEE CS Press, 2010.

[Ins85] Alfred Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1(4):69–91, 1985.

[KKZ09] Hans-Peter Kriegel, Peer Kröger und Arthur Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM Transactions on Knowledge Discovery from Data (TKDD '09)*, 3(1):1–58, 2009.

[Tat13] Andrada Tatu. *Visual Analytics of Patterns in High-Dimensional Data*. Dissertation, University of Konstanz, Juli 2013.

Andrada Tatu wurde am 23. Dezember 1984 in Sibiu (Rumänien) geboren. Nachdem sie dort die deutsche Schule besucht hatte, studierte sie in Konstanz von Okt. 2004 bis Sept. 2007 im Studiengang Information Engineering und erlangte den Bachelor of Science, danach weitere zwei Jahre, um im Sept. 2009 den Master of Science an der Universität Konstanz zu erlangen. In dieser Zeit war sie ab dem dritten Bachelorsemester Stipendiatin der Studienstiftung des deutschen Volkes und Hilfswissenschaftlerin (HiWi) am Lehrstuhl für Datenanalyse und Visualisierung von Prof. Dr. Daniel A. Keim. Durch den engen Kontakt zur Forschungsgruppe begann sie im Okt. 2009 eine Promotion und schloss diese an diesem Lehrstuhl im Juli 2013 ab. Nach einer halbjährigen Tätigkeit als Postdoktorandin entschied sie sich für die Wirtschaft und arbeitet seit Januar 2014 bei PricewaterhouseCoopers AG als Senior Consultant im Bereich Forensic Services / Datenanalyse.

Modulare Analyse praxisrelevanter Sicherheitsprotokolle in UC-Modellen*

Max Tuengerthal

ecsec GmbH, Michelau
max.tuengerthal@ecsec.de

Abstract: Sicherheitsprotokolle, wie SSL/TLS, sind aus unserem Alltag nicht mehr wegzudenken und wir verlassen uns stark auf ihre Sicherheit. Da immer wieder ernstzunehmende Angriffe auf diese Protokolle gefunden werden, ist ihre umfassende Sicherheitsanalyse unabdingbar. Die Komplexität dieser Protokolle und damit der Sicherheitsbeweise stellt jedoch eine große Herausforderung dar. Eine Möglichkeit, dieser Komplexität zu begegnen, ist eine *modulare Sicherheitsanalyse* in sogenannten "universal composability" (UC) Modellen, bei der das zu analysierende Protokoll in Komponenten zerlegt wird, die separat betrachtet werden können. Dieser erfolgversprechender Ansatz wurde bisher allerdings nur selten verwendet, um praxisrelevante Protokolle umfassend und im Detail zu analysieren. Dies hat verschiedene Gründe, auf die wir in dieser Arbeit eingehen werden.

Der Hauptbeitrag dieser Arbeit ist es, Methoden und Techniken zur modularen Sicherheitsanalyse in UC-Modellen bereitzustellen. Es wird damit eine solide Grundlage für eine umfassende Sicherheitsanalyse praxisrelevanter Sicherheitsprotokolle gelegt. Zudem wird der entwickelte Ansatz auf in der Praxis weit verbreitete Sicherheitsprotokolle angewendet, um dessen Nutzen zu demonstrieren.

1 Einleitung

Sicherheitsprotokolle, wie SSL/TLS, IEEE 802.11i (WPA/WPA2), SSH, IPsec, DNSSEC, Kerberos und viele mehr, sind aus unserem Alltag nicht mehr wegzudenken und wir verlassen uns stark auf ihre Sicherheit. Sie werden z. B. für den sicheren Schlüsselaustausch und die sichere Kommunikation in unsicheren Netzwerken, wie dem Internet, genutzt.

Der Entwuf von Sicherheitsprotokollen ist äußerst komplex und fehleranfällig, wie viele Schwachstellen und Angriffe auf aktuelle Sicherheitsprotokolle immer wieder belegen. Die genutzten Angriffsvektoren sind dabei vielfältig: Es gibt Angriffe auf die verwendeten kryptographischen Primitive (Verschlüsselung, Digitale Signaturen, Zertifikate, Nachrichtenauthentifizierungscodes, etc.), wie z. B. den berühmten Bleichenbacher Angriff [Ble98] auf SSL, der eine Schwachstelle in der RSA-PKCS#1-Verschlüsselung ausnutzt, den Angriff auf den CBC-Modus der Verschlüsselung SSL/TLS 1.0 [Bar06], die bekannten Angriffe

*Englischer Titel der Dissertation: "Analysis of Real-World Security Protocols in a Universal Composability Framework" [Tue13].

auf WEP (siehe z. B. [TWP07]) und Angriffe von Paterson et al. auf manche in SSH, IPsec und TLS verwendeten Verschlüsselungsmodi (siehe z. B. [PY06, APW09, DP10, AP13]). Darüber hinaus gibt es Angriffe auf das Protokoll an sich, also auf die logische Struktur des Protokolls. Zum Beispiel Angriffe auf Kerberos [CJS+08], eine Schwachstelle (die sogenannte *renegotiation vulnerability*) bei SSL/TLS [RD09] und Angriffe auf die Authentisierung im von Google verwendeten SAML Single Sign-On Protokoll [ACC+08] sowie im von Mozilla entwickelten Single Sign-On Protokoll *Persona* [FKS14].

Die Bedeutung von Sicherheitsprotokollen auf der einen und die zahlreichen Angriffe auf der anderen Seite zeigen deutlich, dass es unverzichtbar ist, Sicherheitsprotokolle systematisch und umfassend zu analysieren. Für die Protokollanalyse gibt es im Wesentlichen zwei Ansätze: den sogenannten *symbolischen* (oder *formalen*) Ansatz und den *kryptographischen* (oder *komplexitätstheoretischen*) Ansatz. Der kryptographische Ansatz kann weiter unterteilt werden in die Analyse in *Spiel-basierten* und *Simulations-basierten* (oder *"universal composability" (UC)*) Modellen. In jedem dieser Ansätze ist die Grundidee, dass das Sicherheitsprotokoll seine Sicherheitseigenschaft in folgendem Szenario erfüllen muss:

- Das Netzwerk wird komplett oder zum Teil von Angreifern kontrolliert.
- Protokollteilnehmer können mit den Angreifern kollaborieren und vom Protokoll abweichen.
- Mehrere Protokollsitzungen mit den gleichen oder verschiedenen Protokollteilnehmern können parallel ablaufen.

Dieses Szenario erfasst die in vielen Netzwerken, z. B. dem Internet, existierenden Bedrohungen und gilt als Standard in der Informationssicherheit und Kryptographie. Aufgrund dieses starken, aber realistischen, Angreifermodells ist der Entwurf und die Analyse von kryptographischen Protokollen sehr komplex und fehleranfällig. Praxisrelevante Sicherheitsprotokolle sind meist sehr umfangreich, da sie eine Vielzahl kryptographischer Primitive verwenden und aus verschiedenen Unterprotokollen bestehen, die miteinander interagieren. Ihre umfassende Sicherheitsanalyse stellt daher eine große Herausforderung dar.

Um die Komplexität der Analyse in den Griff zu bekommen, scheint eine *modulare* Sicherheitsanalyse unumgänglich. Dabei wird das zu analysierende Protokoll in Komponenten zerlegt, die separat analysiert werden können.

Ein erfolgversprechender Ansatz für die modulare Sicherheitsanalyse ist die Verwendung der genannten UC-Modelle [Can01, PW01]. Der Hauptbeitrag dieser Arbeit besteht darin, Methoden und Techniken zur modularen Sicherheitsanalyse in UC-Modellen bereitzustellen und damit eine Grundlage für eine umfassende Sicherheitsanalyse praxisrelevanter Protokolle zu legen.

Zwar gibt es auch bei Spiel-basierten Modellen, z. B. Bellare-Rogaway Modellen [BR93], Ansätze zur modularen Analyse, z. B. [HY08, MSW10, BFWW11], der Grad an Modularität, der dort bisher erreicht werden konnte, ist aber deutlich geringer als derjenige, der nötig erscheint und in dieser Arbeit angestrebt und erreicht wird. Auf symbolische Modelle gehen wir an dieser Stelle nicht ein und verweisen stattdessen auf Abschnitt 3.3.

Überblick. In Abschnitt 2 stellen wir den Stand der modularen Analyse in UC-Modellen zu Beginn des Dissertationsvorhabens vor. Die Hauptergebnisse der Doktorarbeit präsentieren wir in Abschnitt 3. In Abschnitt 4 fassen wir die Arbeit zusammen und diskutieren offene Fragen und mögliche zukünftige Arbeiten.

2 Modulare Sicherheitsanalyse in UC-Modellen

In UC-Modellen wird die Sicherheit von Protokollen mit Hilfe von sogenannten idealen Protokollen (auch ideale Funktionalitäten genannt) definiert. Die Grundidee ist, dass man ein Protokoll $\mathcal{P}$ *sicher* bzgl. einer idealen Funktionalität $\mathcal{F}$ nennt (man sagt auch: $\mathcal{P}$ *realisiert* $\mathcal{F}$), wenn es zu jedem Angriff auf $\mathcal{P}$ einen Angriff auf $\mathcal{F}$ gibt, so dass eine Umgebung nicht zwischen diesen beiden Angriffen unterscheiden kann: Da $\mathcal{F}$ per Definition keinen wirklichen Angriff zulässt, kann es auch auf $\mathcal{P}$ keine Angriffe geben. Protokolle, Angreifer und Umgebungen werden dabei als probabilistische polynomzeit-beschränkte Algorithmen (Turing Maschinen) modelliert.

Aufgrund dieser Sicherheitsdefinition können übergeordnete Protokollkomponenten auf Basis untergeordneter idealisierter Komponenten (idealer Funktionalitäten) entwickelt und analysiert werden. Kompositionstheoreme erlauben dann die Ersetzung der idealen Funktionalitäten durch ihre Realisierung, so dass ein Protokoll ohne idealisierte Komponenten entsteht. Anstatt also die Sicherheit eines komplexen Protokolls ohne idealisierte Komponenten direkt zu beweisen, können die untergeordneten Komponenten in Isolation und das komplexe Protokoll auf Basis der idealisierten Komponenten analysiert werden. Da die übergeordneten Komponenten selbst ideale Funktionalitäten realisieren, können sie als untergeordnete Komponenten in immer komplexeren Protokollen fungieren. Diese Vorgehensweise wird in den Abbildungen 1 bis 4 am Beispiel des SSL/TLS Protokolls veranschaulicht: Zunächst beweist man, das kryptographische Primitive, wie solche zur Verschlüsselung und für digitale Signaturen, geeignete ideale Funktionalitäten realisieren (Abbildung 2). Man kann dann diese idealen Funktionalitäten verwenden, um zu zeigen, dass das Handshake-Protokoll von SSL/TLS ein sicheres Schlüsselaustauschprotokoll ist, d. h. eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{KE}}$ realisiert (siehe Abbildung 3). Schließlich kann man $\mathcal{F}_{\mathrm{KE}}$ verwenden, um zu zeigen, dass SSL/TLS einen sicheren Kommunikationskanal realisiert, also eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{SC}}$ (siehe Abbildung 4). Durch die Kompositionstheoreme können nun alle idealen Funktionalitäten durch ihre Realisierung ersetzt werden, was dann, falls die einzelnen Beweisschritte jeweils erfolgreich waren, insgesamt zeigen würde, dass das SSL/TLS Protokoll (ohne idealisierte Komponenten) ein sicheres Kommunikationsprotokoll ist (Abbildung 1).

Darüber hinaus erlauben es die Kompositionstheoreme, wie z. B. Canettis Kompositionstheorem im UC-Modell [Can01] oder die Kompositionstheoreme im IITM-Modell von Küsters [Küs06], und Kompositionstheoreme mit *gemeinsamem Zustand* (*joint state*), z. B. das von Canetti und Rabin [CR03], eine Protokollsitzung in Isolation zu betrachten und aus deren Sicherheit, die Sicherheit beliebig vieler, paralleler Protokollsitzungen zu folgern.

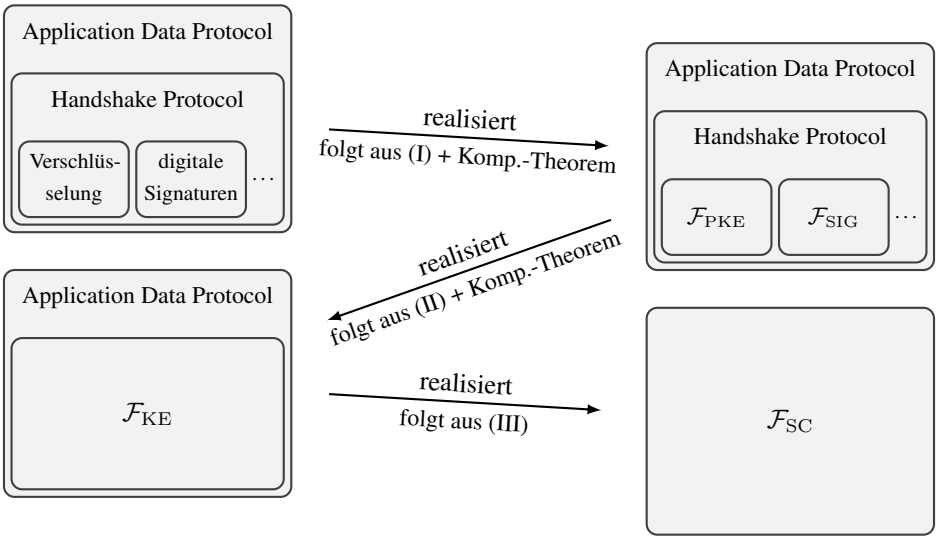


Abbildung 1: Modulare Protokollanalyse in UC-Modellen am Beispiel von SSL/TLS. Die Sicherheit von SSL/TLS folgt aus den Beweisschritten I, II und III (siehe Abbildung 2, 3 bzw. 4) und mit Hilfe von Kompositionstheoremen und der Transitivität der Realisierungs-Relation.

Abbildung 2: Beweisschritt I. Es muss gezeigt werden, dass die im Protokoll verwendeten kryptographischen Primitive geeignete ideale Funktionalitäten realisieren.

Abbildung 3: Beweisschritt II. Es muss gezeigt werden, dass das TLS Handshake Protocol (mit idealisierten kryptographischen Primitiven) eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{KE}}$ realisiert.

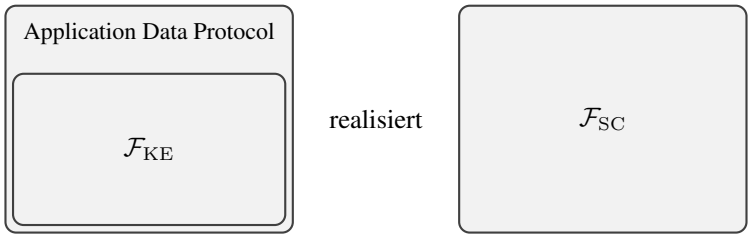


Abbildung 4: Beweisschritt III. Es muss gezeigt werden, dass das TLS Application Data Protocol (mit idealisiertem Schlüsselaustausch) eine geeignete ideale Funktionalität $\mathcal{F}_{\mathrm{SC}}$ realisiert.

Dieser vielversprechende Ansatz wird in der Kryptographie häufig genutzt, um (neue) Protokolle modular zu entwerfen (siehe [Can06] für einen Überblick). Allerdings wurde er bisher nur selten zur Analyse existierender, praxisrelevanter Sicherheitsprotokollen verwendet. Die nennenswertesten Arbeiten hierzu sind die von Gajek et al. [GMP+08] und Backes et al. [BCJ+06]. Gajek et al. [GMP+08] benutzten Canettis UC-Modell, um den kryptographischen Kern von TLS zu analysieren. Allerdings analysierten sie lediglich eine stark modifizierte Variante von TLS. Backes et al. [BCJ+06] analysierten Kerberos, betrachteten aber auch nur eine idealisierte Variante des Orginalprotokolls und benutzten ein unrealistisches Angreifermodell.

Es gab verschiedene Hindernisse für die Verwendung des ansonsten vielversprechenden UC-Ansatzes für die modulare Analyse praxisrelevanter Sicherheitsprotokolle.

Ein Hindernis war das Fehlen geeigneter idealer Funktionalitäten grundlegender kryptographischer Primitive. Obwohl ideale Funktionalitäten für viele kryptographische Aufgaben, wie asymmetrische Verschlüsselung, digitale Signaturen, Authentifizierung und viele andere Aufgaben in UC-Modellen formuliert wurden, zusammen mit ihren Realisierungen, existierte überraschenderweise keine solche Funktionalität für die symmetrische Verschlüsselung und viele andere grundlegende symmetrische kryptographische Verfahren.

Ein weiteres entscheidendes Hindernis war, dass alle bisher vorgestellten Kompositionstheoreme annehmen, dass Parteien, welche an einer gemeinsamen Protokollsitzung teilnehmen, eine im Vorfeld ausgehandelte Sitzungskennung (SID) verwenden. Zwar ist das Verwenden einer solchen SID ein gutes Konstruktionsprinzip, aber existierende Protokolle, insbesondere in der Praxis verwendete, benutzen solche SIDs typischerweise nicht; zumindest nicht explizit oder in der Art und Weise wie von den Theoremen gefordert. Daher können diese Theoreme nicht für die modulare Analyse solcher Protokolle verwendet werden.

3 Hauptbeiträge

Das Hauptziel der Doktorarbeit war es, Techniken, Hilfsmittel und Kompositionstheoreme zur Verfügung zu stellen, welche eine hochmodulare Protokollanalyse in UC-Modellen erlauben, ohne deren Genauigkeit einzuschränken. Dazu mussten insbesondere die oben beschriebenen Hindernisse genau verstanden und überwunden werden.

3.1 Eine ideale Funktionalität für symmetrische kryptographische Primitive

Wie erwähnt, bilden ideale Funktionalitäten die Grundlage für die modulare Protokollanalyse in UC-Modellen. Ideale Funktionalitäten für kryptographische Primitive stellen übergeordneten Protokollen die Funktionalität der Primitive (z. B. Ver- und Entschlüsselung) bereit und garantieren Sicherheit auf „syntaktische“ Art und Weise, z. B. werden Chiffretexte unabhängig von den zu verschlüsselnden Klartexten generiert. Dies vereinfacht die Sicherheitsanalyse, da man auf die syntaktisch garantierten Eigenschaften der idealen Funktionalitäten zurückgreifen kann.

Es existierten bereits viele Funktionalitäten für asymmetrische Primitive, aber keine geeigneten Funktionalitäten für symmetrische Verfahren, insbesondere symmetrische Verschlüsselung. Im Vergleich zu einer idealen Funktionalität für asymmetrischen Verschlüsselung, gibt es verschiedene Schwierigkeiten beim Entwurf einer Funktionalität für symmetrische Verschlüsselung. Bei asymmetrischer Verschlüsselung ist es vernünftig anzunehmen, dass der private Schlüssel niemals die Funktionalität verlässt. Dadurch ist es relativ einfach, geeignete Sicherheitsgarantien zu formulieren. Symmetrische Schlüssel hingegen müssen typischerweise zwischen Protokollteilnehmern ausgetauscht werden, wie z. B. bei Kerberos.

Ein wesentlicher Beitrag der Doktorarbeit war die Entwicklung einer idealen Funktionalität für symmetrische Verschlüsselung [KT11b]. Diese Funktionalität unterstützt außerdem Verfahren zum Ableiten von Schlüsseln, asymmetrische Verschlüsselung und Message Authentication Codes (MAC). Wir haben bewiesen, dass diese Funktionalität mit gängigen Annahmen an übliche kryptographische Verfahren realisiert werden kann.

Diese Funktionalität, welche mit anderen idealen Funktionalitäten, insbesondere solchen, die für reale Protokolle relevant sind (z. B. für digitale Signaturen), kombiniert werden kann, hat ein weites Anwendungsfeld und bietet eine gute Basis für präzise und modulare kryptographische Analyse von praxisrelevanten Sicherheitsprotokollen. Sie vereinfacht, wie erwähnt, aufgrund ihrer syntaktischen Sicherheitsgarantien derartige Analysen sehr.

3.2 Kriterium für Schlüsselaustauschprotokolle und eine Fallstudie an WPA2

Um den Nutzen unserer neuen idealen Funktionalität für die Analyse von praxisrelevanten Sicherheitsprotokollen zu demonstrieren, entwickelten wir ein Kriterium für Schlüsselaustauschprotokolle [KT11b]. Wir zeigten, dass ein Protokoll, welches unser Kriterium erfüllt, sicher im Sinne von UC-Modellen ist, d. h. eine ideale Funktionalität für Schlüsselaustausch realisiert. Da dieses Kriterium auf der neuen Funktionalität basiert, erfordert es lediglich informationstheoretische oder gar rein syntaktische Argumente, anstatt aufwendiger Reduktionsargumente, um das Kriterium zu zeigen.

Als Fallstudie nutzten wir unsere Methode, um zwei zentrale Protokolle des IEEE Standards 802.11i (WPA2),[1] nämlich das 4-Way Handshake (4WHS) Protokoll und das CCM Protokoll, zu analysieren und deren Sicherheit zu beweisen. Dies stellt die erste fundierte kryptographische Analyse dieser Protokolle dar. Diese Fallstudie ist auch deshalb erwähnenswert, weil wir in der Lage waren, das Protokoll sehr präzise zu modellieren. Zum Beispiel konnten die verwendeten Nachrichtenformate exakt nachempfunden werden.

3.3 Kryptographische Korrektheit von Analysen in symbolischen Modellen

Formale Analysen von Sicherheitsprotokollen basierend auf symbolischen Modellen, auch Dolev-Yao-Modelle [DY83] genannt, werden sehr erfolgreich eingesetzt, um Schwachstel-

[1] Dieses Protokoll wird typischerweise in WLAN-Netzen für die sichere Kommunkation zwischen Computer (Laptop/Smartphone/Tablet) und Access Point verwendet.

len in existierenden Protokollen zu finden und Sicherheit mit automatischen Methoden zu beweisen. Eine dabei wichtige Frage ist, ob diese Art der formalen Analyse Sicherheitsgarantien im Sinne der modernen Kryptographie bietet. Initiiert durch die grundlegende Arbeit von Abadi und Rogaway [AR00] wurde diese Frage untersucht und viele positive Ergebnisse zeigen diese sogenannte kryptographische Korrektheit formaler Analysen. Für den Fall von aktiven Angreifern und Protokollen, welche symmetrische Verschlüsselung verwenden, blieb dies allerdings eine Herausforderung.

In dieser Arbeit stellten wir das erste allgemeine Ergebnis zur kryptographischen Korrektheit formaler Analysen für Schlüsselaustauschprotokolle, die symmetrische Verschlüsselung verwenden, vor [KT09]. Dazu entwickelten wir ein symbolisches, automatisch verifizierbares Kriterium und zeigten, dass Schlüsselaustauschprotokolle, welche dieses Kriterium erfüllen, sicher im Sinne von UC-Modellen sind. Unser Ergebnis gilt unter gängigen kryptographischen Annahmen. Im Beweis dieses Ergebnisses nutzten wir unsere neue ideale Funktionalität zur Abstraktion von symmetrischer Verschlüsselung und wir verwendeten Kompositionstheoreme, um Sicherheit für mehrere Protokollsitzungen aus der Sicherheit einer einzelnen Protokollsitzung zu folgern. Dies demonstriert den Nutzen unserer idealen Funktionalität auf dem Gebiet der kryptographischen Korrektheit formaler Analyse.

3.4 Kompositionstheoreme ohne vorherbestimmte Sitzungskennungen

Um dem zweiten in Abschnitt 2 angesprochenen Hindernis zu begegnen, stellten wir ein universelles Kompositionstheorem und eines mit gemeinsamem Zustand vor, welche keine vorherbestimmten Sitzungskennung (SID) annehmen oder verwenden. Der gemeinsame Zustand im Kompositionstheorem ist unsere neue ideale Funktionalität (siehe Abschnitt 3.1). Diese Kompositionstheoreme erlauben es, die Sicherheit von vielen parallelen Protokollsitzungen aus der Sicherheit einer einzelnen Sitzung zu folgern.

Kompositionstheoreme mit gemeinsamem Zustand sind geeignet, um Protokolle zu analysieren, bei denen verschiedenen Sitzungen einen gemeinsamen Zustand haben. Dies ist meist der Fall für grundlegende Schlüsselaustausch- oder Authentifizierungsprotokolle, da diese meist Langzeit-Schlüssel (asymmetrische oder symmetrische Schlüssel, die über viele Protokollsitzungen hinweg verwendet werden) verwenden. Die bisher existierenden Kompositionstheoreme mit gemeinsamem Zustand basierten auf vorherbestimmten SIDs. Diese SIDs wurden genutzt, um den gemeinsamen Zustand quasi wieder aufzuheben. Zum Beispiel dadurch, dass nicht der eigentliche Klartext, sondern Klartext + SID verschlüsselt wird. Der Chiffretext kann dann nicht mehr in eine andere Sitzung eingeschleust werden, da dort auffallen würde, dass er die falsche SID enthält. Das Problem ist, dass viele Protokolle (z. B. SSL/TLS und das 4WHS von WPA2) dies nicht so explizit machen. Dadurch können diese Kompositionstheoreme meist nicht verwendet werden.

Ein entscheidender Beitrag unserer Arbeit war es, ein geeignetes Kriterium (*implizite Disjunktheit* genannt) zu entwickeln, welches diese Quasi-Trennung des Zustandes charakterisiert [KT11a]. Im Wesentlichen drückt das Kriterium aus, dass eine Nachricht, die in

einer Protokollsitzung verschlüsselt (bzw. signiert) wird, nicht in einer anderen Protokollsitzung erfolgreich entschlüsselt (bzw. verifiziert) werden kann.

Praxisrelevante Protokolle erfüllen typischerweise implizite Disjunktheit und dies ist meist leicht zu zeigen (z. B. ist oft nur ein kleiner Teil der verwendeten kryptographischen Primitive zu betrachten), wie wir in Fallstudien zu den Sicherheitsprotokollen SSL/TLS, IEEE 802.11i (WPA2), SSH, IPsec und EAP-PSK gezeigt haben. All diese Protokolle sind implizit disjunkt und daher ist unser Kompositionstheorem mit gemeinsamem Zustand auf sie anwendbar. Um nun also ihre Sicherheit zu beweisen, genügt es, eine einzelne Protokollsitzung in Isolation zu betrachten. Dies vereinfacht den Sicherheitsbeweis stark und erlaubt es nun leichter, die Protokolle im Detail zu analysieren.

4 Zusammenfassung und Ausblick

In dieser Arbeit wurde erläutert, dass es unverzichtbar ist, komplexe, praxisrelevante Sicherheitsprotokolle im Detail zu analysieren. Die existierenden Methoden und Techniken, insbesondere grundlegende ideale Funktionalitäten und Kompositionstheoreme, reichten allerdings für eine solche Analyse nicht aus.

Mit der Entwicklung vielseitig einsetzbarer, grundlegender idealer Funktionalitäten für kryptographische Primitive und mit der Entwicklung geeigneter Kompositionstheoreme liefert diese Arbeit eine solide Grundlage für eine umfassende modulare Sicherheitsanalyse praxisrelevanter Sicherheitsprotokolle. Die Anwendbarkeit und der Nutzen der in dieser Arbeit entwickelten Methoden und Techniken wurde in mehreren Fallstudien an praxisrelevanten Protokollen (SSL/TLS, IEEE 802.11i (WPA2), SSH, IPsec und EAP-PSK) demonstriert. Eine hochmodulare Analyse, wie z. B. in Abbildung 1 beschrieben, ist nun auch für praxisrelevante Sicherheitsprotokolle möglich.

Obwohl sich unsere Anwendungen auf praxisrelevante Protokolle konzentrieren, sind unsere Theoreme und Techniken über diesen Anwendungsfall hinaus von großem Interesse für die Informationssicherheit und Kryptographie, da sie dazu beitragen, die Grenzen des Möglichen bei modularer Analysen auszuloten und zu erweitern.

Unsere Arbeit bietet eine sehr fundierte Basis für zukünftige Forschung. Im Rahmen der Arbeit wurde das Protokoll IEEE 802.11i (WPA2) einer genaueren Analyse unterzogen. Für andere praxisrelevante Protokolle wurde demonstriert, dass die entwickelten Methoden und Techniken prinzipiell anwendbar sind. Eine eingehende (modulare) Analyse dieser Protokolle ist deshalb eine natürliche Fortsetzung der hier begonnenen Arbeit. Dazu ist es auch sinnvoll, den Satz der hier bereits entwickelten idealen Funktionalitäten zu erweitern und die vorgestellten Kompositionstheoreme noch zu verallgemeinern. Diese Arbeiten sind bereits Gegenstand aktueller Forschungsprojekte.

Die hier erzielten Resultate werden auch bereits außerhalb der klassischen Kryptographie zur Analyse von sicherheitskritischen Softwaresystemen direkt auf der Ebene der Programmiersprachen verwendet. Dazu werden die in diesen Systemen benutzten kryptographischen Primitive durch ideale Funktionalitäten ersetzt. Diese so idealisierten Systeme werden dann mit Werkzeugen zur Programmanalyse verifiziert (siehe, z. B., [FKS11, KTG12]).

Literatur

[ACC+08] A. Armando, R. Carbone, L. Compagna, J. Cuéllar und M. L. Tobarra. Formal Analysis of SAML 2.0 Web Browser Single Sign-on: Breaking the SAML-based Single Sign-on for Google Apps. In *FMSE 2008*, Seiten 1–10. ACM, 2008.

[AP13] N. J. AlFardan und K. G. Paterson. Lucky Thirteen: Breaking the TLS and DTLS Record Protocols. In *S&P 2013*, Seiten 526–540. IEEE Computer Society, 2013.

[APW09] M. R. Albrecht, K. G. Paterson und G. J. Watson. Plaintext Recovery Attacks against SSH. In *S&P 2009*, Seiten 16–26. IEEE Computer Society, 2009.

[AR00] M. Abadi und P. Rogaway. Reconciling Two Views of Cryptography (The Computational Soundness of Formal Encryption). In *IFIPTCS 2000*, Jgg. 1872 *LNCS*, Seiten 3–22. Springer, 2000.

[Bar06] G. V. Bard. A Challenging but Feasible Blockwise-Adaptive Chosen-Plaintext Attack on SSL. Bericht 2006/136, Cryptology ePrint Archive, 2006. Online verfügbar: `http://eprint.iacr.org/2006/136`.

[BCJ+06] M. Backes, I. Cervesato, A. D. Jaggard, A. Scedrov und J.-K. Tsay. Cryptographically Sound Security Proofs for Basic and Public-Key Kerberos. In *ESORICS 2006*, Jgg. 4189 *LNCS*, Seiten 362–383. Springer, 2006.

[BFWW11] C. Brzuska, M. Fischlin, B. Warinschi und S. C. Williams. Composability of Bellare-Rogaway Key Exchange Protocol. In *CCS 2011*, Seiten 51–62. ACM, 2011.

[Ble98] D. Bleichenbacher. Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1. In *CRYPTO 1998*, Jgg. 1462 *LNCS*, Seiten 1–12. Springer, 1998.

[BR93] M. Bellare und P. Rogaway. Entity Authentication and Key Distribution. In *CRYPTO 1993*, Jgg. 773 *LNCS*, Seiten 232–249. Springer, 1993.

[Can01] R. Canetti. Universally Composable Security: A New Paradigm for Cryptographic Protocols. In *FOCS 2001*, Seiten 136–145. IEEE Computer Society, 2001.

[Can06] R. Canetti. Security and Composition of Cryptographic Protocols: A Tutorial. Bericht 2006/465, Cryptology ePrint Archive, 2006. `http://eprint.iacr.org/`.

[CJS+08] I. Cervesato, A. D. Jaggard, A. Scedrov, J.-K. Tsay und C. Walstad. Breaking and Fixing Public-key Kerberos. *Inf. Comput.*, 206(2-4):402–424, 2008.

[CR03] R. Canetti und T. Rabin. Universal Composition with Joint State. In *CRYPTO 2003*, Jgg. 2729 *LNCS*, Seiten 265–281. Springer, 2003.

[DP10] J. P. Degabriele und K. G. Paterson. On the (In)Security of IPsec in MAC-then-Encrypt Configurations. In *CCS 2010*. ACM, 2010.

[DY83] D. Dolev und A. C. Yao. On the Security of Public-Key Protocols. *IEEE Transactions on Information Theory*, 29(2):198–208, 1983.

[FKS11] C. Fournet, M. Kohlweiss und P.-Y. Strub. Modular code-based cryptographic verification. In *CCS 2011*, Seiten 341–350. ACM, 2011.

[FKS14] D. Fett, R. Küsters und G. Schmitz. An Expressive Model for the Web Infrastructure: Definition and Application to the BrowserID SSO System. In *S&P 2014*. IEEE Computer Society, 2014.

[GMP+08] S. Gajek, M. Manulis, O. Pereira, A. Sadeghi und J. Schwenk. Universally Composable Security Analysis of TLS. In *ProvSec 2008*, Jgg. 5324 *LNCS*, Seiten 313–327. Springer, 2008.

[HY08] A. Herzberg und I. Yoffe. The Layered Games Framework for Specifications and Analysis of Security Protocols. In *TCC 2008*, Jgg. 4948 *LNCS*, Seiten 125–141. Springer, 2008.

[KT09] R. Küsters und M. Tuengerthal. Computational Soundness for Key Exchange Protocols with Symmetric Encryption. In *CCS 2009*, Seiten 91–100. ACM, 2009.

[KT11a] R. Küsters und M. Tuengerthal. Composition Theorems Without Pre-Established Session Identifiers. In *CCS 2011*, Seiten 41–50. ACM, 2011.

[KT11b] R. Küsters und M. Tuengerthal. Ideal Key Derivation and Encryption in Simulation-based Security. In *CT-RSA 2011*, Jgg. 6558 *LNCS*, Seiten 161–179. Springer, 2011.

[KTG12] R. Küsters, T. Truderung und J. Graf. A Framework for the Cryptographic Verification of Java-like Programs. In *CSF 2012*, Seiten 198–212. IEEE Computer Society, 2012.

[Küs06] R. Küsters. Simulation-Based Security with Inexhaustible Interactive Turing Machines. In *CSFW-19 2006*, Seiten 309–320. IEEE Computer Society, 2006. Vollständige und überarbeitete Version: `http://eprint.iacr.org/2013/025`.

[MSW10] P. Morrissey, N. P. Smart und B. Warinschi. The TLS Handshake Protocol: A Modular Analysis. *Journal of Cryptology*, 23(2):187–223, 2010.

[PW01] B. Pfitzmann und M. Waidner. A Model for Asynchronous Reactive Systems and its Application to Secure Message Transmission. In *S&P 2001*, Seiten 184–201. IEEE Computer Society, 2001.

[PY06] K. G. Paterson und A. K. L. Yau. Cryptography in Theory and Practice: The Case of Encryption in IPsec. In *EUROCRYPT 2006*, Jgg. 4004 *LNCS*, Seiten 12–29. Springer, 2006.

[RD09] M. Ray und S. Dispensa. Renegotiating TLS. November 2009. Online verfügbar: `http://kryptera.se/Renegotiating%20TLS.pdf`.

[Tue13] M. Tuengerthal. *Analysis of Real-World Security Protocols in a Universal Composability Framework*. Logos Verlag Berlin, 2013. Doktorarbeit.

[TWP07] E. Tews, R.-P. Weinmann und A. Pyshkin. Breaking 104 Bit WEP in Less Than 60 Seconds. In *WISA 2007*, Seiten 188–202, 2007.

Dr. Max Tuengerthal studierte Informatik an der Christian-Albrechts-Universität zu Kiel und machte 2007 seinen Abschluss zum Diplom-Informatiker. Danach arbeitete er als wissenschaftlicher Mitarbeiter in der Foundations of Computer and Network Security Group an der ETH Zürich und am Lehrstuhl für Informationssicherheit und Kryptographie an der Universität Trier und promovierte 2013 an der Universität Trier. Sein Forschungsschwerpunkt war das Design und die Analyse von Sicherheitsprotokollen, insbesondere die Entwicklung von Methoden und Techniken zur modularen Protokollanalyse. Seit 2014 ist Max Tuengerthal Berater und Softwareentwickler bei der ecsec GmbH, Michelau.

Vertrauenswürdige und privatheitsbewahrende eingebettete Systeme basierend auf Physically Unclonable Functions*

Christian Wachsmann
Technische Universität Darmstadt
christian.wachsmann@trust.cased.de

Abstract: Die Dissertation präsentiert neuartige Ansätze zum Design und zur Analyse sicherer und privatheitsbewahrender kryptographischer Verfahren für eingebettete Systeme. Sie stellt insbesondere effiziente Authentifikationsprotokolle basierend auf sogenannten Physically Unclonable Functions (PUFs) vor. PUFs erlauben die Erzeugung eindeutiger und unklonbarer Hardware-Fingerprints, die in kryptographische Protokolle eingebunden werden können. Zur Sicherheitsanalyse PUF-basierter Protokolle sowie zur Evaluation der zugrundeliegenden PUF-Implementierungen werden neuartige formale Modelle und Methoden präsentiert. Die aufgestellten Modelle ermöglichen erstmalig die Realisierung beweisbar sicherer PUF-basierter kryptographischer Verfahren und eine Anbindung an komplexitätstheoretische Annahmen im Sinne der modernen Kryptographie. Die Korrektheit der theoretischen Modelle wurde durch die experimentelle Evaluation von PUF-Implementierungen in ASICs validiert.

1 Einleitung

Eingebettete Systeme werden heute in vielen sicherheitskritischen und datenschutzsensiblen Umgebungen eingesetzt. Das Anwendungsspektrum erstreckt sich von Zugangskontrollsystemen, elektronischen Tickets, Sensoren und sogenannten Wearables bis hin zu Automotive-Anwendungen und kritischen Infrastrukturen. In diesen Anwendungen stellt die Authentifikation und die Attestierung, d. h. die Feststellung der eindeutigen Identität und der Integrität der eingebetteten Systeme, eine fundamentale Sicherheitsanforderung dar. Eingebettete Systeme verfügen jedoch meist nur über begrenzte Ressourcen und bieten keine ausreichende Sicherheit, insbesondere hinsichtlich Hardwareangriffen und dem Datenschutz ihrer Nutzer. Eine grundsätzliche Fragestellung in diesem Kontext ist die sichere Bindung von kryptographischen Verfahren an die zugrundeliegende Hardware.

Problemstellung 1: Die Entwicklung von sicheren und privatheitsbewahrenden Authentifikationsverfahren für eingebettete Systeme ist sehr herausfordernd. Dies spiegelt sich in einer Vielzahl von unterschiedlichen Sicherheitsmodellen und Designansätzen wieder (z. B. [Jue05, ADO06, JW07, PV08, BvLDMT09]). Während die in der Praxis eingesetzten Verfahren meist keinen Schutz der Privatsphäre bieten, erfüllen Lösungen aus der Literatur oft die funktionalen Anforderungen praxistauglicher Systeme nicht. Zudem sind

*Englischer Titel der Dissertation: "Trusted and Privacy-preserving Embedded Systems: Advances in Design, Analysis and Application of Lightweight Privacy-preserving Authentication and Physical Security Primitives"

bestehende Sicherheitsmodelle meist nicht miteinander vergleichbar oder gar inkompatibel. So existieren Authentifikationsprotokolle, die zwar in einem Modell formal als sicher bewiesen werden können, in einem anderen jedoch angreifbar sind [JW07]. Um verlässliche Aussagen über die Sicherheit von Authentifikationsverfahren treffen zu können, ist es daher unerlässlich ein einheitliches und praxisnahes Sicherheitsmodell zur Analyse dieser Verfahren zu etablieren.

In diesem Zusammenhang besteht eine grundsätzliche Fragestellung beim Einsatz kryptographischer Verfahren in der sicheren Speicherung der zugrundeliegenden kryptographischen Schlüssel. Insbesondere kosteneffiziente eingebettete Systeme haben oft keinen sicheren Speicher, wodurch die gespeicherten kryptographischen Schlüssel durch Hardwareangriffe ausgelesen werden können. Ein vielversprechender Ansatz diese Systeme vor Hardwareangriffen zu schützen, sind sogenannte Physically Unclonable Functions (PUFs) [MV10]. PUFs basieren auf den im Rahmen von Fertigungstoleranzen entstehenden physikalischen Unterschieden von Hardwarekomponenten, die zwar mit geringem Aufwand gemessen jedoch praktisch nicht reproduziert werden können. Diese Unterschiede sind für jedes Gerät einzigartig und können als Identifikationsmerkmal verwendet werden, sozusagen als physikalischer Fingerabdruck des Gerätes. Dieser Fingerabdruck kann zur Erzeugung kryptographischer Schlüssel verwendet und in sichere kryptographische Protokolle eingebunden werden. Die Vorteile von PUFs bestehen darin, dass kryptographische Schlüssel nicht gespeichert werden müssen, sondern bei Bedarf aus den physikalischen Eigenschaften der PUF erzeugt werden können. Dies ermöglicht zudem eine sichere Bindung dieser Schlüssel und der Software, die diese Schlüssel verwendet, an die zugrundeliegende Hardware-Plattform.

Problemstellung 2: In der Literatur gibt es bereits einige PUF-basierte Authentifikationsverfahren [TB06, BR07]. Jedoch erfordern viele dieser Verfahren die Verfügbarkeit einer Datenbank mit Referenzwerten zur Verifikation des PUF-Fingerabdrucks. Diese Datenbank kann sehr groß werden, insbesondere da jeder Referenzwert nur für eine einzige Authentifikation verwendet werden darf, da sonst Replay-Angriffe möglich sind. Ein anderer Ansatz nutzt die PUF zur Erzeugung kryptographischer Schlüssel zur Verwendung in Standard-Authentifikationsverfahren und erfordert den Einsatz von Fehlerkorrekturmechanismen, um die Reproduzierbarkeit des Schlüssels zu gewährleisten. Die Implementierungen der zugrundeliegenden Dekodierverfahren sind jedoch oft sehr komplex und für eingebettete Systeme ungeeignet.

Problemstellung 3: Die Sicherheit von PUF-basierten Protokollen fundiert auf physikalischen Annahmen, die bisher noch nicht hinreichend untersucht wurden. So werden in der Literatur oft idealisierte Sicherheitsmodelle für PUFs verwendet, die nicht alle Eigenschaften von PUF-Implementierungen berücksichtigen. Zudem gibt es bisher kein allgemeingültiges formales Sicherheitsmodell für PUF-basierte kryptographische Verfahren.

2 Zusammenfassung der Ergebnisse der Dissertation

Die Dissertation [Wac14] stellt neuartige Ansätze zum Design und zur Analyse sicherer und privatheitsbewahrender kryptographischer Authentifikationsverfahren vor, die für verschiedenartige eingebettete Systeme geeignet sind. Die zugrundeliegende Arbeit leis-

tet einen wesentlichen Beitrag zur Entwicklung eines einheitlichen und praxisnahen Sicherheitsmodells für Authentifikationsverfahren für eingebettete Systeme. Konkret zeigen wir am Beispiel eines der bis dato umfassendsten Sicherheitsmodelle für diese Systeme [PV08], dass subtile Aspekte bei der Modellierung von Hardwareangriffen dazu führen können, dass manche Sicherheits- und Datenschutzziele formal nicht gleichzeitig erreicht werden können. Unsere Ergebnisse bilden die Grundlage von wissenschaftlichen Folgearbeiten zum Design und zur Analyse von privatheitsbewahrenden Authentifikationsverfahren (z. B. [Vau10, HPVP11, DR12]).

Beim Design der von uns vorgestellten Authentifikationsverfahren betrachten wir insbesondere praxisnahe Lösungen basierend auf Physically Unclonable Functions (PUFs), die ohne die sichere Speicherung kryptografischer Schlüssel auskommen und somit den Angriffsvektor reduzieren. Zur Sicherheitsanalyse dieser Verfahren sowie zur Evaluation der zugrundeliegenden PUF-Implementierungen präsentieren wir neuartige formale Werkzeuge: ein Evaluations-Framework für PUF-Implementierungen und ein formales Sicherheitsmodell für PUF-basierte kryptographische Verfahren. Unser Evaluations-Framework erlaubt die einheitliche Analyse verschiedener PUF-Typen und eine präzisere Bewertung der von PUFs generierten Entropie als vorherige Methoden. Basierend auf unserem Evaluations-Framework präsentieren wir eine umfangreiche Analyse der wichtigsten Eigenschaften von Implementierungen unterschiedlicher PUF-Typen (Arbiter, Ring Oszillator, SRAM, Flip-Flop und Latch PUFs) in ASIC. Unsere Ergebnisse erlauben erstmals den direkten und fairen Vergleich der Eigenschaften verschiedener PUF-Typen. Unser formales Sicherheitsmodell ermöglicht erstmalig die Realisierung beweisbar sicherer PUF-basierter kryptographischer Verfahren.

Unsere Ergebnisse geben neue Einsichten in die Nutzung physikalischer Eigenschaften von Hardware zur eindeutigen Authentifikation von eingebetteten Systemen. Unsere formalen und praxisnahen Lösungen adressieren die im Bereich der IT-Sicherheit grundsätzliche Problemstellung der eindeutigen und unklonbaren Bindung von Protokollen an die zugrundeliegende Hardware und haben eine Reihe wissenschaftlicher Arbeiten und Veröffentlichungen (z. B. [Vau10, HPVP11, DR12, SSJM12, SKS12, DGK+12]) auf international renommierten Konferenzen erzeugt und motiviert. In den folgenden Abschnitten stellen wir die wichtigsten Ergebnisse der Dissertation detaillierter vor.

3 Hintergrundinformationen zu Physically Unclonable Functions

Eine Physically Unclonable Function (PUF) ist ein physikalisches System, das in ein physikalisches Objekt (z. B. einen Mikrochip) eingebettet ist [MV10]. Es gibt eine Vielzahl von PUF-Typen, welche auf den verschiedensten physikalischen Merkmalen basieren, darunter optische, magnetische und elektrische Effekte. Für die Integration in eingebettete Systeme am geeignetsten sind PUFs basierend auf elektrischen Effekten. Zu den wichtigsten elektrischen PUFs zählen sogenannte verzögerungsbasierte (delay-based) PUFs und speicherbasierte (memory-based) PUFs. Verzögerungsbasierte PUFs basieren auf Signallaufzeiten in elektronischen Schaltungen (z. B. Arbiter und Ring Oszillator PUF) während speicherbasierte PUFs auf der Instabilität von flüchtigen Speicherzellen, wie SRAM, Flip-Flops und Latches basieren.

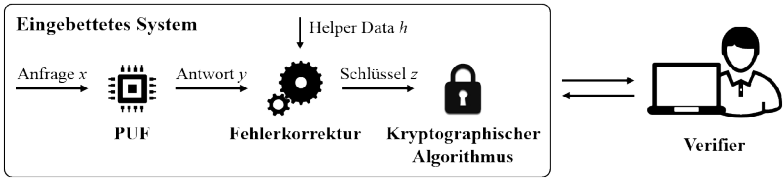


Abbildung 1: Typisches Anwendungsszenario von Physically Unclonable Functions

Wenn eine PUF mit einer Anfrage x (z. B. einem elektrischen Signal) stimuliert wird, antwortet sie mit einer eindeutigen Antwort $y \leftarrow \mathsf{PUF}(x)$, die von den physikalischen Eigenschaften der PUF und x abhängig ist (siehe Abbildung 1). Da die physikalischen Eigenschaften der PUF-Hardware auch von den Betriebsbedingungen (z. B. Schwankungen in der Umgebungstemperatur und der Versorgungsspannung) der PUF-Hardware beeinflusst werden, wird eine PUF auf die selbe Anfrage x immer leicht unterschiedlich antworten.

Die wichtigsten in der Literatur getroffenen Annahmen über PUFs sind *Robustheit*, *physikalische Unklonbarkeit* und *Unberechenbarkeit*. Robustheit bedeutet informell, dass eine mehrfach mit der selben Anfrage x stimulierte PUF immer mit ähnlichen Antworten $y_i = y + e_i$ antwortet, wobei die Unterschiede (z. B. das Hamming-Gewicht von) e_i gering sind. Physikalische Unklonbarkeit bedeutet, dass es praktisch unmöglich ist, zwei PUFs herzustellen, die nicht anhand ihres Anfrage/Antwort-Verhaltens unterschieden werden können. Unberechenbarkeit bezeichnet die Eigenschaft, dass es praktisch unmöglich ist, die Antwort y einer PUF auf eine zufällig gewählte Anfrage x vorherzusagen.

PUFs können zur Erzeugung von kryptographischen Schlüsseln z verwendet und in kryptographische Protokolle eingebunden werden. Um die Reproduzierbarkeit von z zu gewährleisten, werden PUFs meist in Kombination mit Fehlerkorrekturverfahren (z. B. Secure Sketches [DRS04]) eingesetzt (siehe Abbildung 1). Üblicherweise wird vor dem Einsatz der PUF für jede Anfrage x (oder für eine Teilmenge von Anfragen) eine Antwort y als Referenzwert ermittelt. Zusätzlich wird mittels des Fehlerkorrekturverfahrens eine sogenannte Helper Data h für y erzeugt. Später, wenn die PUF auf die Anfrage x mit $y_i = y + e_i$ antwortet, kann das Fehlerkorrekturverfahren mit Hilfe von h den Fehler e_i innerhalb der Grenzen des zugrundeliegenden Dekodierungsverfahrens korrigieren und den Referenzwert y reproduzieren. Hierbei ist zu beachten, dass h zwar partielle Informationen über y preisgibt, jedoch aber nicht geheim gehalten werden muss. Um aus y einen kryptographischen Schlüssel z zu erzeugen, muss die in y enthaltene Entropie (z. B. mittels kryptographischer Hash-Funktionen) extrahiert werden. Hierbei muss der durch das Veröffentlichen von h verursachte Entropieverlust in y berücksichtigt werden. Typischerweise wird der Partei (*Verifier*), die später ein kryptographisches Protokoll mit dem eingebetteten System ausführt, bei der Systeminitialisierung der aus y erzeugte Schlüssel z über einen sicheren Kanal mitgeteilt. Die PUF-Antwort y und Helper Data h können im eingebetteten System abgespeichert werden und müssen nicht geheim gehalten werden.

4 Analyse der Eigenschaften von PUF-Implementierungen

Die Robustheit und die Unberechenbarkeit Eigenschaften von PUFs sind essentiell für deren Integration in sichere und zuverlässige kryptographische Verfahren. In der Literatur finden sich unterschiedliche Ansätze zur Analyse der Eigenschaften von PUF-Implementierungen [TvS+05, HBF09]. Jedoch sind die Ergebnisse dieser Analysen aufgrund der unterschiedlichen Methoden und Testbedingungen nur schwer vergleichbar. Darüberhinaus sind sie nicht konform zu etablierten Sicherheitsmodellen in der modernen Kryptographie.

Wir stellen ein neuartiges Evaluations-Framework vor, das erstmals die einheitliche Analyse der wichtigsten Eigenschaften von Implementierungen unterschiedlicher PUF-Typen ermöglicht. Unser Framework erlaubt zudem eine präzisere Bewertung der Güte von PUFs, insbesondere der generierten Entropie, als vorherige Ansätze.

Unsere Analyse verwendet die Shannon Entropie, die üblicherweise in der Kryptographie betrachtet wird. Insbesondere interessieren wir uns für die minimale Entropie eines einzigen PUF-Antwort-Bits unter der Bedingung, dass alle anderen Bits der Antwort bekannt sind. Dies ermöglicht die Abschätzung einer informationstheoretischen oberen Schranke für die Wahrscheinlichkeit, dass ein starker Angreifer ein einzelnes Bit der PUF-Antwort vorhersagen kann, selbst wenn er alle anderen Bits der Antwort kennt. Formal betrachten wir die bedingte Min-Entropie

$$\mathbf{H}_\infty(Y|W) = -\log_2\big(\max_{x\in X}\{\Pr\big[Y(x)|W(x)\big]\}\big). \tag{1}$$

Hierbei bezeichnet X die Menge aller PUF-Anfragen, $Y(x)$ die Zufallsvariable für das Antwort-Bit y bezüglich einer Anfrage x und $W(x)$ die Zufallsvariable für die Menge aller Antwort-Bits ohne y, also $W(x) = \big\{y'|y' \leftarrow \mathsf{PUF}(x') \wedge x' \in X \setminus \{x\}\big\}$. Die Berechnung dieser Entropie ist praktisch nur schwer durchführbar, da deren Komplexität exponentiell mit der Größe von X und der Größe der Menge aller PUF-Antworten wächst. Um die Entropie dennoch abschätzen zu können, treffen wir die im Folgenden erläuterten Annahmen über die physikalischen Eigenschaften der betrachteten PUFs.

Alle bis dato bekannten elektrischen PUFs bestehen aus einzelnen Elementen (z. B. Speicherzellen), die sich an verschiedenen Positionen im Chip befinden. Unter der Annahme, dass sich weit voneinander entfernte PUF-Elemente gegenseitig kaum beeinflussen und daher keine statistischen Abhängigkeiten aufweisen [HBF09, MV10], lässt sich die bei der Entropieberechnung zu berücksichtigende Menge an PUF-Antwort-Bits signifikant reduzieren. Insbesondere betrachten wir statt der Menge der Antwort-Bits $W(x)$ die deutlich kleinere Menge aller Antwort-Bits $W'(x)$, die von PUF-Elementen in direkter physischer Nähe zum PUF-Element des Antwort-Bits $Y(x)$ erzeugt wurden. Dies ermöglicht eine effiziente und dennoch präzise Abschätzung von Gleichung 1.

Basierend auf unserem Evaluations-Framework haben wir unterschiedliche Implementierungen verschiedener PUF-Typen in ASICs analysiert. Unsere Evaluation basiert auf Daten aus 96 ASICs, die in 65 nm CMOS Technologie mit Industriepartnern hergestellt wurden. Jeder dieser ASICs enthält mehrere Implementierungen unterschiedlicher PUF-Typen, darunter Ring Oszillator, Arbiter, SRAM, Flip-Flop und Latch PUFs.

Unsere Evaluationsergebnisse zeigen, dass alle untersuchten PUFs selbst unter verschie-

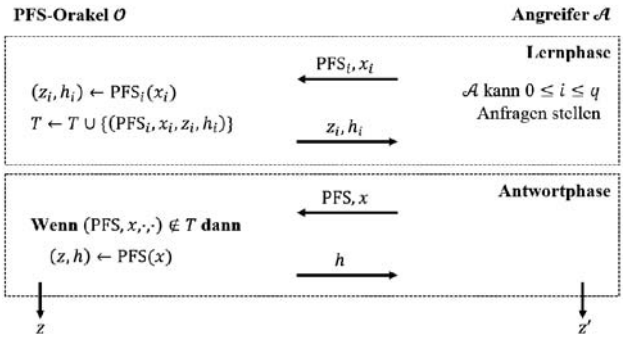


Abbildung 2: Sicherheitsexperiment $\mathbf{Exp}_{\mathcal{A}}^{\text{w-uprd}}(q)$ für Unberechenbarkeit

denen Betriebsbedingungen und hinsichtlich der Materialermüdung ausreichend robust für praktische Anwendungen sind. Jedoch ist die Entropie der PUF-Antworten stark vom PUF-Typ und den Einsatzbedingungen der PUF-Hardware abhängig. Insbesondere die von Arbiter PUFs generierte Entropie ist sehr niedrig, während die Entropie der Flip-Flop und Latch PUF-Antworten durch Temperaturschwankungen beeinflusst wird. Die von Ring Oszillator und SRAM PUFs generierte Entropie ist nahezu ideal und ändert sich bei unterschiedlichen Betriebsbedingungen kaum.

5 Sicherheitsmodell für PUF-basierte kryptographische Verfahren

Wir präsentieren ein formales Sicherheitsmodell für PUFs, welches erstmals eine fundierte Analyse von PUF-basierten kryptographischen Verfahren ermöglicht. Das Modell berücksichtigt die wichtigsten Eigenschaften von PUFs: Robustheit, physikalische Unklonbarkeit und Unberechenbarkeit.

Unser Modell umfasst alle Komponenten, die zum Einsatz von PUFs in kryptographischen Verfahren erforderlich sind. Diese Komponenten fassen wir als *Physical Function System* (PFS) zusammen, das im Wesentlichen einer mit einer PUF ausgestatteten Hardware-Plattform entspricht. Hierbei betrachten wir *physikalische Funktionen* (PF) im Allgemeinen wobei die physikalische Unklonbarkeit nur ein mögliches Merkmal einer PF darstellt. Eine PF besteht aus einer physikalischen Komponente C (z. B. einem Schaltkreis) und einem Evaluationsprozess, der als Schnittstelle zu C dient. Die physikalische Komponente C wird durch einen Herstellungsprozess erzeugt, der gewissen Fertigungstoleranzen unterliegt aus denen die Einzigartigkeit von C resultiert. Um durch Schwankungen in den Betriebsbedingungen der PF-Hardware hervorgerufene Einflüsse auf die PF-Antworten y zu kompensieren, werden PFs meist in Kombination mit einem Extraktor (z. B. einem Fehlerkorrekturverfahren) verwendet [DRS04, MV10]. Der Extraktor kann in zwei Modi betrieben werden: Erzeugung und Rekonstruktion. Im Erzeugung-Modus erzeugt der Extraktor eine Helper Data h und einen Schlüssel z aus y. Im Rekonstruktion-Modus rekonstruiert der Extraktor z aus h und $y_i = y + e_i$.

Basierend auf unserem Systemmodell formalisieren wir Robustheit, physikalische Un-

klonbarkeit und Unberechenbarkeit in Anlehnung an etablierte kryptographische Modelle. Im Folgenden beschreiben wir beispielhaft die Formalisierung von Unberechenbarkeit, die auf dem in Abbildung 2 dargestelltem Sicherheitsexperiment $\mathbf{Exp}_{\mathcal{A}}^{\text{w-uprd}}(q)$ basiert. In diesem Experiment darf ein Angreifer $\mathcal{A}$ in einer Lernphase zunächst bis zu q Anfragen x_i an verschiedene PFS_i stellen und erhält die jeweiligen Ausgaben (z_i, h_i). Die in der Lernphase gewonnene Information kann $\mathcal{A}$ in der Antwortphase des Experiments nutzen, um eine ihm bisher *unbekannte* Ausgabe z eines PFS vorherzusagen. Ein PFS ist unberechenbar, wenn $\mathcal{A}$ dies nur mit geringer Wahrscheinlichkeit gelingt, formal:

Definition 1 (Unberechenbarkeit)**.** *Sei $T = \emptyset$, $\lambda, q \in \mathbb{N}$ mit $q \geq 0$ und $\mathbf{Exp}_{\mathcal{A}}^{\text{w-uprd}}(q)$ das in Abbildung 2 dargestellte Sicherheitsexperiment. Weiterhin bezeichne ρ die Robustheit des PFS. Ein PFS ist* (λ, q)-unberechenbar*, wenn gilt:*

$$\Pr\left[z = z' | (z, z') \leftarrow \mathbf{Exp}_{\mathcal{A}}^{\text{w-uprd}}(q)\right] \leq \lambda \cdot \rho$$

Die Definition ist ähnlich zur Definition von kryptographischen Pseudorandom Functions (PRFs). Jedoch gibt es subtile aber entscheidende Unterschiede, die von den PUF-Modellen in der Literatur nicht betrachtet wurden. Zum einen muss die Helper Data h einbezogen werden, die $\mathcal{A}$ Informationen über z preisgeben könnte. Zum anderen kann $\mathcal{A}$ im Gegensatz zur Sicherheitsdefinition von PRFs mit mehreren verschiedenen PFS_i interagieren, was die Möglichkeit berücksichtigt, dass $\mathcal{A}$ basierend auf den Ausgaben eines PFS Aussagen über die Ausgaben eines anderen PFS treffen könnte.

Unser Sicherheitsmodell dient als Grundlage für Folgearbeiten zum Design von PUFs und PUF-basierter kryptographischer Verfahren (z. B. [SSJM12, SKS12, DGK+12]).

6 Physikalisch-kryptographische Verfahren für eingebettete Systeme

Bestehende PUF-basierte kryptographische Verfahren erfordern entweder eine große Datenbank mit Referenzwerten zur Verifikation der PUF-Ausgaben oder die Implementierung von komplexen Dekodierverfahren. Beides ist für praktische Anwendungen eingebetteter Systeme ungeeignet. Wir stellen ein PUF-basiertes Authentifikationsverfahren für eingebettete Systeme vor, das ressourcenschonend implementierbar ist und keine große Referenzdatenbank benötigt.

Das Protokoll ist in Abbildung 3 dargestellt. Bei der Initialisierung des Verfahrens wird mindestens eine Anfrage x und die dazugehörige Antwort y der Physical Function (PF) des eingebetteten Systems $\mathcal{P}$ in der Datenbank D des Verifiers $\mathcal{V}$ gespeichert. $\mathcal{V}$ startet das Protokoll indem er eine zufällig gewählte PF-Anfrage x und einen zufällig gewählten Wert c mit n-Bit Länge an $\mathcal{P}$ schickt. Daraufhin ermittelt $\mathcal{P}$ die Antwort y seiner PF, erzeugt daraus eine Helper Data h und einen Schlüssel z und schickt h zusammen mit dem mittels einer kryptographischen Hash-Funktion berechneten Wert r an $\mathcal{V}$. Schließlich rekonstruiert $\mathcal{V}$ den von $\mathcal{P}$ verwendeten Schlüssel z, berechnet die Hash-Funktion und akzeptiert $\mathcal{P}$ nur dann, wenn das Ergebnis mit dem von $\mathcal{P}$ erhaltenem r übereinstimmt.

In unserem Protokoll wird der Extraktor anders als in der Literatur üblich eingesetzt, um auf der Seite von $\mathcal{V}$ den von $\mathcal{P}$ verwendeten Schlüssel z zu rekonstruieren. Standard-

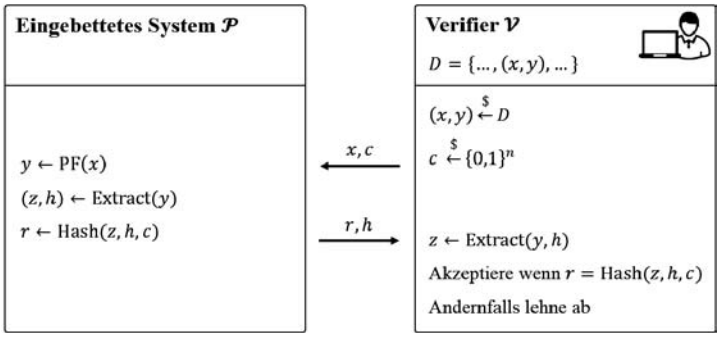


Abbildung 3: PUF-basiertes Authentifikationsprotokoll

Implementierungen des Extraktors sind in diesem Szenario nicht anwendbar, da ein Angreifer mehrere Helper Data zu PUF-Antworten auf die selbe Anfrage x erhalten und somit genug Informationen sammeln kann, um den Schlüssel z zu rekonstruieren. Daher erfordert dieser Ansatz die Verwendung spezieller Extractoren [Boy04], die auch in diesem Szenario als sicher gelten.

Wir zeigen die Sicherheit unseres Protokolls formal nach dem in der Kryptographie üblichen komplexitätstheoretischen Prinzip der Reduktion. Konkret zeigen wir, dass ein Angreifer $\mathcal{A}$, der eine Nachricht (r, h) erzeugen kann, die der Verifier $\mathcal{V}$ akzeptiert, entweder die Sicherheit des Extraktors, der Hash-Funktion oder die Unberechenbarkeit der PUF verletzen kann. Basierend auf der Annahme, dass die verwendete Hash-Funktion und der Extraktor ihre Sicherheitsansprüche erfüllen und auf unseren Evaluationsergebnissen, die zeigen, dass die Antworten geeigneter PUF-Implementierungen eine nahezu ideale Entropie erreichen und somit praktisch nicht berechenbar sind, folgern wir, dass ein solcher Angreifer in der Praxis nicht existiert.

7 Zusammenfassung und Ausblick

Wir präsentieren sichere, effiziente und praxisnahe PUF-basierte kryptographische Protokolle und formale Werkzeuge, die eine fundierte Sicherheitsanalyse dieser Verfahren ermöglichen. Konkret stellen wir ein neuartiges formales Sicherheitsmodell für PUF-basierte kryptographische Verfahren und ein Evaluations-Framework für PUF-Implementierungen vor. Unsere Ergebnisse sind die Grundlage für mehrere wissenschaftliche Folgearbeiten zur Erfassung der Eigenschaften neuartiger PUF-Typen und beim Design PUF-basierter kryptographischer Verfahren.

Unsere Arbeit liefert Anknüpfungspunkte für aktuelle Forschung. Während bisherige Implementierungen verzögerungsbasierter PUFs meist als zusätzliche Schaltung implementiert werden müssen, stellen wir ein neuartiges PUF-Design vor, das es ermöglicht vorhandene Schaltungen in eingebetteten Systemen neben ihrer eigentlichen Funktion auch als PUF zu verwenden [KKSW14]. Ein weiterer Anknüpfungspunkt ist die Erweiterung unseres PUF-Evaluations-Frameworks hinsichtlich Hardwareangriffen. In diesem Kontext

zeigen wir, dass speicherbasierte PUFs unter bestimmten Bedingungen durch die Kombination von Hardwareangriffen und Kryptanalyse angegriffen werden können und präsentieren Gegenmaßnahmen [OSW13].

Eine weitere grundsätzliche Fragestellung neben der Authentifikation von eingebetteten Systemen ist die Verifizierung der Integrität von entfernten Rechnerplattformen (Attestierung). In diesem Zusammenhang analysieren wir Attestierungsprotokolle für eingebettete Systeme mit stark begrenzten Ressourcen [ASSW13]. Durch die Integration von PUFs binden wir diese Protokolle an die zugrundeliegende Hardware, wodurch bestimmte Angriffe verhindert werden [SSW11].

Literatur

[ADO06] G. Avoine, E. Dysli und P. Oechslin. Reducing Time Complexity in RFID Systems. In *Selected Areas in Cryptography (SAC)*, Jgg. 3897 of *LNCS*, Seiten 291–306. Springer, 2006.

[ASSW13] F. Armknecht, A.-R. Sadeghi, S. Schulz und C. Wachsmann. A Security Framework for the Analysis and Design of Software Attestation. In *ACM Conference on Computer and Communications Security (CCS)*, Seiten 1–12. ACM, 2013.

[Boy04] X. Boyen. Reusable Cryptographic Fuzzy Extractors. In *ACM Conference on Computer and Communications Security (CCS)*, Seiten 82–91. ACM, 2004.

[BR07] L. Bolotnyy und G. Robins. Physically Unclonable Function-based Security and Privacy in RFID Systems. In *Conference on Pervasive Computing and Communications (PerCom)*, Seiten 211–220. IEEE, 2007.

[BvLDMT09] M. Burmester, T. van Le, B. De Medeiros und G. Tsudik. Universally Composable RFID Identification and Authentication Protocols. *ACM Transactions on Information and Systems Security*, 12(4), 2009.

[DGK+12] F. Durvaux, B. Gérard, S. Kerckhof, F. Koeune und F. Standaert. Intellectual Property Protection for Integrated Systems Using Soft Physical Hash Functions. In *Information Security Applications*, Jgg. 7690 of *LNCS*, Seiten 208–225. Springer, 2012.

[DR12] T. Deursen und S. Radomirović. Insider Attacks and Privacy of RFID Protocols. In *European Conference on Public Key Infrastructures, Services and Applications (EuroPKI)*, Jgg. 7163 of *LNCS*, Seiten 91–105. Springer, 2012.

[DRS04] Y. Dodis, L. Reyzin und A. Smith. Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. In *Advances in Cryptology (EUROCRYPT)*, Jgg. 3027 of *LNCS*, Seiten 523–540. Springer, 2004.

[HBF09] D. Holcomb, W. P. Burleson und K. Fu. Power-up SRAM State as an Identifying Fingerprint and Source of True Random Numbers. *IEEE Transactions on Computers*, 58(9):1198–1210, 2009.

[HPVP11] J. Hermans, A. Pashalidis, F. Vercauteren und B. Preneel. A New RFID Privacy Model. In *European Symposium on Research in Computer Security (ESORICS)*, Jgg. 6879 of *LNCS*, Seiten 568–587. Springer, 2011.

[Jue05] A. Juels. Minimalist Cryptography for Low-cost RFID Tags (Extended Abstract). In *Security in Communication Networks (SCN)*, Jgg. 3352 of *LNCS*, Seiten 149–164. Springer, 2005.

[JW07] A. Juels und S. A. Weis. Defining Strong Privacy for RFID. In *Conference on Pervasive Computing and Communications (PerCom)*, Seiten 342–347. IEEE, 2007.

[KKSW14] J. Kong, F. Koushanfar, A.-R. Sadeghi und C. Wachsmann. PUFatt: Embedded Platform Attestation Based on Novel Processor-Based PUFs. In *Design Automation Conference (DAC)*. 2014.

[MV10] R. Maes und I. Verbauwhede. Physically Unclonable Functions: A Study on the State of the Art and Future Research Directions. In *Towards Hardware-Intrinsic Security*, Seiten 3–37. Springer, 2010.

[OSW13] Y. Oren, A.-R. Sadeghi und C. Wachsmann. On the Effectiveness of the Remanence Decay Side-Channel to Clone Memory-based PUFs. In *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, Jgg. 8086 of *LNCS*, Seiten 107–125. Springer, 2013.

[PV08] R. I. Paise und S. Vaudenay. Mutual Authentication in RFID: Security and Privacy. In *ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, Seiten 292–299. ACM, 2008.

[SKS12] S. Shariati, F. Koeune und F. Standaert. Security Analysis of Image-based PUFs for Anti-counterfeiting. In *Communications and Multimedia Security*, Jgg. 7394 of *LNCS*, Seiten 26–38. Springer, 2012.

[SSJM12] S. Shariati, F.-X. Standaert, L. Jacques und B. Macq. Analysis and Experimental Evaluation of Image-based PUFs. 2(3):189–206, 2012.

[SSW11] S. Schulz, A.-R. Sadeghi und C. Wachsmann. Short Paper: Lightweight Remote Attestation Using Physical Functions. In *ACM Conference on Wireless Network Security (WiSec)*, Seiten 109–114. ACM, 2011.

[TB06] P. Tuyls und L. Batina. RFID-tags for Anti-counterfeiting. In *Topics in Cryptology (CT-RSA)*, Jgg. 3860 of *LNCS*, Seiten 115–131. Springer, 2006.

[TvS+05] P. Tuyls, B. Škorić, S. Stallinga, A. H. M. Akkermans und W. Ophey. Information-Theoretic Security Analysis of Physical Uncloneable Functions. In *Financial Cryptography and Data Security (FC)*, Jgg. 3570 of *LNCS*, Seite 578. Springer, 2005.

[Vau10] S. Vaudenay. Privacy Models for RFID Schemes. In *Radio Frequency Identification: Security and Privacy Issues (RFIDSec)*, Jgg. 6370 of *LNCS*, Seite 65. Springer, 2010.

[Wac14] C. Wachsmann. *Trusted and Privacy-preserving Embedded Systems: Advances in Design, Analysis and Application of Lightweight Privacy-preserving Authentication and Physical Security Primitives*. Dissertation, TU Darmstadt, 2014.

Christian Wachsmann hat im September 2013 seine Promotion an der TU Darmstadt mit Auszeichnung abgeschlossen. Zurzeit ist er am Intel Collaborative Research Institute for Secure Computing an der TU Darmstadt beschäftigt. Der Schwerpunkt seiner Arbeit liegt in der Konzeption, Entwicklung, formalen Modellierung und Sicherheitsanalyse von Sicherheitsarchitekturen und kryptographischen Protokollen zur Überprüfung der Softwareintegrität (Attestierung) von eingebetteten Systemen und hardwarebasierten Sicherheitsmechanismen zur Erkennung von Produktfälschungen. Christian Wachsmann ist Autor von über 30 wissenschaftlichen Veröffentlichungen in international renommierten Fachzeitschriften und IT-Sicherheitskonferenzen.

GI-Edition Lecture Notes in Informatics

Dissertations

Vol. D-1: Ausgezeichnete Informatikdissertationen 2000
Vol. D-2: Ausgezeichnete Informatikdissertationen 2001
Vol. D-3: Ausgezeichnete Informatikdissertationen 2002
Vol. D-4: Ausgezeichnete Informatikdissertationen 2003
Vol. D-5: Ausgezeichnete Informatikdissertationen 2004
Vol. D-6: Ausgezeichnete Informatikdissertationen 2005
Vol. D-7: Ausgezeichnete Informatikdissertationen 2006
Vol. D-8: Ausgezeichnete Informatikdissertationen 2007
Vol. D-9: Ausgezeichnete Informatikdissertationen 2008
Vol. D-10: Ausgezeichnete Informatikdissertationen 2009
Vol. D-11: Ausgezeichnete Informatikdissertationen 2010
Vol. D-12: Ausgezeichnete Informatikdissertationen 2011
Vol. D-13: Ausgezeichnete Informatikdissertationen 2012
Vol. D-14: Ausgezeichnete Informatikdissertationen 2013

P-1 Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001.

P-2 Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001.

P-3 Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Language to Information Systems, NLDB'2001.

P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensorgestützte Chirurgie; Workshop des SFB 414.

P-5 Andy Schürr (Hg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems.

P-6 Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001.

P-7 Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001.

P-8 Reinhard Keil-Slawik, Johannes Magenheim (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.

P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeitstagung.

P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience.

P-11 Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002.

P-12 Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002.

P-13 Jan von Knop, Peter Schirmbacher and Viljan Mahni_ (Hrsg.): The Changing Universities – The Role of Technology.

P-14 Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Se-mantic Web – XSW 2002.

P-15 Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine.

P-16 J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Busi-ness 2002.

P-17 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze –Die Verletzbarkeit meistern, 16. DFN Arbeitstagung.

P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2002.

P-19 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund.

P-20 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3. Okt. 2002 in Dortmund (Ergänzungsband).

P-21 Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen.

P-22 Sigrid Schubert, Johannes Magenheim, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation.

P-23 Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit

P-24 Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen

P-25 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003

P-26 Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web

P-27 Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin

P-28 Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Er-fahrungen und Visionen

P-29 Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems

P-30 Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications

P-31 Arslan Brömme, Christoph Busch (Eds.): BIOSIG 2003: Biometrics and Electronic Signatures

P-32 Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003

P-33 Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft

P-34 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1)

P-35 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2)

P-36 Rüdiger Grimm, Hubert B. Keller, Kai Rannenberg (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik

P-37 Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik

P-38 E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003

P-39 Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003

P-40 Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004

P-41 Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing

P-42 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste

P-43 Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistance, Scalability, Transactions – Database Mechanisms for Mobile Applications

P-44 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning. E-Services

P-45 Bernhard Rumpe, Wofgang Hesse (Hrsg.): Modellierung 2004

P-46 Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment

P-47 Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society

P-48 Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications

P-49 G. Schiefer, P. Wagner, M. Morgenstern, U. Rickert (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven

P-50 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-51 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-52 Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik

P-53 Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004

P-54 Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration

P-55 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science und Grid Adhoc-Netze Medienintegration

P-56 Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government

P-57 Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen

P-58 Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schweiggert (Hrsg.): Testing of Component-Based Systems and Software Quality

P-59 J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Ranneberg, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments

P-60 Steffen Friedrich (Hrsg.): Unterrichtskonzepte für inforrmatische Bildung

P-61 Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen

P-62 Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit

P-63 Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and ist Applications

P-64 Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005

P-65 Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolffried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web

P-66 Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik

P-67 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1)

P-68 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2)

P-69 Robert Hirschfeld, Ryszard Kowalcyk, Andreas Polze, Matthias Weske (Hrsg.): NODe 2005, GSEM 2005

P-70 Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005)

P-71 Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005

P-72 Klaus P. Jantke, Klaus-Peter Fähnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment

P-73 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): “Heute schon das Morgen sehen“

P-74 Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology

P-75 Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture

P-76 Thomas Kirste, Birgitta König-Riess, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz

P-77 Jana Dittmann (Hrsg.): SICHERHEIT 2006

P-78 K.-O. Wenkel, P. Wagner, M. Morgenstern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel

P-79 Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006

P-80 Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce

P-81 Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS´06

P-82 Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006

P-83 Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics

P-84 Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications

P-85 Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems

P-86 Robert Krimmer (Ed.): Electronic Voting 2006

P-87 Max Mühlhäuser, Guido Rößling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik

P-88 Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODe 2006, GSEM 2006

P-90 Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur

P-91 Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006

P-92 Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006

P-93 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1

P-94 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2

P-95 Matthias Weske, Markus Nüttgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen

P-96 Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies

P-97 Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Mangament & IT-Forensics – IMF 2006

P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices

P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik

P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik

P-101 Stefan Böttinger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten

P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments

P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)

P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme

P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007

P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops

P-107 Heinrich C. Mayr, Dimitris Karagiannis (eds.) Information Systems Technology and its Applications

P-108 Arslan Brömme, Christoph Busch, Detlef Hühnlein (eds.) BIOSIG 2007: Biometrics and Electronic Signatures

P-109 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 1

P-110 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 2

P-111 Christian Eibl, Johannes Magenheim, Sigrid Schubert, Martin Wessner (Hrsg.) DeLFI 2007: 5. e-Learning Fachtagung Informatik

P-112 Sigrid Schubert (Hrsg.) Didaktik der Informatik in Theorie und Praxis

P-113 Sören Auer, Christian Bizer, Claudia Müller, Anna V. Zhdanova (Eds.) The Social Semantic Web 2007 Proceedings of the 1st Conference on Social Semantic Web (CSSW)

P-114 Sandra Frings, Oliver Göbel, Detlef Günther, Hardo G. Hase, Jens Nedon, Dirk Schadt, Arslan Brömme (Eds.) IMF2007 IT-incident management & IT-forensics Proceedings of the 3rd International Conference on IT-Incident Management & IT-Forensics

P-115 Claudia Falter, Alexander Schliep, Joachim Selbig, Martin Vingron and Dirk Walther (Eds.) German conference on bioinformatics GCB 2007

P-116 Witold Abramowicz, Leszek Maciszek (Eds.)
Business Process and Services Computing
1st International Working Conference on Business Process and Services Computing
BPSC 2007

P-117 Ryszard Kowalczyk (Ed.)
Grid service engineering and manegement
The 4th International Conference on Grid Service Engineering and Management
GSEM 2007

P-118 Andreas Hein, Wilfried Thoben, Hans-Jürgen Appelrath, Peter Jensch (Eds.)
European Conference on ehealth 2007

P-119 Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.)
Enterprise Modelling and Information Systems Architectures
Concepts and Applications

P-120 Adam Pawlak, Kurt Sandkuhl, Wojciech Cholewa,
Leandro Soares Indrusiak (Eds.)
Coordination of Collaborative Engineering - State of the Art and Future Challenges

P-121 Korbinian Herrmann, Bernd Bruegge (Hrsg.)
Software Engineering 2008
Fachtagung des GI-Fachbereichs Softwaretechnik

P-122 Walid Maalej, Bernd Bruegge (Hrsg.)
Software Engineering 2008 - Workshopband
Fachtagung des GI-Fachbereichs Softwaretechnik

P-123 Michael H. Breitner, Martin Breunig, Elgar Fleisch, Ley Pousttchi, Klaus Turowski (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Technologien, Prozesse, Marktfähigkeit
Proceedings zur 3. Konferenz Mobile und Ubiquitäre Informationssysteme
(MMS 2008)

P-124 Wolfgang E. Nagel, Rolf Hoffmann, Andreas Koch (Eds.)
9th Workshop on Parallel Systems and Algorithms (PASA)
Workshop of the GI/ITG Speciel Interest Groups PARS and PARVA

P-125 Rolf A.E. Müller, Hans-H. Sundermeier, Ludwig Theuvsen, Stephanie Schütze, Marlies Morgenstern (Hrsg.)
Unternehmens-IT:
Führungsinstrument oder Verwaltungsbürde
Referate der 28. GIL Jahrestagung

P-126 Rainer Gimnich, Uwe Kaiser, Jochen Quante, Andreas Winter (Hrsg.)
10th Workshop Software Reengineering (WSR 2008)

P-127 Thomas Kühne, Wolfgang Reisig, Friedrich Steimann (Hrsg.)
Modellierung 2008

P-128 Ammar Alkassar, Jörg Siekmann (Hrsg.)
Sicherheit 2008
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
2.-4. April 2008
Saarbrücken, Germany

P-129 Wolfgang Hesse, Andreas Oberweis (Eds.)
Sigsand-Europe 2008
Proceedings of the Third AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems

P-130 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
1. DFN-Forum Kommunikations-technologien Beiträge der Fachtagung

P-131 Robert Krimmer, Rüdiger Grimm (Eds.)
3rd International Conference on Electronic Voting 2008
Co-organized by Council of Europe, Gesellschaft für Informatik and E-Voting. CC

P-132 Silke Seehusen, Ulrike Lucke, Stefan Fischer (Hrsg.)
DeLFI 2008:
Die 6. e-Learning Fachtagung Informatik

P-133 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 1

P-134 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 2

P-135 Torsten Brinda, Michael Fothe, Peter Hubwieser, Kirsten Schlüter (Hrsg.)
Didaktik der Informatik –
Aktuelle Forschungsergebnisse

P-136 Andreas Beyer, Michael Schroeder (Eds.)
German Conference on Bioinformatics
GCB 2008

P-137 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2008: Biometrics and Electronic Signatures

P-138 Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau, Klaus Turowski (Hrsg.)
Synergien durch Integration und Informationslogistik
Proceedings zur DW2008

P-139 Georg Herzwurm, Martin Mikusz (Hrsg.)
Industrialisierung des Software-Managements
Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -wartung im Fachbereich Wirtschaftsinformatik

P-140 Oliver Göbel, Sandra Frings, Detlef Günther, Jens Nedon, Dirk Schadt (Eds.)
IMF 2008 - IT Incident Management & IT Forensics

P-141 Peter Loos, Markus Nüttgens, Klaus Turowski, Dirk Werth (Hrsg.)
Modellierung betrieblicher Informationssysteme (MobIS 2008)
Modellierung zwischen SOA und Compliance Management

P-142 R. Bill, P. Korduan, L. Theuvsen, M. Morgenstern (Hrsg.)
Anforderungen an die Agrarinformatik durch Globalisierung und Klimaveränderung

P-143 Peter Liggesmeyer, Gregor Engels, Jürgen Münch, Jörg Dörr, Norman Riegel (Hrsg.)
Software Engineering 2009
Fachtagung des GI-Fachbereichs Softwaretechnik

P-144 Johann-Christoph Freytag, Thomas Ruf, Wolfgang Lehner, Gottfried Vossen (Hrsg.)
Datenbanksysteme in Business, Technologie und Web (BTW)

P-145 Knut Hinkelmann, Holger Wache (Eds.)
WM2009: 5th Conference on Professional Knowledge Management

P-146 Markus Bick, Martin Breunig, Hagen Höpfner (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Entwicklung, Implementierung und Anwendung
4. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2009)

P-147 Witold Abramowicz, Leszek Maciaszek, Ryszard Kowalczyk, Andreas Speck (Eds.)
Business Process, Services Computing and Intelligent Service Management
BPSC 2009 · ISM 2009 · YRW-MBP 2009

P-148 Christian Erfurth, Gerald Eichler, Volkmar Schau (Eds.)
9th International Conference on Innovative Internet Community Systems
I^{2}CS 2009

P-149 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
2. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung

P-150 Jürgen Münch, Peter Liggesmeyer (Hrsg.)
Software Engineering
2009 - Workshopband

P-151 Armin Heinzl, Peter Dadam, Stefan Kirn, Peter Lockemann (Eds.)
PRIMIUM
Process Innovation for Enterprise Software

P-152 Jan Mendling, Stefanie Rinderle-Ma, Werner Esswein (Eds.)
Enterprise Modelling and Information Systems Architectures
Proceedings of the 3rd Int'l Workshop EMISA 2009

P-153 Andreas Schwill, Nicolas Apostolopoulos (Hrsg.)
Lernen im Digitalen Zeitalter
DeLFI 2009 – Die 7. E-Learning Fachtagung Informatik

P-154 Stefan Fischer, Erik Maehle Rüdiger Reischuk (Hrsg.)
INFORMATIK 2009
Im Focus das Leben

P-155 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2009:
Biometrics and Electronic Signatures
Proceedings of the Special Interest Group on Biometrics and Electronic Signatures

P-156 Bernhard Koerber (Hrsg.)
Zukunft braucht Herkunft
25 Jahre »INFOS – Informatik und Schule«

P-157 Ivo Grosse, Steffen Neumann, Stefan Posch, Falk Schreiber, Peter Stadler (Eds.)
German Conference on Bioinformatics 2009

P-158 W. Claupein, L. Theuvsen, A. Kämpf, M. Morgenstern (Hrsg.)
Precision Agriculture
Reloaded – Informationsgestützte Landwirtschaft

P-159 Gregor Engels, Markus Luckey, Wilhelm Schäfer (Hrsg.)
Software Engineering 2010

P-160 Gregor Engels, Markus Luckey, Alexander Pretschner, Ralf Reussner (Hrsg.)
Software Engineering 2010 – Workshopband
(inkl. Doktorandensymposium)

P-161 Gregor Engels, Dimitris Karagiannis Heinrich C. Mayr (Hrsg.)
Modellierung 2010

P-162 Maria A. Wimmer, Uwe Brinkhoff, Siegfried Kaiser, Dagmar Lück-Schneider, Erich Schweighofer, Andreas Wiebe (Hrsg.)
Vernetzte IT für einen effektiven Staat
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2010

P-163 Markus Bick, Stefan Eulgem, Elgar Fleisch, J. Felix Hampe, Birgitta König-Ries, Franz Lehner, Key Pousttchi, Kai Rannenberg (Hrsg.)
Mobile und Ubiquitäre Informationssysteme
Technologien, Anwendungen und Dienste zur Unterstützung von mobiler Kollaboration

P-164 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2010: Biometrics and Electronic Signatures Proceedings of the Special Interest Group on Biometrics and Electronic Signatures

P-165 Gerald Eichler, Peter Kropf, Ulrike Lechner, Phayung Meesad, Herwig Unger (Eds.)
10th International Conference on Innovative Internet Community Systems (I^2CS) – Jubilee Edition 2010 –

P-166 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
3. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung

P-167 Robert Krimmer, Rüdiger Grimm (Eds.)
4th International Conference on Electronic Voting 2010
co-organized by the Council of Europe, Gesellschaft für Informatik and E-Voting.CC

P-168 Ira Diethelm, Christina Dörge, Claudia Hildebrandt, Carsten Schulte (Hrsg.)
Didaktik der Informatik
Möglichkeiten empirischer Forschungsmethoden und Perspektiven der Fachdidaktik

P-169 Michael Kerres, Nadine Ojstersek Ulrik Schroeder, Ulrich Hoppe (Hrsg.)
DeLFI 2010 - 8. Tagung der Fachgruppe E-Learning der Gesellschaft für Informatik e.V.

P-170 Felix C. Freiling (Hrsg.)
Sicherheit 2010
Sicherheit, Schutz und Zuverlässigkeit

P-171 Werner Esswein, Klaus Turowski, Martin Juhrisch (Hrsg.)
Modellierung betrieblicher Informationssysteme (MobIS 2010)
Modellgestütztes Management

P-172 Stefan Klink, Agnes Koschmider Marco Mevius, Andreas Oberweis (Hrsg.)
EMISA 2010
Einflussfaktoren auf die Entwicklung flexibler, integrierter Informationssysteme
Beiträge des Workshops der GI-Fachgruppe EMISA (Entwicklungsmethoden für Informationssysteme und deren Anwendung)

P-173 Dietmar Schomburg, Andreas Grote (Eds.)
German Conference on Bioinformatics 2010

P-174 Arslan Brömme, Torsten Eymann, Detlef Hühnlein, Heiko Roßnagel, Paul Schmücker (Hrsg.)
perspeGKtive 2010
Workshop „Innovative und sichere Informationstechnologie für das Gesundheitswesen von morgen“

P-175 Klaus-Peter Fähnrich, Bogdan Franczyk (Hrsg.)
INFORMATIK 2010
Service Science – Neue Perspektiven für die Informatik
Band 1

P-176 Klaus-Peter Fähnrich, Bogdan Franczyk (Hrsg.)
INFORMATIK 2010
Service Science – Neue Perspektiven für die Informatik
Band 2

P-177 Witold Abramowicz, Rainer Alt, Klaus-Peter Fähnrich, Bogdan Franczyk, Leszek A. Maciaszek (Eds.)
INFORMATIK 2010
Business Process and Service Science – Proceedings of ISSS and BPSC

P-178 Wolfram Pietsch, Benedikt Krams (Hrsg.)
Vom Projekt zum Produkt
Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -wartung im Fachbereich Wirtschafts-informatik (WI-MAW), Aachen, 2010

P-179 Stefan Gruner, Bernhard Rumpe (Eds.)
FM+AM`2010
Second International Workshop on Formal Methods and Agile Methods

P-180 Theo Härder, Wolfgang Lehner, Bernhard Mitschang, Harald Schöning, Holger Schwarz (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW)
14. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme“ (DBIS)

P-181 Michael Clasen, Otto Schätzel, Brigitte Theuvsen (Hrsg.)
Qualität und Effizienz durch informationsgestützte Landwirtschaft, Fokus: Moderne Weinwirtschaft

P-182 Ronald Maier (Hrsg.)
6th Conference on Professional Knowledge Management
From Knowledge to Action

P-183 Ralf Reussner, Matthias Grund, Andreas Oberweis, Walter Tichy (Hrsg.)
Software Engineering 2011
Fachtagung des GI-Fachbereichs Softwaretechnik

P-184 Ralf Reussner, Alexander Pretschner, Stefan Jähnichen (Hrsg.)
Software Engineering 2011 Workshopband (inkl. Doktorandensymposium)

P-185 Hagen Höpfner, Günther Specht, Thomas Ritz, Christian Bunse (Hrsg.)
MMS 2011: Mobile und ubiquitäre Informationssysteme Proceedings zur 6. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2011)

P-186 Gerald Eichler, Axel Küpper, Volkmar Schau, Hacène Fouchal, Herwig Unger (Eds.)
11th International Conference on Innovative Internet Community Systems (I^2CS)

P-187 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
4. DFN-Forum Kommunikations-technologien, Beiträge der Fachtagung 20. Juni bis 21. Juni 2011 Bonn

P-188 Holger Rohland, Andrea Kienle, Steffen Friedrich (Hrsg.)
DeLFI 2011 – Die 9. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. 5.–8. September 2011, Dresden

P-189 Thomas, Marco (Hrsg.)
Informatik in Bildung und Beruf
INFOS 2011
14. GI-Fachtagung Informatik und Schule

P-190 Markus Nüttgens, Oliver Thomas, Barbara Weber (Eds.)
Enterprise Modelling and Information Systems Architectures (EMISA 2011)

P-191 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2011
International Conference of the Biometrics Special Interest Group

P-192 Hans-Ulrich Heiß, Peter Pepper, Holger Schlingloff, Jörg Schneider (Hrsg.)
INFORMATIK 2011
Informatik schafft Communities

P-193 Wolfgang Lehner, Gunther Piller (Hrsg.)
IMDM 2011

P-194 M. Clasen, G. Fröhlich, H. Bernhardt, K. Hildebrand, B. Theuvsen (Hrsg.)
Informationstechnologie für eine nachhaltige Landbewirtschaftung
Fokus Forstwirtschaft

P-195 Neeraj Suri, Michael Waidner (Hrsg.)
Sicherheit 2012
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 6. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)

P-196 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2012
Proceedings of the 11th International Conference of the Biometrics Special Interest Group

P-197 Jörn von Lucke, Christian P. Geiger, Siegfried Kaiser, Erich Schweighofer, Maria A. Wimmer (Hrsg.)
Auf dem Weg zu einer offenen, smarten und vernetzten Verwaltungskultur
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2012

P-198 Stefan Jähnichen, Axel Küpper, Sahin Albayrak (Hrsg.)
Software Engineering 2012
Fachtagung des GI-Fachbereichs Softwaretechnik

P-199 Stefan Jähnichen, Bernhard Rumpe, Holger Schlingloff (Hrsg.)
Software Engineering 2012
Workshopband

P-200 Gero Mühl, Jan Richling, Andreas Herkersdorf (Hrsg.)
ARCS 2012 Workshops

P-201 Elmar J. Sinz Andy Schürr (Hrsg.)
Modellierung 2012

P-202 Andrea Back, Markus Bick, Martin Breunig, Key Pousttchi, Frédéric Thiesse (Hrsg.)
MMS 2012:Mobile und Ubiquitäre Informationssysteme

P-203 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
5. DFN-Forum Kommunikations-technologien
Beiträge der Fachtagung

P-204 Gerald Eichler, Leendert W. M. Wienhofen, Anders Kofod-Petersen, Herwig Unger (Eds.)
12th International Conference on Innovative Internet Community Systems (I2CS 2012)

P-205 Manuel J. Kripp, Melanie Volkamer, Rüdiger Grimm (Eds.)
5th International Conference on Electronic Voting 2012 (EVOTE2012)
Co-organized by the Council of Europe, Gesellschaft für Informatik and E-Voting.CC

P-206 Stefanie Rinderle-Ma, Mathias Weske (Hrsg.)
EMISA 2012
Der Mensch im Zentrum der Modellierung

P-207 Jörg Desel, Jörg M. Haake, Christian Spannagel (Hrsg.)
DeLFI 2012: Die 10. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V.
24.–26. September 2012

P-208 Ursula Goltz, Marcus Magnor, Hans-Jürgen Appelrath, Herbert Matthies, Wolf-Tilo Balke, Lars Wolf (Hrsg.)
INFORMATIK 2012

P-209 Hans Brandt-Pook, André Fleer, Thorsten Spitta, Malte Wattenberg (Hrsg.)
Nachhaltiges Software Management

P-210 Erhard Plödereder, Peter Dencker, Herbert Klenk, Hubert B. Keller, Silke Spitzer (Hrsg.)
Automotive – Safety & Security 2012
Sicherheit und Zuverlässigkeit für automobile Informationstechnik

P-211 M. Clasen, K. C. Kersebaum, A. Meyer-Aurich, B. Theuvsen (Hrsg.)
Massendatenmanagement in der Agrar- und Ernährungswirtschaft
Erhebung – Verarbeitung – Nutzung
Referate der 33. GIL-Jahrestagung
20. – 21. Februar 2013, Potsdam

P-212 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2013
Proceedings of the 12th International Conference of the Biometrics Special Interest Group
04.–06. September 2013
Darmstadt, Germany

P-213 Stefan Kowalewski, Bernhard Rumpe (Hrsg.)
Software Engineering 2013
Fachtagung des GI-Fachbereichs Softwaretechnik

P-214 Volker Markl, Gunter Saake, Kai-Uwe Sattler, Gregor Hackenbroich, Bernhard Mit schang, Theo Härder, Veit Köppen (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW) 2013
13. – 15. März 2013, Magdeburg

P-215 Stefan Wagner, Horst Lichter (Hrsg.)
Software Engineering 2013 Workshopband (inkl. Doktorandensymposium)
26. Februar – 1. März 2013, Aachen

P-216 Gunter Saake, Andreas Henrich, Wolfgang Lehner, Thomas Neumann, Veit Köppen (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW) 2013 –Workshopband
11. – 12. März 2013, Magdeburg

P-217 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
6. DFN-Forum Kommunikations-technologien
Beiträge der Fachtagung
03.–04. Juni 2013, Erlangen

P-218 Andreas Breiter, Christoph Rensing (Hrsg.)
DeLFI 2013: Die 11 e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. (GI)
8. – 11. September 2013, Bremen

P-219 Norbert Breier, Peer Stechert, Thomas Wilke (Hrsg.)
Informatik erweitert Horizonte
INFOS 2013
15. GI-Fachtagung Informatik und Schule
26. – 28. September 2013

P-220 Matthias Horbach (Hrsg.)
INFORMATIK 2013
Informatik angepasst an Mensch, Organisation und Umwelt
16. – 20. September 2013, Koblenz

P-221 Maria A. Wimmer, Marijn Janssen, Ann Macintosh, Hans Jochen Scholl, Efthimios Tambouris (Eds.)
Electronic Government and Electronic Participation
Joint Proceedings of Ongoing Research of IFIP EGOV and IFIP ePart 2013
16. – 19. September 2013, Koblenz

P-222 Reinhard Jung, Manfred Reichert (Eds.)
Enterprise Modelling
and Information Systems Architectures
(EMISA 2013)
St. Gallen, Switzerland
September 5. – 6. 2013

P-223 Detlef Hühnlein, Heiko Roßnagel (Hrsg.)
Open Identity Summit 2013
10. – 11. September 2013
Kloster Banz, Germany

P-224 Eckhart Hanser, Martin Mikusz, Masud Fazal-Baqaie (Hrsg.)
Vorgehensmodelle 2013
Vorgehensmodelle – Anspruch und Wirklichkeit
20. Tagung der Fachgruppe Vorgehensmodelle im Fachgebiet Wirtschaftsinformatik (WI-VM) der Gesellschaft für Informatik e.V.
Lörrach, 2013

P-225 Hans-Georg Fill, Dimitris Karagiannis, Ulrich Reimer (Hrsg.)
Modellierung 2014
19. – 21. März 2014, Wien

P-226 M. Clasen, M. Hamer, S. Lehnert, B. Petersen, B. Theuvsen (Hrsg.)
IT-Standards in der Agrar- und Ernährungswirtschaft Fokus: Risiko- und Krisenmanagement
Referate der 34. GIL-Jahrestagung
24. – 25. Februar 2014, Bonn

P-227 Wilhelm Hasselbring, Nils Christian Ehmke (Hrsg.)
Software Engineering 2014
Fachtagung des GI-Fachbereichs Softwaretechnik
25. – 28. Februar 2014
Kiel, Deutschland

P-228 Stefan Katzenbeisser, Volkmar Lotz, Edgar Weippl (Hrsg.)
Sicherheit 2014
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 7. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
19. – 21. März 2014, Wien

P-230 Arslan Brömme, Christoph Busch (Eds.)
BIOSIG 2014
Proceedings of the 13th International Conference of the Biometrics Special Interest Group
10. – 12. September 2014 in Darmstadt, Germany

P-231 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
7. DFN-Forum Kommunikationstechnologien
16. – 17. Juni 2014
Fulda

P-232 E. Plödereder, L. Grunske, E. Schneider, D. Ull (Hrsg.)
INFORMATIK 2014
Big Data – Komplexität meistern
22. – 26. September 2014
Stuttgart

P-233 Stephan Trahasch, Rolf Plötzner, Gerhard Schneider, Claudia Gayer, Daniel Sassiat, Nicole Wöhrle (Hrsg.)
DeLFI 2014 – Die 12. e-Learning Fachtagung Informatik
der Gesellschaft für Informatik e.V.
15. – 17. September 2014
Freiburg

P-234 Fernand Feltz, Bela Mutschler, Benoît Otjacques (Eds.)
Enterprise Modelling and Information Systems Architectures
(EMISA 2014)
Luxembourg, September 25-26, 2014

P-235 Robert Giegerich, Ralf Hofestädt, Tim W. Nattkemper (Eds.)
German Conference on Bioinformatics 2014
September 28 – October 1
Bielefeld, Germany

P-236 Martin Engstler, Eckhart Hanser, Martin Mikusz, Georg Herzwurm (Hrsg.)
Projektmanagement und Vorgehensmodelle 2014
Soziale Aspekte und Standardisierung
Gemeinsame Tagung der Fachgruppen Projektmanagement (WI-PM) und Vorgehensmodelle (WI-VM) im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V., Stuttgart 2014

P-237 Detlef Hühnlein, Heiko Roßnagel (Hrsg.)
Open Identity Summit 2014
4.–6. November 2014
Stuttgart, Germany

The titles can be purchased at:

Köllen Druck + Verlag GmbH
Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn
Fax: +49 (0)228/9898222
E-Mail: druckverlag@koellen.de