GESELLSCHAFT
FÜR INFORMATIK

Steffen Hölldobler et al. (Hrsg.)

Ausgezeichnete Informatikdissertationen 2019

Im Auftrag der GI herausgegeben durch die Mitglieder des Nominierungsausschusses

Steffen Hölldobler (Vorsitzender), Technische Universität Dresden
Sven Apel, Universität des Saarlandes
Abraham Bernstein, Universität Zürich
Felix Freiling, Universität Erlangen-Nürnberg
Hans-Peter Lenhof, Universität des Saarlandes
Gustaf Neumann, Wirtschaftsuniversität Wien
Rüdiger Reischuk, Universität zu Lübeck
Kay Uwe Römer, TU Graz
Björn Scheuermann, Humbold-Universität zu Berlin
Nicole Schweikardt, Humbold-Universität zu Berlin
Myra Spiliopoulou, Otto-von-Guericke Universität Magdeburg
Sabine Süsstrunk, École Polytechnique Fédérale de Lausanne
Klaus Wehrle, RWTH Aachen

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Dissertations
Series of the Gesellschaft für Informatik (GI)
Volume D-20
ISBN 978-3-88579-775-3

Volume Editors
Prof. Dr. Steffen Hölldobler (Chair), Technische Universität Dresden
01062 Dresden, Deutschland, sh43@posteo.de

Sven Apel, Universität des Saarlandes
Abraham Bernstein, Universität Zürich
Felix Freiling, Universität Erlangen-Nürnberg
Hans-Peter Lenhof, Universität des Saarlandes
Gustaf Neumann, Wirtschaftsuniversität Wien
Rüdiger Reischuk, Universität zu Lübeck
Kay Uwe Römer, TU Graz
Björn Scheuermann, Humbold-Universität zu Berlin
Nicole Schweikardt, Humbold-Universität zu Berlin
Myra Spiliopoulou, Otto-von-Guericke Universität Magdeburg
Sabine Süsstrunk, École Polytechnique Fédérale de Lausanne
Klaus Wehrle, RWTH Aachen

Series Editorial Board
Andreas Oberweis, KIT Karlsruhe,
(Chairman, andreas.oberweis@kit.edu)
Torsten Brinda, Universität Duisburg-Essen, Germany
Dieter Fellner, Technische Universität Darmstadt, Germany
Ulrich Flegel, Infineon, Germany
Ulrich Frank, Universität Duisburg-Essen, Germany
Michael Goedicke, Universität Duisburg-Essen, Germany
Ralf Hofestädt, Universität Bielefeld, Germany
Wolfgang Karl, KIT Karlsruhe, Germany
Michael Koch, Universität der Bundeswehr München, Germany
Peter Sanders, Karlsruher Institut für Technologie (KIT), Germany
Andreas Thor, HFT Leipzig, Germany
Ingo Timm, Universität Trier, Germany
Karin Vosseberg, Hochschule Bremerhaven, Germany
Maria Wimmer, Universität Koblenz-Landau, Germany

© Gesellschaft für Informatik, Bonn 2021
printed by Köllen Druck+Verlag GmbH, Bonn

This book is licensed under a Creative Commons BY-SA 4.0 licence.

Vorwort

Die Gesellschaft für Informatik e.V. (GI) vergibt gemeinsam mit der Schweizer Informatik Gesellschaft (SI) und der Österreichischen Computergesellschaft (OCG) jährlich einen Preis für eine hervorragende Dissertation im Bereich der Informatik. Deutsche, österreichische und schweizer Universitäten und Hochschulen schlagen jeweils eine ausgezeichnet bewertete Dissertation vor, die zur Weiterentwicklung im Bereich Informatik und / oder in den Anwendungsgebieten beiträgt oder die Wechselwirkung zwischen Informatik und Gesellschaft untersucht. Somit sind die im Auswahlverfahren vorgeschlagenen Kandidatinnen und Kandidaten bereits „Preisträger“ ihrer Hochschule.

Die 25 Einreichungen für das Jahr 2019 belegen die Bedeutung des Dissertationspreises. Leider konnten wir in diesem Jahr aufgrund der Corona-Pandemie das Kolloquium zum Dissertationspreis nicht im Leibniz-Zentrum für Informatik Schloss Dagstuhl durchführen. Vielmehr haben wir online-Vorträge der Nominierten organisiert, die verteilt über mehrere Tage abliefen. Das wissenschaftliche Niveau der Vorträge war sehr hoch, die sich daran anschließenden Diskussionen aber leider doch zeitlich sehr beschränkt.

Wie in jedem Jahr fiel es dem Nominierungsausschuss sehr schwer, eine Dissertation auszuwählen, die durch den Preis besonders gewürdigt wird. Mit der Präsentation aller vorgeschlagenen Arbeiten in diesem Band wird die Ungerechtigkeit, eine aus mehreren ebenbürtigen Dissertationen hervorzuheben, etwas ausgeglichen. Der 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 genannten Gesellschaften zeichnen Herrn Dr. Jakub Tarnawski für seine Arbeit „New Graph Algorithms via Polyhedral Techniques“ mit dem Dissertationspreis 2019 aus. Dr. Tarnawski hat zwei herausragende Ergebnisse erzielt: den ersten Approximationsalgorithmus für das asymmetrische Problem des Handlungsreisenden, der polynomielle Laufzeit aufweist und einen Approximationsfaktor garantiert, der unabhängig von der Anzahl der Knoten im Graph ist, und einen deterministischen parallelen Algorithmus für das perfekte Matching. Mit dieser Preisverleihung wird eine herausragende algorithmische Arbeit mit signifikanten methodischen Innovationen gewürdigt.

Ein großer Dank gilt dem Nominierungsausschuss für die sehr effiziente und konstruktive Arbeit. Desweiteren möchte ich mich besonders bedanken bei Frau Sylvia Wünsch für die Organisation der online-Vorträge, Herrn Rüdiger Reischuk für die technischen Vorbereitung und Durchführung der Vorträge, Herrn Dr. Stefan Sobernig und Frau Dr. Lena Reinfelder für die Zusammenstellung des Bandes und dem Team der Gesellschaft für Informatik e.V. für die technische Unterstützung des Auswahlverfahrens.

Steffen Hölldobler
Dresden im November 2020

Kandidat*innen für den GI-Dissertationspreis 2019

Dr. Arp, Daniel	Technische Universität Braunschweig
Dr. Baltes, Sebastian	The University Adelaide
M.Sc. Bannach, Max	Universität zu Lübeck
Dr. Behnke, Gregor	Albert-Ludwigs-Universität Freiburg
Dr.-Ing. Bornschlegl, Marco Xaver	Fernuniversität Hagen
Dr. Eisenbach, Markus	Technische Universität Ilmenau
Dr. Engel, Heiko	Goethe-Universität Frankfurt a. M.
Dr. Fillbrunn, Alexander	Universität Konstanz
Dr. Goli, Mehran	Universität Bremen
Dr. Goranci, Gramoz	Universität Wien
Dr.-Ing. Hammernik, Kerstin	Technische Universität Graz
Dr.-Ing. Hantke, Simone	Technische Universität München
Dr. Kaminski, Benjamin	RWTH Aachen
Dr. rer. nat. Kohl, Lisa Maria	Karlsruher Institut für Technologie
Dr. Malavolta, Giulio	Friedrich-Alexander-Universität Erlangen-Nürnberg
Dr.-Ing. Meuschke, Monique	Otto-von-Guericke-Universität Magdeburg
Dr. Morris, Christopher	Technische Universität Dortmund
Dr. Müller, Florian	Technische Universität Darmstadt
Dr.-Ing. Pensel, Maximilian	Technische Universität Dresden
Dr. rer. nat. Späth, Johannes	Universität Paderborn
Dr. Tarnawski, Jakub	EPFL - Ecole Polytechnique Federale de Lausanne
Dr. Wahl, Florian	Universität Passau
Dr. rer. nat. Wingerath, Wolfram	Universität Hamburg
Dr. Winkler, Kyrill	Technische Universität Wien
Dr. Zulehner, Alwin	Johannis Kepler Universität Linz

Mitglieder des Nominierungsausschusses für den GI-Dissertationspreis 2019

Prof. Dr. Steffen Hölldobler (Vorsitzender)	Technische Universität Dresden
Prof. Dr. Sven Apel	Universität des Saarlandes
Prof. Dr. Abraham Bernstein	Universität Zürich
Prof. Dr.-Ing. Felix Freiling	Universität Erlangen-Nürnberg
Prof. Dr. Hans-Peter Lenhof	Universität des Saarlandes
Prof. Dr. Gustaf Neumann	Wirtschaftsuniversität Wien
Prof. Dr. Rüdiger Reischuk	Universität zu Lübeck
Prof. Dr. Kay Uwe Römer	TU Graz
Prof. Dr. Björn Scheuermann	Humbold-Universität zu Berlin
Prof. Dr. Nicole Schweikardt	Humbold-Universität zu Berlin
Prof. Dr. Myra Spiliopoulou	Otto-von-Guericke-Universität Magdeburg
Prof. Dr. Sabine Süsstrunk	École Polytechnique Fédérale de Lausanne
Prof. Dr. Klaus Wehrle	RWTH Aachen

Inhaltsverzeichnis

Arp, Daniel
Erkennung mobiler Schadsoftware mit maschinellen Lernverfahren 9

Baltes, Sebastian
Arbeitsgewohnheiten und Expertise von Softwareentwicklern 19

Bannach, Max
Parallele Parametrisierte Algorithmen 29

Behnke, Gregor
Hierarchisches Planen durch Propositionale Logik 39

Bornschlegl, Marco
Erweiterte Visuelle Benutzerschnittstellen für Big-Data-Analysen 49

Eisenbach, Markus
Personenwiedererkennung mittels maschineller Lernverfahren 59

Engel, Heiko
Auslesekarte zur schnellen Verarbeitung von Detektordaten 69

Fillbrunn, Alexander
Widening mit Hashbasierter Partitionierung des Hypothesenraums 79

Goli, Mehran
Automatisierte Analyse virtueller Prototypen auf der ESL 89

Goranci, Gramoz
Algorithmen für graphbasierte dynamische Probleme 99

Hammernik, Kerstin
Variationsnetzwerke für die medizinische Bildrekonstruktion 109

Hantke, Simone
Intelligentes, spielorientiertes Crowdsourcing für Audioverarbeitung 119

Kaminski, Benjamin
Erweiterte Verifikationskalküle für Probabilistische Programme 129

Kohl, Lisa
Verbesserte Kommunikationskomplexität in der Kryptographie 139

Malavolta, Giulio
Kryptographische Uhren und Anwendungen 149

Meuschke, Monique
Visuelle Risikoanalyse und Therapieplanung zerebraler Aneurysmen 159

Morris, Christopher
Lernen mit Graphen: Kern- und neuronale Methoden 169

Müller, Florian
Around-Body Interaction 179

Pensel, Maximilian
Schlussfolgern in Defeasible Beschreibungslogiken 189

Späth, Johannes
Synchronisierte Pushdown Systeme für Datenfluss-Analysen 199

Tarnawski, Jakub
Neue Graphen-Algorithmen mittels polyedrischer Methoden 209

Wahl, Florian
Methoden zum Monitoring des zirkadianen Rhythmus im Alltag 219

Wingerath, Wolfram
Skalierbare Push-basierte Echtzeitanfragen für Pull-basierte DBs 229

Winkler, Kyrill
Lösbarkeit von Consensus unter Nachrichtengegnern 239

Zulehner, Alwin
Entwurfsautomatisierung für Quantencomputer 249

Effiziente und erklärbare Erkennung mobiler Schadsoftware mittels maschineller Lernmethoden[1]

Daniel Arp[2]

Abstract: Die Verbreitung von Smartphones und Tablets hat in den vergangenen Jahren stark zugenommen. Aufgrund ihrer hohen Popularität haben sich diese Geräte jedoch zugleich auch zu einem lukrativen Ziel für Autoren von Schadsoftware entwickelt, weshalb mittlerweile täglich neue Schadprogramme gefunden werden, insbesondere für das Android-Betriebssystem. Durch das hohe Aufkommen bösartiger Applikationen bieten aktuelle Erkennungsmethoden wie Antivirenprogramme jedoch oft keinen ausreichenden Schutz, da sie meist signaturbasiert arbeiten und somit auf die Bereitstellung zeitnaher Updates für die Erkennung neuer Schadsoftware angewiesen sind.

In dieser Dissertation wird eine richtungsweisende Methode zur Erkennung von mobiler Schadsoftware vorgestellt, die eine effektive Erkennung direkt auf dem Mobilgerät ermöglicht. Hierfür werden Techniken des maschinellen Lernens und der statischen Code-Analyse derart kombiniert, dass selbst auf Mobilgeräten mit geringer Rechenleistung eine zuverlässige Erkennung von Schadsoftware innerhalb weniger Sekunden möglich wird. Im Vergleich zu anderen Methoden kann eine deutliche Steigerung der Erkennungsleistung erzielt werden und das Verfahren liefert darüber hinaus als erstes seiner Art interpretierbare Entscheidungen zurück.

1 Problembeschreibung

Smartphones und Tablets erfreuen sich nach wie vor steigender Beliebtheit, sodass diese Geräte mittlerweile von mehr als 2,5 Milliarden Menschen weltweit verwendet werden [Ne19]. Zu dieser rasanten Verbreitung hat nicht zuletzt Google's Betriebssystem *Android* erheblich beigetragen, welches mittlerweile auf mehr als 85% der Mobilgeräte eingesetzt wird. Durch die hohe Popularität dieses Betriebssystems haben sich diese Geräte allerdings zugleich auch zu einem lukrativen Angriffsziel für Autoren von Schadsoftware entwickelt. So konnte beispielsweise der Antivirenhersteller *GData* im Jahr 2018 mehr als 4 Millionen neuer Instanzen mobiler Schadsoftware identifizieren und bislang ist kein Rückgang dieses Trends zu beobachten [Bu19].

Obwohl für die Erkennung von Schadsoftware (auch *Malware* genannt) bereits verschiedene Lösungsansätze existieren, bieten diese häufig keinen ausreichenden Schutz. Ein Hauptnachteil aktueller Erkennungssoftware liegt etwa darin, dass sie Schadapplikationen meist nur anhand bestimmter Muster, sogenannter Signaturen, identifizieren, die zuvor manuell erstellt werden müssen und somit für neue Schadapplikationen nicht vorhanden sind [Wr17, MKK07]. Durch das hohe Aufkommen neuer Malware können passende Er-

[1] Englischer Titel der Dissertation: "Efficient and Explainable Detection of Mobile Malware with Machine Learning"

[2] Technische Universität Braunschweig, d.arp@tu-braunschweig.de

CC BY SA

kennungssignaturen jedoch häufig nicht rechtzeitig bereitgestellt werden, sodass Mobilgeräte trotz installiertem Antivirenprogramm infiziert werden können.

Eine vielversprechende Forschungsrichtung, um auf die Bedrohung durch mobile Malware zu reagieren, stellt der Einsatz maschineller Lernmethoden dar. Diese Algorithmen erlauben die automatische Extraktion effektiver Erkennungsmuster aus großen Datenbeständen, bringen jedoch auch mehrere Herausforderungen mit sich, die den Einsatz dieser Techniken auf Mobilgeräten bislang verhindert haben. Zum einen benötigen diese Techniken erhebliche Rechenressourcen und sind damit normalerweise für den Einsatz direkt auf dem Mobilgerät nicht geeignet. Weiterhin sind die gelernten Modelle meist nicht interpretierbar, sodass Malware-Analysten und Smartphone-Benutzer die vom System getroffenen Entscheidungen nicht nachvollziehen können. Ferner sind zudem Angriffe auf diese Systeme denkbar, die ihre Erkennungsrate in der Praxis deutlich reduzieren können.

1.1 Beiträge der Dissertation

In dieser Dissertation wird ein richtungsweisendes Verfahren zur Erkennung von mobiler Schadsoftware vorgestellt, das die zuvor diskutierten Herausforderungen beim Einsatz maschineller Lernmethoden auf Mobilgeräten adressiert und somit als erste Methode eine effiziente, effektive und zugleich interpretierbare Erkennung von Schadsoftware direkt auf dem Mobilgerät ermöglicht. Die Dissertation kann dabei grob in drei unterschiedliche Schwerpunkte unterteilt werden [Ar19]:

1. *Ultraschallbasierte Schadsoftware.* Um die Notwendigkeit zur Entwicklung neuer Erkennungsansätze zu motivieren, werden zunächst die Ergebnisse einer Studie über eine fortgeschrittene Schadsoftware präsentiert, die mittels eines ultraschallbasierten Seitenkanals persönliche Daten von Smartphone-Nutzern ausspioniert. In dieser Studie konnten insgesamt 234 Applikationen mit der entsprechenden Funktionalität identifiziert werden [Ar17]. In Zusammenarbeit mit Google konnten allerdings alle betroffenen Applikationen aus dem *GooglePlay Store* entfernt werden, sodass diese kein Sicherheitsrisiko mehr darstellen. Im Rahmen der Studie werden die Schwächen signaturbasierter Ansätze anhand konkreter Beispielen erläutert und die Notwendigkeit zur Entwicklung lernbasierter Erkennungsansätze ersichtlich.
2. *Lernbasierte Erkennung.* Anschließend wird eine lernbasierte Methode names DREBIN vorgestellt, die eine zuverlässige Erkennung verschiedener Typen von Schadsoftware direkt auf dem Mobilgerät erlaubt. Hierfür analysiert DREBIN einen großen Datenbestand aus gutartigen und bösartigen Applikationen zunächst statisch und leitet die notwendigen Muster zur Erkennung von Schadsoftware mittels maschineller Lernmethoden automatisch ab. Das daraus resultierende Erkennungsmodell kann anschließend auf dem Mobilgerät verwendet werden. In einer Vielzahl von Experimenten wird gezeigt, dass DREBIN bei geringer Laufzeit meist deutlich bessere Ergebnisse erzielt als vergleichbare Methoden [Ar14].
3. *Analyse des Erkennungsmodells.* Das Erkennungsmodell wird im Detail analysiert, wobei unter anderem die Erklärbarkeit sowie die Robustheit der Methode gegenüber

gezielten Angriffen untersucht wird. Im Gegensatz zu anderen Verfahren liefert DREBIN aussagekräftige Erklärungen für seine Entscheidungen, was durch mehrere Experimente untermauert wird. Weiterhin ist es eines der ersten Erkennungssysteme, dessen Robustheit gegenüber realistischen Angriffen systematisch untersucht und verbessert wird [Ar14, De19].

Im Folgenden wird die Studie über ultraschallbasierte Malware (Abschnitt 2), sowie die Methode zur Erkennung von Schadsoftware kurz vorgestellt (Abschnitt 3). Abschließend werden in Abschnitt 4 die wichtigsten Ergebnisse der Dissertation kurz zusammengefasst. Für eine detaillierte Diskussion der Ergebnisse sei auf die Dissertation verwiesen [Ar19].

2 Ultraschallbasierte Schadapplikationen

Um die Notwendigkeit zur Entwicklung neuer Erkennungsmethoden zu motivieren und an die Thematik der mobilen Schadsoftware heranzuführen, werden zunächst die Ergebnisse einer Studie über die Verwendung eines ultraschallbasierten Seitenkanals in mobilen Applikationen (kurz *Apps*) vorgestellt. Hierbei kann in einem Fall gezeigt werden, dass diese Technologie von einem der Anbieter missbraucht wird, um das Konsumverhalten von Mobilnutzern heimlich auszuspionieren.

In der Studie werden Software-Lösungen von drei verschiedenen Anbieter analysiert, die im Frequenzbereich zwischen 18 kHz und 20 kHz Informationen übertragen. Die Übertragung von Informationen in diesem ultraschallnahen Frequenzbereich hat den Vorteil, dass sie für die meisten Menschen nicht hörbar ist und dadurch für verschiedene Zwecke verwendet werden kann. Abbildung 1 zeigt etwa ein Beispiel, in dem zusätzliche Informationen zusammen mit einem Musikstück übertragen werden. Während der Einsatz dieser Technologie bei zwei der Anbieter mit dem Einverständnis der Mobilnutzer geschieht, kann bei der Analyse des *Software Development Kits (SDK)* des Anbieters *SilverPush* festgestellt werden, dass personenbezogene Daten der Nutzer ohne deren Wissen gesammelt und an den Anbieter gesendet werden.

Das SDK dieses Anbieters kann in verschiedene mobile Apps integriert werden und erlaubt dem Anbieter durch die Verwendung des Ultraschall-Seitenkanals ein geräteübergreifendes Profil des Konsumverhaltens einer Nutzerin zu erstellen. Dies ermöglicht wiederum das Anzeigen personalisierter Werbung. Hierfür wird von Apps mit diesem SDK ein Hintergrundprozess gestartet, der kontinuierlich die Umgebungsgeräusche nach speziellen, hochfrequenten Signalen absucht. Diese sogenannten *Audio Beacons* werden im Audiosignal verschiedener Medien, wie etwa Fernsehwerbung, eingebettet und erlauben SilverPush das Konsumverhalten einer Benutzerin geräteübergreifend zu verfolgen. Weiterhin sammelt die Spionagesoftware eine Vielzahl von sensiblen Daten der Geräte, einschließlich der Telefonnummern von Nutzern, und sendet diese zusammen mit anderen personenbezogenen Informationen an die Entwickler von SilverPush.

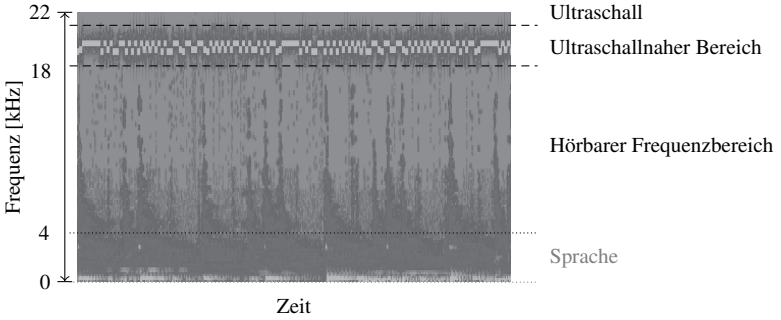


Abb. 1: Beispiel eines im Frequenzbereich zwischen 18 und 20 kHz übertragenen Signals.

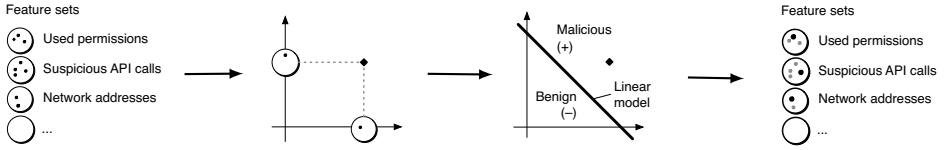
Ergebnisse. Neben einer detaillierten Analyse der Schadsoftware wird auch ihre Verbreitung untersucht, da der Anbieter nicht bekanntgegeben hat mit welchen Firmen er kollaboriert. Hierzu wird ein Datenbestand mit mehr als 1,3 Millionen mobilen Applikationen analysiert. Mit Hilfe speziell auf die Erkennung dieser Schadsoftware zugeschnittenen Techniken können insgesamt 234 Android-Applikationen identifiziert werden, die das SDK enthalten, wobei einige Applikationen millionenfach aus dem GooglePlay Store heruntergeladen wurden. Durch eine Zusammenarbeit mit Google konnten alle Applikationen mit dem Silverpush-SDK aus dem Store entfernt werden, sodass von dieser speziellen Spionagesoftware keine Gefahr für die Privatsphäre von Mobilnutzern mehr ausgehen sollte. Durch die hohe Konzentration sensibler Nutzerdaten bleiben Smartphones allerdings weiterhin ein beliebtes Ziel von Malware-Autoren und dubiosen Werbefirmen. Nicht zuletzt zeigt der Fall auch die Notwendigkeit für die Entwicklung von neuen Erkennungstechniken für mobile Schadsoftware.

3 Lernbasierte Erkennung mobiler Schadsoftware

Die im vorangegangenen Abschnitt beschriebene Spionagesoftware zeigt eindrucksvoll die rasante Weiterentwicklung im Bereich der mobilen Malware. Eine aussichtsreiche Richtung dieser Entwicklung zu begegnen stellt der Einsatz maschineller Lernmethoden dar, da diese Techniken die automatische Extraktion effektiver Erkennungsmuster aus großen Datenbeständen ermöglichen. Die in der Dissertation vorgestellte Methode namens DREBIN nutzt diese Techniken, um eine effektive, effiziente und zugleich interpretierbare Erkennung von mobiler Schadsoftware zu bewerkstelligen. Die Details der Methode werden in den folgenden Abschnitten erläutert.

3.1 Überblick der Methode

Zur Erkennung von Schadsoftware auf einem Mobilgerät muss DREBIN Applikationen in eine Repräsentation überführen, die für die Anwendung von maschinellen Lernalgorithmen geeignet ist. Hierzu wird eine Applikation zunächst statisch analysiert und relevante



(a) Statische Analyse (b) Vektorraum-Einbettung (c) Lernbasierte Erkennung (d) Erklärung

Abb. 2: Schematische Darstellung der verschiedenen Schritte des Erkennungssystems

Merkmale aus der Applikation extrahiert, die es anschließend erlauben, Android-Apps in eine neuartige Vektorrepräsentation zu überführen. In dem daraus resultierenden hochdimensionalen Vektorraum können Applikationen schließlich weiter analysiert und klassifiziert werden. Ein Überblick über diesen Prozess ist schematisch in Abbildung 2 dargestellt. Dabei werden im Detail folgende Schritte durchlaufen:

a) *Statische Analyse.* Im ersten Schritt führt DREBIN eine effiziente Code-Analyse der Android-Applikation durch und extrahiert dabei unterschiedliche Merkmalstypen aus verschiedenen Bestandteilen einer Anwendung.

b) *Vektorraum-Einbettung.* Anschließend werden die Merkmale in einen hochdimensionalen Vektorraum abgebildet, der es ermöglicht, nach relevanten Mustern und markanten Kombinationen von Merkmalen zu suchen.

c) *Lernbasierte Erkennung.* Durch die Überführung in den Vektorraum wird die Anwendung von Lernalgorithmen möglich, um Schadsoftware effektiv zu erkennen.

d) *Interpretation.* Schließlich werden Merkmale identifiziert, die auf die Entscheidung einen hohen Einfluss hatten und der Benutzerin oder Malware-Analystin präsentiert.

Die folgenden Abschnitte legen die einzelnen Schritte der Methode etwas ausführlicher dar. Abschließend wird die Robustheit der Methode diskutiert. Für Details sei auf die Ausführungen in der Dissertation [Ar19] beziehungsweise der entsprechenden Publikationen [Ar14, De19] verwiesen.

3.1.1 Statische Analyse mobiler Applikationen

Zunächst wird die mobile Applikation direkt auf dem Mobilgerät statisch analysiert. Aufgrund der Hardwarelimitierungen von Mobilgeräten muss dies besonders effizient und ressourcensparend erfolgen, um der Benutzerin in annehmbarer Zeit ein Ergebnis präsentieren zu können. Dafür analysiert DREBIN die sogenannte *Manifest*-Datei und den disassemblierten *Dalvik Bytecode* einer App. Hierbei werden insgesamt 8 verschiedene Merkmalskategorien aus der Anwendung extrahiert.

- *Android-Manifest.* Die Manifest-Datei ist fester Bestandteil einer jeden Android-Applikation und enthält verschiedene Meta-Informationen, die während der Instal-

lation und zur Laufzeit einer Anwendung benötigt werden. Aus dieser Datei extrahiert DREBIN verschiedene Informationen, wie die angeforderten Berechtigungen einer Applikation und die Namen der Komponenten, aus der sich die Anwendung zusammensetzt.

- *Dalvik Bytecode.* Android-Applikationen werden meist in Java oder Kotlin geschrieben und dann in sogenannten Dalvik Bytecode übersetzt. DREBIN disassembliert zunächst den Bytecode einer Applikation und extrahiert anschließend verschiedene Merkmale, die zur Erkennung von Schadsoftware relevant sein können. Dies schließt beispielweise bestimmte Aufrufe der Android-API mit ein, die häufig in mobiler Schadsoftware zu finden sind.

Im Gegensatz zu vorherigen Methoden führt DREBIN eine sehr breite Analyse der Applikationen durch und kann so erstmals verschiedene Varianten von Schadsoftware modellieren und erkennen.

3.1.2 Einbettung in den Vektorraum

Jede der von DREBIN extrahierten Merkmalskategorien bildet eine Menge von Zeichenketten. Da die meisten maschinellen Lernmethoden allerdings auf Vektorräumen agieren, müssen Android-Applikationen entsprechend erst als Vektoren repräsentiert werden können. Um zu einer passenden Vektorrepräsentation zu gelangen, werden die einzelnen Merkmalskategorien zunächst zu einer Gesamtmenge S vereint, die die Menge aller zur Trainingszeit extrahierten Merkmale beschreibt.

Die vereinte Menge S erlaubt anschließend die Definition eines $|S|$-dimensionalen Vektorraums, in dem jede Dimension entweder den Wert 0 oder 1 annimmt. Um eine Android-Applikation $z \in \mathscr{Z}$ in diesen Vektorraum zu überführen, wird eine Mappingfunktion $\varphi : \mathscr{Z} \to \{0,1\}^{|S|}$ verwendet. Diese bildet die Applikation z aus der Menge aller Android-Applikationen $\mathscr{Z}$ auf einen Vektor $\varphi(z) \in \{0,1\}^{|S|}$ ab, in welchem die entsprechenden Dimensionen aller aus der App extrahierten Merkmale auf 1 und aller nicht-vorhandenen Merkmale auf 0 gesetzt werden. Die Vektorrepräsentation $\varphi(z)$ einer Applikation z könnte beispielweise wie folgt aussehen:

$$\varphi(z) \mapsto \begin{pmatrix} \cdots \\ 0 \\ 1 \\ \cdots \\ 1 \\ 0 \\ \cdots \end{pmatrix} \begin{matrix} \cdots \\ \texttt{HARDWARE::android.hardware.wifi} \\ \texttt{HARDWARE::android.hardware.telephony} \\ \cdots \\ \texttt{REQ_PERMISSION::SEND_SMS} \\ \texttt{REQ_PERMISSION::DELETE_PACKAGES} \\ \cdots \end{matrix}$$

Durch diese neuartige Einbettung von Android-Applikationen wird es möglich, diese effizient zu analysieren und dadurch häufige Muster mobiler Schadsoftware automatisch zu

extrahieren. Der Ansatz baut dabei auf Konzepten aus der Sprachverarbeitung [SWY75] und Bioinformatik [LEN02] auf und verbindet so moderne Konzepte der Datenanalyse mit aktuellen Problemstellungen aus der IT-Sicherheit.

3.1.3 Erkennung von Schadsoftware

In dem resultierenden Vektorraum liegen Applikationen geometrisch dichter beieinander, die ähnliche Merkmale aufweisen, als Applikationen, die sehr verschieden sind. Im nächsten Schritt muss daher eine Funktion gefunden werden, die in diesem Raum gutartige und schädliche Applikationen geometrisch voneinander trennen kann.

Für diese Aufgabe existiert eine Vielzahl von Lernalgorithmen, jedoch erfordern viele einen hohen Rechenaufwand und nur wenige erlauben eine Interpretation der zurückgelieferten Ergebnisse. DREBIN verwendet daher einen bereits gut verstandenen Lernalgorithmus — die lineare *Support Vector Machine* (SVM) [Va79]. Diese ist für die Anwendung besonders gut geeignet, da durch die gezielte Ausnutzung charakteristischer Eigenschaften des Algorithmus eine sehr effiziente und zugleich erklärbare Erkennung von Schadsoftware möglich wird.

Die Verwendung der linearen SVM erlaubt schließlich die Klassifizierung von Apps direkt auf dem Mobilgerät. Hierfür kann der Umstand ausgenutzt werden, dass nur die in einer Applikation vorhandenen Merkmale und auch diese lediglich bei ihrem ersten Auftreten im Bytecode in die Entscheidung einfließen. Somit steigt die Laufzeit zur Erkennung von Apps nicht einmal linear mit der Größe des Bytecodes an.

3.1.4 Interpretation der Ergebnisse

Ein großer Nachteil vieler maschineller Lernmethoden besteht darin, dass die von ihnen zurückgelieferten Modelle meist nicht interpretierbar sind [SP10]. DREBIN ist hingegen in der Lage, Erklärungen für seine Entscheidungen zurückzuliefern.

Das Erkennungsmodell wurde bei DREBIN so gewählt, dass es sich um eine lineare Funktion handelt und somit für jedes extrahierte Merkmal der Einfluss an der Gesamtentscheidung exakt bestimmt werden kann. Während der Berechnung der Erkennungsfunktion werden dazu die k Merkmale gespeichert, die den größten Einfluss auf die Entscheidung hatten. Diese können anschließend in verständlich aufbereiteter Form der Smartphone-Nutzerin oder einer Analystin präsentiert werden. Somit wird es möglich nachzuvollziehen, weshalb eine App von DREBIN als gutartig oder bösartig eingestuft wurde.

3.1.5 Angriffe auf das Lernmodell

Die Sicherheit lernbasierter Systeme gegen gezielte Angriffe ist ein wichtiger Bestandteil aktueller Forschung. Auch in der Dissertation wird die Durchführbarkeit von *Evasion-*

angriffen [De19] auf das Erkennungsmodell untersucht, da diese Art von Angriffen eine realistische Bedrohung für lernbasierte Erkennungssysteme in der Praxis darstellt.

Bei einem Evasionangriff manipuliert die Angreiferin gezielt die Merkmale einer Applikation, um eine Falsch-Klassifikation des Systems herbeizuführen. Der Angriff ist hierbei allerdings gewissen Limitierungen unterworfen, da die Angreiferin in der Praxis nicht beliebige Merkmale manipulieren kann, ohne die Funktionalität einer App zu beeinflussen. In der Dissertation wird gezeigt, dass diese Angriffe unter bestimmten Umständen tatsächlich die Erkennungsleistung des Systems reduzieren können, da einzelne Merkmale auf die Entscheidungsfunktion der SVM einen hohen Einfluss haben können. Gelingt es der Angreiferin eben diese Merkmale zu ermitteln und zu manipulieren, sinkt die Erkennungsleistung des Systems signifikant.

Um hier entgegenzuwirken, kann etwa das Optimierungsproblem der SVM derart modifiziert werden, sodass sichergestellt wird, dass eine Entscheidung nicht nur von wenigen Merkmalen abhängt. Der resultierende Klassifikator, genannt Sec-SVM, weist dadurch eine deutlich höhere Robustheit gegenüber gezielten Angriffen auf.

4 Zusammenfassung der wichtigsten Ergebnisse

In der Dissertation wird eine neuartige Methode namens DREBIN vorgestellt, die eine effiziente, effektive und interpretierbare Erkennung von mobiler Schadsoftware direkt auf dem Mobilgerät ermöglicht. Dies wird durch zahlreiche Experimente untermauert, in denen unter anderem die Erkennungsleistung, Interpretierbarkeit, sowie Robustheit gegen gezielte Angriffe untersucht wird. Zur Evaluation werden mehrere Datensätze verwendet, die sich aus insgesamt mehr als 450.000 Android-Applikationen aus dem Zeitraum zwischen 2010 und 2017 zusammensetzen.

Erkennungsleistung. Beim Vergleich mit anderen Methoden kann gezeigt werden, dass DREBIN meist deutlich bessere Ergebnisse erzielt und eine zuverlässige Erkennung von Schadsoftware ermöglicht, einschließlich der in Abschnitt 2 diskutierten ultraschallbasierten Spionage-Applikationen. In den meisten Experimenten kann eine Erkennungsrate von über 90% bei einer geringen Falsch-Positiv-Rate von gerade einmal 1% erreicht werden. Dies entspricht einem Falsch-Alarm pro 100 installierten Applikationen, was auch im praktischen Einsatz akzeptabel zu sein scheint, insbesondere wenn die Detektion direkt auf dem Gerät erfolgt.

Auch bei einem Vergleich mit populären Antivirenscannern erzielt DREBIN vergleichbare und in vielen Fällen sogar bessere Ergebnisse. Während dieser Untersuchung zeigt sich auch, dass Antivirenprodukte häufig eine gewisse Vorlaufzeit für die Bereitstellung effektiver Erkennungssignaturen benötigen. Dies kann im schlimmsten Fall zu einer unerkannten Infektion des Mobilgeräts führen.

Laufzeit. Zur Evaluation des Laufzeitverhaltens wird ein Prototyp der Applikation implementiert und die Dauer zur Ausgabe eines Ergebnisses auf fünf Mobilgeräten mit unterschiedlicher Rechenleistung untersucht. Die Laufzeit wird hierbei mit 100 populären Applikationen aus dem GooglePlay Store durchgeführt. Im Durchschnitt dauert es auf den Geräten weniger als 15 Sekunden bis eine Entscheidung zurückgeliefert werden kann und selbst auf sehr rechenschwachen Geräten nie länger als eine Minute.

Erklärbarkeit. Im Gegensatz zu allen zuvor vorgestellten lernbasierten Erkennungsmethoden für mobile Malware stellt die Interpretierbarkeit einen essentiellen Bestandteil von DREBIN dar. In einer Evaluation mit populären Malware-Familien kann gezeigt werden, dass DREBIN in der Lage ist, automatisch charakteristische Merkmale dieser Familien zu identifizieren und zur Erkennung zu nutzen. Dieses Ergebnis wird sowohl von den präsentierten Erklärungen als auch einer detaillierten Analyse der Generalisierungseigenschaften des gelernten Modells gestützt.

Robustheit. Zwar kann gezeigt werden, dass Evasionangriffe unter bestimmten Voraussetzungen die Erkennungsleistung von DREBIN beeinflussen können, allerdings existieren unterschiedliche Möglichkeiten den Einfluss solcher Angriffe stark zu reduzieren. So verhindert etwa der Einsatz der Sec-SVM, dass eine Entscheidung nur von einer kleinen Zahl von Merkmalen abhängt, die der Angreifer unter Umständen manipulieren kann. Infolgedessen erkennt die Sec-SVM beispielsweise noch weit über 80% der Schadsoftware, selbst wenn der Angreifer mehrere Merkmale manipulieren kann.

5 Fazit

Zusammenfassend zeigen die in dieser Dissertation vorgestellten Ergebnisse, dass die Verwendung maschineller Lernverfahren einen vielversprechenden Ansatz darstellt, um die Sicherheit mobiler Geräte zu verbessern. Zwar können auch diese Techniken die Bedrohung durch mobile Schadanwendungen nicht vollends beseitigen, aber zumindest die Infektion von Mobilgeräten deutlich zu erschweren. Die vorgestellte Methode leistet hier einen wichtigen Beitrag und dient Wissenschaftlern weltweit als Grundlage für weiterführende Forschung in diesem Bereich.

Literaturverzeichnis

[Ar14] Arp, Daniel; Spreitzenbarth, Michael; Hübner, Malte; Gascon, Hugo; Rieck, Konrad: Drebin: Efficient and Explainable Detection of Android Malware in Your Pocket. In: Proc. of Network and Distributed System Security Symposium (NDSS). 2014.

[Ar17] Arp, Daniel; Quiring, Erwin; Wressnegger, Christian; Rieck, Konrad: Privacy Threats through Ultrasonic Side Channels on Mobile Devices. In: Proc. of IEEE European Symposium on Security and Privacy (EuroS&P). 2017.

[Ar19] Arp, Daniel: Efficient and Explainable Detection of Mobile Malware with Machine Learning. Dissertation, Technische Universität Braunschweig, 2019.

[Bu19] Bundeskriminalamt: Bundeslagebild Cybercrime 2018. Bericht, 2019.

[De19] Demontis, A.; Melis, M.; Biggio, B.; Maiorca, D.; Arp, D.; Rieck, K.; Corona, I.; Giacinto, G.; Roli, F.: Yes, Machine Learning Can Be More Secure! A Case Study on Android Malware Detection. IEEE Transactions on Dependable and Secure Computing (TDSC), 2019.

[LEN02] Leslie, Christina S.; Eskin, Eleazar; Noble, William Stafford: The Spectrum Kernel: A String Kernel for SVM Protein Classification. In: Proc. of the Pacific Symposium on Biocomputing (PSB). S. 566–575, 2002.

[MKK07] Moser, Andreas; Kruegel, Christopher; Kirda, Engin: Limits of Static Analysis for Malware Detection. In: Proc. of Annual Computer Security Applications Conference (ACSAC). 2007.

[Ne19] Newzoo: Global Mobile Market Report. Bericht, 2019.

[SP10] Sommer, R.; Paxson, V.: Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. In: Proc. of IEEE Symposium on Security and Privacy (S&P). 2010.

[SWY75] Salton, G.; Wang, A.; Yang, C. S.: A Vector Space Model for Automatic Indexing. Communication of the ACM (CACM), 18(11):613–620, 1975.

[Va79] Vapnik, V.N.: Estimation of Dependences Based on Empirical Data [in Russian]. Nauka, 1979.

[Wr17] Wressnegger, Christian; Freeman, Kevin; Yamaguchi, Fabian; Rieck, Konrad: Automatically Inferring Malware Signatures for Anti-Virus Assisted Attacks. In: Proc. of ACM Asia Conference on Computer Computer and Communications Security (ASIA CCS). 2017.

Daniel Arp, geboren am 19. März 1986 in Berlin, ist derzeit Post-Doktorand am Institut für Systemsicherheit der Technischen Universität Braunschweig. Er hat Technische Informatik an der Technischen Universität Berlin studiert und anschließend an der Georg-August-Universität Göttingen und der Technische Universität Braunschweig im Fachbereich IT-Sicherheit promoviert. In seiner Dissertation stellt er ein neuartiges Verfahren zur Erkennung mobiler Schadsoftware vor und wurde für seinen Beitrag im Bereich der Malwareerkennung dafür bereits vom Wirtschaftsministerium Niedersachsen als “KI Talent 2019” ausgezeichnet. Während seiner Promotion veröffentlichte er zahlreiche wissenschaftliche Beiträge, die insgesamt mehr als 2.000 Mal zitiert wurden.

Arbeitsgewohnheiten und Expertise von Softwareentwicklern: Empirische Studien zur Verwendung von Skizzen, Implikationen von Code-Plagiaten und dem Aufbau von Expertise[1]

Sebastian Baltes[2]

Abstract: Eine wichtige Voraussetzung für die Optimierung von Arbeitsabläufen und die Entwicklung neuer Werkzeuge in Softwareprojekten ist es, die Arbeitsweise von Softwareentwicklern und die daraus resultierenden Bedürfnisse zu kennen und zu verstehen. Diese Dissertation beschreibt empirische Untersuchungen dreier noch wenig erforschter Bereiche der Softwareentwicklung, stellt neue Werkzeuge vor und gibt Handlungsempfehlungen basierend auf den durchgeführten Untersuchungen. Zunächst wird illustriert wie Softwareentwickler Skizzen und Diagramme in ihrer täglichen Arbeit einsetzen. Anschließend wird der Umgang von Softwareentwicklern mit Code-Fragmenten beschrieben, die auf der populären Onlineplattform Stack Overflow bereitgestellt werden. Nach einer detaillierten Erläuterung dieser beiden Aspekte der täglichen Arbeit von Softwareentwicklern stellen wir ein erstes Modell vor, das empirisch fundiert wichtige Eigenschaften eines Experten in der Softwareentwicklung darstellt und Faktoren auflistet, die die kontinuierliche Weiterbildung von Softwareentwicklern unterstützen. Neben diesen drei Forschungsrichtungen gehen wir auf methodische Erkenntnisse ein und stellen den offenen Datensatz SOTorrent vor, der im Rahmen dieser Dissertation entstand.

1 Einleitung

Ein "work habit", also eine Arbeitsgewohnheit, kann die tägliche Arbeit von Softwareentwicklern positiv wie negativ beeinflussen. Diese Gewohnheiten und die daraus resultierenden Bedürfnisse zu kennen und zu verstehen ist eine notwendige Voraussetzung dafür, bestehende Softwareentwicklungsprozesse und -werkzeuge zu verbessern. Die Software-Engineering-Forschungscommunity wird allerdings oft dafür kritisiert, nicht die Probleme zu adressieren, auf die Entwickler in der Praxis tatsächlich stoßen. Gleichzeitig treffen Entwickler Entscheidungen eher auf Grundlage ihrer persönlichen Erfahrung als auf Grundlage von empirisch fundierten Erkenntnissen. Um diese Lücke zwischen der akademischen Welt und der Praxis zu schließen – oder zumindest zu verkleinern – präsentieren wir in dieser Arbeit [Ba19] empirisch fundierte Ergebnisse und Werkzeuge, die sich mit verschiedenen Arbeitsgewohnheit von Softwareentwicklern beschäftigen. Zusätzlich beleuchten wir was

[1] Englischer Titel der Dissertation: "Software Developers' Work Habits and Expertise: Empirical Studies on Sketching, Code Plagiarism, and Expertise Development"

[2] QAware GmbH, Deutschland, und University Adelaide, Australien, sebastian.baltes@adelaide.edu.au

CC BY SA

Experten in der Softwareentwicklung auszeichnet und welche Faktoren die Entstehung einer solchen Expertise begünstigen oder behindern können. Im Folgenden wird zunächst ein Überblick über die zentralen Kapitel der Dissertation gegeben, bevor die Ergebnisse zusammengefasst und ein Ausblick auf zukünftige Forschungsrichtungen gegeben wird.

2 Skizzieren: Wie verwenden Softwareentwickler Skizzen und Diagramme?

Kommunikation ist allgegenwärtig in der Softwareentwicklung. Anforderungen werden von potentiellen Benutzern zu den Softwareentwicklern kommuniziert, die Architektur der Software wird innerhalb des Entwicklungsteams kommuniziert, Entwickler kommunizieren während Pair-Programming-Sitzungen miteinander, und nach der Bereitstellung der Software werden Probleme von Benutzern zu den Entwicklern kommuniziert. An solchen Informationsflüssen sind diverse Kommunikationskanäle beteiligt, so zum Beispiel "face-to-face"-Kommunikation, E-Mails, Videokonferenzen, oder andere Kollaborationswerkzeuge. Insbesondere wenn alle Entwickler gemeinsam an einem Ort "colocated" arbeiten, verwenden Sie – oft informelle – Skizzen und Diagramme zur Kommunikation. Diese visuellen Artefakte umfassen verschiedene Medienarten wie Papier, Whiteboards oder Computerwerkzeuge und unterstützen Entwickler beim Entwerfen neuer und beim Verstehen bestehender Softwaresysteme. Nichtsdestotrotz fehlte zu Beginn unserer Forschung zum Thema Skizzen und Diagramme in der Softwareentwicklung ein umfassendes Bild darüber, wie und warum Entwickler diese visuellen Artefakte verwenden. Deshalb befasst sich der erste Teil dieser Dissertation zunächst damit, verschiedene Dimensionen von Skizzen und Diagrammen in der Softwareentwicklung aus verwandten Arbeiten und eigenen empirischen Studien abzuleiten. Auf Grundlage dieser Dimensionen entwarfen wir dann ein Studiendesign, das es uns erlaubte zu erforschen, wie Entwickler Skizzen und Diagramme in ihrer täglichen Arbeit verwenden. Unsere Forschung beinhaltete eine explorative Feldstudie in drei Softwareunternehmen, eine Onlineumfrage unter 394 Softwareentwicklern und eine Beobachtungsstudie in einer kontrollierten Umgebung mit sechs Pair-Programming-Teams. Nachdem wir unsere Ergebnisse und damit den "state of practice" beschreiben, leiten wir daraus Bedürfnisse von Entwicklern im Umgang mit Skizzen und Diagrammen ab und präsentieren weiterhin zwei Werkzeugprototypen, die wir auf Grundlage unserer empirischen Ergebnisse entwickelt haben. Der Inhalt des entsprechenden Kapitels dieser Dissertation basiert auf vier wissenschaftlichen Publikationen, die alle einen Peer-Review-Begutachtungsprozess durchlaufen haben [BD14, BSD14, BHD17, Ba15].

Forschungsbeiträge:

- Eine **Klassifizierung** von Skizzen und Diagrammen in der Softwareentwicklungspraxis, die auf verwandten Arbeiten und einer eigenen Feldstudie in drei Softwareunternehmen beruht [BD14].
- Eine empirische Untersuchung der 11 identifizierten Dimensionen von Skizzen und Diagrammen in der Softwareentwicklung, basierend auf einer **Onlineumfrage** mit 394 Softwareentwicklern [BD14].

- Eine Analyse darüber, wie Entwickler in einer **Pair-Programming**-Umgebung miteinander kommunizieren, wenn sie **Performanzprobleme** beheben, und insbesondere eine Analyse der **Rolle von Skizzen** in diesem Szenario [Ba15].
- Eine Beschreibung von zwei **Werkzeugprototypen**, die die tatsächlichen Arbeitsabläufe von Entwicklern beim Erstellen und Verwenden von Skizzen und Diagrammen besser unterstützten [BSD14, BHD17].

3 Code-Plagiate: Stack-Overflow-Codefragmente in GitHub-Projekten

Stack Overflow ist heutzutage das populärste Onlineforum für Softwareentwickler. Viele der dort publizierten Beiträge enthalten Codefragmente, die Entwickler in ihre eigenen Projekte übertragen können. Die Verwendung dieser Codefragmente wirft aber Fragen bezüglich deren Wartbarkeit und möglichen Lizenzproblemen auf. Die Creative-Commons-Lizenz, die Stack Overflow verwendet (CC BY-SA), verlangt eine Zuschreibung des kopierten Inhalts ("attribution"), also das Hinzufügen einer Referenz auf die ursprüngliche Frage oder Antwort auf der Plattform, und verlangt die Verwendung einer kompatiblen Lizenz für Werke, die auf dem kopierten Inhalt aufbauen ("share-alike"). Während es eine hitzige Diskussion zum Lizenzmodell von Stack Overflow gab, insbesondere zur Lizenzierung von Quellcode auf der Plattform und der notwendigen Zuschreibung, gab es wenig empirische Evidenz darüber, wie häufig Code von Stack Overflow ohne die verlange Zuschreibung kopiert wird. Um diese Forschungslücke zu schließen, führten wir eine großangelegte empirische Studie durch, um die Verwendung und Zuschreibung von nicht-trivialen Java Codefragmenten aus Stack-Overflow-Antworten in öffentlichen GitHub-Projekten zu untersuchen. Mit Hilfe von drei verschiedenen Data-Mining-Ansätzen triangulierten wir eine Abschätzung für das Verhältnis zwischen zugeschriebenen und nicht zugeschriebenen Verwendungen solcher Codefragmente. Um diese quantitativen Abschätzungen zu komplementieren, befragten wir außerdem Softwareentwickler in zwei Onlineumfragen. In den drei untersuchten Stichproben von GitHub-Projekten lag der Anteil von Projekten mit Referenzen zu Stack-Overflow-Inhalten zwischen 3.3% und 11.9%. Unsere Untersuchungen zeigen, dass nur 1.8% aller untersuchten GitHub-Projekte, die Code von Stack Overflow enthalten, den Code entsprechend der Lizenzbedingungen von CC BY-SA verwenden. Unsere konservative Abschätzung des Anteils von Codefragmenten, die mit der verlangten Zuschreibung der Quelle verwendet werden, liegt bei 25%. Die Mehrheit nicht-trivialer Codefragmente von Stack Overflow werden also ohne Zuschreibung in öffentlichen GitHub-Projekten verwendet. Dies deckt sich auch mit den Ergebnissen unserer Onlineumfragen, in denen etwa die Hälfte der befragten Entwickler zugaben, Code von Stack Overflow ohne die notwendige Zuschreibung zu verwenden. Etwa zwei Drittel der Umfrageteilnehmer gaben allerdings auch an, dass ihnen die Lizenzsituation von Code auf der Plattform Stack Overflow und deren Implikationen nicht bewusst war. Der Inhalt des entsprechenden Kapitels dieser Dissertation basiert hauptsächlich auf einer wissenschaftlichen Zeitschriftenpublikation, die einen Peer-Review-Begutachtungsprozess durchlaufen hat [BD18b]. Außerdem wurden vorläufige Ergebnisse zuvor auf einer Konferenz präsentiert [BKD17].

Forschungsbeiträge:

- Eine fundierte Beschreibung der **Lizenzsituation** von Codefragmenten, die auf der Onlineplattform Stack Overflow veröffentlicht wurden.
- Eine **triangulierte Abschätzung** des **Anteils korrekt zugeschriebener** Codefragmente von Stack Overflow in öffentlichen GitHub-Projekten.
- Eine Analyse potentieller **Lizenzkonflikte** für GitHub-Projekte, die nicht-triviale Codefragmente von Stack Overflow enthalten.
- Eine qualitative Analyse darüber, **wie Softwareentwickler** auf Inhalte von Stack Overflow **verweisen**.
- Eine **Onlineumfrage** die suggeriert, dass vielen **Softwareentwicklern** die Lizenzierung von Inhalten auf Stack Overflow und deren Implikationen **nicht bewusst** ist.

4 Softwareentwicklungsexpertise: Der Weg zu einer ersten Theorie

Softwareentwicklung beinhaltet sehr unterschiedliche Aufgaben, angefangen von der Analyse von Anforderungen über die Implementierung neuer Funktionalitäten bis hin zum Beheben von Fehlern in existierender Software. Um ein Experte in all diesen Aufgaben zu sein, benötigt man verschiedene Fähigkeiten, Wissen und Erfahrung. Bestehende Studien beleuchteten individuelle Aspekte der Softwareentwicklungsexpertise, es fehlte allerdings eine umfassende und übergreifende Theorie zu diesem abstrakten Konzept. Im entsprechenden Kapitel der Dissertation beschreiben wir die erste konzeptionelle Theorie über Softwareentwicklungsexpertise. Unser Modell basiert auf den Ergebnissen von Onlineumfragen mit 335 Softwareentwicklern und verwandten Arbeiten aus Informatik und Psychologie. Wir konzentrierten uns zunächst primär auf Programmiertätigkeiten, unsere Arbeit führte aber bereits in diesem frühen Stadium zu wertvollen Einsichten für andere Forscher, Softwareentwickler und Arbeitgeber. Die Theorie beschreibt wichtige Eigenschaften, die maßgeblich für Expertise in der Softwareentwicklung sind, und Faktoren, die das entstehen einer solchen Expertise fördern oder behindern können. Außerdem zeigen die quantitativen Ergebnisse unserer Onlineumfrage, dass die Einschätzung der eigenen Expertise bei Softwareentwicklern stark kontextabhängig ist und dass Erfahrung nicht notwendigerweise mit Expertise korreliert. Der Inhalt des entsprechenden Kapitels dieser Dissertation basiert hauptsächlich auf einer wissenschaftlichen Publikation, die einen Peer-Review-Begutachtungsprozess durchlaufen hat [BD18a].

Forschungsbeiträge:

- Eine erste **konzeptionelle Theorie** über **Softwareentwicklungsexpertise** basierend auf einer Onlineumfrage mit 335 Softwareentwicklern sowie auf verwandten Arbeiten über Expertise aus Informatik und Psychologie.
- Quantitative Ergebnisse, die auf die **Kontextabhängigkeit** von **Expertise-Selbsteinschätzungen** von Softwareentwicklern hindeuten.
- Ein Ansatz zur **Theoriebildung**, der induktive und deduktive Schritte kombiniert und den andere Software-Engineering-Forscher adaptieren können.

5 Methodische Einsichten: Online-Umfragen mit Softwareentwicklern

Onlineumfragen so wie wir sie für diese Dissertation durchgeführt haben, gelten als wichtiges Werkzeug, um den aktuellen Stand der Praxis zu untersuchen. Insbesondere in der Software-Engineering-Forschung werden solche Umfragen als Werkzeug zum Erforschen und Beschreiben verschiedener Charakteristiken breiter Populationen verwendet. Nichtsdestotrotz stellt das Erreichen professioneller Softwareentwickler mit solchen Umfrage eine schwierige Aufgabe dar. Mit Ausnahme einzelner Unternehmen oder Institutionen, die es Forschern erlauben eine Liste ihrer Angestellten zum Ziehen einer Zufallsstichprobe zu verwenden, ist eine zufällige Auswahl von Umfrageteilnehmern in den meisten Fällen unmöglich. Forscher greifen daher oft auf verfügbare Teilnehmer zurück, was man als *convenience sampling* bezeichnet. Die Verwendung solcher nicht-zufälligen Methoden zur Auswahl von Umfrageteilnehmern führt häufig zu verzerrten Stichproben mit eingeschränkter externer Validität. Um diese Einschränkungen und potentiellen Verzerrungen abzumildern, benötigen Forscher detailliertes Wissen über die Population der Softwareentwickler, die sie untersuchen möchten – allerdings sind die dazu erforderlichen Daten oft nicht verfügbar. Hinzu kommt, dass manche der Methoden zur Auswahl von Umfrageteilnehmern, die Forscher in der Vergangenheit verwendeten, ethisch problematisch sind. Dazu zählt beispielsweise das Kontaktieren von Entwicklern mit Hilfe der E-Mail-Adresse, die sie auf GitHub hinterlegt haben – dies taten Sie allerdings nicht für den Zweck der Kontaktaufnahme durch Forscher. In dieser Dissertation fassen wir die methodischen und ethischen Erkenntnisse zusammen, die wir mit verschiedenen Onlineumfragen gewonnen haben. Der Inhalt des entsprechenden Kapitels dieser Dissertation basiert hauptsächlich auf einer wissenschaftlichen Publikation, die einen Peer-Review-Begutachtungsprozess durchlaufen hat [BD16].

6 Forschungsbeiträge:

- **Erfahrungsberichte** zu verschiedenen Methoden der Rekrutierung von Umfrageteilnehmern.
- Motivation dafür, eine globale **Datenbank mit demographischen Daten über Softwareentwickler** aus verschiedenen Studien aufzubauen, die zum Vergleich herangezogen werden kann, wenn die externe Validität von nicht-zufälligen Stichproben eingeschätzt werden soll.
- Erläuterung **ethischer Implikationen** verschiedener Ansätze zum Kontaktieren von Entwicklern, die von Forschern verwendet wurden und werden.

7 Offene Forschungsdaten: Aufbau und Pflege des SOTorrent-Datensatzes

Für alle Studien, die im Kontext dieser Dissertation durchgeführt wurden, stellt der Autor ergänzende Materialen zur Verfügung, die es anderen Forschern ermöglichen, die Ergebnisse nachzuprüfen. So publizierten wir die (anonymisierten) Daten unserer Studien im langfristig verfügbaren Onlinearchiv Zenodo. Außerdem veröffentlichten wir die Software und Skripte zum Abfragen und Auswerten dieser Daten auf GitHub und archivierten sie zusätzlich auf

Zenodo. Pre- oder Postprints aller Forschungspapiere sind online frei verfügbar. Neben diesen allgemeinen Bemühungen um offene Wissenschaft möchten wir an dieser Stelle den im Rahmen dieser Dissertation entstandenen Datensatz *SOTorrent* hervorheben, den wir entwickelt und veröffentlicht haben, um zukünftige Forschung über Codefragmente auf Stack Overflow zu erleichtern. Dieser Datensatz erlaubt es Forschern, die Evolution von Code auf Stack Overflow auf Ebene individueller Text- und Codeblöcke zu untersuchen, was mit dem offiziell von Stack Overflow zur Verfügung gestellten Datensatz nicht möglich ist. Neben der Verwendung des Datensatzes für unsere eigene Forschung haben wir ihn früh auch für andere Forscher zugänglich gemacht. Dies führte dazu, dass SOTorrent als offizielle *Mining Challenge* der *16th International Conference on Mining Software Repositories* (MSR 2019) ausgewählt wurde [BTD19]. Der Inhalt des entsprechenden Kapitels dieser Dissertation basiert auf zwei wissenschaftlichen Publikationen, die einen Peer-Review-Begutachtungsprozess durchlaufen haben. Ein Konferenzpapier, das die Architektur und Algorithmen hinter dem Datensatz zusammen mit ersten Ergebnissen vorstellt [Ba18] und ein Papier, das die MSR Mining Challenge beschreibt [BTD19]. Wir stellen außerdem zusätzliche Auswertungen vor, die wir für ein noch unveröffentlichtes Forschungspapier durchgeführt haben.

Forschungsbeiträge:

- Ein offener **Datensatz**, der es Forschern erlaubt die Evolution von Inhalten auf Stack Overflow zu untersuchen. Außerdem erleichtert der Datensatz die Analyse der Verbindung von Stack Overflow mit anderen Plattformen wie GitHub durch das Extrahieren von Hyperlinks.
- Eine fundierte **Evaluierung von 134 String-Ähnlichkeitsmetriken** hinsichtlich ihrer Anwendbarkeit für die Rekonstruierung der Versionshistorie von Stack Overflow Text- und Codeblöcken.
- Eine **erste Auswertung der Evolution** von Inhalten auf Stack Overflow, die einen Zusammenhang von Änderungen an Beiträgen und zugehörigen Kommentaren motiviert.
- Eine Auswertung von **Codeduplikaten innerhalb von Stack Overflow** mit einer Diskussion möglicher Lizenzkonflikte und Implikationen für die Wartbarkeit.

8 Zusammenfassung der Ergebnisse und Ausblick

In der hier vorgestellten Dissertation verwendete der Autor verschiedene Forschungsdesigns, um bisher wenig erforschte Aspekte der täglichen Arbeit von Softwareentwicklern zu untersuchen. Zunächst wurde erforscht, wie Softwareentwickler *Skizzen und Diagramme* in ihrer täglichen Arbeit verwenden, mit dem Ziel, Anforderungen für verbesserte Werkzeugunterstützung abzuleiten. Zwei Prototypen für Werkzeuge wurden implementiert und evaluiert. Im zweiten Forschungsprojekt analysierten wir die Angewohnheit von Entwicklern, *nicht-triviale Codefragmente* aus dem populären Onlineforum Stack Overflow in quelloffene GitHub-Projekte zu *kopieren*, ohne sich an die Bedingungen von Stack Overflows Lizenz zu halten. Zusätzlich untersuchten wir, inwiefern Entwicklern die Lizenzsituation und deren Implikationen bewusst ist. Während diese beiden Projekte Analysen von positiven beziehungsweise negativen Angewohnheiten von Entwicklern darstellen, fokussierte das dritte Projekt auf Verhaltensweisen, die Entwickler dabei unterstützen können, Experten in

bestimmten Softwareentwicklungsaufgaben zu werden. Neben Faktoren, die den Aufbau einer solchen Expertise über die Zeit beeinflussen, entwickelten wir eine erste konzeptionelle Theorie zum Strukturieren des abstrakten Konzepts der *Softwareentwicklungsexpertise*.

Skizzieren Um einen Überblick über die Verwendung von *Skizzen und Diagrammen* in der Softwareentwicklung zu erhalten, führten wir mehrere qualitative und quantitative Studien durch, die zeigen, dass Softwareentwickler in ihrer täglichen Arbeit regelmäßig Skizzen und Diagramme erstellen und wiederverwenden. Unsere Onlineumfrage mit 394 Teilnehmern ergab, dass diese Skizzen und Diagramme oft informell sind, aber trotzdem als wertvolle Ressource angesehen werden, die verschiedene Aspekte des Softwareentwicklungsprozesses dokumentieren. Wir zeigen auf, wie diese Artefakte mit dem Quellcode auf verschiedenen Abstraktionsebenen zusammenhängen und dass etwa die Hälfte der erstellten Skizzen und Diagramme potentiell hilfreich zum Verstehen des zugehörigen Quellcodes sind. Da Softwaredokumentation oft nicht vorhanden oder nicht aktuell ist, können diese visuellen Artefakte Softwaredokumentation wie Quellcodekommentare oder Textdokumente ergänzen. Die meisten Skizzen und Diagramme werden jedoch auf analogen Medien wie Papier oder Whiteboards erstellt. Viele dieser analogen Skizzen werden archiviert, unsere Studienteilnehmer erwähnten aber auch technische Probleme wie fehlende Werkzeugunterstützung, um Skizzen zusammen mit dem zugehörigen Quellcode zu archivieren. Basierend auf diesen Ergebnissen entwickelten und evaluierten wir den Werkzeugprototypen *SketchLink*, der Entwickler beim Archivieren von und Zugreifen auf Skizzen, die mit Quellcode in Verbindung stehen, unterstützt. Bezüglich der Evolution von Skizzen deuten unsere qualitativen Ergebnisse darauf hin, dass es üblich ist, Skizzen zunächst auf analogen Medien wie Papier oder Whiteboards zu erstellen. Nach potentiell mehreren Iterationen werden die zunächst analogen Artefakte dann häufig digital archiviert. Um solche analog-digitalen Arbeitsabläufe zu unterstützten, entwickelten wir einen zweiten Werkzeugprototypen namens *LivelySketches*, der die Transitionen zwischen verschiedenen analogen oder digitalen Medien erleichtert. Ein möglicher Schritt für zukünftige Arbeiten ist es, die Funktionalitäten der beiden Prototypen zu vereinigen und das resultierende Werkzeug dann in einem größeren Softwareprojekt zu evaluieren. Außerdem beobachten wir, dass Ansätze wie *graphic facilitation* und *sketchnoting* immer beliebter werden. Potentielle Anwendungsfälle solche Techniken in Softwareprojekten zu studieren ist ein weitere Richtung für zukünftige Arbeiten. Tatsächlich haben wir bereits "graphic facilitators" mit Erfahrung in Softwareprojekten befragt und deren Aussagen mit denen von Softwareentwicklern und -architekten mit Skizziererfahrung verglichen. Basierend auf diesen Interviews werden wir Empfehlungen für die Anwendung von informellen Visualisierungs- und Skizziertechniken in verschiedenen Phasen der Softwareentwicklung ableiten.

Code-Plagiate Unsere Untersuchungen zu Kopien von *Stack-Overflow-Code-Fragmenten* in Java-Projekten auf GitHub ergab, dass höchstens ein Viertel der Fragmente wie von Stack Overflows Lizenz CC BY-SA gefordert die Quelle zuschreiben ("attribution"). Ein

weiteres Ergebnis ist, dass zwischen 3.3% und 11.9% der analysierten GitHub-Projekte in mindestens einer Datei auf Stack Overflow verwiesen. In unserer Studie erfüllten nur 1.8% der Projekte mit Kopien von Stack-Overflow-Codefragmenten beide Kriterien der Lizenz – verwiesen also auf die Quelle ("attribution") und verwendeten eine kompatible Lizenz ("share-alike"). Für die restlichen 98.2% der Projekte kann insbesondere die Verletzung des "share-alike"-Kriteriums zu Lizenzkonflikten mit den in der Dissertation diskutierten Konsequenzen – von möglichen Unterlassungsklagen bis hin zu Schadensersatzforderungen – führen. Die beiden Onlineumfragen, die wir in diesem Kontext durchgeführt haben, zeigen, dass viele Entwickler zugeben, Code von Stack Overflow ohne Zuschreibung zu kopieren. Vielen Umfrageteilnehmern war aber gleichzeitig nicht bewusst, wie Inhalte auf Stack Overflow lizenziert sind und welche Implikationen dies für die Verwendung in eigenen Projekten hat. Im Zuge dieses Forschungsprojekts entwickelten wir auch den offenen Datensatz *SOTorrent*, den wir seitdem regelmäßig aktualisieren. Neben eigenen Studien zur Wartung und Aktualisierung von Codefragmenten auf Stack Overflow verfolgen wir die Arbeiten anderer Forschergruppen, die unseren Datensatz verwenden. Eine zukünftige Forschungsrichtung, die nicht auf Code von Stack Overflow beschränkt ist, ist es, bessere Werkzeugunterstützung für Softwareentwickler zu schaffen, um ihnen die Lizenz- und Wartungsimplikationen von Codefragmenten aus dem Internet zu verdeutlichen. So könnten in Zukunft Continuous-Integration-Werkzeuge automatisch überprüfen, ob Commits nicht-triviale Codefragmente von Onlinequellen zu einem Projekt hinzufügen. Solche Werkzeuge können helfen, die notwendige Zuschreibung hinzuzufügen, sie können aber auch Entwickler über die Lizenzkompatibilität und die entsprechenden Anforderungen aufklären. Diese Ansätze können dabei helfen, lizenzrechtliche Probleme für Open-Source-Projekte, die durch die absichtliche oder unabsichtliche Verwendung von lizenziertem Onlinecode entstehen, zu verhindern.

Softwareentwicklungsexpertise In unserer Forschung zur *Softwareentwicklungsexpertise* identifizierten wir verschiedene Charakteristiken von Experten in der Softwareentwicklung, zeigten aber auch Faktoren auf, die den Aufbau einer solchen Expertise über die Zeit positiv wie negativ beeinflussen können. Neben der Entwicklung einer ersten konzeptionellen Theorie fanden wir heraus, dass die Expertise-Selbsteinschätzung von Entwicklern kontextabhängig ist und nicht unbedingt mit der Erfahrung in Jahren korreliert. Wir empfehlen Forschern, wenn sie Studienteilnehmer nach einer Selbsteinschätzung fragen, möglichst explizit zu beschreiben, was in ihrer spezifischen Studie Anfänger von Experten unterscheidet. Neben dem Entwurf neuer Studien erlaubt es unsere Theorie, Ergebnisse vergangener Studie neu einzuordnen, da verschiedener Einflussfaktoren auf Verhalten und Leistung aufgezeigt werden. Softwareentwickler können mit Hilfe unserer Ergebnisse lernen, was andere Entwickler von Experten erwarten und welche Verhaltensweisen für den gezielten Aufbau von Expertise hilfreich sind. So erläutern wir beispielsweise das Konzept der "deliberate practice" und weisen auf die Wichtigkeit anspruchsvoller Ziele, einer unterstützenden Arbeitsumgebung und des Feedbacks von Arbeitskollegen hin. Erfahrenere Entwickler können lernen, was von guten Mentoren erwartet wird. So werden sie als wichtige Quelle für Feedback

betrachtet und sollten vor allem geduldig und unvoreingenommen sein. Arbeitgeber können von unsere Studie lernen, was typische Faktoren der Demotivierung ihrer Mitarbeiter sind und wie sie eine Arbeitsumgebung schaffen können, die Lernen und Weiterentwicklung unter ihren Angestellten fördert. Neben offensichtlichen Strategien wie Fortbildungen und Konferenzbesuche suggerieren unsere Ergebnisse, dass der Informationsfluss zwischen verschiedenen Entwicklungsteams und auch zwischen Entwicklungsteams und anderen Teilen des Unternehmens ein wichtiger Faktor ist. Arbeitgeber sollten sich ebenfalls darum bemühen, eine gute Mischung aus Kontinuität und Wandel in ihren Softwareentwicklungsprozess zu integrieren, da wenig anspruchsvolle Aufgaben ein Hauptdemotivationsfaktor für Softwareentwickler sind. Eine wichtige zukünftige Forschungsrichtung ist die *Rolle älterer Softwareentwickler*. Insbesondere in Industrieländern führt der demographische Wandel zu einer alternden Erwerbsbevölkerung. Trotz dieser Tatsache sind die Hürden und Herausforderungen älterer Entwickler in einem kompetitiven Feld wie der Softwareentwicklung größtenteils unbekannt. Unsere Studienteilnehmer nannten bereits verschiedene altersbezogene Herausforderungen. Wir möchten diese weiter untersuchen, um sie wo möglich zu adressieren und abzumildern, damit erfahrende Entwickler sich nicht dazu entscheiden, ab einem gewissen Alter die Softwareindustrie zu verlassen.

Im Verlauf dieser Dissertation beschrieben wir nicht nur unsere Studiendesigns und Ergebnisse, sondern versuchten immer auch aufzuzeigen wie unsere empirischen Ergebnisse dazu verwendet werden können, nächste Schritte in Forschung oder Praxis zu motivieren. So haben wir unsere Studienergebnisse beispielsweise dazu verwendet: (1) neuartige Werkzeuge zur Unterstützung der Arbeitsabläufe von Entwicklern im Zusammenhang mit Skizzen und Diagrammen zu entwerfen, (2) Open-Source-Entwickler über mögliche Lizenzkonflikte in ihren Projekten zu informieren, (3) eine erste konzeptionelle Theorie über Softwareentwicklungsexpertise abzuleiten, welche sowohl in der Forschung als auch in der Praxis angewandt werden kann, (4) auf altersbezogene Probleme von Softwareentwicklern hingewiesen, (5) Aufmerksamkeit unter Forschern über ethische Implikationen ihrer Befragungsstrategien zu schaffen, und (6) einen offenen Datensatz bereitzustellen, der für zukünftige Forschungsprojekte verwendet werden kann. Unsere Arbeiten unterstützen Forscher und Menschen aus der Praxis darin, Entscheidungen über die Verwendung bestimmter Werkzeuge oder Abläufe im Softwareentwicklungsprozess auf Grundlage von Daten und empirischen Ergebnissen – statt “aus dem Bauch heraus” – zu treffen.

Literaturverzeichnis

[Ba15] Baltes, Sebastian; Moseler, Oliver; Beck, Fabian; Diehl, Stephan: Navigate, Understand, Communicate: How Developers Locate Performance Bugs. In: ESEM 2015. 2015.

[Ba18] Baltes, Sebastian; Dumani, Lorik; Treude, Christoph; Diehl, Stephan: SOTorrent: Reconstructing and Analyzing the Evolution Stack Overflow Posts. In: 15th International Conference on Mining Software Repositories (MSR 2018). 2018.

[Ba19] Baltes, Sebastian: Software Developers' Work Habits and Expertise: Empirical Studies on Sketching, Code Plagiarism, and Expertise Development. Doctoral Dissertation. University of Trier, Germany, 2019.

[BD14] Baltes, Sebastian; Diehl, Stephan: Sketches and diagrams in practice. In: FSE 2014. 2014.

[BD16] Baltes, Sebastian; Diehl, Stephan: Worse Than Spam: Issues In Sampling Software Developers. In: ESEM 2016. 2016.

[BD18a] Baltes, Sebastian; Diehl, Stephan: Towards a Theory of Software Development Expertise. In: ESEC/FSE 2018. 2018.

[BD18b] Baltes, Sebastian; Diehl, Stephan: Usage and Attribtion of Stack Overflow Code Snippets in GitHub Projects. Empirical Software Engineering, Online First:1–37, 2018.

[BHD17] Baltes, Sebastian; Hollerich, Fabrice; Diehl, Stephan: Round-Trip Sketches: Supporting the Lifecycle of Software Development Sketches from Analog to Digital and Back. In: VISSOFT 2017. 2017.

[BKD17] Baltes, Sebastian; Kiefer, Richard; Diehl, Stephan: Attribution required: Stack overflow code snippets in GitHub projects. In: ICSE 2017 Companion Volume. 2017.

[BSD14] Baltes, Sebastian; Schmitz, Peter; Diehl, Stephan: Linking sketches and diagrams to source code artifacts. In: FSE 2014. 2014.

[BTD19] Baltes, Sebastian; Treude, Christoph; Diehl, Stephan: SOTorrent: Studying the Origin, Evolution, and Usage of Stack Overflow Code Snippets. In: MSR 2019. 2019.

Sebastian Baltes arbeitet als Senior Software Engineer bei der QAware GmbH in Deutschland und als Adjunct Lecturer in der School of Computer Science der University of Adelaide in Australien. Er wurde im April 2019 an der Universität Trier mit der Dissertation "Software Developers' Work Habits and Expertise" als Dr. rer. nat. promoviert. Zuvor studierte er an den Universitäten in Trier, Saarbrücken und Växjö, Schweden, Informatik (B.Sc. und M.Sc.). In seiner Forschung widmet er sich der empirischen Analyse der Arbeitsweise von Softwareentwicklern, um deren tägliche Arbeit durch verbesserte Werkzeuge und Handlungsempfehlungen produktiver zu gestalten. Die Hauptmotivation seiner Forschung ist, dass in der Softwareentwicklung immer noch zu viele Entscheidungen eher meinungs- als datenbasiert getroffen werden. Diese Lücke zwischen empirischer Forschung und Praxis gilt es zu schließen, sowohl durch das Erforschen relevanter Probleme als auch durch geeignete Wissenschaftskommunikation der Ergebnisse zurück zu den Entwicklern in der Praxis.

Parallele Parametrisierte Algorithmen[1]

Max Bannach[2]

Abstract: Die parametrisierte Algorithmik ist ein Schlüsselbereich des modernen Algorithmenentwurfes sowie der Komplexitätstheorie. Die fast ausschließlich untersuchte Ressource in diesem Bereich ist dabei die sequentielle Zeit, obwohl die Parallelverarbeitung ein zentrales und vielfach untersuchtes Teilgebiet der klassischen Algorithmik ist. Sowohl das Identifizieren eines geeigneten Parameters als auch die direkte Beschleunigung durch Parallelisierung verfolgen das gleiche Ziel: möglichst viele Instanzen eines an sich nicht effizient lösbaren Problems dennoch zu lösen. Es ist daher naheliegend, beide Forschungsgebiete miteinander zu verbinden – und genau diese Art von Integration ist das Ziel dieser Arbeit. Ich präsentiere eine Vielzahl von parametrisierten Komplexitätsklassen und entwickle eine Sammlung von parallelen parametrisierten Basisalgorithmen. Dabei werden nahezu alle Techniken, die die parametrisierte Komplexitätstheorie zu bieten hat, von einem parallelen Standpunkt aus untersucht – unter anderem Color Coding, beschränkte Suchbäume, Kernelisierung, strukturelle Zerlegungen von Graphen sowie algorithmische Metasätze. Außerdem illustriere ich, wie die letzten zwei Techniken in der Praxis genutzt werden können, indem ich zwei Software-Bibliotheken vorstelle: eine zum Berechnen optimaler Baumzerlegungen und eine für die Modellprüfung eines Fragmentes der monadischen Prädikatenlogik zweiter Stufe.

1 Einleitung

In meiner Dissertation versuche ich, einen neuen Zweig in der Komplexitätstheorie zu motivieren: *parallele, parametrisierte Algorithmen*. Als Grundlage für diese Theorie nutze ich parametrisierte Schaltkreiskomplexitätsklassen, die analog zur klassischen Schaltkreiskomplexität definiert sind. Anschließend zeige ich, dass fast alle gängigen Techniken der parametrisierten Komplexitätstheorie in diesem Modell nutzbar sind. Unter anderem betrachten wir im Rahmen der Dissertation die Color Coding Technik, parallele Versionen von beschränkten Suchbäumen und Kernelisierungen, sowie algorithmische Metasätze.

Diese Zusammenfassung ist, ebenso wie die Dissertation, in zwei Hauptteile aufgeteilt. Im ersten Teil der Dissertation (hier Kapitel 2) beschäftigen wir uns mit den theoretischen Grundlagen paralleler, parametrisierter Algorithmen. Wir führen das zugrunde liegende Maschinenmodell ein und implementieren darauf gängige Konzepte der parametrisierten Komplexitätstheorie. Im zweiten Teil der Arbeit (hier Kapitel 3) versuche ich die theoretischen Erkenntnisse des ersten Teils mittels *Algorithm Engineering* in effiziente Werkzeuge zu überführen. Die vorgeschlagenen Ansätze werden experimentell evaluiert und die Ergebnisse genutzt, um weiterführende Forschungsrichtungen in der Theorie aufzuzeigen.

[1] Englischer Titel der Dissertation [Ba19]: Parallel Parameterized Algorithms

[2] Institut für Theoretische Informatik, Universität zu Lübeck, bannach@tcs.uni-luebeck.de

CC BY SA

2 Theorie paralleler, parametrisierter Algorithmen

Wie der Titel der Arbeit es vermuten lässt, beschäftigen wir uns mit *parametrisierten Problemen.* Dies sind Mengen $L \subseteq \{(w,k) \mid w \in \{0,1\}^*, k \in \mathbb{N}\}$ in denen wir w die *Instanz* und k den *Parameter* nennen. Ein solches Problem heißt *fixed-parameter tractable,* wenn es eine berechenbare Funktion f, eine Konstante c und einen Algorithmus A gibt, sodass A für jedes Paar (w,k) in Zeit $f(k) \cdot |w|^c$ entscheidet, ob $(w,k) \in^? L$. Ziel der Theorie in meiner Arbeit ist es, ein analoges Konzept zu definieren: *fixed-parameter parallel tractable.* Dazu ersetzen wir in der Definition den Algorithmus A durch eine Familie von Boolschen-Schaltkreisen $(C_{n,k})_{n,k\in\mathbb{N}}$, die aus `Und`-, `Oder`- und `Nicht`-Gattern bestehen. Wir sprechen von AC-Schaltkreisen, wenn die `Und`- und `Oder`-Gatter beliebig viele Eingänge besitzen, anderenfalls von NC-Schaltkreisen. Ein Schaltkreis C mit n Eingängen und m Ausgängen beschreibt auf natürliche Art und Weise eine Funktion $\sigma\colon \{0,1\}^n \to \{0,1\}^m$, beziehungsweise eine Menge, falls $m = 1$. Wir schreiben $C(x)$ für die Ausgabe des Schaltkreises bei Eingabe $x \in \{0,1\}^n$ und definieren die folgenden Komplexitätsklassen:

Definition 1. *Ein parametrisiertes Problem L ist in* para-AC^i *für $i \geq 1$, falls es eine berechenbare Funktion f, eine Konstante c und eine* uniforme *Familie $(C_{n,k})_{n,k\in\mathbb{N}}$ von* AC*-Schaltkreisen gibt, sodass für alle $(w,k) \in \{0,1\}^* \times \mathbb{N}$ gilt:*

1. $C_{|w|,k}(w) = 1 \Leftrightarrow (w,k) \in L$;
2. $\text{size}(C_{n,k}) \leq f(k) \cdot n^c$;
3. $\text{depth}(C_{n,k}) \leq f(k) + c \log^i n$.

Die Definition lässt etwas Spielraum darüber, was wir mit Uniformität, sowie der Größe und Tiefe eines Schaltkreises meinen. Alle Resultate, die wir betrachten, gelten für jede "vernünftige" Definition der Größe und Tiefe, sowie für die stärkste Form der Uniformität (DLOGTIME-uniformity). Die Klasse para-AC^0 ist wie oben definiert, verlangt aber, dass die Tiefe konstant ist (unabhängig von k und n). Ersetzen wir in der Bedingung der Tiefe das Plus durch ein Mal, so sprechen wir von para-AC$^{i\uparrow}$ – wobei wir para-AC$^{0\uparrow}$ eine Tiefe von $f(k)$ gewähren. Analog definieren wir para-FAC^i und para-FAC$^{i\uparrow}$ für Funktionsprobleme. In allen Definitionen lässt sich AC auch durch NC und TC ersetzen.

Eine Familie von para-AC^i Schaltkreisen entspricht grob einem parallelen Algorithmus, der ein Problem in paralleler Zeit $f(k) + c \log^i n$ unter Verwendung von $f(k) \cdot n^c$ Arbeit löst. Daher sagen wir, dass ein Problem *fixed-parameter parallel tractable* ist, wenn es in para-AC^i liegt. Für die "up-Klassen" gilt para-AC$^i \subseteq$ para-AC$^{i\uparrow} \subseteq$ para-AC$^{i+\varepsilon}$; die anderen Klassen erben die Hierarchie der klassischen Komplexitätstheorie:

$$\text{para-AC}^0 \subseteq \text{para-NC}^1 \subseteq \text{para-AC}^1 \subseteq \cdots \subseteq \text{para-NC} = \text{para-AC} = \bigcup_{i=1}^{\infty} \text{para-AC}^i \subseteq \text{FPT}.$$

2.1 Das vielfältigste Werkzeug: Color Coding

Die erste Technik der parametrisierten Komplexitätstheorie, die wir betrachten, ist *Color Coding*. Diese von Alon, Yuster und Zwick [AYZ95] vorgestellte Technik, beschreibt ein randomisiertes Verfahren um eine kleine Teilstruktur in einer größeren Trägerstruktur zu finden. Stellen wir uns zum Beispiel vor, dass wir k disjunkte in einem großen Graphen G finden wollen. Die Technik sieht vor, die Knoten von G *zufällig* mit k Farben zu färben und dann ein monochromatisches Dreieck jeder Farbe zu suchen. Es ist leicht sequentiell in Zeit $O(n^3)$ oder parallel in Zeit $O(1)$ (also in AC^0) zu testen, ob es *ein* Dreieck *einer* Farbe gibt. Finden wir für jede Farbe ein solches Dreieck, so wissen wir aufgrund der monochromatischen Eigenschaft, dass alle Dreiecke disjunkt sind.

Diese Beobachtungen führen zu einem *randomisierten* para-AC^0-Algorithmus, denn die Wahrscheinlichkeit, dass eine potentielle Lösung von k disjunkten Dreiecken sinnvoll gefärbt wird, lässt sich durch $1/k^{3k}$ abschätzen. Im Umkehrschluss bedeutet dies, dass $f(k)$ Wiederholungen genügen, um die Wahrscheinlichkeit beliebig nah an 1 zu bringen.

Mit einem Trick aus der Schaltkreiskomplexität lässt sich dieses Verfahren derandomisieren. Mit Hashfunktionen zeigt man üblicherweise, dass AC^0-Schaltkreise bis zu polylogarihmischen Zahlen zählen können [NRW90]. Wir wandeln diese Technik ab und zeigen:

Satz 2. *Color Coding kann in* para-AC^0 *derandomisiert werden.*

2.2 Beschränkte Suchbäume parallel durchsuchen

Beschränkte Suchbäume sind Suchbäume mit konstantem Verzweigungsgrad c und Tiefe k. Da diese auf natürliche Weise in Zeit $O(c^k)$ durchsucht werden können, bilden sie eine Grundlage für viele parametrisierte Algorithmen. Intuitiv sollte es möglich sein, solche Suchbäume parallel in Zeit $O(k)$ zu durchsuchen, wenn Arbeit $O(c^k)$ investiert wird. Wir zeigen formal, dass dies für eine allgemeine Familie von Problemen gilt.

Sei $\mathscr{F}$ eine Familie von Graphen. Die Eingabe für $\mathscr{F}$-MODULATOR ist ein Graph G und eine Zahl k. Gesucht ist eine Menge $X \subseteq V(G)$ mit $|X| \leq k$, sodass $G[V \setminus X] \in \mathscr{F}$. Das Problem $\mathscr{F}$-EDITING ist analog, erlaubt es aber k Kanten hinzuzufügen oder zu löschen.

Die erste Familie von Graphen, die wir studieren, ist die Familie der H-freien Graphen. In diesem Fall verallgemeinern die obigen Probleme bereits viele kombinatorische Probleme, wie VERTEX-COVER ($H = \circ\!\!-\!\!\circ$) oder CLUSTER-EDITING ($H = \circ\!\!-\!\!\circ\!\!-\!\!\circ$).

Satz 3. *Für jeden festen Graphen H gilt:*

1. p_k-*H-free*-MODULATOR $\in$ para-$\text{AC}^{0\uparrow}$;
2. p_k-*H-free*-EDITING $\in$ para-$\text{AC}^{0\uparrow}$.

Im obigen Satz ist jeweils k der Parameter. Wollen wir diese Ergebnisse weiter verallgemeinern, liegt es nahe, dass auch "H als Parameter" betrachtet werden sollte. Dies ist

im Allgemeinen schwierig, da wir nicht umher kommen, dass H eine Clique sein könnte. Beschränken wir jedoch die Struktur von H, so lässt sich zeigen:

Satz 4. *Für einen Graphen H sei $\mathscr{F}$ die Menge der Graphen G, für die es* keinen *Homomorphismus von H nach G gibt. Sei H ein Graph mit konstanter Baumweite, dann gilt:*

$$\text{p}_{|H|+k}\text{-}\mathscr{F}\text{-MODULATOR} \in \text{para-AC}^{0\uparrow}.$$

Mit der Hilfe von Color Coding lässt sich dieser Satz auch noch dahingehend verallgemeinern, dass wir statt eines Homomorphismus auch eine *Einbettung* von H nach G verbieten können. Eine interessante Inkarnation des Problems ist dann PATH-VERTEX-COVER – der Spezialfall für $H = P_c$.

2.3 Sequentielle Kernel parallelisieren

Kernelisierungen sind ein weiterer Baustein der parametrisierten Komplexitätstheorie. Dabei handelt es sich um Funktionen $K\colon \{0,1\}^* \times \mathbb{N} \to \{0,1\}^* \times \mathbb{N}$ mit folgenden drei Eigenschaften: Sei $K(w,k) = (w',k')$, dann gilt (i) $(w,k) \in L \Leftrightarrow (w',k') \in L$, (ii) $|w'| \leq f(k)$ und (iii) $k' \leq f(k)$. Wie üblich ist f hier eine beliebige berechenbare Funktion.

Umgangssprachlich kann man sich Kernelisierungen als "Vorverarbeitung mit Garantie" vorstellen. Daher fordert man üblicherweise, dass K selbst in polynomieller Zeit berechenbar ist. Es ist bekannt, dass ein entscheidbares Problem genau dann eine in polynomieller Zeit berechenbare Kernelisierung hat, wenn es in FPT liegt. Dieses Resultat lässt sich auf die gesamte parametrisierte Schaltkreishierarchie verallgemeinern:

Satz 5. *Sei L ein entscheidbares Problem. Die folgenden Aussagen sind äquivalent:*

1. $L \in \text{para-AC}^i$;
2. L besitzt eine Kernelisierung K mit $K \in \text{FAC}^i$.

Dieser Satz ist insofern etwas überraschend, als dass durch die Literatur hinweg Kernelisierungen immer *sehr* sequentiell beschrieben werden: üblicherweise gibt es eine Menge von Reduktionsregeln, die sequentiell immer wieder angewendet werden. Anderseits sagt der Satz, dass jeder parallele Algorithmus eine parallele Kernelisierung impliziert.

Ein Problem, an dem dies im Rahmen der Dissertation deutlich wird, ist HITTING-SET. Die Eingabe ist ein d-Hypergraph $H = (V,E)$, also ein Hypergraph in dem $|e| \leq d$ für jede Kante $e \in E$ gilt, sowie eine Zahl $k \in \mathbb{N}$. Gesucht ist eine Menge $X \subseteq V$ mit $|X| \leq k$ und $X \cap e \neq \emptyset$ für alle $e \in E$. Chen, Flum und Huang zeigten, dass sich ein Kernel für dieses Problem mit FAC-Schaltkreisen der Tiefe $O(d)$ berechnen lässt [CFH17]. Da auch hier eine Reduktionsregel immer wieder angewendet wird, stellten sie die Vermutung auf, dass solche Schaltkreise auch immer eine Tiefe von $\Omega(d)$ benötigen würden – was hieße, dass eine parallele Kernelisierung nur für konstante d möglich wäre.

Diese Vermutung lässt sich widerlegen. Eine Runde der Kernelisierung von Chen, Flum und Huang lässt sich per Color Coding in konstanter Zeit implementieren – was zunächst die Laufzeit von $O(d)$ nicht verbessert. Wir zeigen, dass die iterative Anwendung von Color Coding in eine *einzige* Anwendung von Color Coding kollabiert werden kann. Dazu müssen zwar *deutlich* mehr Farben genutzt werden, aber die Anzahl der benötigten Farben lässt sich durch eine Funktion in k und d beschränken. Es folgt:

Satz 6. *Das Problem* $\mathrm{p}_{k,d}$-HITTING-SET *besitzt eine Kernelisierung K mit* $K \in \mathrm{FAC}^0$.

Korollar 7. $\mathrm{p}_{k,d}$-HITTING-SET $\in$ para-AC^0.

Leider ist der Satz von rein theoretischem Interesse, denn der produzierte Kernel ist hochgradig exponentiell in k und d. Anderseits ist das Ergebnis aber in sofern scharf, als dass p_k-HITTING-SET W[2]-vollständig und p_d-HITTING-SET NP-vollständig ist.

Die bisherigen Ergebnisse sagen uns, dass wir oft Kernel parallel berechnen können – nicht jedoch, wie klein diese Kernel sind. Wie man erwarten kann, gibt es einen interessanten Zusammenhang zwischen Schaltkreistiefe und Kernelgröße. Beispielsweise gilt:

Satz 8. *Für* p_k-VERTEX-COVER *lässt sich ein Kernel der Größe (i)* 2^k *in* FAC^0*, (ii)* k^2 *in* FTC^0 *und (iii)* $2k$ *in* FP *berechnen.*

Der in polynomieller Zeit berechenbare Kernel der Größe $2k$ ist der kleinste Kernel, der aktuell bekannt ist. Seine Berechnung basiert auf dem Satz von Nemhauser und Trotter [NJ74]. Die Frage, ob sich Kernel basierend auf diesem Satz auch parallel berechnen lassen können, lässt sich an die gleiche Frage für das Paarungsproblem in Graphen binden:

Satz 9. *Das Berechnen eines Kernels auf Basis des Satzes von Nemhauser und Trotter ist* NC*-äquivalent zum Berechnen einer perfekten Paarung in bipartiten Graphen.*

Korollar 10. *Ein* $2k$*-Kernel für* p_k-VERTEX-COVER *ist in* FRNC *und* quasi-FNC *berechenbar. Er lässt sich in* FNC *berechnen, wenn das Paarungsproblem in* NC *liegt.*

2.4 Parallele algorithmische Metasätze

Algorithmische Metasätze oder Metatheoreme sind Aussagen der Form: “Alle Probleme die sich in einer bestimmten Logik formulieren lassen, lassen sich effizient auf Strukturen mit bestimmten Eigenschaften lösen.” Das bekannteste Beispiel ist der Satz von Courcelle – nachdem sich alle Probleme, die in monadischer Prädikatenlogik zweiter Stufe formulierbar sind, auf Strukturen konstanter Baumweite in linearer Zeit lösen lassen [Co90]. Es gibt einige Ansätze, diesen Satz zu parallelisieren [EJT12a, EJT12b]. Wir zeigen, dass dies auch dann noch möglich ist, wenn die Baumweite der Struktur ein Parameter ist. Um Algorithmen wie jenen im Satz von Courcelle zu implementieren, benötigen wir Algorithmen, die entsprechende Zerlegungen von Graphen berechnen.

Satz 11. *Es gibt eine Familie von* para-$\mathrm{FAC}^{0\uparrow}$*-Schaltkreisen, die jeden Graphen G auf eine Baumtiefenzerlegung der Weite* $O(2^k)$ *von G abbildet.*

Satz 12. *Es gibt eine Familie von* para-FAC$^{2\uparrow}$*-Schaltkreisen, die jeden Graphen G auf eine optimale Baumzerlegung von G abbildet.*

Mit Hilfe dieser Zerlegungen lassen sich die Aussagen in Tabelle 1 zeigen. Dabei basieren alle Ergebnisse für die Logik zweiter Stufe auf der Simulation eines *Multimengen-Baumautomaten,* angelehnt an ein ähnliches Resultat für konstante Baumweite [EJT12a]. Das Resultat zur Logik erster Stufe ist analog zu einem Ergebnis von Flum und Grohe, das besagt, dass dieses Problem in Platz $f(k) + c\log(n)$ lösbar ist [FG03]. Diese Klassen (para-AC$^{0\uparrow}$ und para-L) liegen diagonal zueinander: para-AC$^{0\uparrow}$ enthält das parametrisierte Distanzproblem, während para-L das klassische Erreichbarkeitsproblem enthält – die entgegengesetzten Richtungen gelten allerdings nur, wenn Komplexitätsklassen kollabieren.

Tab. 1: Die verschiedenen Metasätze, die im Rahmen meiner Dissertation bewiesen werden. Der Parameter ist immer die Länge der Formel *zusammen* mit einem strukturellen Parameter. Von oben nach unten: der Maximalgrad, die Vertex-Cover-Zahl, die Baumtiefe und die Baumweite.

Logik	Parameter	Komplexität
Prädikatenlogik erster Stufe	$\lvert\varphi\rvert + \Delta$	para-AC$^{0\uparrow}$
monadische Prädikatenlogik zweiter Stufe	$\lvert\varphi\rvert + \mathrm{vc}$	para-AC^{0}
monadische Prädikatenlogik zweiter Stufe	$\lvert\varphi\rvert + \mathrm{td}$	para-AC$^{0\uparrow}$
monadische Prädikatenlogik zweiter Stufe	$\lvert\varphi\rvert + \mathrm{tw}$	para-AC$^{2\uparrow}$

3 Von der Theorie zur Praxis und zurück

Im Forschungsgebiet der parametrisierten Komplexitätstheorie nimmt das Interesse daran, theoretische Resultate in praktische Algorithmen zu verwandeln, seit einigen Jahren immer weiter zu. Dieses Phänomen gipfelte 2016 mit der Einführung der PACE (der *Parameterized Algorithms and Computational Experiments Challenge*). In diesem jährlich stattfindendem Wettbewerb ist es das Ziel, ein parametrisiertes Problem in möglichst kurzer Zeit optimal zu lösen. In den vergangenen Jahren wurden zum Beispiel die Probleme TREEWIDTH, FEEDBACK-VERTEX-SET und VERTEX-COVER betrachtet. Im zweiten Teil meiner Dissertation beschreibe ich Programme, mit denen ich bei der PACE im Jahr 2016 und 2017 teilgenommen habe. Außerdem diskutieren wir neue theoretische Erkenntnisse, die aus den Einreichungen resultieren.

3.1 Jdrasil: Eine modulare Bibliothek zum Berechnen von Baumzerlegungen

Jdrasil, der Name ist ein Portmanteau aus “Java” und “Yggdrasil,” ist eine modulare Bibliothek zum Berechnen von Baumzerlegungen. Die Bibliothek enthält nahezu alle exakten und viele heuristische Algorithmen zum Berechnen von Baumzerlegungen, sowie eine Vielzahl von Vor- und Nachverarbeitungsroutinen. Damit soll es Forschern ermöglicht werden, verschiedene Algorithmen gegeneinander zu vergleichen. Zum Beispiel ist es

möglich, den Branch-and-Bound Algorithmus von Gogate und Dechter [GD04] mit dem SAT-Ansatz von Berg und Järvisalo [BJ14] zu vergleichen. Letzterer wurde im Rahmen der Arbeit noch durch einige Änderungen an der SAT-Formulierung verbessert.

Die Bibliothek ist mit einer parallelen Architektur entwickelt worden. Unabhängig davon, ob ein Algorithmus (wie jener von Gogate und Dechter) parallel implementiert wird, wird Jdrasil den Graphen *vor* dem Berechnen der Baumzerlegung schon in kleinere Graphen zerlegen. Für diese werden dann *parallel* Baumzerlegungen gesucht. Diese Strategie nutzt das Konzept der *sicheren Separatoren,* das von Bodlaender und Koster vorgestellt wurde [BK06]. Die Arbeit untersucht Heuristiken, um solche Separatoren effizient zu finden.

Sowohl 2016 als auch 2017 konnte Jdrasil den dritten Platz der PACE belegen (in den folgenden Jahren war TREEWIDTH nicht Teil des Wettbewerbes). Die Gewinner in beiden Jahren implementierten einen Algorithmus von Hisao Tamaki [Ta17]. Im Rahmen meiner Dissertation konnte ich zeigen, dass die zugrunde liegende *positive-instance driven dynamic programming* Strategie für eine allgemeine Familie von Problemen einsetzbar ist – und nicht nur für Baumweite. In Vertex-Pursuit-Evasion-Spielen versucht ein Team von *k Suchern* einen einzelnen *Gesuchten* zu fangen. Indem man variiert, wie sich die Sucher und der Gesuchte bewegen dürfen (zum Beispiel nur entlang der Topologie des Graphen), erhält man Spiele die zu strukturellen Parametern von Graphen korrespondieren.

Satz 13. *Für eine allgemeine Form des Vertex-Pursuit-Evasion-Spiels kann, bei Eingabe eines Graphen $G = (V,E)$ und einer Zahl $k \in \mathbb{N}$, in Zeit $O(|\mathscr{B}(Q)|^2 \cdot |V|^2)$ entschieden werden, ob k Suchende eine Gewinnstrategie besitzen.*

Hier bezeichnet $\mathscr{B}(Q)$ die Gewinnregion, *dass bedeutet die Menge der Konfigurationen, von denen aus die Suchenden eine Gewinnstrategie besitzen.*

Der Satz veranschaulicht, woher die Strategie ihren Namen hat. Der zugrunde liegende Algorithmus listet ausschließlich positive Konfigurationen des Spiels auf. Konfigurationen von denen aus die Suchenden keine Gewinnstrategie besitzen, werden während der Berechnung nie betrachtet. Die Allgemeinheit des Spiels liefert nun folgendes Resultat:

Korollar 14. *Gegeben ein Graph $G = (V,E)$ und eine Zahl $k \in \mathbb{N}$. Es gibt einen Algorithmus der in Zeit $O(|\mathscr{B}(Q)|^2 \cdot |V|^2)$ entscheidet, ob die $\{$ Baumweite, Pfadweite, Baumtiefe, q-verzweigte Baumweite, Abhängigkeits-Baumweite $\}$ von G höchstens k ist.*

In der gleichen Zeit kann eine entsprechende Zerlegung ausgegeben werden.

3.2 Jatatosk: Ein Modellprüfer für ein Fragment der Prädikatenlogik

Basierend auf Jdrasil wurde Jatatosk entwickelt. Der Name ist erneut ein Portmanteau – diesmal aus "Java" und "Ratatosk," dem Eichhörnchen das auf Yggdrasil Nachrichten von den Wurzeln zu den Blättern transportiert. Jatatosk überprüft, mit Hilfe dynamischer Programmierung auf Baumzerlegungen, ob eine logische Struktur ein Modell einer Formel

monadischer Prädikatenlogik zweiter Stufe ist. Um die *extreme* kombinatorische Explosion, die Implementierungen vom Satz von Courcelle notwendigerweise aufweisen [FG04], zu umgehen, beschränken wir uns auf ein Fragment der Logik.

Das Fragment wurde nicht (wie sonst üblich) aus "der Perspektive der Logik" gewählt, um möglichst aussagestark zu sein; sondern aus "Sicht der Implementierung." Dazu arbeiten wir auf einer *sehr schönen Baumzerlegung,* die neben Join-, Forget- und Leaf-Bags auch Edge-Bags besitzt. Auf solchen Zerlegungen lässt sich eine Formel "event-driven" überprüfen: anstatt zu Beginn der Berechnung einen Baumautomaten zu konstruieren, testen wir Eigenschaften lokal für eingefügte Kanten. Entsprechend wurde das Fragment so gewählt, dass es nur an diesen Stellen eine Aussagekraft besitzt. Im Detail besteht es aus Formeln der folgenden Bauart, wobei die χ_i quantorenfreie Formeln erster Stufe sind:

$$\varphi = \exists X_1, \dots \exists X_q \bigwedge_{i=1}^{\ell} \psi_i \quad \text{für} \quad \psi_i \in \{ \forall x \forall y\, E(x,y) \to \chi_i,\ \forall x \exists y\, E(x,y) \wedge \chi_i, \\ \exists x \forall y\, E(x,y) \to \chi_i, \exists x \exists y\, E(x,y) \wedge \chi_i,\ \forall x\, \chi_i,\ \exists x\, \chi_i \}.$$

Ein Problem, das auf natürliche Art mit solchen Formeln ausgedrückt werden kann, ist die Frage ob ein gegebener Graph mit 3 Farben gefärbt werden kann:

$$\varphi_{\text{3col}} = \exists R \exists G \exists B \forall x \forall y E(x,y) \to \bigwedge_{C \in \{R,G,B\}} \neg C(x) \vee \neg C(y).$$

Damit auch Optimierunsprobleme wie VERTEX-COVER oder DOMINATING-SET beschrieben werden können, bietet Jatatosk eine gewichtete Variante. Hierzu können für eine Struktur $G = (V,E)$ Gewichte $w_i \colon V \to \mathbb{N}$ definiert werden. Für eine Formel $\varphi(X_1, \dots, X_p)$ mit freien, monadischen Variablen zweiter Stufe, sucht Jatatosk eine Belegung $S_i \subseteq V$, die $\sum_{i=1}^{p} \sum_{v \in V} w_i(v)$ unter $G \models \varphi(S_1, \dots, S_p)$ minimiert. Wir erhalten für die beiden Probleme:

$$\varphi_{\text{vc}}(S) = \forall x \forall y E(x,y) \to S(x) \vee S(y); \qquad \varphi_{\text{ds}}(S) = \forall x \exists y E(x,y) \wedge S(y).$$

Es gibt Eigenschaften, die sich in monadischer Prädikatenlogik zweiter Stufe ausdrücken lassen, nicht aber im oben beschriebenen Fragment. Ist eine solche Eigenschaft effizient per dynamischer Programmierung auf Baumzerlegungen testbar, so scheint es sinnvoll, sie direkt in der Logik verfügbar zu machen. Jatatosk unterstützt die Quantorenerweiterungen $\exists^{\text{connected}} X$ und $\exists^{\text{forest}} X$, die über zusammenhängende Teilmengen beziehungsweise Wälder quantifizieren. Mit Hilfe dieser Quantoren ist es möglich, FEEDBACK-VERTEX-SET und Eigenschaften wie "der Graph besitzt ein Dreieck als Minor" zu beschreiben:

$$\varphi_{\text{fvs}}(S) = \exists^{\text{forest}} F \forall x \left(S(x) \vee F(x) \right);$$

$$\begin{aligned} \varphi_{\text{triangle-minor}} = {} & \exists^{\text{connected}} R \exists^{\text{connected}} G \exists^{\text{connected}} B \,. \\ & \left(\forall x \left(\neg R(x) \vee \neg G(x) \right) \wedge \left(\neg G(x) \vee \neg B(x) \right) \wedge \left(\neg B(x) \vee \neg R(x) \right) \right) \\ & \wedge \left(\exists x \exists y\, E(x,y) \wedge R(x) \wedge G(y) \right) \wedge \left(\exists x \exists y\, E(x,y) \wedge G(x) \wedge B(y) \right) \\ & \wedge \left(\exists x \exists y\, E(x,y) \wedge B(x) \wedge R(y) \right). \end{aligned}$$

Formel	Laufzeit
φ_{3col}	$O^*(3^k)$
$\varphi_{\text{vc}}(S)$	$O^*(2^k)$
$\varphi_{\text{ds}}(S)$	$O^*(8^k)$
$\varphi_{\text{triangle-minor}}$	$O^*(k^{6k})$
$\varphi_{\text{fvs}}(S)$	$O^*(2^k k^{2k})$

Das schlanke Design von Jatatosk erlaubt es, die *exakte Laufzeit* von Jatatosk direkt aus der *Syntax* einer Formel abzulesen. Im Kontrast dazu, müssen bei anderen Implementierungen für jede Formel einzeln nicht-triviale Analysen durchgeführt werden [KLR11]. Die Tabelle am Rand fasst die Laufzeiten von Jatatosk für alle hier vorgestellten Formeln zusammen, wobei k jeweils die Baumweite plus eins ist. Um die Effizienz von Jatatosk zu bewerten, habe ich es im Rahmen der Arbeit experimentell gegen zwei ähnliche Projekte verglichen. Sequoia ist eine spieltheoretische Implementierung vom Satz von Courcelle [KLR11] und D-Flat ist ein Programm, das es erlaubt dynamische Programme auf Baumzerlegungen zu spezifizieren. Abbildung 1 zeigt einen Kaktusplot, der die Programme vergleicht.

Abb. 1: Ein Kaktusplot, der die Programme Jatatosk, Sequoia und D-Flat auf über 250 Instanzen vergleicht. Auf der x-Achse ist die Zeit, die die Programme *pro Instanz* zur Verfügung haben, aufgetragen; auf der y-Achse die Anzahl der Instanzen, die in der jeweiligen Zeit gelöst werden konnten.

4 Ausblick

In dieser Arbeit haben wir das faszinierende Feld der parametrisierten Komplexitätstheorie von einem parallelen Standpunkt aus betrachtet. Dazu haben wir parallele Subklassen von FPT definiert und analysiert, inwieweit sich klassische Techniken der parametrisierten Komplexitätstheorie in diesen umsetzen lassen. Daraus ergeben sich viele spannende Fragen, sowohl in der Komplexitätstheorie als auch in der Algorithmik. Zum Beispiel haben alle in dieser Arbeit vorgestellten Kernel, die in FAC^0 berechnet werden können, exponentielle Größe. Es gibt allerdings aktuell kein Argument, das Kernel polynomieller Größe ausschließt. Daher wäre es interessant, natürliche Probleme zu finden, für die FAC^0-Schaltkreise Kernel polynomieller Größe berechnen können. Von einem algorithmischen Standpunkt sei erwähnt, dass die Resultate dieser Arbeit immer kleine parallele Zeit (also kleine Schaltkreistiefe) zum Ziel hatten – ebenso spannend ist es aber, die *Arbeit* paralleler Algorithmen zu minimieren (also die Größe der Schaltkreise).

Literaturverzeichnis

[AYZ95] Alon, Noga; Yuster, Raphael; Zwick, Uri: Color-Coding. Journal of the ACM, 42(4):844–856, 1995.

[Ba19] Bannach, Max: Parallel Parameterized Algorithms. Dissertation, Universität zu Lübeck, Germany, 2019.

[BJ14] Berg, Jeremias; Järvisalo, Matti: SAT-Based Approaches to Treewidth Computation: An Evaluation. In: ICTAI. 2014.

[BK06] Bodlaender, Hans L.; Koster, Arie M. C. A.: Safe Separators for Treewidth. Discrete Mathematics, 306(3):337–350, 2006.

[CFH17] Chen, Yijia; Flum, Jörg; Huang, Xuangui: Slicewise Definability in First-Order Logic with Bounded Quantifier Rank. In: CSL. 2017.

[Co90] Courcelle, Bruno: Graph Rewriting: An Algebraic and Logic Approach. In: Formal Models and Semantics, Handbook of Theoretical Computer Science. Elsevier, Amsterdam, Netherlands and MIT Press, Cambridge, Massachusetts, 1990.

[EJT12a] Elberfeld, Michael; Jakoby, Andreas; Tantau, Till: Algorithmic Meta Theorems for Circuit Classes of Constant and Logarithmic Depth. In: STACS. 2012.

[EJT12b] Elberfeld, Michael; Jakoby, Andreas; Tantau, Till: Logspace Versions of the Theorems of Bodlaender and Courcelle. In: FOCS. 2012.

[FG03] Flum, Jörg; Grohe, Martin: Describing Parameterized Complexity Classes. Information and Computation, 187(2):291–319, 2003.

[FG04] Frick, Markus; Grohe, Martin: The Complexity of First-Order and Monadic Second-Order Logic Revisited. Annals of Pure and Applied Logic, 130(1-3):3–31, 2004.

[GD04] Gogate, Vibhav; Dechter, Rina: A Complete Anytime Algorithm for Treewidth. In: UAI. 2004.

[KLR11] Kneis, Joachim; Langer, Alexander; Rossmanith, Peter: Courcelle's Theorem - A Game-Theoretic Approach. Discrete Optimization, 8:568–594, 2011.

[NJ74] Nemhauser, George L.; Jr., Leslie E. Trotter: Properties of Vertex Packing and Independence System Polyhedra. Mathematical Programming, 6(1):48–61, 1974.

[NRW90] Newman, Ilan; Ragde, Prabhakar; Wigderson, Avi: Perfect Hashing, Graph Entropy, and Circuit Complexity. In: CoCo. 1990.

[Ta17] Tamaki, Hisao: Positive-Instance Driven Dynamic Programming for Treewidth. In: ESA. 2017.

Max Bannach wurde am 26. Dezember 1990 in Halle an der Saale geboren. Im Jahr 2009 legte er sein Abitur mit den Leistungskursen Mathematik und Biologie am Bismarck Gymnasium Genthin ab. Anschließend studierte er bis 2014 Informatik an der Universität zu Lübeck. Von 2014 bis 2019 promovierte er am Institut für Theoretische Informatik der gleichen Universität und ist seit 2020 dort als Postdoktorand angestellt. Sein Forschungsinteresse gilt hauptsächlich der parametrisierten und deskriptiven Komplexitätstheorie – mit einem Fokus auf parallele Algorithmen, Schaltkreise und algorithmische Graphentheorie.

Hierarchisches Planen durch Propositionale Logik – Hohe Flexibilität, Vielseitigkeit und Flexibilität[1]

Gregor Behnke[2]

Abstract: Planungsbasierte Assistenzsysteme bieten ihren Nutzern flexible und individualisierte Unterstützung bei der Bewältigung schwieriger Probleme. Für einen erfolgreichen Assistenten wird ein effizientes und flexibles Planungssystem benötigt. Wir demonstrieren, wie mittels einer Transformation in Aussagenlogik der momentan schnellste HTN-Planer konstruiert werden kann und veranschaulichen, auf welche Weise dieser garantiert optimale Lösungen finden kann. Weiterhin zeigen wir, dass die Wünsche des Nutzers in Linearer Temporaler Logik ausgedrückt werden können – und wie dieser in den Planungsprozess einbezogen werden kann. Letztlich stellen wir dar, wie die entwickelten Technologien in Kooperation mit der Robert Bosch GmbH in einem industriellen Kontext verwendet werden. Der im Rahmen dieser Zusammenarbeit entwickelte Assistent ROBERT hilft seinen Nutzern bei der Durchführung von Do-It-Yourself Heimwerker-Projekten.

1 Einführung

Planungstechniken der Künstlichen Intelligenz ermöglichen es, individualisierte und situations-adaptive Unterstützung für vielfältige Problemstellungen anzubieten. Planungsbasierte Assistenzsysteme geben individuelle und kontextabhängige Anweisungen zur Lösung von komplexen Problemen. Sie können z.B. bei technischen Problemen, Haushaltsreparaturen, Do-It-Yourself-Projekten oder schwierigen Montageaufgaben Hilfestellung leisten. Als Basis für diese Assistenz dienen die von Planungssystemen generierten Handlungspläne, die Sequenzen von einzelnen Aktionen sind. Wenn diese ausgeführt werden, lösen sie die vom Nutzer gestellte Aufgabe. Der Handlungsplan dient als Grundlage für eine Schritt-für-Schritt Anleitung, die dem Nutzer präsentiert wird. Der Plan wird individuell für die konkrete Situation, in der sich der Nutzer befindet (z.B. vorhandene Werkzeuge und Materialien), generiert. Der Planer passt die dem Nutzer gegebenen Anweisungen perfekt an die aktuelle Situation an. So verhindert er, dass die Aufgabe etwa wegen fehlenden Materials oder eines fehlenden Werkzeugs scheitert. Planungsbasierte Assistenz sollte den aktuellen Wünschen und Präferenzen des Benutzers Rechnung tragen. Der Assistent muss verhindern, dass er dem Nutzer Instruktionen gibt, die dieser nicht wünscht oder für unangebracht hält. Dies wird durch die Einbeziehung des Nutzers in den Planungsprozess erreicht – durch *gemischt-initiatives* Planen [Sm12]. Im Rahmen dieser Dissertation wurden Planungsprobleme betrachtet, die mittels hierarchischer Task Netzwerke (HTN, [EHN96]) modelliert wurden. Für die Unterstützung (menschlicher) Nutzer eignen sich HTNs besonders gut, da ihre Art Probleme zu beschreiben strukturell der menschlichen Problemlösestrategie ähnelt [By77, WM05].

[1] Englischer Titel der Dissertation: „Hierarchical Planning through Propositional Logic – Highly Efficient, Versatile, and Flexible“

[2] Albert-Ludwigs-Universität Freiburg, Institut für Informatik, behnkeg@informatik.uni-freiburg.de

cc by sa

Das Hauptaugenmerk dieser Arbeit [Be19a] liegt auf der Entwicklung eines effizienten HTN-Planungssystems, das in der Lage ist, Pläne ausreichend schnell zu finden (Kapitel 3). Hierzu wurde ein neuer Lösungsansatz verwendet: die Transformation des Planungsproblems in Aussagenlogische Formeln. Aus erfüllenden Belegungen dieser Formeln wird anschließend der Plan extrahiert. Die Transformation basiert auf neuartigen Datenstrukturen (Path Decomposition Graphs und Solution Order Graphs), die es erlauben, die Struktur eines HTN-Planungsproblems kompakt darzustellen. Außerdem wurde gezeigt, wie die entwickelten Technologien verwendet werden können, um garantiert *optimale* Pläne zu finden. Dies ist insbesondere für Assistenzsysteme wichtig, da nicht-optimale Pläne zu Anweisungen führen können, die einem Nutzer unsinnig erscheinen. Dies führt dazu, dass der Assistent in Zukunft nicht mehr verwendet wird.

Um Anfragen und Anforderungen des Nutzers in den Planungsprozess einbinden zu können, müssen diese zunächst geeignet repräsentiert werden. Hierzu verwenden wir Formeln in Linearer Temporaler Logik (LTL). Wir untersuchen sowohl die theoretischen als auch die praktischen Aspekte des HTN-Planens unter gegebenen LTL-Formeln (Kapitel 4) .

In einem DFG-geförderten Transferprojekt mit der Robert Bosch GmbH wurde gezeigt, dass die entwickelten Techniken auch in einem industriellen Kontext erfolgreich eingesetzt werden können (Kapitel 5). Es wurde das Assistenzsystem ROBERT entwickelt, das Nutzer durch ein handwerkliches Do-It-Yourself-Projekt führt. Mit ROBERT können Nutzer selbst komplexeste Projekte erfolgreich beenden.

2 Hierarchisches Planen

Beim Planen mit hierarchischen Task-Netzwerken (HTN, [EHN96]) werden zwei Arten von Aktivitäten unterschieden: primitive Aktionen und abstrakte Tasks. Während primitive Aktionen über eine Zustands-Übergangs-Semantik definiert sind, repräsentieren abstrakte Tasks komplexere Handlungsfolgen. Ihre Semantik ist über sogenannte Dekompositionsmethoden definiert. Eine solche Methode erlaubt die Ausführung eines abstrakten Task t, indem statt t eine partiell geordnete Menge tn von anderen Tasks und Aktionen ausgeführt wird. Diese Menge tn nennt man *Task Netzwerk*. Das Ziel eines HTN-Planungsproblems ist durch einen abstrakten Task gegeben, der ausgeführt werden soll. Dies könnte z.B. `baue-ein-Vogelhaus` sein. Um einen Plan zu erhalten, muss nun eine Dekompositionsmethode auf den initialen Task angewendet werden. Das Resultat ist ein Task Netzwerk, das wiederum abstrakte Tasks enthalten kann. Auch auf diese müssen Dekompositionsmethoden angewendet werden. Dieser Prozess wird wiederholt, bis ein Task-Netzwerk gefunden wird, das nur primitive Aktionen enthält und dessen Aktionen im gegebenen initialen Zustand in der gegebenen Reihenfolge ausführbar sind. Folglich ist ein Plan genau dann eine Lösung für ein HTN-Planungsproblem, wenn er ausführbar ist und es eine Folge von Anwendungen von Dekompositionsmethoden – eine *Ableitung* – gibt, die ihn aus dem initialen abstrakten Task entstehen lässt.

3 Hierarchisches Planen mittels Aussagenlogik

Die Entwicklung effizienter Planer für HTN-Planungsprobleme ist besonders schwierig. Algorithmen müssen beide Dimensionen der Problemstellung – Zustandssemantik der Aktionen und Dekompositionsmethoden – und ihre Wechselwirkungen berücksichtigen. Traditionellerweise werden HTN Planungsprobleme durch Suche – im Plan- oder Progressionsraum – gelöst. Wir präsentieren eine neue Möglichkeit zur Lösung von HTN-Planungsproblemen: die Transformation in Aussagenlogische Formeln. Dabei wird eine Formel $\mathscr{F}$ gesucht, die genau dann erfüllbar ist, wenn das gegebene HTN-Planungsproblem eine Lösung besitzt. Eine erfüllende Belegung kann mit Hilfe eines der existierenden, effizienten SAT Solver gefunden werden. Aus dieser Belegung kann anschließend ein Plan extrahiert werden. Dieser kann die Grundlage für eine Schritt-für-Schritt Anleitung sein.

Da HTN Planung im Allgemeinen unentscheidbar ist [EHN96, GB11], kann es eine solche Transformation im Allgemeinen nicht geben. Stattdessen muss für ein individuelles Planungsproblem eine (potentiell unendliche) Sequenz $(\mathscr{F}_k)$ von Aussagenlogischen Formeln angegeben werden. Sofern das Planungsproblem eine Lösung hat, wird eine dieser Formeln erfüllbar sein – ebenso wie alle folgenden. Wir können folglich die $\mathscr{F}_k$ sequentiell auf Erfüllbarkeit testen.

Für die k-te generierte Formel betrachten wir eine Beschränkung des HTN-Planungsproblems: die k-Tiefenbeschränkung. Wir suchen mittels der k-te Formel also nur diejenigen Pläne, die über eine Zerlegung von höchstens der Tiefe k aus dem initialen abstrakten Task abgeleitet werden können. Das heißt, es ist eine Formel $\mathscr{F}_k$ zu konstruieren, die genau dann erfüllbar ist, wenn ein solcher Plan existiert. Diese Formel muss die möglichen Pläne und deren Ableitungen vom initialen abstrakten Task repräsentieren. Eine explizite Beschreibung all dieser möglichen Ableitungen ist nicht möglich, da es bis zu $\mathscr{O}(2^{2^k})$ solcher Ableitungen geben kann. Folglich benötigen wir eine kompakte Repräsentation der möglichen Pläne und ihrer Ableitungen.

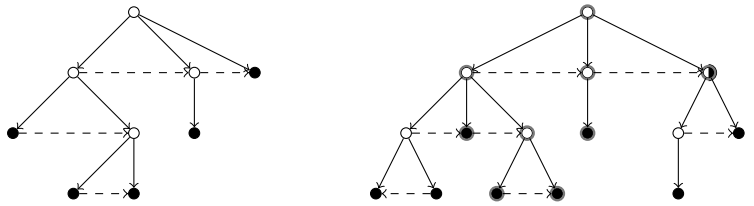


Abb. 1: Eine mögliche Ableitung eines HTN Planes (links) und ein Path Decomposition Tree (PDT, rechts). Weiße Knoten beschreiben abstrakte Tasks, schwarze primitive Aktionen. Der PDT enthält die links gezeigte Ableitung als Teilbaum. Die entsprechenden Knoten sind rot eingefärbt.

Hierzu führen wir das Konzept der Path Decomposition Trees (PDTs) ein. Der k-PDT eines HTN-Planungsproblems ist ein Baum, der alle möglichen k-tiefenbeschränkten Ableitungen als Teilbäume enthält. Abbildung 1 zeigt eine Ableitung und einen PDT, der die Ableitung als Teilbaum enthält. Eine erfüllende Belegung der zu konstruierenden Formel $\mathscr{F}_k$ wird nun einen solchen Teilbaum auswählen. Hierfür hat es für jeden Knoten v des PDT und Task t eine Variable t_v. Falls für einen Knoten v eine der Variablen t_v wahr ist,

repräsentiert dies, dass der Knoten Teil der ausgewählten Ableitung ist. Die einzelnen t_v kodieren, welcher Task an der entsprechenden Stelle in der Ableitung vorkommt.

Neben der Wahl einer konkreten Ableitung über die Belegung der a_v Variablen muss weiterhin sichergestellt werden, dass die resultierenden primitiven Aktionen im initialen Zustand ausführbar sind. Hierfür können existierende Kodierungen aus dem klassischen Planen [RHN06] verwendet werden [BHB19a, BHB18a, BHB18b]. Die abgeleiteten primitiven Aktionen müssen weiterhin in einer mit der Ableitung kompatiblen Reihenfolge ausführbar sein. Hierzu wurden drei verschiedene Möglichkeiten der Kodierung in Aussagenlogik entwickelt. Zunächst haben wir eine Spezialkodierung für den Fall entwickelt, dass alle Dekompositionsmethoden Sequenzen von Tasks und Aktionen sind [BHB18a]. Die zweite, allgemeine Kodierung repräsentiert die durch die Ableitung implizierten Ordnungsbeschränkungen mithilfe zusätzlicher Variablen [BHB18b]. Im Allgemeinen gibt es keine eindeutige Ordnung zwischen den Blättern des PDT – welche die primitiven Aktionen repräsentieren, die aus der Ableitung entstehen. Die resultierende Kodierung in Aussagenlogik hat $\mathcal{O}(n^4)$ Klauseln, wobei n die Anzahl der Blätter des PDT ist.

Wir haben gezeigt, dass dies signifikant verbessert werden kann. Hierzu schlagen wir vor, gemeinsam mit dem PDT eine partielle Ordnung seiner Blätter zu bestimmen. Diese wird durch einen gerichteten azyklischen Graphen, den Solution Order Graph (SOG), repräsentiert [BHB19a]. Die Knoten des SOG sind die Blätter die PDT, also die möglicherweise in einem abgeleiteten Plan enthaltenen primitiven Aktionen. Der SOG ist dann derart zu bestimmen, dass jedes konkrete ableitbare Task Netz ein induzierter Teilgraph des SOG ist. Hierdurch wird garantiert, dass, falls ein Blatt des PDT Teil der konkret gewählten Ableitung ist, dessen relative Ordnung zu anderen ebenfalls aus der Ableitung entstandenen Tasks immer die selbe, also statisch ist. Dies erlaubt eine erheblich kompaktere Kodierung mit lediglich $\mathcal{O}(n^2\Delta)$ vielen Klauseln, wobei Δ der maximale Ein-Grad eines Knotens im SOG ist. Neben dieser Anwendung zur Erstellung einer kompakteren aussagenlogischen Kodierung ist die Struktur des SOG auch vielversprechend für zukünftige Untersuchungen. Da er *alle* (k-tiefenbeschränkten) Lösungen eines HTN-Planungsproblems kompakt repräsentiert, kann er die Basis für die Extraktion von Eigenschaften des Planungsproblems, z.B. von HTN-spezifischen Mutexen o.ä. sein.

Fraglich ist nun, wie ein solcher PDT und der zugehörige SOG effizient berechnet werden können. Es ist praktisch unmöglich, alle Ableitungen zu berechnen und anschießend einen passenden PDT zu bestimmen. Stattdessen verwenden wir einen Algorithmus, der auf lokaler Expansion basiert. Wir beginnen mit einem PDT, der lediglich einen Wurzelknoten w enthält und weisen diesem eine Menge $\alpha(w) = \{a_I\}$ von möglichen Tasks und Aktionen zu, die für w genau den initialen abstrakten Task enthält. Nun betrachten wir wiederholt ein Blatt b des aktuellen PDTs und dessen Task-Menge $\alpha(b)$. Wir bestimmen dann alle auf die abstrakten Tasks in $\alpha(b)$ anwendbaren Methoden und deren Tasknetze. Wir interpretieren diese als eine Familie von Graphen (G_i). Anschließend bestimmen wir einen Graphen G^*, so dass alle G_i induzierte Teilgraphen von G^* sind. Wie wir gezeigt haben, ist bereits die Bestimmung von G^* $\mathbb{NP}$-vollständig[3] [BHB19a], jedoch nötig um einen SOG aus dem PDT zu extrahieren. Wir verwenden deshalb einen Greedy-Algorithmus, um G^* im Planer

[3] Das Entscheidungsproblem fragt, ob ein G^* mit weniger als x Knoten existiert.

zu bestimmen. Wir fügen nun die Knoten von G^* als Kinder von b in den PDT ein und bestimmen die Menge α für jeden Knoten in G^*, basierend auf den Homomorphismen der G_i in G^*. Wir wiederholen diesen Prozess, bis alle Blätter des PDT die Tiefe k haben.

Der beschriebene Prozess der Erstellung eines PDT/SOG-Paares wurde automatisiert und auf seiner Basis ein SAT-basierter HTN-Planer auch praktisch umgesetzt. Dieser Planer wurde auf einer Menge von Benchmark-Instanzen gegen andere State-of-the-Art HTN-Planungssysteme verglichen[4]. Die jeweils pro Domäne und insgesamt gelösten Instanzen sind in Tabelle 1 dargestellt. Es lässt sich erkennen, dass der neu entwickelte SAT-basierte HTN-Planer signifikant mehr Instanzen als bisherige Techniken löst. Insbesondere schwerere und komplexere Planungsprobleme lassen sich mit dem neuen Ansatz leichter und deutlich schneller lösen.

	#instances	SAT-F ∃ expMC	SAT-F ∃ MapleLCM	SAT-F ∃ CaDiCaL	SAT-F ∃ cryptominisat	SAT-F expMC	SAT-F MapleLCM	SAT-F CaDiCaL	SAT-F cryptominisat	SAT-tree ∃ expMC	SAT-tree ∃ MapleLCM	SAT-tree ∃ CaDiCaL	SAT-tree ∃ cryptominisat	SAT-tree expMC	SAT-tree MapleLCM	SAT-tree CaDiCaL	SAT-tree cryptominisat	PANDApro lm-cut	PANDApro FF	PANDApro ADD	TDG-m greedy A*	TDG-c greedy A*	UMCP H	UMCP BF	UMCP DF	HTN2STRIPS jasper	HTN2STRIPS FD-SS 2018	HTN2STRIPS SaarPlan	HTN2STRIPS LAPKT-BFWS	HTN2STRIPS MpC	SHOP2	FAPE
UM-TRANSLOG	22	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	**22**	19	17	17	17	6	**22**	-
SATELLITE	25	**25**	**25**	**25**	**25**	24	24	**25**	**25**	**25**	**25**	**25**	**25**	**25**	**25**	**25**	**25**	**25**	24	23	**25**	21	23	18	20	23	19	14	12	0	22	22
WOODWORKING	11	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	**11**	10	9	9	8	10	6	6	6	5	5	5	5	4	8	0
SMARTPHONE	7	**7**	**7**	**7**	**7**	**7**	**7**	6	**7**	6	6	**7**	**7**	6	6	6	**7**	5	5	5	5	5	4	4	4	6	6	5	5	4	4	-
PCP	17	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	11	**12**	11	**12**	11	**12**	9	10	11	9	8	0	0	0	3	3	3	3	0	0	-
ENTERTAINMENT	12	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	**12**	11	11	**12**	9	9	6	5	5	5	5	5	4	4	5	-
ROVER	20	10	**11**	9	8	5	6	4	4	4	4	4	6	4	4	4	5	4	3	4	2	2	0	0	0	5	5	4	4	4	3	3
TRANSPORT	30	**22**	20	20	20	15	14	15	17	**22**	20	19	21	15	15	15	18	9	11	7	1	1	0	1	0	19	17	13	13	3	0	-
total	144	**121**	120	118	117	108	108	107	110	114	112	111	116	106	107	106	112	95	95	93	81	78	61	56	57	85	77	66	63	25	64	25/56

Tab. 1: Anzahl der von verschiedenen HTN-Planungssystemen gelösten Benchmark-Instanzen [BHB19a]. SAT-tree beschreibt die Kodierung mittels zusätzlicher Variablen. SAT-F beschreibt die Kodierung auf Basis des SOG. Kodierungen mit ∃ verwenden die klassische ∃-step Kodierung [RHN06], diejenigen ohne die Kautz&Selman-Kodierung [KS96]. Zuletzt wird der jeweils verwendete SAT-Solver genannt. PANDApro [Hö19, Hö18], TDG [Be17], UMCP [EHN94], HTN2STRIPS [Al16], SHOP2 [Na03] und FAPE [Dv14] sind HTN-Planer aus der Literatur.

Die bis hierhin beschriebene Technik zum Finden von Plänen mittels einer Transformation in Aussagenlogik ermöglicht nur das Finden *irgendeines* Planes. Insbesondere in planungsbasierten Assistenzsystemen ist es jedoch von großer Bedeutung, einen Kosten-optimalen Plan zu finden. Nicht-optimale Pläne enthalten regelmäßig überflüssige Aktionen. Womöglich könnte ein solcher Plan den Nutzer anweisen, eine Batterie aus einem Gerät zu entfernen und sofort wieder einzusetzen. Derartige Anweisungen sind für den Nutzer klar als unsinnig zu erkennen. Dies führt zu einem Verlust an Vertrauen in die Fähigkeiten des Assistenten und kann einen Abbruch der Interaktion mit ihm verursachen.

Mittels der entwickelten aussagenlogischen Kodierung kann auch ein Kosten-optimaler Plan gefunden werden [BHB19b]. Hierzu wird zunächst ein beliebiger Plan gesucht. Dessen Kosten C^* bilden eine obere Schranke für die Kosten des optimalen Plans. Anschließend wird der optimale Plan mittels weiterer SAT-Aufrufe unter einer Kosten-Beschränkung und binärer Suche gefunden. Für eine gegebene Kosten-Schranke C ist hierzu eine Tiefen-

[4] Dabei erhielt jeder Planer für jede Instanz 10 Minuten Zeit und 4 GB RAM auf einem Intel Xeon E5-2660.

beschränkung $k(C)$ derart zu wählen, dass jeder Plan mit höchstens Kosten C eine Tiefe von höchstens $k(C)$ besitzt. Wir haben gezeigt, wie eine solche Tiefenbeschränkung approximiert werden kann [BHB19b]. Experimente haben erwiesen, dass diese Approximation in der Regel sehr genau ist und dass der resultierende, optimale HTN-Planer ebenfalls andere Ansätze zum Finden optimaler HTN-Pläne signifikant dominiert. Wir haben damit insgesamt die Grundlage für die Verwendung von effizienten HTN-Planungssystemen für planungsbasierte Assistenzsysteme gelegt.

4 Änderung von Plänen

Planungsbasierte Assistenzsysteme müssen sowohl einen schnellen, effizienten Planer als auch einen flexiblen Planer besitzen. Dieser muss in der Lage sein, auf die Wünsche und Präferenzen des Nutzers einzugehen. Im Rahmen eines planungsbasierten Assistenzsystems wird der Nutzer diese bezüglich eines konkreten, vom Assistenten vorgeschlagenen Plans äußern, d.h. seine Wünsche werden die Gestalt einer Anfrage zur Änderung eines Planes haben. Wir haben zunächst die theoretischen Grundlagen derartiger Wünsche untersucht. Hier stellt sich zunächst die Frage, wie derartige Wünsche geeignet formal repräsentiert werden können. Wir schlagen vor, dafür Formeln in Linearer Temporaler Logik (LTL[5] [Pn77]) zu verwenden. So kann beispielsweise in einem Heimwerkerszenario der Wunsch „Ich möchte alle Sägekanten schleifen.“ als $\mathrm{G}(\mathit{sägen}(x) \rightarrow \mathrm{E}\mathit{schleifen}(x))$ modelliert werden. Derartige Formeln können mittels Techniken des maschinellen Lernens aus natürlich-sprachlichen Aussagen des Nutzers extrahiert werden.

Der Ausgangspunkt für unsere theoretischen Untersuchungen war das *Planverifikationsproblem*. Dieses stellt die Frage, ob eine gegebene Sequenz von primitiven Aktionen eine Lösung für ein gegebenes HTN-Planungsproblem darstellt. Es handelt sich dabei gleichzeitig um den speziellsten Wunsch, den ein Nutzer ausdrücken kann (eine Formel der Form $a_1 \wedge X(a_2 \wedge X(\ldots))$). Wir haben gezeigt, dass das Planverifikationsproblem $\mathbb{NP}$-vollständig ist [BHB15]. Basierend auf diesem Ergebnis haben wir die Berechnungskomplexität der elementarsten Anfragen zur Änderung eines Plans (Hinzufügen, Entfernen, Austauschen, Neuordnen von Aktionen) untersucht und gezeigt, dass diese dieselbe Komplexität wie das jeweilige Planexistenzproblem haben [Be16]. In der Dissertation wurde dieses Ergebnis auf allgemeine LTL-Formeln erweitert. Nach unserem besten Wissen – ist dies die erste komplexitätstheoretische Analyse des gemischten-initiativen HTN-Planens.

Um Wünsche und Anfragen des Nutzers nun auch praktisch beantworten zu können, haben wir entsprechende Techniken entwickelt. Zunächst haben wir einen Planverifizierer für HTN-Pläne erarbeitet, der auf einer Transformation des Problems in Aussagenlogik basiert [BHB17]. Er ist der erste Verifikator für Lösungen von HTN-Planungsproblemen. Planverifikation ist neben der Anwendung in planungsbasierten Assistenzsystemen auch für die Forschung von hoher Bedeutung, da sie die *unabhängige* Überprüfung von entwickelten Planungssystemen ermöglicht und so zur Objektivierung der Forschung beiträgt. Die Publikation unseres Planverifizierers hat bereits weitere Forschung anderer Gruppen angestoßen hat [BMC18].

[5] Technisch betrachten wir LTL über finiten Sequenzen, also LTLf [DGV13].

Der Planer muss nun in der Lage sein, einen Plan zu finden, der sowohl eine Lösung des eigentlichen Planungsproblems ist, als auch eine LTL Formel ϕ erfüllt. Als Basis verwenden wir den im vorherigen Kapitel beschriebenen HTN Planer auf Basis von Aussagenlogik. Der deklarative Charakter der Kodierung – sie modelliert schlicht, *was* eine Lösung ist – erlaubt es, Kodierungen für weitere Einschränkungen, wie z.B. LTL Formeln, hinzuzufügen. Da sich die LTL-Formeln (momentan) nur auf die primitiven Aktionen eines Planes beziehen, sind die für das klassische Planen entwickelten Kodierungen für LTL-Formeln auch auf HTN-Planungsprobleme anwendbar. Wir verwenden die Kodierung von Mattmüller und Rintanen [MR07]. Leider unterstützt diese Kodierung nicht alle syntaktisch gültigen LTL-Formeln, sondern nur solche ohne den Next-Operator X. Dieser Operator ist schwierig zu handhaben, da sein Vorhandensein die Ausnutzung der Stotteräquivalenz [La83] verbietet. Er ist jedoch in der Praxis notwendig, da er das einzige Mittel ist, um festzulegen, dass eine bestimmte Aufgabe ausgeführt werden muss, wenn die unmittelbar nächste Aktion – z.B. in einer Benutzeranforderung "Ich will als nächstes sägen" – ausgeführt wird. Wir konnten zeigen, dass der X-Operator trotz der fehlenden Stotteräquivalenz effizient in die Kodierung integriert werden kann [BB18]. Die hierfür notwendigen Änderungen führten zu weiteren Modifikationen an der bisherigen Kodierung, die diese ebenfalls effizienter machten.

Abb. 2: ROBERT erklärt deinem Nutzer wie er einen 3mm Metallbohrer erkennt und diesen in die Bohrmaschine einsetzt.

5 Planungsbasierte Assistenz in der Praxis

Im Rahmen der Dissertation wurden theoretische und technische Konzepte erarbeitet und umgesetzt. Darüber hinaus wurden diese in der praktischen, industrienahen Anwendung umgesetzt und evaluiert. Im Rahmen eines DFG-Transferprojekts gemeinsam mit der Robert Bosch GmbH wurde der planungsbasierte Assistent ROBERT entwickelt [Be19b, Be18]. Er unterstützt bei der Durchführung von Do-It-Yourself Heimwerkerprojekten. Ohne geeignete Assistenz überfordern diese schnell durch komplexe Anforderungen hinsichtlich angemessener Vorgehensweisen, Wahl der passenden Werkzeuge und Arbeitsmittel. Der multimediale und situations-adaptive Heimwerkerassistent ROBERT leitet seinen Nutzer

schrittweise durch das Heimwerkerprojekt. Die gegebene Anleitung wird durch den in dieser Dissertation entwickelten Planer bestimmt. Das dabei verwendete Planungsmodell ist sehr komplex, da es die verschiedensten Möglichkeiten zur Verwendung von Materialien und Werkzeugen beschreiben muss. Aktuell ist nur der hier präsentierte HTN-Planer in der Lage, die von ROBERT benötigten Pläne in einem akzeptablen Zeitrahmen zu finden.

ROBERT präsentiert die gefundenen Handlungsschritte als eine multimediale Anleitung, deren Inhalte dynamisch anhand der durchzuführenden Handlung bestimmt werden. Ein Beispiel für eine solche Anleitung wird in Abbildung 2 gezeigt. ROBERT ist auch in der Lage, Fragen des Nutzers mittels seines Hintergrundwissens passend zu beantworten. Insgesamt ermöglicht ROBERT so eine vollkommen neuartige Unterstützung – vor allem von bisher unerfahrenen Heimwerkern, denen bis jetzt nur textuelle Anleitungen und nicht individualisierte YouTube-Videos zur Verfügung standen. Durch seine detaillierte und vertrauenswürdige Assistenz ermöglicht ROBERT es Personen sogar, ihre „Angst“ vor elektrischen Heimwerkergeräten, wie z.B. einer Stichsäge, abzulegen und diese erstmals erfolgreich zu verwenden. ROBERT wurde in gemeinsam mit der Robert Bosch GmbH durchgeführten Studien mehrfach erfolgreich mit Probanden getestet [Be20, Be19b].

6 Zusammenfassung

Damit die Planer die Grundlage für eine sinnvolle und individualisierte Assistenz bieten können, müssen sie sowohl schnell als auch mit der Fähigkeit ausgestattet sein, Pläne entsprechend der Wünschen ihrer Nutzer zu ändern. Wir haben gezeigt, wie diese beiden Herausforderungen für hierarchische Planungsprobleme angegangen werden können. Zunächst wurde ein höchst effizienter neuer Ansatz zur Generierung von HTN-Plänen erarbeitet: HTN-Planen mittels Aussagenlogik. Hierzu wurden Path Decomposition Trees (PDTs) und Solution Order Graphs (SOGs) entwickelt, die eine kompakte Repräsentation eines k-tiefenbeschränkten HTN-Planungsproblems ermöglichen. Auf Basis von PDT und SOG entwickelten wir drei Kodierungen des Planungsproblems in Aussagenlogik. Eine empirische Evaluation des resultierenden Planers zeigte, dass er signifikant bessere Ergebnisse als der bisherige State-of-the-art im HTN-Planen zeigt. Weiterhin modifizierten wir den Planer so, dass er garantiert optimale Pläne finden kann. Anschließend wurden die von einem Nutzer geäußerten Wünsche zur Änderung eines Planes untersucht. Es wurde zunächst Lineare Temporal Logik als eine geeignete Repräsentation für diese Wünsche identifiziert. Anschließend wurde die Berechnungskomplexität der Beachtung dieser Wünsche während der Generierung von Handlungsplänen untersucht. Letztlich wurde gezeigt, wie diese als LTL-Formeln ausgedrückten Wünsche in den Planungsprozess eingebunden werden können. Dabei wurde die bisher für das Planen bekannte Kodierung von LTL in Aussagenlogik verbessert. Die entwickelten Techniken wurden in der praktischen Anwendung erprobt. Gemeinsam mit der Robert Bosch GmbH wurde der planungsbasierte Assistent ROBERT entwickelt, der Heimwerker bei Do-It-Yourself Projekten unterstützt. Mithilfe der entwickelten Planungstechnologien kann ROBERT seine Anweisungen in Realzeit auf die konkrete Situation des Nutzers anpassen und vorhandene Materialien und Werkzeuge optimal verwenden. Außerdem kann ROBERT die Wünsche und Anweisungen des Nutzers in die gegebenen Anweisungen mit einbeziehen.

Literaturverzeichnis

[Al16] Alford, Ron; Behnke, Gregor; Höller, Daniel; Bercher, Pascal; Biundo, Susanne; Aha, David W.: Bound to Plan: Exploiting Classical Heuristics via Automatic Translations of Tail-Recursive HTN Problems. In: ICAPS. 2016.

[BB18] Behnke, Gregor; Biundo, Susanne: X and more Parallelism: Integrating LTL-Next into SAT-based Planning with Trajectory Constraints While Allowing for Even More Parallelism. Inteligencia Artificial, 21(62):75–90, 2018.

[Be16] Behnke, Gregor; Höller, Daniel; Bercher, Pascal; Biundo, Susanne: Change the Plan – How hard can that be? In: ICAPS. 2016.

[Be17] Bercher, Pascal; Behnke, Gregor; Höller, Daniel; Biundo, Susanne: An Admissible HTN Planning Heuristic. In: IJCAI. 2017.

[Be18] Behnke, Gregor; Schiller, Marvin; Kraus, Matthias; Bercher, Pascal; Schmautz, Mario; Dorna, Michael; Minker, Wolfgang; Glimm, Birte; Biundo, Susanne: Instructing Novice Users on How to Use Tools in DIY Projects. In: IJCAI-ECAI. 2018.

[Be19a] Behnke, Gregor: Hierarchical planning through propositional logic: highly efficient, versatile, and flexible. dissertation, Ulm University, 2019.

[Be19b] Behnke, Gregor; Schiller, Marvin; Kraus, Matthias; Bercher, Pascal; Schmautz, Mario; Dorna, Michael; Dambier, Michael; Minker, Wolfgang; Glimm, Birte; Biundo, Susanne: Alice in DIY-Wonderland or: Instructing novice users on how to use tools in DIY projects. AI Communications, 32(1):31–57, 2019.

[Be20] Behnke, Gregor; Bercher, Pascal; Kraus, Matthias; Schiller, Marvin; Mickeleit, Kristof; Häge, Timo; Dorna, Michael; Dambier, Michael; Minker, Wolfgang; Glimm, Birte; Biundo, Susanne: New Developments for Robert – Assisting Novice Users Even Better in DIY Projects. In: ICAPS. 2020.

[BHB15] Behnke, Gregor; Höller, Daniel; Biundo, Susanne: On the Complexity of HTN Plan Verification and Its Implications for Plan Recognition. In: ICAPS. 2015.

[BHB17] Behnke, Gregor; Höller, Daniel; Biundo, Susanne: This is a solution! (... but is it though?) – Verifying solutions of hierarchical planning problems. In: ICAPS. 2017.

[BHB18a] Behnke, Gregor; Höller, Daniel; Biundo, Susanne: totSAT – Totally-Ordered Hierarchical Planning through SAT. In: AAAI. 2018.

[BHB18b] Behnke, Gregor; Höller, Daniel; Biundo, Susanne: Tracking Branches in Trees – A Propositional Encoding for solving Partially-Ordered HTN Planning Problems. In: ICTAI. 2018.

[BHB19a] Behnke, Gregor; Höller, Daniel; Biundo, Susanne: Bringing Order to Chaos – A Compact Representation of Partial Order in SAT-based HTN Planning. In: AAAI. 2019.

[BHB19b] Behnke, Gregor; Höller, Daniel; Biundo, Susanne: Finding Optimal Solutions in HTN Planning – A SAT-based Approach. In: IJCAI. 2019.

[BMC18] Barták, Roman; Maillard, Adrien; Cardoso, Rafael C.: Validation of Hierarchical Plans via Parsing of Attribute Grammars. In: ICAPS. 2018.

[By77] Byrne, Richard: Planning meals: Problem solving on a real data-base. Cognition, 1977.

[DGV13] De Giacomo, Giuseppe; Vardi, Moshe Y.: Linear Temporal Logic and Linear Dynamic Logic on Finite Traces. In: IJCAI. 2013.

[Dv14] Dvorak, Filip; Bit-Monnot, Arthur; Ingrand, Félix; Ghallab, Malik: A flexible ANML actor and planner in robotics. In: PlanRob. 2014.

[EHN94] Erol, Kutluhan; Hendler, James; Nau, Dana: UMCP: A Sound and Complete Procedure for Hierarchical Task-Network Planning. In: AIPS. 1994.

[EHN96] Erol, Kutluhan; Hendler, James; Nau, Dana: Complexity results for HTN planning. Annals of Mathematics and AI, 18(1):69–93, 1996.

[GB11] Geier, Thomas; Bercher, Pascal: On the Decidability of HTN Planning with Task Insertion. In: IJCAI. 2011.

[Hö18] Höller, Daniel; Bercher, Pascal; Behnke, Gregor; Biundo, Biundo: A Generic Method to Guide HTN Progression Search with Classical Heuristics. In: ICAPS. 2018.

[Hö19] Höller, Daniel; Bercher, Pascal; Behnke, Gregor; Biundo, Susanne: HTN Planning as Heuristic Progression Search. JAIR, 2019.

[KS96] Kautz, Henry; Selman, Bart: Pushing the Envelope: Planning, Propositional Logic, and Stochastic Search. In: AAAI. 1996.

[La83] Lamport, Leslie: What Good is Temporal Logic? In: Information Processing. 1983.

[MR07] Mattmüller, Robert; Rintanen, Jussi: Planning for Temporally Extended Goals as Propositional Satisfiability. In: IJCAI. 2007.

[Na03] Nau, Dana; Au, Tsz-Chiu; Ilghami, Okhtay; Kuter, Ugur; Murdock, J.; Wu, Dan; Yaman, Fusun: SHOP2: An HTN Planning System. Journal of Artificial Intelligence Research (JAIR), 20:379–404, 2003.

[Pn77] Pnueli, Amir: The Temporal Logic of Programs. In: SFCS. 1977.

[RHN06] Rintanen, Jussi; Heljanko, Kejio; Niemelä, Ilkka: Planning as satisfiability: parallel plans and algorithms for plan search. Artificial Intelligence, 170(12-13):1031–1080, 2006.

[Sm12] Smith, David E.: Planning As an Iterative Process. In: AAAI. 2012.

[WM05] Ward, Geoff; Morris, Robin: Introduction to the psychology of planning. Kapitel 1, 2005.

Gregor Behnke ist Jahrgang 1992. Nach dem Abitur im Jahr 2008 hat er bis 2014 an der Universität Rostock Informatik studiert. Er erwarb dort sowohl einen Bachelor als auch einen Master of Science. Nach seinem Studium war er wissenschaftlicher Mitarbeiter am Institut für Künstliche Intelligent der Universität Ulm von Prof. Dr. Susanne Biundo-Stephan. Er arbeitete dort im Rahmen des SFB/TRR 62 „Eine Companion-Technologie für kognitive technische Systeme“ am Hierarchischen Planen und dessen Einsatz in gemischt-initiativen Planungssystemen. Weiterhin war er im Transferprojekt „Do it yourself, but not alone: Companion-Technologie für die Heimwerkerunterstützung“ beschäftigt und wendete dort die von ihm entwickelten Planungstechnologien zur Unterstützung von Heimwerkern an. Im Dezember 2019 schloss er seine Promotion an der Universität Ulm mit der Note *summa cum laude* ab. Seit Januar 2020 ist er als PostDoc wissenschaftlicher Mitarbeiter am Lehrstuhl für Grundlagen der Künstlichen Intelligenz von Prof. Dr. Bernhard Nebel an der Albert-Ludwigs-Universität Freiburg.

Erweiterte visuelle Benutzerschnittstellen zur Unterstützung verteilter Cloud-basierter Big-Data-Analysen[1]

Marco Xaver Bornschlegl[2]

Abstract: Benutzerfreundliche Schnittstellen, die sich sowohl durch intuitive Bedienung als auch durch ein leichtes Erlernen charakterisieren, bieten den Anwendern von Big-Data-Analysis Software oftmals nur allgemeine Informationen sowie eine geringe Interaktionsmöglichkeit. Im Gegensatz dazu und insbesondere in Situationen, in denen Anwender mehr Kontrolle über unterschiedliche Aspekte der Software benötigen, bieten End-User-Empowered-Schnittstellen spezialisierte Interaktionsmöglichkeiten, die eine größere Benutzungsvielfalt erlauben. Speziell in Big-Data-Analysis-Anwendungen ist es jedoch wichtig, den Anwendern eine Kontext-sensitive Benutzungsschnittstelle zur Verfügung zu stellen, deren Verhalten sich anhand der unterschiedlichen Anwender sowie deren individueller Anwendungsszenarien anpasst. Um diese Forschungsdiskrepanz in Bezug auf Kontext-Sensitivität sowie Informationsvisualisierung bei Big-Data-Analysis-Anwendungsszenarien zu schließen, wurde mit IVIS4BigData ein theoretisches Referenzmodell entwickelt, welches als modernes und innovatives Rahmenwerk für verteilte Big-Data-Analysis-Anwendungsszenarien dient.

1 Einführung und Motivation

Die Verfügbarkeit von Daten hat sich innerhalb der vergangenen zehn Jahre dramatisch verändert. Die weite Verbreitung von internetfähigen mobilen Geräten sowie die Evolution von Web 2.0 Technologien tragen zu einer großen Menge an Daten (bekannt unter dem Begriff Big-Data) bei [Fr14]. Aufgrund der Tatsache, dass wir im *„Informationszeitalter leben“* [Wo11], ist eine kognitiv effiziente Wahrnehmung und Interpretation von Wissen und Information zur Erkennung versteckter Muster, unbekannter Korrelationen oder anderer nützlicher Informationen innerhalb dieser großen Datenmenge (einer Vielzahl an Datentypen [Pr13]) eine große Herausforderung.

Moderne Cloud-Technologien sowie verteilte Computerarchitekturen führen heutzutage zu einer fast unbegrenzten Leistung bei der Speicherung und der Verarbeitung von Daten. Für einen Zugriff auf große und komplexe Datenmengen über eine Vielzahl von verteilten Datenquellen, um diese für die Anwender mithilfe innovativer Strategien visualisieren zu können, werden allerdings moderne Technologien benötigt [Fr15]. Im Gegensatz dazu wandeln traditionelle Big-Data-Analysemethoden die Daten allerdings oftmals direkt zu einer visuellen Repräsentation um, ohne den Anwendern eine Interaktion innerhalb der einzelnen Analyseprozessschritte zu ermöglichen. Weiterhin führt die aktuelle Entwicklung

[1] Englischer Titel der Dissertation: „Advanced Visual Interfaces Supporting Distributed Cloud-Based Big Data Analysis“

[2] FernUniversität Hagen, Fakultät für Mathematik und Informatik, D-58085 Hagen, marco-xaver.bornschlegl@fernuni-hagen.de

cc by sa

innerhalb der Informationstechnologie zu einem stetigen Wachstum an Computer-Systemen sowie der Anwenderzahl [Co09]. Innerhalb des Entwicklungsprozesses von visuellen Benutzungsschnittstellen für Visual-Analytics-Anwendungen kommt es dadurch zu einer wichtigen Entscheidung, welche der beteiligten Parteien – der Mensch oder die Software – die Interaktion kontrolliert. Benutzerfreundliche Schnittstellen, die sich sowohl durch intuitive Bedienung als auch durch ein leichtes Erlernen charakterisieren [Ch14], *„bieten den Anwendern oftmals nur allgemeine Informationen sowie eine geringe Interaktionsmöglichkeit“* [Ng04]. Im Gegensatz dazu und in Situationen in denen Anwender mehr Kontrolle über unterschiedliche Aspekte der Software benötigen, bieten End-User-Empowered-Schnittstellen [Fi12] spezialisierte Interaktionsmöglichkeiten, die eine größere Benutzungsvielfalt erlauben. Speziell in Big-Data-Analysis-Anwendungen ist es somit wichtig, den Anwendern eine Kontext-sensitive Benutzungsschnittstelle zur Verfügung zu stellen [Fi12], deren Verhalten sich anhand der unterschiedlichen Anwender sowie deren individuellen Anwendungsszenarien anpasst [Ro16].

Um diese Forschungsdiskrepanz in Bezug auf Kontext-Sensitivität sowie Informationsvisualisierung bei Big-Data-Analysis-Anwendungsszenarien zu schließen, wurde mit IVIS4-BigData ein theoretisches Referenzmodell innerhalb dieser Dissertation mit dem Thema **Advanced Visual Interfaces Supporting Distributed Cloud-Based Big Data Analysis** [Bo19] entwickelt, welches die neuen Gegebenheiten der modernen Informationstechnologie berücksichtigt und als modernes und innovatives Rahmenwerk für verteilte Big-Data-Analysis-Anwendungsszenarien dient.

2 State of the Art in Wissenschaft und Technik

Informationsvisualisierung (IVIS) ist ein Resultat *„aus den Forschungen in den Bereichen Mensch-Maschine Kommunikation, Informatik, Grafik, visuellem Design, Psychologie sowie Geschäftsmethoden“* [TC+06]. Gemäß Rainer Kuhlen kann IVIS jedoch auch, aufgrund der fehlenden direkten Verbindung zwischen zwei Personen, aus der Frage nach dem Austausch von Ideen und Informationen zwischen unterschiedlichen Personen hergeleitet werden [Ku04]. Shneiderman betont zusätzlich, dass *„das Durchsuchen von Informationssammlungen mit zunehmendem Umfang immer schwieriger wird. Eine einzelne Informationsseite ist zwar leicht zu durchsuchen, aber wenn die Informationen den Umfang eines Buches, einer Bibliothek oder sogar größer haben, kann es schwierig sein, bekannte Elemente zu finden oder nach Informationen zu suchen, um sich einen Überblick zu verschaffen“* [Sh96]. Aufgrund dieser Argumente und der Art und Weise, wie der menschliche Verstand Informationen verarbeitet, *„ist es schneller die Bedeutung vieler Datenpunkte zu erfassen, wenn sie in Diagrammen und Grafiken angezeigt werden, als über Stapel von Tabellenkalkulationen zu blättern oder Seiten von Auswertungen zu lesen“* [SA12]. Auch bei sehr großen Datenmengen können Muster somit schnell und einfach erkannt werden.

Mit zunehmender Bildschirmauflösung sowie Verarbeitungsgeschwindigkeit der Computer Anfang der 90er Jahre erkannte Shneiderman, *„dass Informationsvisualisierung und grafische Oberflächen wahrscheinlich eine wachsende Rolle spielen werden“* [Sh96]. Damals

waren Benutzeroberflächen größtenteils textorientiert. Da Menschen über bemerkenswerte Wahrnehmungsfähigkeiten verfügten, die in dieser Zeit bei der Gestaltung der Benutzeroberflächen größtenteils ungenutzt blieben, ergaben sich bei der der Erforschung visueller Ansätze interessante neue Möglichkeiten [Sh96]. Als Softwaredesigner gerade erforschten, wie die neuen schnellen und hochauflösenden Farbdisplays optimal verwendet werden können, definierten Card et al. [CMS99] ein Referenzmodell der Informationsvisualisierung (vgl. Abb. 1), um die Diskussion über Informationsvisualisierungssysteme zu vereinfachen sowie diese Systeme vergleichen und gegenüberstellen zu können.

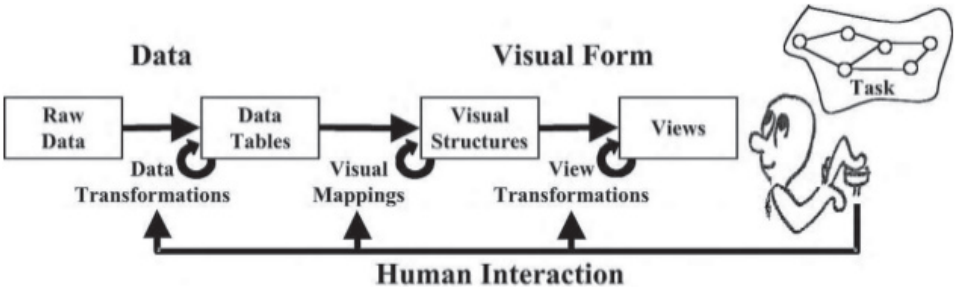


Abb. 1: Referenzmodell der Informationsvisualisierung [CMS99]

Ausgehend von einer Rohdatenquelle auf der linken Seite hin zu einer visuellen Datenrepräsentation für den Anwender auf der rechten Seite, beschreibt dieses Modell eine Reihe von Datentransformationen. Darüber hinaus beschreiben zusätzliche Pfeile ausgehend vom Anwender hin zu den einzelnen Transformationen die Anpassung dieser Transformationen durch benutzerdefinierte Interaktionen [CMS99].

Auch wenn sich dieses Referenzmodell als ein angesehener Standard etabliert hat, beschreibt es jedoch nur die Erfassung von Rohdaten aus einer einzigen Datenquelle. Weiterhin lässt es visuelle Interaktionsmöglichkeiten auf Zwischenergebnisse der einzelnen Datentransformationen zur Darstellung für den Anwender vermissen. Heutzutage ist es jedoch notwendig, mehrere verteilte, heterogene sowie interdisziplinäre domänen- oder organisationsübergreifende Datenquellen durch geeignete Informationsintegrationsansätze zu verbinden und zu integrieren, um auf diese Weise eine globale Verarbeitung in Big-Data- und Informationsvisualisierungs-Anwendungen zu gewährleisten. Darüber hinaus ist es notwendig, unterschiedlichen Anwender-Stereotypen visuellen Zugriff auf die Zwischenergebnisse der einzelnen Prozessschritte zu ermöglichen, um diese anzeigen, bearbeiten, korrigieren und kommunizieren zu können. Somit fehlen diesem Referenzmodell direkte manipulative Interaktionsmöglichkeiten zur Unterstützung der Zusammenarbeit zwischen verschiedenen Anwender-Stereotypen innerhalb der verschiedenen Prozessschritte und der Anpassung der jeweiligen Transformationen durch benutzergesteuerte Interaktionen. Um den neuen Rahmenbedingungen von Big-Data-Analyseanwendungen in der gegenwärtigen Situation gerecht zu werden und die verteilten und heterogenen Rohdaten auf den verschiedenen Prozessschritten der Informationsvisualisierung unterschiedlichen Anwender-Stereotypen zur Verfügung zu stellen, ist es notwendig, dieses Referenzmodell zu verfeinern.

3 Konzeptionelle Modellierung und Design

Basierend auf den identifizierten Herausforderungen des Referenzmodells der Informationsvisualisierung, ist im Rahmen der AVI Konferenz 2016 ein Workshop veranstaltet [Bo16b] worden, durch den die bereits erkannten Herausforderungen validiert sowie neue Herausforderungen identifiziert werden konnten; um somit ein systematisches Road-Mapping hin zu einem neuen Referenzmodell der Informationsvisualisierung für moderne Big-Data-Analyseanwendungen durchführen zu können. Ausgehend von den Ergebnissen dieses Expert-Round-Tables, bestehend aus akademischen und industriellen Forschern und Anwendern der Bereiche Big-Data, Visual-Analytics und Informationsvisualisierung, wurde das konzeptionelle IVIS4BigData -Referenzmodell (vgl. Abb. 2) abgeleitet und im Rahmen des Road-Mapping Prozesses des Workshops qualitativ bewertet [Bo16a].

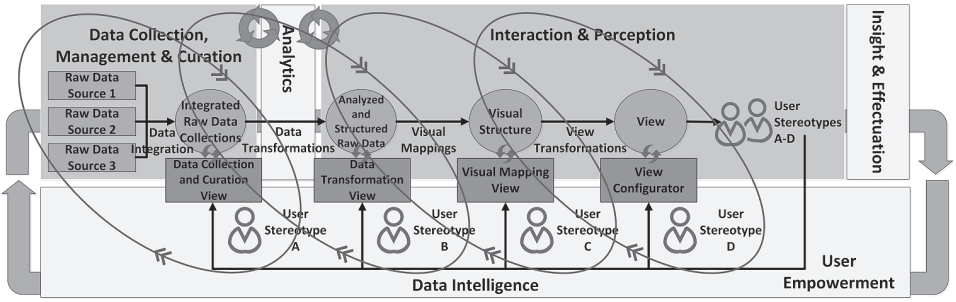


Abb. 2: IVIS4BigData Referenzmodell

Analog zum Referenzmodell der Informationsvisualisierung ist die IVIS-Pipeline des IVIS4BigData-Referenzmodells in eine Reihe von Datentransformationen unterteilt. Aufgrund der Anforderung einer direkten manipulativen Interaktionsmöglichkeit unterschiedlicher Anwender-Stereotypen innerhalb der einzelnen Prozessschritte und deren Anpassung und Konfiguration der jeweiligen Daten-Transformationen, bietet jedes Segment innerhalb der IVIS4BigData-Pipeline den unterschiedlichen Anwender-Stereotypen eine, auf dem End-User-Empowerment-Prinzip [Fi94] bestehende, graphische Schnittstelle. Dadurch ermöglicht IVIS4BigData den unterschiedlichen Anwender-Stereotypen (Domänen-Spezialisten) von Big-Data-Analyseanwendungen, Co-Designer der Anwendung zu werden, um somit eine höhere Unabhängigkeit von Computer-Spezialisten zu erhalten [Fi12; Fi94].

Aus diesem Grund ist die IVIS4BigData-Pipeline in vier aufeinanderfolgende Prozessschritte (Ellipsen) unterteilt, mit denen unterschiedliche Anwender-Stereotypen jede Daten-Transformation interaktiv spezifizieren, konfigurieren, simulieren, optimieren und ausführen können. Ausgehend von den Rohdaten auf der linken Seite werden die vier aufeinanderfolgenden IVIS4BigData-Prozessschritte - wobei jeder Prozessschritt eine bestimmte IVIS4BigData-Transformation (Daten-Integration, Daten-Transformation, Visuelle-Zuordnung und View-Transformation) repräsentiert - folgendermaßen definiert:

Advanced Visual Interfaces
Information Visualization for Big Data, Informationsvisualisierung für Big-Data

- **Data Collection, Management, and Curation:** Harmonisierung und Semantische-Integration einzelner, verteilter und heterogener Rohdatenquellen in ein einheitliches Schema (Daten-Integration aus lokalen Quellschemata in ein globales Schema).
- **Analytics:** Verteilte und Cloud-basierte Big-Data-Analyse der integrierten Rohdaten (Daten-Transformation).
- **Visualization:** Definition und Erstellung einer geeigneten visuellen Struktur auf Basis der strukturierten Daten (Visuelle-Zuordnung).
- **Perception and Effectuation:** Ermöglichung der Interaktion mit entsprechenden Ansichten der generierten visuellen Strukturen, um eine geeignete Interpretation der Big-Data-Analyseergebnisse zu ermöglichen (View-Transformation).

Während die einzelnen Prozessschritte (Ellipsen) SECI Zyklen entsprechen, verdeutlicht die Iteration um den ganzen Analyseprozess, dass es sich bei IVIS4BigData nicht nur um einen einmaligen Prozess handelt. Somit können die Ergebnisse aus vorherigen Analyseprozessen als Eingabe für eine neue Prozess-Iteration verwendet werden. Auf diese Weise ermöglicht IVIS4BigData die Kombination von Data-Mining-Ansätzen, mit deren Hilfe unbekannte Muster (ill-defined Information-Need) im ersten Schritt identifiziert werden, mit Knowledge-Management-Ansätzen. Dadurch können in weiteren Iterationen die bereits identifizierten Muster (well-defined Information-Need) automatisch erkannt werden.

Zur Modellierung des End-User-Empowerments ist es notwendig, *„den Anwendern die Informationen bereitzustellen, die als Ergebnis auf ihre - auf Teilspezifikationen und Teilkonstruktionen basierenden - Problemlösungsaktivitäten resultieren, um diese bei der Verfeinerung der Definition ihres Analyseprozesses wiederverwenden zu können“* [FN92]. Um dies zu realisieren werden die, von Fischer und Nakakoji definierten, Elemente einer facettenreichen Architektur [FN92] verwendet (vgl. Abb. 3 und 4).

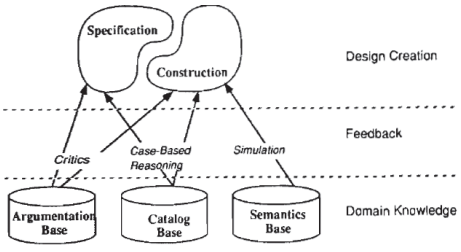


Abb. 3: Elemente einer facettenreichen Architektur [FN92]

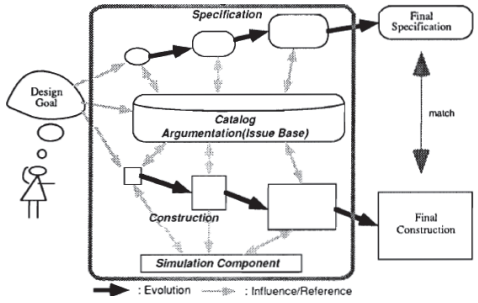


Abb. 4: Spezifikation- und Konstruktion innerhalb einer facettenreichen Architektur [FN92]

Während die linke Abbildung 3 die verschiedenen Arten der Interaktion mit Funktionspfeilen zwischen der funktionsübergreifenden Domain-Knowledge-Layer und den beiden oberen

SECI Model of Knowledge Dimensions [NT95] (SECI = Socialization, Externalization, Combination, and Internalization)

Interaktions-Layer darstellt, beschreibt die rechte Abbildung 4 den Spezifikations- und Konstruktionsprozess innerhalb der Problemlösungsaktivitäten der Anwender. *„Ausgehend von einem vagen Entwurfsziel wechseln Anwender zwischen den Komponenten in der Umgebung hin und her“* [FN92]. Indem er die verfügbaren Informationen in einer Argumentationskomponente und einem Katalog sowie das Feedback einer Simulationskomponente verwenden, entwickelt der Anwender gemeinsam mit dem System inkrementelle Spezifikation und Konstruktion, infolgedessen ein passendes Paar aus Spezifikation und Konstruktion das Ergebnis darstellt [FN92].

Mithilfe der, in diesen beiden Ansätze gezeigten, Elemente und Funktionen, ist es nun möglich, die einzelnen Prozessschritte von IVIS4BigData (Ellipsen) in ein funktionales und konzeptionelles Rahmenwerk zu überführen. Auf dessen Basis und unter Einhaltung der End-User-Empowerment-Prinzipien können somit konkrete Use-Case-Spezifikationen für die jeweiligen Aktionen der interaktiven und benutzerdefinierten Transformationsprozesse der einzelnen Prozessschritte definiert werden (vgl. Abb. 5).

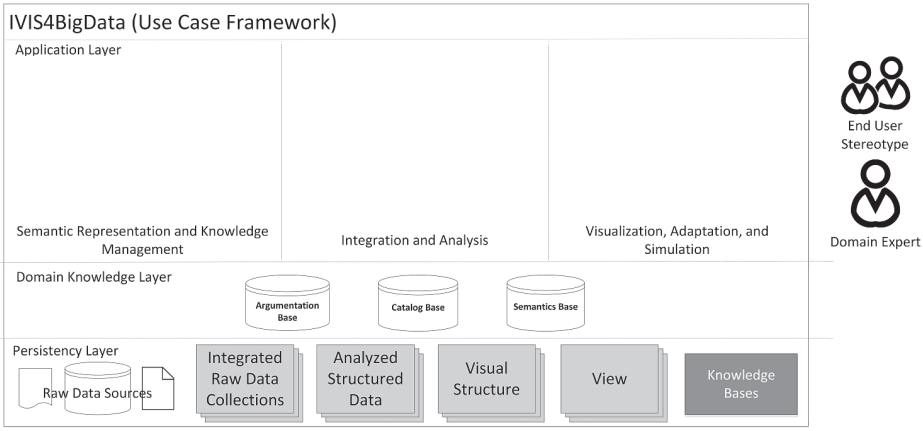


Abb. 5: IVIS4BigData Use-Case Framework

Während das mittlere Domain-Knowledge-Layer dem ursprünglichen Modell der Elemente einer facettenreichen Architektur entspricht, haben Fischer und Nakakoji eine separates Feedback-Layer definiert. Da Feedback in der heutigen Zeit ein integraler Bestandteil eines Spezifikations- und Konstruktionsprozesses ist, wird das ursprüngliche Feedback-Layer in das Application-Layer, welches wiederum dem ursprünglichen Design-Creation-Layer entspricht, innerhalb des IVIS4BigData Use-Case-Framework integriert.

Das zentrale Application-Layer, in dem sich alle Spezifikations- und Konstruktionsaktivitäten der einzelnen IVIS4BigData-Prozessschritte befinden, ist in drei Bereiche unterteilt. Während der linke Semantic Representation and Knowledge-Management Bereich alle Aktivitäten zur Spezifikation und Konstruktion der jeweiligen Daten-, Analyse- oder Visualisierungs-Prozessschritte enthält, beinhaltet der zentrale Integration and Analysis Bereich alle Aktivitäten um den spezifizierten und konstruierten IVIS4BigData-Prozesschritt

auszuführen. Weiterhin enthält dieser Bereich auch die zentrale Transformation der einzelnen IVIS4BigData-Prozessschritte (Daten-Integration, Daten-Transformation, Visuelle-Zuordnung und View-Transformation). Der rechte Visualization, Adaptation, and Simulation Bereich dieses Layers enthält alle Aktivitäten zur Unterstützung des Spezifikations- und Konstruktionsprozesses. Auf diese Weise werden in diesem zentralen Application-Layer alle Spezifikations-, Konstruktions- und Feedback-Aktivitäten von Fischers und Nakakoji's facettenreicher Architektur auf detaillierte und effektive Weise zusammengefasst.

Um die verteilte Verarbeitungsfähigkeit von IVIS4BigData abzudecken, wird außerdem ein zusätzliches Persistency-Layer unter den beiden oberen Layer eingeführt. Die Zuordnung der Knowledge-Bases innerhalb des Persistency-Layer statt innerhalb des Domain-Knowledge-Layer ergibt sich aus der Eigenschaft, dass die Knowledge-Bases sowohl Endanwendern als auch Domänenexperten bei der Nutzung domänenspezifischer Funktionalitäten entlang der verschiedenen IVIS4BigData-Prozessschritte unterstützen, jedoch auch gleichzeitig als Grundlage für die Konfiguration des gesamten IVIS4BigData-Prozesses dienen müssen. Aufgrund der Möglichkeit, verteilte Architekturen und Cloud-Dienste für die Datenspeicherung sowie für die Datenverarbeitung zu verwenden, wird durch die Zuordnung der einzelnen IVIS4BigData-Prozessschritte in diesem unteren Layer auch die Bedeutung der Verwaltung und Steuerung der Daten während und nach eines IVIS4BigData-Prozessschrittes hervorgehoben.

4 Zusammenfassung der Ergebnisse und Diskussion

Zusätzlich zu der Definition und Evaluation des theoretischen IVIS4BigData-Referenzmodells im Rahmen eines Expert-Round-Table-Workshops auf der AVI 2016 Konferenz, sowie der Design-Evaluation der entwickelten IVIS4BigData Service-orienterten Architektur inklusive der graphischen Benutzeroberflächen mithilfe einer Thinking-Aloud-Methodik, wurde die resultierende IVIS4BigData Proof-of-Concept-Referenzimplementierung mithilfe zweier konkreter Anwendungsfälle aus der Industrie aus Anwender- und Anwendungssicht evaluiert.

In Bezug auf die theoretische Evaluation im Rahmen des Expert-Round-Table-Workshops, bestehend aus initialem Fragebogen sowie einer anschließenden Diskussionsrunde, bestätigten sich die identifizierten Anpassungsnotwendigkeiten des Referenzmodells für Informationsvisualisierung, um den neuen Rahmenbedingungen von Big-Data-Analyseanwendungen in der gegenwärtigen Situation gerecht zu werden und die verteilten und heterogenen Rohdaten auf den verschiedenen Prozessschritten der Informationsvisualisierung unterschiedlichen Anwender-Stereotypen zur Verfügung zu stellen. Auch wenn einige Teilnehmer im Rahmen ihrer aktuellen Forschung nur Teile des IVIS4BigData-Referenzmodells verwenden, waren sich jedoch alle einig, dass dieses Modell einen globalen Rahmen für die IT-Infrastruktur ihrer virtuellen Forschungsumgebung sowie einen allgemeinen Rahmen für die Unterstützung verteilter Big-Data-Analyseanwendungen darstellen kann, um die aufkommende

Anomalien-Detektion für Car-to-Cloud-Datenanwendungen
Predictive-Maintenance Analyse für Roboter-Sensordaten

Wissensgenerierung in der interdisziplinären Forschungszusammenarbeit sowie domänen- oder organisationsübergreifende Business-Intelligence- und Innovations-Szenarien zu unterstützen.

Während die erste praktische Evaluation der IVIS4BigData-Referenzimplementierung im Rahmen des konkreten Anwendungsfalls Anomalien-Detektion für Car-to-Cloud-Datenanwendungen die allgemeine Anwendbarkeit belegen sollte, indem auf Basis von synthetischen Testdaten eines führenden internationalen Automobilkonzerns, verborgene, aber bekannte Anomalien erkannt werden sollten, fokussierte sich die Evaluation des zweiten Anwendungsfalls Predictive-Maintenance-Analyse für Roboter-Sensordaten auf die Identifikation sowie auf die anschließende Erkennung von bisher unbekannten Anomalien innerhalb der Daten eines führenden internationalen Anbieters für Mechatronik-Lösungen. Somit konnte nachgewiesen werden, dass IVIS4BigData die Kombination von Data-Mining-Ansätzen, mit deren Hilfe unbekannte Muster (ill-defined Information-Need) im ersten Schritt identifiziert werden, mit Knowledge-Management-Ansätzen, durch die, in weiteren Iterationen, die bereits identifizierten Muster (well-defined Information-Need) automatisch erkannt werden, ermöglicht. Zusätzlich zu dem Aspekt der technischen Funktionalität diente die zweite praktische Evaluation jedoch auch dazu, dass Anwendungsdesign zu verifizieren. Da die Service-orientierte Architektur der Referenzimplementierung auf Basis des Konzepts eines Virtual-Research-Environment definiert wurde, konnte durch die erneute Instanziierung einer weiteren Referenzapplikation auf Basis derselben darunterliegenden Service-, Ressourcen- und Infrastruktur-Layer nachgewiesen werden, dass beide Referenzapplikationen einzelnen Anwendungsfällen innerhalb des gemeinsamen Virtual-Research-Environments entsprechen.

Zusammenfassend und auf Basis der Ergebnisse der theoretischen und praktischen Evaluierungen kann daher der Schluss gezogen werden, dass IVIS4BigData die Rahmenbedingungen von Big-Data-Analyseanwendungen in der gegenwärtigen Situation erfüllt. Durch direkte manipulative Interaktionsmöglichkeiten unterschiedlicher Anwender-Stereotypen innerhalb der einzelnen Prozessschritten und deren Anpassung und Konfiguration der jeweiligen Daten-Transformationen, bietet jedes Segment innerhalb der IVIS4BigData-Pipeline den unterschiedlichen Anwender-Stereotypen eine, auf dem End-User-Empowerment-Prinzip bestehende, graphische Schnittstelle. Dadurch ermöglicht IVIS4BigData den unterschiedlichen Anwender-Stereotypen (Domänen-Spezialisten) von Big-Data-Analyseanwendungen, Co-Designer der Anwendung zu werden, um somit eine höhere Unabhängigkeit von Computer-Spezialisten zu erhalten.

Literaturverzeichnis

[Bo16a] Bornschlegl, M. X.; Engel, F. C.; Bond, R.; Hemmje, M. L.: Advanced Visual Interfaces. Supporting Big Data Applications, AVI 2016 Workshop, AVI-BDA 2016, Bari, Italy, June 7–10, 2016, Revised Selected Papers, English, 2016, XIII, 290.

[Bo16b] Bornschlegl, M. X.; Manieri, A.; Walsh, P.; Catarci, T.; Hemmje, M. L.: Road Mapping Infrastructures for Advanced Visual Interfaces Supporting Big Data Applications in Virtual Research Environments. In: Proceedings of the International Working Conference on Advanced Visual Interfaces, AVI 2016, Bari, Italy, June 7-10, 2016. S. 363–367, 2016.

[Bo19] Bornschlegl, M. X.: Advanced Visual Interfaces Supporting Distributed Cloud-Based Big Data Analysis, Diss., Hagen, Germany: University of Hagen, Faculty of Mathematics, Computer Science, Chair of Multimedia und Internet Applications, 2019.

[Ch14] Christensson, P.: User-Friendly Definition, Accessed: 2017-10-15, Jan. 2014, URL: https://techterms.com/definition/user-friendly.

[CMS99] Card, S. K.; Mackinlay, J. D.; Shneiderman, B.: Information Visualization. In (Card, S. K.; Mackinlay, J. D.; Shneiderman, B., Hrsg.): Readings in Information Visualization. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, S. 1–34, 1999, ISBN: 1-55860-533-9.

[Co09] Costabile, M. F.; Mussio, P.; Parasiliti Provenza, L.; Piccinno, A.: Supporting End Users to Be Co-designers of Their Tools. In: End-User Development. Bd. 5435, Lecture Notes in Computer Science, Springer Berlin Heidelberg, S. 70–85, 2009, ISBN: 978-3-642-00425-4.

[Fi12] Fischer, G.: Context-Aware Systems: The 'Right' Information, at the 'Right' Time, in the 'Right' Place, in the 'Right' Way, to the 'Right' Person. In: Proceedings of the International Working Conference on Advanced Visual Interfaces. AVI '12, ACM, Capri Island, Italy, S. 287–294, 2012, ISBN: 978-1-4503-1287-5.

[Fi94] Fischer, G.: In Defense of Demassification: Empowering Individuals. Human-Computer Interaction 9/1, S. 66–70, 1994.

[FN92] Fischer, G.; Nakakoji, K.: Beyond the Macho Approach of Artificial Intelligence: Empower Human Designers - Do Not Replace Them. Knowledge-Based Systems 5/1, S. 15–30, 1992, ISSN: 0950-7051.

[Fr14] Freiknecht, J.: Big Data in der Praxis. Carl Hanser Verlag GmbH & Co. KG, München, Deutschland, 2014, ISBN: 978-3-446-44177-4.

[Fr15] Fraunhofer Institute for Computer Graphics Research IGD: Visual Business Analytics, Accessed: 2015-12-02, 2015, URL: http://www.igd.fraunhofer.de/en/Institut/Abteilungen/Informationsvisualisierung-und-Visual-Analytics/Visual-Business-Analytics.

[Ku04] Kuhlen, R.: Informationsethik: Umgang mit Wissen und Information in elektronischen Räumen. UVK-Verlag-Ges., 2004, ISBN: 9783825224547.

[Ng04] Ng, A.: User Friendliness? User Empowerment? How to Make a Choice?, Techn. Ber., Graduate School of Library und Information Science, University of Illinois at Urbana-Champaign, 2004.

[NT95] Nonaka, I.; Takeuchi, H.: The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation. Oxford University Press, 1995, ISBN: 0-19-509269-4.

[Pr13] Prajapati, V.: Big Data Analytics with R and Hadoop. Packt Publishing, 2013, ISBN: 978-1-782-16328-2.

[Ro16] Rouse, M.: Context Awareness, Accessed: 2017-10-15, Mai 2016, URL: http://whatis.techtarget.com/definition/context-awareness.

[SA12] SAS Institute Inc.: Data Visualization - What it is and why it is Important, Accessed: 2015-12-21, Juli 2012, URL: http://www.sas.com/en%5C_us/insights/big-data/data-visualization.html.

[Sh96] Shneiderman, B.: The Eyes Have it: A Task by Data Type Taxonomy for Information Visualizations. In: Visual Languages, 1996. Proceedings., IEEE Symposium on. S. 336–343, Sep. 1996.

[TC+06] Thomas, J. J.; Cook, K. et al.: A Visual Analytics Agenda. Computer Graphics and Applications, IEEE 26/1, S. 10–13, Jan. 2006, ISSN: 0272-1716.

[Wo11] Wood, J.; Andersson, T.; Bachem, A.; Best, C.; Genova, F.; Lopez, D. R.; Los, W.; Marinucci, M.; Romary, L.; Van de Sompel, H.; Vigen, J.; Wittenburg, P.; Giaretta, D.; Hudson, R. L.: Riding the Wave: How Europe can Gain from the Rising Tide of Scientific Data. Final Report of the High Level Expert Group on Scientific Data; A Submission to the European Commission./, 2011.

Dr.-Ing. **Marco Xaver Bornschlegl** studierte Technische Informatik (B.Sc.) sowie Informationstechnik (M.Sc.) an der Hochschule Mannheim. Bis zum erfolgreichen Abschluss seines Studiums war er Stipendiat der Stiftung der Deutschen Wirtschaft (sdw), dass sich an herausragende, sozial engagierte Studierende und Doktoranden richtet, die Potenzial zur Übernahme von Führungsverantwortung in Wirtschaft und Gesellschaft aufweisen. Darüber hinaus erhielt er während seines Studiums zwei weitere Stipendien von führenden mittelständigen Unternehmen aus der Metropolregion Rhein-Neckar. Er schloss sein Bachelor- und Masterstudium jeweils mit Auszeichnung ab und erhielt für seine Masterthesis eine Auszeichnung von einem führenden deutschen Ingenieurverband (IfKom e.V.). Auf der SEDE-Konferenz 2017 (San Diego, CA) wurde sein Forschungsbericht mit dem Best-Paper-Award ausgezeichnet. 2018 sowie 2019 wurde er mit dem IDEAward für digitale Pioniere und Innovatoren ausgezeichnet, die sich erfolgreich den Herausforderungen der digitalen Transformation stellen und damit die Nachhaltigkeit der Deutschen und Österreichischen Wirtschaft stärken. Seine mit Summa Cum Laude abgeschlossene Promotion wurde mit dem Bernd-Pederzani-Gedächtnispreis ausgezeichnet, der sich an herausragende Promovenden der FernUniversität Hagen richtet. Seit 2016 ist er CISO und seit 2020 leitet er die Direktion Application Services & Data Science der STRABAG SE. Weiterhin ist er seit 2019 Habilitations-Kandidat an der FernUniversität Hagen.

Personenwiedererkennung mittels maschineller Lernverfahren für öffentliche Einsatzumgebungen

Markus Eisenbach[1]

Abstract: Die erscheinungsbasierte Personenwiedererkennung in öffentlichen Einsatzumgebungen ist eines der schwierigsten, noch ungelösten Probleme der Bildverarbeitung. Viele Teilprobleme können nur gelöst werden, wenn Methoden des maschinellen Lernens mit Methoden der Bildverarbeitung kombiniert werden. Das entwickelte Verfahren zur erscheinungsbasierten Personenwiedererkennung wird exemplarisch anhand zweier Einsatzszenarien — Videoüberwachung und Robotik — evaluiert. Die Qualität des umgesetzten Verfahrens wird anhand von zwölf Kriterien charakterisiert, die einen Vergleich mit biometrischen Verfahren ermöglichen. Durch den Einsatz maschineller Lernverfahren für alle Abarbeitungsschritte der erscheinungsbasierten Personenwiedererkennung wird in den betrachteten unüberwachten, öffentlichen Einsatzfeldern eine Erkennungsleistung erzielt, die sich mit biometrischen Verfahren messen kann.

1 Einleitung

Die heutige Zeit ist geprägt von einer Vielzahl technischer Neuerungen, die technischen Systemen ein immer höheres Maß an Autonomie ermöglichen. Unter anderem wird im Bereich der Pflege und Gesundheitsvorsorge verstärkt am Einsatz von Servicerobotern geforscht, welche zum Beispiel im häuslichen Umfeld bei der Betreuung von Demenzpatienten [Sc13], bei der Rehabilitation von Schlaganfallpatienten [Gr17b] oder bei der Rehabilitation von Patienten nach Hüftoperationen [Sc19] eingesetzt werden.

Um den Kontakt zwischen Mensch und Roboter aufrecht zu erhalten muss eine Wiedererkennung des Nutzers zu jeder Zeit möglich sein. Bei der Wiedererkennung des Nutzers im Gesundheits- und Pflegebereich gibt es jedoch einige Einschränkungen. Es ist zu beachten, dass es sich bei diesen Anwendungsbereichen um unkontrollierte Szenarien handelt. Dies bedeutet, dass keine ständige Kooperation der Personen erwartet werden kann, um zum Beispiel eine Erkennung anhand biometrischer Merkmale, wie Fingerabdruck oder Gesicht, zu ermöglichen. In diesen Anwendungsbereichen müssen technische Systeme daher auf die gesamte visuelle Erscheinung der Personen achten, inklusive deren Kleidung. [Go14b][2]

Dies entspricht auch dem menschlichen Vorgehen, was anhand von Abb. 1 ersichtlich wird. Die Unterscheidung der drei Kinder wird deutlich erleichtert, wenn zusätzlich zum Gesicht auch die Kleidung betrachtet wird. Das gesamte Erscheinungsbild spielt also bei der menschlichen Wahrnehmung eine deutlich größere Rolle als einzelne biometrische Merkmale, wie beispielsweise das Gesicht.

In der Literatur ist diese erscheinungsbasierte Wiedererkennung von Personen durch ein technisches System wie folgt definiert:

[1] Fachgebiet Neuroinformatik und Kognitive Robotik, TU Ilmenau, markus.eisenbach@tu-ilmenau.de
[2] [Go14b], Preface, S. v

CC BY SA

Abb. 1: Zuordnung von Personen anhand des Gesichts und der Kleidung. Bildquelle: [GC08]

„Personenwiedererkennung ist das Problem der Erkennung und Zuordnung einer Person an verschiedenen physischen Orten über die Zeit, nachdem die Person zuvor irgendwo anders visuell beobachtet wurde." [Go14b][3] Die „Hauptaufgabe der Personenwiedererkennung ist das Messen der Ähnlichkeit zwischen zwei personenzentrierten Bildregionen, um Vorhersagen zu ermöglichen, ob diese Regionen die gleiche Person darstellen trotz Veränderungen der Beleuchtung, des Blickwinkels, störender Hintergründe, Verdeckungen sowie unterschiedlicher Bildqualität und Auflösung." [MSJ14][4]

Wie aus dem letzten Teil der Definition hervorgeht, soll die erscheinungsbasierte Wiedererkennung auch bei unterschiedlichen Umwelteinflüssen funktionieren, die jedoch oft nicht bekannt sind. Das heißt, die Wiedererkennung einer Person muss auch möglich sein

- wenn die Person nicht kooperiert und biometrische Merkmale der Person nicht erkennbar sind (unüberwachtes Szenario),
- wenn die Person teilweise oder zeitweise vollständig verdeckt ist (Verdeckung),
- wenn die Person aus verschiedenen Perspektiven beobachtet wird, das heißt zum Beispiel erst frontal, dann von hinten (Blickwinkel),
- wenn die Person in verschiedenen Posen beobachtet wird, zum Beispiel erst sitzend, dann stehend (Pose),
- wenn die Person weit von der Kamera entfernt ist und der Bildausschnitt daher nur niedrig aufgelöst ist (Auflösung),
- wenn die Bilder der Person unscharf sind, zum Beispiel durch die Bewegung der Kamera, die an einem mobilen Roboter angebracht ist (Bildqualität),
- wenn die Kleidung der Person durch wechselnde Beleuchtungsbedingungen oder Schattenwürfe unterschiedlich erscheint (Umwelteinflüsse) und
- wenn mehrere Personen ähnliche Kleidung tragen (Varianz).

Aufgrund dieser großen Herausforderung gilt die Personenwiedererkennung als eines der schwierigsten, noch ungelösten Probleme der Bildverarbeitung. Viele Teilprobleme können

[3] [Go14b], Preface, S. v: engl. *„Person re-identification is the problem of recognising and associating a person at different physical locations over time after the person had been previously observed visually elsewhere."*

[4] [MSJ14], S. 23: engl. *„[…] one key issue of person re-identification is […] to measure the similarity between two person-centered image regions, allowing to predict if these represent the same person despite changes in illumination, viewpoints, background clutter, occlusion, and image quality/resolution."*

nur gelöst werden, wenn Methoden des maschinellen Lernens mit Methoden der Bildverarbeitung kombiniert werden. Dies wird auch deutlich anhand der relativ großen Anzahl an Publikationen in den letzten zehn Jahren zu diesem Thema in allen relevanten größeren Konferenzen und Journalen der beiden Forschungsbereiche. [Go14b]

2 Anwendungen

Das Lösen des Wiedererkennungsproblems stellt allein schon eine große Herausforderung dar. Zusätzlich bietet es laut [Go14a] auch noch ein enormes Potential für eine große Bandbreite praktischer Anwendungen. Im Rahmen dieser Arbeit werden exemplarisch zwei Anwendungsgebiete adressiert. Zum einen die Suche nach Personen in einem Multikamerasystem an öffentlichen Plätzen, zum anderen die Betreuung von Schlaganfallpatienten durch einen Serviceroboter. Beide Szenarien haben gemeinsam, dass Personen in Echtzeit visuell erfasst und durch Merkmale charakterisiert werden müssen, damit später eine Wiedererkennung ohne größeren Zeitaufwand stattfinden kann. Nachfolgend werden die beiden Anwendungen genauer beschrieben.

2.1 Videoüberwachung

Die Videoüberwachung von Flughäfen und Fluglandeplätzen wurde im Rahmen des Forschungsprojekts APFel[5] betrachtet. Das Ziel dieses Projekts war die Unterstützung eines Operateurs bei der kameraübergreifenden Verfolgung ausgewählter Personen. Ein unterstützendes Analysetool (Abb. 2) kann dabei helfen, eine ausgewählte Person im Blick zu behalten. Dabei sind Zeitpunkte, an denen die Zielperson vom System wiedererkannt wurde, grün hervorgehoben.

Viele Überwachungskameras sind so ausgerichtet, dass sie möglichst große Bereiche erfassen. Dadurch sind Details, wie das Gesicht, nicht erkennbar und können bei diesen Übersichtskameras auch nicht für die Erkennung von Personen verwendet werden. Die erscheinungsbasierte Wiedererkennung ist hierbei die entscheidende Komponente, um diesen Vorgang teilautomatisiert umzusetzen. Eine besondere Schwierigkeit stellen dabei Unterschiede in Beleuchtung und Blickwinkel dar.

Abb. 2: Anwendung der Wiedererkennung im Bereich Videoüberwachung.

[5] APFel: Analyse von Personenbewegungen an Flughäfen mittels zeitlich rückwärts- und vorwärtsgerichteter Videodatenströme. Forschungsprojekt gefördert vom Bundesministerium für Bildung und Forschung unter dem Förderkennzeichen 13N10797. Laufzeit: 01.01.2010 – 31.03.2014

Übertragbarkeit des Szenarios: Anhand dieses Forschungsprojekts wurde stellvertretend die Anwendung der Wiedererkennung zur Auswertung von Videos in einem Multikamerasystem untersucht. Daraus ergeben sich praktische Anwendungsmöglichkeiten für

- die Überwachung von kritischen Infrastrukturen, wie zum Beispiel Flughäfen, Bahnhöfen, U-Bahn-Stationen, Einkaufszentren, aber auch öffentlichen Plätzen,
- die Auswertung forensischer Daten,
- das Tracking von Personen über mehrere Räume in Smart-Home-Anwendungen und
- die Erfassung kundenspezifischer Informationen im Einzelhandel.

2.2 Nutzererkennung durch einen Serviceroboter

Im Rahmen des Forschungsprojekts ROREAS[6] wurde ein Serviceroboter zur Begleitung und Unterstützung von Schlaganfallpatienten während ihrer Rehabilitation entwickelt. Damit der Roboter seinem aktuellen Nutzer folgen kann, muss er ihn von anderen Personen unterscheiden können. In eindeutigen Fällen kann der Nutzer dazu multimodal getrackt werden. Die erscheinungsbasierte Wiedererkennung ist wichtig, um den Patienten in Fluren mit hohem Personenaufkommen folgen zu können, auch wenn diese zwischenzeitlich vollständig verdeckt und somit für den Roboter nicht sichtbar waren (Abb. 3). Die größten Schwierigkeiten stellen dabei wechselnde Beleuchtungsbedingungen und teilweise Verdeckungen dar.

Abb. 3: Anwendung der Wiedererkennung im Bereich Servicerobotik.

Übertragbarkeit des Szenarios: Anhand dieses Forschungsprojekts wurde stellvertretend die Anwendung der Wiedererkennung in dem neuen Anwendungsfeld Servicerobotik untersucht. Dieses wurde bisher in der Literatur kaum betrachtet. Praktische robotische Anwendungsmöglichkeiten ergeben sich für

- die Begleitung von Personen im medizinischen Kontext,
- das Lotsen von Personen in öffentlichen Einsatzumgebungen, wie zum Beispiel Einkaufszentren, Museen, Flughäfen oder Bürogebäuden und
- die Unterscheidung von Personen in Einsatzumgebungen mit eingeschränktem Nutzerkreis, wie zum Beispiel Seniorenresidenzen oder Smart-Home-Umgebungen.

[6] ROREAS: Interaktiver Robotischer Reha-Assistent für das Lauf- und Orientierungstraining von Patienten nach Schlaganfällen. Forschungsprojekt gefördert vom Bundesministerium für Bildung und Forschung unter dem Förderkennzeichen 16SV6133. Laufzeit: 01.07.2013 – 31.03.2016

3 Zielstellung

Diese beiden Anwendungsszenarien — Videoüberwachung und Servicerobotik — haben gemeinsam, dass eine Wiedererkennung unter unkontrollierten Bedingungen (keine Kooperation der Personen) unter Einfluss verschiedener Umwelteinflüsse robust funktionieren muss. Hieraus leiten sich die Anforderungen an das Wiedererkennungssystem ab, das im Rahmen dieser Arbeit entwickelt wurde:

- **Verwendung erscheinungsbasierter Merkmale**, die geeignet sind, um Personen anhand ihrer Kleidung wiederzuerkennen.
- **Echtzeitfähige Verarbeitung** aller Wiedererkennungsschritte bei gleichzeitiger Verwendung möglichst weniger Rechenressourcen, vor allem im Falle der Servicerobotik.
- **Bevorzugte Verwendung maschineller Lernverfahren** für alle Teilkomponenten der Wiedererkennung, um möglichst große Flexibilität und Praktikabilität zu erreichen.

Die Dissertation hatte entsprechend das Ziel des Entwurfs einer echtzeitfähigen erscheinungsbasierten Personenwiedererkennung, die flexibel für verschiedene Anwendungen einsetzbar ist.

4 Systemüberblick

Der generelle Ablauf der Wiedererkennung ist an die Verarbeitungskette in einem biometrischen System angelehnt (siehe Abb. 4). Zuerst werden zu erkennende Personen sensorisch über eine oder mehrere Kameras erfasst. Durch Vorverarbeitungsschritte lassen sich aus dem Ausgangsbild personenzentrierte Bildausschnitte extrahieren. Für die Merkmalsextraktion wird ein Neuronales Netzwerk derart trainiert, dass aus den Bildausschnitten diskriminative Merkmalsvektoren berechnet werden können, die ein schnelles Matching ermöglichen. Das generierte Template beschreibt die Zielperson anhand mehrerer Merkmalsvektoren. Es wird während des Matchings mit Merkmalsvektoren der zur Anwendungsphase beobachteten Personen verglichen. Eine gelernte Metrik kompensiert dabei szenariospezifische Umwelteinflüsse. Durch eine geeignete Fusion komplementärer

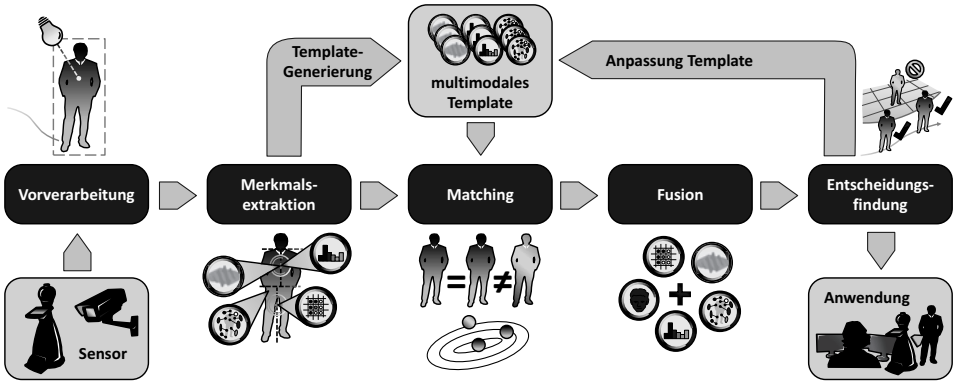


Abb. 4: Verarbeitungskette für echtzeitfähige Personenwiedererkennung.

Merkmale kann die Wiedererkennungsleistung gesteigert werden. Bei der Entscheidungsfindung müssen alle berechneten Matching Scores verrechnet werden. Eine Suchraumeinschränkung kann dabei die Verarbeitungsgeschwindigkeit und die Erkennungsleistung steigern.

Die eigentlichen Wiedererkennungsschritte — Merkmalsextraktion, Template-Generierung, Matching, Fusion und Entscheidungsfindung — finden sich auch in einem biometrischen System wieder. Durch den modularen Aufbau ist der Austausch von biometrischen Merkmalen gegen erscheinungsbasierte Merkmale ohne Probleme möglich. Außer für die Merkmalsextraktion können für alle Wiedererkennungsschritte Komponenten verwendet werden, die sowohl für die Verarbeitung biometrischer als auch erscheinungsbasierter Merkmale geeignet sind. Insbesondere durch Verfahren des maschinellen Lernens ist dies problemlos möglich.

5 Erreichte Verbesserungen gegenüber dem State of the Art

Schwerpunktmäßig wurde in dieser Arbeit untersucht, für welche Teilaspekte der Personenwiedererkennung Verfahren des maschinellen Lernens eingesetzt werden können und wie diese zu gestalten sind. Es wurden Fortschritte gegenüber dem State of the Art zu allen Teilaspekten des Wiedererkennungssystems erreicht:

- **Vorverarbeitung**: Durch den Einsatz von Convolutional Neural Networks konnte die Personendetektion deutlich verbessert werden [Ei16]. Der entwickelte Ansatz erzielt deutlich bessere Generalisierungseigenschaften in unbekannten Einsatzgebieten als der State of the Art zur Detektion mittels Neuronaler Netzwerke. Dies erhöht die Flexibilität bezüglich der Anwendungsszenarien. Eine Laufzeitoptimierung des Convolutional Neural Networks ermöglicht die Anwendung auf einer Jetson TX1 [Ei17] und somit den Einsatz auf einem Roboter.

  Um Beleuchtungseinflüsse auf die Kleidungsfarbe wiederzuerkennender Personen zu korrigieren, wurde datengetrieben eine Beleuchtungskarte gelernt und bei verändernden Bedingungen adaptiert [Ei13]. Der vorgestellte Ansatz macht im Gegensatz zu den meisten State-of-the-Art-Verfahren zur Farbkonstanz keine Annahmen über die Einsatzumgebung und ist daher für ein reales Einsatzfeld besser geeignet.
- **Merkmalsextraktion**: Es wurde gezeigt, dass geeignete Merkmale für die Wiedererkennung rein datengetrieben gelernt werden können. Die Genauigkeit bei der Schätzung der mittels Convolutional Neural Networks gelernten semantischen Attribute übertraf für die meisten Attribute den State of the Art [Ei19].

  Durch den Einsatz moderner Fehlerfunktionen konnten mit tiefen Neuronalen Netzwerken Merkmalsvektoren gelernt werden, die eine sehr gute Unterscheidung von Personen ermöglichen [Ei19]. Aktuelle Softmax-Loss-Weiterentwicklungen wurden dabei erstmals im Kontext der erscheinungsbasierten Wiedererkennung verwendet. Die erzielten Wiedererkennungsraten liegen auf menschlichem Niveau, trotz einer im Vergleich zum State of the Art eher einfachen Netzwerkarchitektur.
- **Template-Generierung**: Es wurde ein Verfahren entwickelt, das zur Laufzeit für eine ausgewählte Person geeignete Merkmale auswählt, um diese Person von anderen zu unterscheiden [Ei12]. Dadurch kann das Template kompakt gehalten werden

und die Laufzeit sowie die Wiedererkennungsleistung verbessert werden. Einfache, personenspezifisch ausgewählte Merkmale waren komplexeren Merkmalen deutlich überlegen.

- **Matching**: Zur Verbesserung des Metric Learning wurde eine Vorverarbeitung der Trainingsdaten vorgestellt [Ei15b]. Außerdem wurde ein *Re-Ranking*-Verfahren entwickelt, das die Sortierung basierend auf Score-Werten auch ohne menschliche Rückmeldung verbessert [Ei19], was auch eine deutliche Verbesserung für die nachfolgende Fusion von Merkmalen bringt.
- **Fusion**: Die Fusion auf Score Level wurde erstmalig im Kontext der erscheinungsbasierten Wiedererkennung von Personen evaluiert [Ei15a]. Hierzu wurden zahlreiche State-of-the-Art-Verfahren verglichen und ein eigener Ansatz eingebracht, der das Fusionsergebnis deutlich verbessern kann. Außerdem erfolgte erstmalig eine Kombination von Score-Level-Fusion und Metric Learning, die eine (teilweise deutliche) Verbesserung der State-of-the-Art-Erkennungsraten erzielt.
- **Entscheidungsfindung**: Zur Entscheidung, welche Person der aktuelle Nutzer eines Roboters ist, wurde ein neuartiger Ansatz vorgestellt. Dieser berücksichtigt für mehrere Beobachtungen nicht nur die reinen Matching Scores, sondern rechnet diese geeignet in Wahrscheinlichkeiten um, bezieht zusätzlich Ranking-Informationen ein und entscheidet dann anhand eines probabilistischen Mehrheitsvotums [Ei15b]. Dieses konsistente probabilistische Framework stellt einen deutlichen Vorteil gegenüber den in der Literatur verwendeten einfachen Heuristiken dar.

  Des Weiteren wurde die Entscheidungsfindung durch eine enge Kopplung mit dem Personentracking [We16] verbessert. Das Tracking zuvor identifizierter Personen ermöglichte eine spatio-temporale Suchraumeinschränkung für die Wiedererkennung. Dadurch konnte die Sicherheit bei der Wiedererkennung erhöht werden, wodurch das Template häufiger adaptiert werden konnte. Dies führte durch die größere Varianz der Ansichten wiederum zu besseren Wiedererkennungsraten.
- **Einbindung in Anwendung**: Es wurde erstmalig ein Wiedererkennungssystem realisiert, dass in Echtzeit auf einem mobilen Roboter den aktuellen Nutzer anhand dessen Kleidung erkennt [Ei15b].

  Bezüglich der Videoüberwachung konnte gezeigt werden, dass die erscheinungsbasierte Wiedererkennung eine kameraübergreifende Verfolgung von Personen ermöglicht und dazu führt, dass ein Operateur eine Situation deutlich schneller einschätzen kann [Ko13].

6 Bewertung und Fazit

Die Wiedererkennungsleistung kann auf Benchmarkdatensätzen anhand von erstellten Rankings bewertet werden, zum Beispiel anhand von CMC- oder SRR-Kurven. Diese häufig im State-of-the-Art anzutreffende Bewertung ist jedoch sehr einseitig, da sie zwar die Erkennungsleistung bewertet, jedoch die Praktikabilität des Ansatzes und die Skalierbarkeit außer Acht lässt. Im Rahmen dieser Arbeit wurde die Wiedererkennungsleistung daher anhand von zwölf Gütekriterien bewertet, die auch für biometrische Merkmale angesetzt werden.

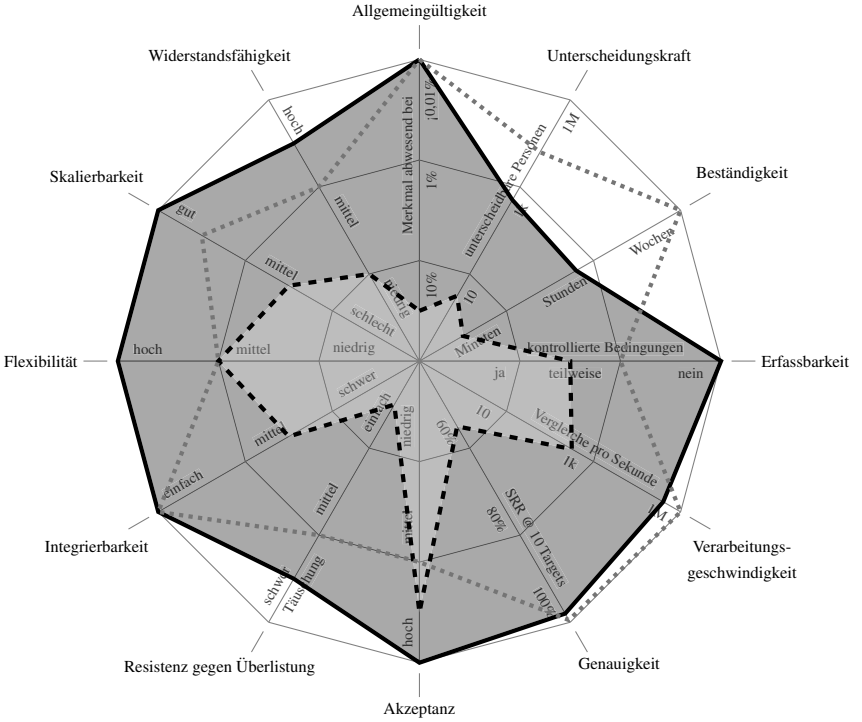


Abb. 5: Gütekriterien zur Beurteilung des Wiedererkennungssystems.

Neben den zwei Kriterien „Unterscheidungskraft“ und „Genauigkeit“, die sich auf die oben genannten Benchmarkergebnisse beziehen, wird die Wiedererkennungsleistung bezüglich zehn weiterer Kriterien bewertet (siehe Abb. 5). Die gestrichelte Linie zeigt das Ergebnis bei einer schlechten Wahl von Einzelkomponenten, die in der Literatur verwendet werden. Durch die im Rahmen dieser Arbeit entwickelten Verfahren konnten alle Gütekriterien deutlich verbessert werden (durchgezogene Linie). Tab. 1 zeigt, welche Teile des Wiedererkennungssystems einen Einfluss auf die einzelnen Gütekriterien ausüben.

Der exemplarische Vergleich mit einer Gesichtserkennung in den betrachteten Szenarien (gepunktete, grüne Linie in Abb. 5) zeigt, dass die erreichte Wiedererkennungsleistung für die meisten Kriterien auf dem Niveau biometrischer Merkmale liegt. Die in dieser Arbeit gelernten erscheinungsbasierten Merkmale erreichen bezüglich der Unterscheidungskraft das menschliche Niveau. Es können etwa 660 Personen eindeutig anhand ihrer Kleidung unterschieden werden. Damit wird jedoch nicht das Niveau einer Gesichtserkennung erreicht, durch die mehr als 10000 Personen eindeutig unterschieden werden können. Die Beständigkeit der erscheinungsbasierten Merkmale konnte durch mehrere Mechanismen auf die Aufenthaltsdauer der Person im Einsatzfeld gesteigert werden. Auch bewusste Kleidungswechsel werden durch ein Tracking und ein Template-Update geeignet behandelt. Die Beständigkeit biometrischer Merkmale über mehrere Jahre hinweg kann jedoch mit einer erscheinungsbasierten Wiedererkennung nicht erreicht werden.

	Vorverarbeitung	Merkmalsextraktion	Template-Generierung	Matching	Fusion	Entscheidungsfindung	Einbindung in Anwendung
Allgemeingültigkeit		✓					
Unterscheidungskraft		✓		✓	✓		
Beständigkeit	✓	✓	✓	✓			
Erfassbarkeit	✓	✓			✓		
Verarbeitungsgeschwindigkeit		✓	✓	✓		✓	✓
Genauigkeit		✓	✓	✓	✓	✓	
Akzeptanz	✓	✓					✓
Resistenz gegen Überlistung	✓		✓		✓	✓	✓
Integrierbarkeit					✓		✓
Flexibilität							✓
Skalierbarkeit			✓	✓	✓	✓	✓
Widerstandsfähigkeit					✓		✓

Tab. 1: Verbesserung der Wiedererkennung durch in dieser Arbeit beschriebene Verfahren.

Literaturverzeichnis

[Ei12] Eisenbach, M.; Kolarow, A.; Schenk, K.; Debes, K.; Groß, H.-M.: View Invariant Appearance-based Person Reidentification Using Fast Online Feature Selection and Score Level Fusion. In: IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance (AVSS). IEEE, S. 184–190, 2012.

[Ei13] Eisenbach, M.; Scheiner, P.; Kolarow, A.; Schenk, K.; Groß, H.-M.; Weinreich, I.: Learning Illumination Maps for Color Constancy in Person Reidentification. In: Workshop Farbbildverarbeitung (FWS). GFaI, S. 103–114, 2013.

[Ei15a] Eisenbach, M.; Kolarow, A.; Vorndran, A.; Niebling, J.; Groß, H.-M.: Evaluation of Multi Feature Fusion at Score-Level for Appearance-based Person Re-Identification. In: IEEE Int. Joint Conf. on Neural Networks (IJCNN). IEEE, S. 469–476, 2015.

[Ei15b] Eisenbach, M.; Vorndran, A.; Sorge, S.; Groß, H.-M.: User Recognition for Guiding and Following People with a Mobile Robot in a Clinical Environment. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS). IEEE, S. 3600–3607, 2015.

[Ei16] Eisenbach, M.; Seichter, D.; Wengefeld, T.; Groß, H.-M.: Cooperative Multi-Scale Convolutional Neural Networks for Person Detection. In: World Congress on Computational Intelligence (WCCI). IEEE, S. 267–276, 2016.

[Ei17] Eisenbach, M.; Stricker, R.; Seichter, D.; Vorndran, A.; Wengefeld, T.; Groß, H.-M.: Speeding up Deep Neural Networks on the Jetson TX1. In: IJCNN-Workshop on Computational Aspects of Pattern Recognition and Computer Vision with Neural Systems (CAPRI). Springer, S. 11–22, 2017.

[Ei19] Eisenbach, M.: Personenwiedererkennung mittels maschineller Lernverfahren für öffentliche Einsatzumgebungen. Dissertation, TU Ilmenau, 2019.

[GC08] Gallagher, A.C.; Chen, T.: Clothing cosegmentation for recognizing people. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). S. 1–8, 2008.

[Go14a] Gong, S.; Cristani, M.; Loy, C.C.; Hospedales, T.M.: The Re-Identification Challenge. Springer, Kapitel 1, S. 1–20, 2014. Person Re-Identification.

[Go14b] Gong, S.; Cristani, M.; Yan, S.; Loy, C.C.: Person Re-Identification. Springer, 2014.

[Gr17a] Groß, H.-M.; Meyer, S.; Stricker, R.; Scheidig, A.; Eisenbach, M.; Müller, S.; Trinh, T.Q.; Wengefeld, T.; Bley, A.; Martin, C.; Fricke, C.: Mobile Robot Companion for Walking Training of Stroke Patients in Clinical Post-stroke Rehabilitation. In: IEEE Int. Conf. on Robotics and Automation (ICRA). IEEE, S. 1028–1035, 2017.

[Gr17b] Groß, H.-M.; Scheidig, A.; Debes, K.; Einhorn, E.; Eisenbach, M.; Müller, S.; Schmiedel, T.; Trinh, T.Q.; Weinrich, C.; Wengefeld, T.; Bley, A.; Martin, C.: ROREAS: Robot Coach for Walking and Orientation Training in Clinical Post-Stroke Rehabilitation. Prototype Implementation and Evaluation in Field Trials. Autonomous Robots (AR), 41(3):679–698, 2017.

[Ko13] Kolarow, A.; Schenk, K.; Eisenbach, M.; Dose, M.; Brauckmann, M.; Debes, K.; Groß, H.-M.: APFel: The Intelligent Video Analysis and Surveillance System for Assisting Human Operators. In: IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance (AVSS). IEEE, S. 195–201, 2013.

[MSJ14] Ma, B.; Su, Y.; Jurie, F.: Discriminative Image Descriptors for Person Re-Identification. Springer, Kapitel 2, S. 23–42, 2014. Person Re-Identification.

[Sc13] Schröter, C.; Müller, S.; Volkhardt, M.; Einhorn, E.; Huijnen, C.; van den Heuvel, H.; van Berlo, A.; Bley, A.; Groß, H.-M.: Realization and User Evaluation of a Companion Robot for People with Mild Cognitive Impairments. In: IEEE Int. Conf. on Robotics and Automation (ICRA). IEEE, S. 1145–1151, 2013.

[Sc19] Scheidig, A.; Jäschke, B.; Schütz, B.; Trinh, T.Q.; Vorndran, A.; Mayfarth, A.; Groß, H.-M.: May I Keep an Eye on Your Training? Gait Assessment Assisted by a Mobile Robot. In: Int. Conf. on Rehabilitation Robotics (ICORR). IEEE/RAS-EMBS, 2019.

[SSE19] Sesselmann, M.; Stricker, R.; Eisenbach, M.: Einsatz von Deep Learning zur automatischen Detektion und Klassifikation von Fahrbahnschäden aus mobilen LiDAR Daten. AGIT – Jour. für Angewandte Geoinformatik, 2019.

[We16] Wengefeld, T.; Eisenbach, M.; Trinh, T.Q.; Groß, H.-M.: May I be your Personal Coach? Bringing Together Person Tracking and Visual Re-identification on a Mobile Robot. In: Int. Symposium on Robotics (ISR). VDE, S. 141–148, 2016.

Markus Eisenbach schloss 2009 sein Studium an der TU Ilmenau als Diplom-Informatiker ab. Anschließend arbeitete er als wissenschaftlicher Mitarbeiter am Fachgebiet Neuroinformatik und Kognitive Robotik der TU Ilmenau. Seine Promotion zum Doktor-Ingenieur schloss er 2019 mit dem Prädikat „Summa cum laude“ ab. Seit 2019 leitet er die Nachwuchsforschergruppe „Maschinelles Lernen“. Seine Forschungsschwerpunkte umfassen maschinelles Lernen, Deep Learning, Robotik sowie Automatisierung von Bild- und Signalverarbeitung in industriellen Anwendungen. Auszeichnungen: 2014 wurde er für besondere Leistungen im Ehrenamt ausgezeichnet. Für seine Lehrtätigkeiten an der TU Ilmenau bekam er mehrere Preise: Die Lehrveranstaltung „Angewandte Neuroinformatik“ wurde 2016, 2017 und 2019 auf Grundlage von Studierendenbewertungen als bestes Seminar der Fakultät für Informatik und Automatisierung ausgezeichnet. 2019 bekam er den Lehrpreis der TU Ilmenau verliehen. Folgende Publikationen wurden ausgezeichnet: Auf der International Conference on Robotics and Automation (ICRA) war [Gr17a] Finalist des Best Paper Awards on Human-Robot Interaction (HRI). Der Beitrag [SSE19] wurde mit dem Best Student Paper Award ausgezeichnet.

Entwicklung einer Auslesekarte zur schnellen Verarbeitung von Detektordaten

ALICE RUN2 Auslese-Upgrade und Evaluierung von Datenfluss-Hardwarebeschreibung für Ausleseanwendungen der Hochenergiephysik[1]

Heiko Engel[2]

Abstract: Im Rahmen dieser Arbeit wurde eine FPGA-basierte Ausleselösung für die ALICE Data Acquisition und High Level Trigger Systeme in RUN 2 entwickelt, die durch ihre generische Architektur weit über die geplanten Anwendungsfälle und Zeiträume hinaus zum Einsatz kommt. Ein zentraler Bestandteil ist hierbei die hardwareseitige Vorverarbeitung von Detektordaten bereits im FPGA der Auslesekarte, ohne die eine Online-Aufbereitung der Daten in diesem Maße nicht möglich gewesen wäre. Durch die Umsetzung des verwendeten Vorverarbeitungsalgorithmus in einer Datenflussbeschreibung konnte darüber hinaus eine Implementierung erarbeitet werden, bei der sich Ressourcenverbrauch und Latenz kaum von der händischen Implementierung unterscheiden. Die deutliche reduzierte Entwicklungszeit, der geringere Umfang, die leichtere Wartbarkeit, sowie die einfachere Partitionierung konnten somit zeigen, dass Hardwarebeschreibungslösungen auf höheren Abstraktionsebenen auch in der Hochenergiephysik enorme Einsparungen bringen können.

1 Einleitung

Die Grundlagenforschung versucht mit einer Vielzahl von Experimenten zu unterschiedlichsten physikalischen Aspekten ein tieferes Verständnis unserer Welt zu fördern. Die Teilchenkollisionsexperimente der Hochenergiephysik tragen hierbei zum zentralen Verständnis der Struktur der Materie bei. All diese Experimente sind hoch-komplexe Systeme mit einer Vielzahl von Komponenten, Detektoren und Technologien. Sie nutzen unterschiedliche physikalische Eigenschaften der Kollisionsprodukte aus, um diese zu detektieren, deren Spuren, Energien oder Impulse zu messen und daraus die physikalischen Stoßprozesse zu rekonstruieren. Eine gemeinsame Herausforderung dieser Experimente ist die Koordination, Synchronisation und Auslese dieser massiv-parallelen Systeme bei Ereignisraten von mehreren Kilohertz. Eine Technologie, die bei vielen Ausleseketten an verschiedenen Stellen eingesetzt wird, sind programmierbare Hardware-Bausteine in der Form von "Field Programmable Gate Arrays" (FPGAs). Mit FPGAs kann benutzerdefinierte Logik umgesetzt werden, die für kleine bis mittlere Stückzahlen deutlich günstiger als ASICs ist und jederzeit an geänderte Bedingungen angepasst werden kann. Außerdem können FPGAs durch ihre fast beliebige Parallelität eine Rechenleitung bieten, die in vielen Anwendungsfällen auch von CPU, GPU oder DSP Systemen nicht erreichbar

[1] Englischer Titel der Dissertation: Development of a Read-Out Receiver Card for Fast Processing of Detector Data – ALICE HLT Run 2 Readout Upgrade and Evaluation of Dataflow Hardware Description for High Energy Physics Readout Applications [En19]

[2] Infrastruktur und Rechnersysteme in der Informationsverarbeitung, Goethe-Universität Frankfurt a. M.

CC BY SA

ist. FPGAs haben in den letzten Jahren ihren festen Platz in den Ausleseketten von Physikexperimenten gefunden. Mit immer größeren und schnelleren Chips werden außerdem immer komplexere Verarbeitungsschritte in Hardware möglich.

Der Large Hadron Collider (LHC, [EB08]) an der Europäischen Organisation für Kernforschung (CERN) ist derzeit der leistungsstärkste Teilchenbeschleuniger der Welt. Dutzende verschiedene Experimenten sind daran angebunden. ALICE [Aa08] ist eines der vier größten Experimente am LHC und auf die Erforschung von Schwerionenkollisionen sowie des dabei entstehenden Quark-Gluon-Plasma spezialisiert. ALICE besteht aus 19 Sub-Detektoren, die mit Hilfe eines komplexen Synchronisations-, Trigger- und Auslesesystems zeitgleich betrieben werden. Über 500 serielle optische Links stellen den Datentransport zwischen den Detektoren und dem Data Acquisition (DAQ) System sicher. Parallel wird eine Kopie der Detektordaten an den High Level Trigger (HLT, [Ri07]) geschickt, der die Daten rekonstruiert, kalibriert und komprimiert. Durch die Kompression und Speicherung vorverarbeiteter Daten können die ursprünglichen Rohdaten verworfen und damit die Datenmenge zur dauerhaften Speicherung signifikant reduziert werden. Nur diese Kompression macht es möglich, die bei den Schwerionenkollisionen in ALICE anfallenden Datenmengen überhaupt speichern zu können.

Eine zentrale Komponente dieser Auslesearchitektur ist die Schnittstelle zwischen den optischen Detektorlinks und den Rechnerclustern der DAQ und HLT Systeme. In der ersten LHC Runperiode (RUN 1) zwischen 2009 und 2012 wurden hierfür speziell entwickelte FPGA-basierte Auslesekarten eingesetzt, die Read-Out Receiver Cards (RORCs). In der HLT-Anwendung war neben der reinen Datenauslese zudem eine Datenvorverarbeitung in Form eines Hardware Cluster Finder Algorithmus integriert. Diese hoch-parallele FPGA-basierte Vorverarbeitung ersparte beträchtliche Mengen an CPU Rechenleistung im Vergleich zu äquivalenten Verarbeitungsschritten auf CPUs in Software.

2 Motivation und Ziele der Arbeit

Nach dem Ende der ersten LHC Runperiode wurden mehrere ALICE Detektorsysteme weiter ausgebaut und für höhere Daten- und Ereignisraten in der zweiten LHC Runperiode (RUN 2) ab 2015 vorbereitet. Die DAQ und HLT Systeme wurden zum größten Teil mit neuer Hardware ersetzt, um den Anforderungen von RUN 2 gerecht werden zu können. Die bisherigen FPGA Auslesekarten konnten auf Grund der höheren Linkraten sowie einer veralteten Host-Schnittstelle nicht mehr mit der verbesserten Detektorhardware und den neuen Server-Maschinen verwendet werden. Ein Ziel dieser Arbeit war daher die Bereitstellung und der Betrieb einer Ausleselösung für ALICE in RUN 2, die den erhöhten Anforderungen gerecht wird, mit den bestehenden Schnittstellen kompatibel ist, eine zeitgemäße und zukunftssichere Technologie bietet und für Erweiterungen der Verarbeitungsschritte in Hardware gerüstet ist. Die Auswahl der Hardware war von Anfang an darauf ausgelegt, dass dieselbe Plattform in den DAQ und HLT Systemen in RUN 2 zum Einsatz kommt. Firmware und Software Implementierungen erfolgten für beide Gruppen separat, da die Anwendungen und die aus RUN 1 wiederverwendeten Technologien deutlich

verschieden sind. Die Firmware-Entwicklungen für den HLT Produktivbetrieb sowie die Integration ins HLT Datenverarbeitungssystem waren ebenfalls Ziele dieser Arbeit.

In einem ersten Schritt musste eine Hardware gefunden werden, die die erhöhten Anforderungen erfüllen kann, in entsprechender Stückzahl rechtzeitig verfügbar ist und in die bestehende Experiment-Infrastruktur integriert werden kann. Hierzu wurden Hardware-Technologien evaluiert, Prototypen erstellt und getestet sowie eine experimentweite Beschaffung organisiert und koordiniert werden.

Ein zweiter Aspekt war die Integration der neuen Auslesekarten in die Hardware- und Softwareumgebung des HLT. Dies umfasste die Entwicklung der Auslesefirmware sowie die Einbindung in das bestehende verteilte HLT Datenverarbeitungssystem. Weiterhin war die Integration, Wartung und Weiterentwicklung der FPGA-basierten Datenvorverarbeitung in Form des Hardware Cluster Finders erforderlich, der auch in RUN 2 einen zentralen Bestandteil der HLT Datenverarbeitungskette bildet. Diese beiden Arbeitspakete waren streng vom Zeitplan des Experiments abhängig. Rechtzeitig vor Beginn der zweiten LHC Runperiode musste eine zuverlässige Hardwarelösung in den Produktivsystemen von DAQ und HLT installiert sein. Diese musste in die Auslese- und Datenverarbeitungssoftware integriert sein und gründlich getestet für die Aufzeichnung von Experimentdaten bereitstehen.

Ein dritter Aspekt dieser Arbeit betrifft die Beschreibung von Datenverarbeitungsschritten in Hardware. Die gängigen Hardwarebeschreibungsmethoden arbeiten üblicherweise auf der Register-Transfer-Ebene, die Datenbewegungen und -Verarbeitungsschritte zwischen Registern Takt-synchron beschreibt. Die Implementierung von Algorithmen auf dieser Ebene ist möglich und wird seit Jahren so betrieben. Auch der Hardware Cluster Finder ist auf dieser Ebene beschrieben. Jedoch bringt die Beschreibung eines Algorithmus auf dieser Ebene einen hohen Aufwand an Entwicklung, Verifikation und Wartung mit sich. Bereits kleine Änderungen am Algorithmus können große Arbeiten an der Implementierung bedeuten. In den letzten Jahren wurden einige Werkzeuge zur Hardwarebeschreibung auf algorithmischer Ebene in Form von High-Level Synthese (HLS) veröffentlicht, die insbesondere den Aufwand von Entwicklung, Verifikation und Wartung vereinfachen sollen. Die meisten dieser Ansätze verwenden Teilmengen imperativer Programmiersprachen, die mit Anweisungen erweitert wurden, um die zeitliche und räumliche Parallelität einer Hardware-Implementierung auszudrücken. Ein vielversprechender Ansatz, der sich konzeptionell von den populärsten Ansätzen etwas abgrenzt, ist die Datenfluss-Programmierung. Datenfluss-Architekturen hatten es zwar nie in kommerziell erfolgreiche Rechnersysteme geschafft, die Konzepte lassen sich jedoch vergleichsweise einfach auf FPGAs abbilden, um damit hoch-effektive Datenverarbeitungssysteme zu entwickeln. Am Beispiel des Hardware Cluster Finder sollte im Rahmen dieser Arbeit der Datenfluss-Ansatz zur High-Level Synthese untersucht werden. Die bestehende Implementierung des Hardware Cluster Finders auf Register-Transfer-Ebene diente hierbei als direkte Referenz, um den Datenfluss-Ansatz zu evaluieren und die Realisierbarkeit einer algorithmischen Hardwarebeschreibung für die Datenverarbeitung einer Experimentauslese zu untersuchen.

3 Die C-RORC Hardware Plattform

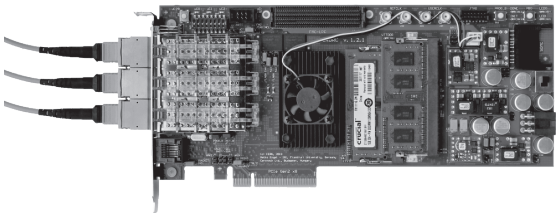


Abb. 1: Foto der C-RORC Karte.

Im Rahmen dieser Arbeit wurde die Common Read-Out Receiver Card (C-RORC) als eine gemeinsame Auslesekarte für die ALICE DAQ und HLT Systeme in RUN 2 entwickelt. Diese Karte musste in der Lage sein, die optischen Detektorlinks sowohl mit der in RUN 1 verwendeten Signalrate, als auch mit erhöhten Raten für die in RUN 2 erweiterten Detektoren auslesen zu können. Die Schnittstelle zum Host-Rechner musste hierbei genügend Bandbreite zur Verfügung stellen, um die eingehenden Detektordaten weiterzuleiten. Im Zuge des technischen Fortschritts war eine Lösung mit erhöhter Link-Dichte gewünscht, da dies die Anzahl der benötigten Host-Rechner zur Installation der Karten reduziert. Weiterhin mussten ausreichend FPGA Ressourcen verfügbar sein, um die aus RUN 1 im HLT bereits eingesetzte Datenvorverarbeitung in Form des Hardware Cluster Finders in RUN 2 mit den erhöhten Anforderungen weiterführen und erweitern zu können. Für die HLT-Anwendung wurde Speicher auf der Karte benötigt, um Detektordaten zu Testzwecken abspielen zu können. Eine spezielle elektrische Schnittstelle wurde von der DAQ-Anwendung benötigt, um die bestehende Flusskontrolle mit dem Triggersystem weiterverwenden zu können. Neben diesen unbedingt notwendigen technischen Voraussetzungen waren eine kompakte Bauform, eine dynamische Auswahloption der seriellen Signalraten sowie ein zuverlässiges und flexibles Konfigurationssystem für die effiziente Integration und den Betrieb in den Produktivsystemen vorgesehen.

Weder zum Zeitpunkt der Hardware-Auswahl noch zu einem späteren Zeitpunkt war eine kommerzielle Hardware-Plattform verfügbar, die diese Anforderungen auch nur annähernd erfüllen konnte. Hardware-Entwicklungen anderer Experimente waren entweder noch nicht weit genug fortgeschritten, oder nicht kompatibel mit den ALICE Anforderungen. Aus diesem Grund wurde die C-RORC als eine eigene FPGA-basierte Auslesekarte konzipiert und entwickelt. Die Link-Dichte, die generischen Komponenten für den anwendungsspezifischen Einsatz, sowie die Innovationen im Bereich Remote-Wartbarkeit waren zum damaligen Zeitpunkt in diesem Formfaktor einzigartig. Ein Foto der Karte ist in Abbildung 1 gezeigt.

Die C-RORC wurde im Rahmen dieser Arbeit als FPGA Auslesekarte für ALICE konzipiert, umgesetzt und integriert. Eine Entwicklung von eigenen Karten bringt immer mehr Herausforderungen insbesondere für kleine Entwicklerteams mit sich. Die Komplexität steigt durch immer mehr Funktionsumfang und schnellere Schnittstellen der Komponenten. Über tausend Pins allein am FPGA und Firmwareentwicklung bereits für die Stromversorgung einer Karte sind keine Seltenheit. Im selben Maße wächst der Umfang der Dokumentation sowie die Kosten und das erforderliche Wissen für passende Entwicklungs-, Simulations- und Messwerkzeuge. Nicht selten is die nächte Generation FPGAs verfügbar, bevor eine Eigenentwicklung wirklich in Produktion gehen kann.

Die Konzeption der ALICE Auslesehardware für RUN 2 wurde 2010 begonnen. Die ersten beiden Prototypen-Serien waren Anfang und Mitte 2013 verfügbar. Zu diesem Zeitpunkt hat sich das ATLAS Experiment dazu entschieden, dieselbe Hardware für ihr Auslesesystem in RUN 2 einzusetzen. Die Serienproduktion der Karten erfolgte daraufhin in Zusammenarbeit zwischen den ALICE und ATLAS Gruppen. Knapp 400 C-RORCs wurden bis Mitte 2014 gefertigt, wobei Teile der Koordination und ein großer Teil der Entwicklung einer Testumgebung zur Verifikation aller Karten bereits beim Hersteller im Rahmen dieser Arbeit geleistet wurden. Die Karten wurden im Herbst 2014 in die ALICE und ATLAS Produktivsysteme eingebaut und waren bis zum Ende von RUN 2 im Dezember 2018 zentrale Bestandteile der jeweiligen Experiment-Auslese.

Neben den Hauptanwendungen in den RUN 2 Auslesesystemen von ALICE und ATLAS hat die Karte eine Anwendung in verschiedenen Test- und Evaluationssystemen gefunden. Teilweise sollen die Karten auch in RUN 3 weiterverwendet werden. Die nach RUN 2 in ALICE nicht mehr benötigten Karten sind bereits für andere Experimente vorgesehen.

4 Integration in den ALICE High-Level Trigger

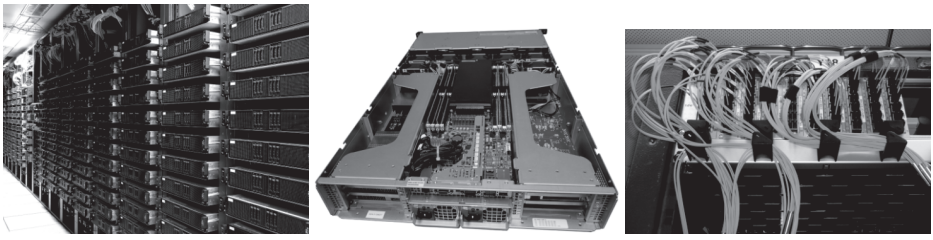


Abb. 2: Links: Foto des ALICE HLT Clusters zu Beginn von RUN 2. Mitte: Offener HLT Rechnerknoten mit C-RORC in rot. Rechts: Optische Aufteilkabel zur Glasfaser-Installation.

Nach Ende der ersten LHC Runperiode wurde die gesamte HLT-Hardware ersetzt um die Daten- und Ereignisraten in RUN 2 verarbeiten zu können. Der HLT besteht in RUN 2 aus 180 identischen Rechner-Knoten die jeweils eine GPU enthalten und mit InfiniBand und Ethernet miteinander verbunden sind. 74 dieser Knoten enthalten außerdem eine C-RORC als zentrale Ein- und Ausgangsschnittstelle für Detektordaten in und aus dem HLT. Die 74 C-RORCs bilden hierbei die Verbindung zwischen über 500 Glasfaserleitungen und den HLT Rechnerknoten. Bis zu 12 Detektorlinks werden pro C-RORC ausgelesen. Glasfaser-Aufteilkabel verbinden die parallel-optischen Schnittstellen der C-RORCs mit der bestehenden Experiment-Infrastruktur. Fotos des Clusters, eines HLT Rechners sowie der Glasfaserinstallation sind in Abbildung 2 gezeigt. Die C-RORCs sind in zwei Datenflussrichtungen im HLT integriert. Sie bilden die Eingangsschnittstelle des HLT, um die Detektordaten von den optischen Links entgegenzunehmen, gegebenenfalls bereits im FPGA mit dem Hardware Cluster Finder vorzuverarbeiten, und an den Host-Rechner weiterzugeben. Weiterhin stellen die C-RORCs die Ausgangsschnittstelle des HLT, indem sie die vom HLT komprimierten und rekonstruierten Daten vom Host-Rechner lesen und über die optischen Links an das DAQ System schicken. Die Signalrate der optischen Links ist zur Laufzeit konfigurierbar. Die Schnittstelle zum Host-Rechner ist mit einer eigenen Direct Memory

Access (DMA) Implementierung zum direkten Zugriff auf den Host-Speicher realisiert. Im Host-Rechner bildet ein Kernelmodul die Grundlage für die Kommunikation mit der Karte per Software sowie die Reservierung von DMA Speicher. Sämtliche Interaktionen mit der Karte erfolgen aus dem Benutzer-Kontext mit einer Treiber-Bibliothek. Die C-RORCs sind ins HLT Datenverarbeitungssystem als Datenquellen und -Senken unter Verwendung dieser Bibliothek integriert. Jeder Detektorlink ist von einem eigenständigen Prozess verwaltet. Dies ermöglicht es, beliebige Link-Konfigurationen unabhängig voneinander zu betreiben. Sämtliche Firmware-Varianten, die gesamte Integration ins HLT-System sowie große Teile der Treiber-Bibliothek wurden im Rahmen dieser Arbeit entwickelt.

5 Hardware Cluster Finding

Der Hardware Cluster Finder ist ein Algorithmus, der in den Rohdaten des größten Detektors in ALICE, der Zeit-Projektionskammer (TPC, [Al10]), nach charakteristischen Signaturen, sogenannten "Clustern", sucht und deren Parameter berechnet und ausgibt. Die Rohdaten werden nach der Auslese verworfen und nur die komprimierten Cluster werden gespeichert. Eine erste Implementierung dieses Algorithmus als parallele Verarbeitungspipeline direkt im Auslese-FPGA war bereits während der ersten LHC Runperiode in den HLT Auslesekarten integriert.

Die Hardware-Vorverarbeitung der Daten war hierbei schon deutlich effizienter als eine vergleichbare Software-Implementierung auf CPUs und konnte die benötigte CPU Rechenleistung im HLT Cluster signifikant reduzieren. Der Algorithmus kann mit laufenden gewichteten Summen beschrieben werden, die sich effizient mit FPGAs berechnen lassen. Abbildung 3 zeigt ein Beispiel der aus Rohdaten berechneten Cluster-Informationen.

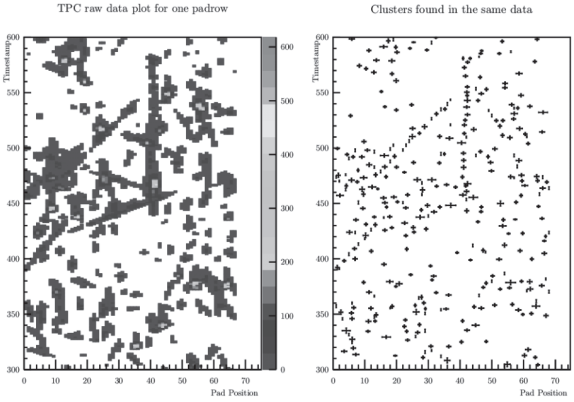


Abb. 3: TPC Rohdaten mit farbkodierten Ladungswerten (links) und die daraus extrahierten Clusterpositionen und -Breiten (rechts).

Die bestehende FPGA-Implementierung aus den RUN 1 Auslesekarte wurde in einem ersten Schritt auf die C-RORC portiert. Sechs Cluster Finder Instanzen sind hierbei pro C-RORC integriert. Die Daten von 216 TPC Ausleselinks werden von 36 C-RORCs im HLT vorverarbeitet. Eine zentrale Erweiterung für ALICE in RUN 2 war das Upgrade der

TPC Ausleseelektronik mit der Readout Control Unit 2 (RCU2, [Al13]). Mit einer höher parallelisierten Auslesearchitektur konnte etwa die doppelte Datenrate vom Detektor zu den RCUs ausgelesen werden. Um diese Datenrate an die DAQ und HLT Onlinesysteme weiterreichen zu können, wurde die Signalrate der optischen Links entsprechend erhöht. Dies hatten deutliche Auswirkungen auf die Datenvorverarbeitung mit dem Cluster Finder in den HLT C-RORCs. Zum einen musste der Cluster Finder mit einer überproportional erhöhten Taktrate betrieben werden, da die verfügbare Bandbreite mit der neuen Detektor Hardware deutlich effektiver ausgenutzt wird. Dies hatte direkten Einfluss auf die maximale Latenz der Berechnungsschritte in Hardware und erforderte eine Optimierung sowohl der Datenpuffer als auch der Pipeline-Architektur um die erforderlichen Taktraten zu erreichen. Zum anderen war eine Abwärtskompatibilität mit dem vorherigen Datenformat gewünscht, um weiterhin beliebige Daten zu Testzwecken vom Kartenspeicher abspielen zu können. Beide Aspekte stellten hohe Ansprüche an die Firmwareentwicklung, da die erforderlichen Taktraten zusammen mit der kombinatorischen Komplexität des Algorithmus bereits nahe an die technischen Grenzen des FPGAs kamen. Diese Firmware-Variante wurde mit der Installation der RCU2 Karten ins Experiment Anfang 2016 im HLT Produktivsystem eingesetzt.

In Zusammenarbeit zwischen den HLT und TPC Gruppen konnte zudem durch die Simulation und Analyse der Roh- und Clusterdaten ein verbesserter Cluster Finding Algorithmus erarbeitet werden, der rauschinduzierte Detektorsignale deutlich besser unterdrücken kann. Die Hardware-Implementierung dieses Algorithmus wurde im Rahmen dieser Arbeit umgesetzt. Messungen haben gezeigt, dass sich die Anzahl gefundener Cluster dadurch um etwa 21 % (Daten mit der Gasmischung von 2017) bzw. 31 % (Daten mit der Gasmischung von 2016) im Vergleich zur vorherigen Implementierung reduziert. Die Qualität der Physikresultate ist von der geringeren Anzahl Cluster nicht betroffen und liefert teilweise sogar bessere Resultate als zuvor. Die reduzierte Anzahl an Clustern bedeutet direkt eine erhöhte Datenkompression im HLT und damit ein reduziertes Datenvolumen für die dauerhafte Datenspeicherung. Diese Firmwarevariante war von Mitte 2017 an im HLT aktiv und hat zusammen mit softwareseitigen Anpassungen die Datenkompressionsrate des HLT auf einen Faktor von über acht erhöht.

Die Bestimmung der Rechenleistung des Hardware Cluster Finders im Vergleich zu einer reinen Softwarelösung benötigt eine definierte Testumgebung. Die Software Referenz-Implementierung in Bezug auf die Physik-Resultate ist der Offline Cluster Finder, der volle Events auf einem einzelnen Knoten verarbeitet. Der Hardware Cluster Finder ist im Gegensatz dazu auf 216 Instanzen in 36 C-RORCs verteilt, die parallel die Daten von jeweils nur einem der 216 optischen Links pro Instanz verarbeiten. Ein Durchsatz-Vergleich von Hardware und Software-Lösung ist jedoch auf Knoten-Ebene möglich. Eine C-RORC mit sechs Cluster Finder Instanzen kann volle TPC-Ereignisse bearbeiten. Der Durchsatz dieses Setups kann mit dem maximalen Durchsatz von parallelen Offline Cluster Finder Instanzen auf demselben Knoten verglichen werden. Die Hardware Implementierung in einer einzelnen C-RORC unter Benutzung von vier CPU Kernen hat sich dabei als etwa einen Faktor zehn schneller als die reine Softwareimplementierung, oder äquivalent zu 480 CPU Kernen erwiesen. Die enormen Einsparungen an CPU Rechenleistung im HLT durch die Datenvorverarbeitung in Hardware sind damit offensichtlich.

6 Datenfluss-Implementierung des Cluster Finders

Die Detektoren und das Experiment wurden über RUN 2 hinweg kontinuierlich weiterentwickelt. Die Firmware-Entwicklungen, insbesondere am Hardware Cluster Finder, haben auch im Rahmen dieser Arbeit aufgezeigt, wie aufwendig bereits kleinere algorithmische Änderungen umzusetzen sein können. Nicht selten müssen dabei Firmware-Teile von Grund auf neu implementiert und verifiziert werden. Ein Grund hierfür ist, dass die Hardwarebeschreibung in der Regel auf Register-Transfer-Ebene stattfindet und damit die Partitionierung eines Algorithmus in Raum und Zeitrichtung komplett manuell erfolgen muss. Kleine algorithmische Änderungen können daher große strukturelle Auswirkungen auf diese Partitionierung haben. Ansätze zur High-Level Synthese versprechen dieses Problem zu lösen, indem sie die Hardwarebeschreibung auf die algorithmische Ebene heben und die Partitionierung auf Register-Transfer-Ebene einem Compiler überlassen. Im Rahmen dieser Arbeit wurde hierzu der Datenfluss-Ansatz untersucht. Ein Framework aus FPGA-Karte, einem Compiler sowie Softwarebibliotheken erlaubt es, Algorithmen aus einer Datenflussbeschreibung heraus unkompliziert auf der Hardware zu betreiben. Implementierungsdetails wie der Datentransfer zu und vom FPGA, Speicherreservierung, Flusskontrolle in der Firmware usw. werden funktionsfertig bereitgestellt und deren Interna bleiben vor dem Nutzer verborgen. Die Datenflussbeschreibung des Algorithmus wird vom Compiler in einen Datenfluss-Graphen übersetzt und mit Hilfe einer internen Bibliothek von Firmwarebausteinen auf das FPGA abgebildet. Die Datenfluss-Beschreibung erfolgt in dem hier verwendeten Framework in einer Sprache mit Java-ähnlichem Syntax, die einerseits auf eine in Hardware abbildbare Teilmenge reduziert und andererseits um Aspekte der Steuerung von räumlicher und zeitlicher Parallelität auf der Hardware erweitert wurde. Der Compiler stellt mit Regeln sicher, dass die somit beschriebene Funktionalität als effiziente Verarbeitungs-Pipeline im FPGA implementiert werden kann. Die Verarbeitungsschritte werden in “Kernels” eingebunden. “Manager” beschreiben die Struktur auf Systemebene und verbinden Kernels untereinander, mit Schnittstellen zum Host-Rechner, zum Speicher auf der Karte oder zu Netzwerkkomponenten. Das Framework bringt eine Simulations- und Entwicklungsumgebung mit, in der sowohl der Datenfluss-, als auch der Host-Code entwickelt, verifiziert und auf der Zielhardware ausgeführt werden kann.

Der Hardware Cluster Finder Algorithmus wurde in dieser Arbeit mit dem Datenfluss-Framework umgesetzt. Die Struktur dieser Implementierung hat sich dabei im Vergleich zur bereits bestehenden hoch-optimierten low-level Implementierung kaum verändert. Der Algorithmus konnte unkompliziert in der Datenfluss-Sprache beschrieben werden. Die Beschreibung von mathematischen Operationen unabhängig von den Datentypen, Latenzen und mit den gängigen Operatoren aus der Softwareprogrammierung vereinfachte und beschleunigte die Entwicklung beträchtlich. Zur Integration der Datenfluss-Implementierung des Cluster Finders in die HLT-Umgebung wurden verschiedene Möglichkeiten untersucht. Eine Ersetzung der C-RORC mit einer reinen Datenfluss-Implementierung der gesamten Ausleselösung war auf Grund von Beschränkungen der Datenfluss-Hardware sowie des Frameworks nicht möglich. Die Auslagerung des Cluster Findings in dedizierte Datenfluss-Hardware wäre nur mit deutlich erhöhten Kosten und Komplexität möglich gewesen. Ein vielversprechender Ansatz war jedoch die Integration des Datenfluss Cluster Finders in die bestehende HLT C-RORC Hardware-, Firmware- und Softwareumgebung

über die eigenständigen Netzlisten. Dies konnte exemplarisch gezeigt werden. Der Durchsatz der händischen Implementierungen und der mit der Datenfluss-Umgebung entwickelten Implementierungen ist konstruktionsbedingt identisch. Auch die maximal möglichen Taktraten sind ähnlich. Der Ressourcenverbrauch beider Varianten hat sich als beinahe identisch erwiesen. Der Quellcode-Umfang der Datenfluss-Implementierung ist auf Grund der höheren Abstrahierung deutlich geringer und damit leichter zu entwickeln, zu erweitern und zu warten.

Ein großer Vorteil des Datenfluss-Ansatzes ist die Möglichkeit, Erweiterungen und Änderungen am Algorithmus schnell und einfach zu evaluieren. In dieser Arbeit wurden beispielhaft die Auswirkungen auf den Ressourcenverbrauch durch die Variation der Fixed-Point Genauigkeit sowie das Verschieben von Verarbeitungsschritten zwischen Software und Hardware untersucht. Beides sind essentielle Schritte für die Planung einer Hardware-Implementierung, die mit der üblichen Register-Transfer Beschreibung der Hardware extrem zeitaufwändig sind. Auf der Datenfluss- / Algorithmusebene waren diese Evaluationen in kurzer Zeit möglich.

7 Ergebnisse, Schlüsse und Ausblick

Die Ziele der Arbeit waren die Entwicklung einer gemeinsamen Ausleselösung für die ALICE DAQ und HLT Systeme in RUN 2, die Integration und Betreuung dieser in der HLT Anwendung sowie die Untersuchung von Ansätzen zur Hardwarebeschreibung auf algorithmischer Ebene. Diese Ziele wurden erreicht und teilweise übertroffen. Es wurde eine Hardwareplattform entwickelt, die neben den eigentlichen Zielanwendungen in den ALICE DAQ und HLT Systemen außerdem im ATLAS Experiment, sowie zahlreichen weiteren Testsystemen und Experimenten ihre Anwendung gefunden hat. Die Parallelnutzung von Hardwareentwicklungen über LHC-Großexperimente hinweg war zu diesem Zeitpunkt ein Novum. Weiterhin ist die Karte für den Einsatz in RUN 3 vorgesehen. Die Karte wurde im Rahmen dieser Arbeit über den gesamten Produktlebenszyklus hinweg von der Konzeption an über Entwicklung, Koordination, Produktion, Test, Firmware- und Softwareentwicklung sowie Integration und Produktivbetrieb im HLT betreut. Sie war rechtzeitig für den Betrieb in den RUN 2 Produktivsystemen verfügbar, erfüllt alle Anforderungen und hatte bis zum Ende von RUN 2 keine technischen Ausfälle zu verzeichnen.

Die Firmware und Software der HLT Anwendung wurde für RUN 2 vorbereitet und über die gesamte Betriebsdauer hinweg kontinuierlich weiterentwickelt. Ein zentraler Aspekt in der HLT Anwendung ist die Datenvorverarbeitung mit dem Hardware Cluster Finder im Auslese-FPGA. Dieser Algorithmus konnte auf die neue Hardware portiert werden, wurde im Rahmen des TPC-Elektronikupdates auf den doppelten Durchsatz optimiert und hat mit einem verbesserten Algorithmus zu einer deutlich erhöhten HLT Datenkompressionsrate bei teilweise sogar verbesserten Physik-Parametern beigetragen. Im Vergleich zu äquivalenten Verarbeitungsschritten in Software, ist eine HLT FPGA-Auslesekarte mit Hardware Cluster Finder etwa einen Faktor 10 schneller als eine parallelisierte Softwareverarbeitung auf einem HLT Knoten.

Die Firmware-Entwicklungen um den Hardware Cluster Finder waren durchweg erfolgreich und mit sehr positiven Resultaten für das Gesamtexperiment verbunden, jedoch auch sehr aufwändig. Änderungen an der Auslesearchitektur sowie dem Algorithmus hatten die Neuentwicklung mehrerer zentraler Komponenten mit hohem Verifikationsaufwand erfordert. Die Beschreibung des Algorithmus mit dem Datenfluss-Ansatz hat sich hierbei als effiziente, kompakte und leicht wartbare Alternative gezeigt. Sowohl der Ressourcenverbrauch, als auch der Durchsatz waren vergleichbar mit einer händischen Implementierung. Beides konnte mit einem Bruchteil der Entwicklungszeit und des Code-Umfangs erreicht werden. Evaluationen von Verarbeitungsschritten in Hardware waren damit sehr einfach möglich. Für die Integration ins Gesamtsystem konnte die Datenfluss-Beschreibung in eine eigenes Firmware- und Softwareumgebung eingebunden werden. Dieser Ansatz verdeutlicht, wie konventionelle Firmware-Entwicklung und HLS-Ansätze zukünftig schrittweise zusammengebracht werden können, um die Vorteile der Hochsprachen mit bestehenden Bausteinen zu kombinieren. Am Beispiel des Hardware Cluster Finders konnte gezeigt werden, wie eine eine algorithmische Hardwarebeschreibung den Entwicklungsaufwand beträchtlich vereinfacht. Sie reduzierte die Entwicklungszeit sowie den Umfang der Beschreibung. Damit erleichterte sie außerdem die Wartung und Weiterentwicklung. Die Umsetzung des Algorithmus mit der Datenfluss-Beschreibung war dies in diesem Fall sogar ohne Einbußen bei Ressourcenverbrauch oder Taktrate möglich.

Literaturverzeichnis

[Aa08] Aamodt, K. et al.: The ALICE experiment at the CERN LHC. JINST, 3:S08002, 2008.

[Al10] Alme, J. et al.: The ALICE TPC, a large 3-dimensional tracking device with fast readout for ultra-high multiplicity events. NIM-A, 622(1):316–367, Oct 2010.

[Al13] Alme, J et al.: RCU2 – The ALICE TPC readout electronics consolidation for Run2. JINST, 8(12):C12032, 2013.

[EB08] Evans, Lyndon; Bryant, Philip: LHC Machine. JINST, 3(08):S08001, 2008.

[En19] Engel, H.: Development of a Read-Out Receiver Card for Fast Processing of Detector Data – ALICE HLT Run 2 Readout Upgrade and Evaluation of Dataflow Hardware Description for High Energy Physics Readout Applications. Dissertation, Goethe-University Frankfurt a. M., 2019.

[Ri07] Richter, M. et al.: High Level Trigger Applications for the ALICE Experiment. IEEE Transactions on Nuclear Science, 55(1):133–138, 2007.

Dr. Heiko Engel hat an der Ruprecht-Karls Universität in Heidelberg Physik studiert und das Studium mit Diplom abgeschlossen. Im Anschluss entwickelte und betreute er an der Johann Wolfgang Goethe-Universität in Frankfurt a. M. Auslesehardware für das ALICE Experiment am CERN. Im Kontext dieser Arbeiten schloss er 2019 am Lehrstuhl von Prof. Dr. Udo Kebschull seine Promotion in Informatik mit Auszeichnung ab. Seit Abschluss der Arbeiten am CERN arbeitet er an programmierbaren Hardwarelösungen für die Industrie.

Effektives Widening mit Hashbasierter Partitionierung des Hypothesenraums

Alexander Fillbrunn[1]

Abstract: Vielen Algorithmen im Data Mining basieren auf gierigem Verhalten, um ein ausreichend gutes Modell schnell zu finden. Bei der Verwendung solcher Greedy-Algorithmen besteht allerdings die Gefahr, dass diese in lokalen Optima stecken bleiben. Mit Hilfe von Widening, einer Technik um den Hypothesenraum breiter zu durchsuchen, kann diese Gefahr verringert werden, indem parallel mehrere, möglichst unterschiedliche, Modelle erzeugt werden. Bisherige Verfahren im Widening leiden jedoch unter zweierlei Problemen: dem Overhead durch die Kommunikation zwischen den parallelen Recheneinheiten und der Notwendigkeit, Modelle bezüglich ihrer Ähnlichkeit miteinander vergleichen zu können. In der vorliegen Arbeit wird mit dem Bucket-Selektor ein randomisiertes und modellunabhängiges Widening-Verfahren vorgestellt, das schneller zu ähnlich guten Ergebnissen gelangt wie bisherige Verfahren.

1 Einleitung

Mit dem Aufkommen von Multiprozessorsystemen, Rechenclustern und verteilten Systemen wie Hadoop, Spark und Flink begann vor einigen Jahren die Arbeit daran, bisher sequentielle Algorithmen an diese neuen Gegebenheiten anzupassen und die vorhandene parallele Rechenkraft effizient auszunutzen. Das Hauptaugenmerk wurde dabei fast ausschließlich darauf gelegt, die Algorithmen zu beschleunigen und dabei die gleichen Ergebnisse wie bisher zu erzielen. Wenig Beachtung wurde dagegen dem Ansatz geschenkt, die zusätzlichen Recheneinheiten zu verwenden, um die Ergebnisse existierender Algorithmen zu verbessern und dabei die bisherige Geschwindigkeit beizubehalten. Auch im maschinellen Lernen fokussierte sich die Forschung lange Zeit auf die möglichen Geschwindigkeitszuwächse, obwohl auch die Genauigkeit der dort verwendeten Vorhersagealgorithmen von einer Parallelisierung profitieren kann.

Ein im maschinellen Lernen häufig anzutreffendes Problem ist die Suche nach einer den vorhandenen Daten zugrundeliegenden Struktur, beziehungsweise einem Modell, das diese Struktur beschreibt und das später für Vorhersagen oder für das bessere Verständnis der vorliegenden Daten herangezogen werden kann. Ein Algorithmus, der ein solches Modell automatisch lernt, legt sich meistens am Anfang des Lernprozesses auf einen Modelltyp fest, der durch Anpassung verschiedener Parameter so optimiert wird, dass die Struktur der Daten durch das Modell erklärt werden kann. Die optimalen Parameter für einen Modelltyp zu finden ist allerdings ein aufwändiger Prozess, der meist das Durchsuchen vieler verschiedener Wertekombinationen voraussetzt. Durch Ausprobieren kleiner Veränderungen der Parameter kann ein Algorithmus ein Modell Schritt für Schritt verbessern, nicht im-

[1] KNIME GmbH, alexander.fillbrunn@knime.com

CC BY SA

mer führt dies aber zu einem optimalen Ergebnis. Stellt man sich den Parameterraum eines Modells als Landschaft vor, in der die Qualität des von den Parametern beschriebenen Modells die Höhe bestimmt, dann verhindern verschiedene topologische Strukturen die Konvergenz zum besten Modell, dem *globalen Optimum*. Das beste Beispiel für eine solche hinderliche Struktur ist ein *lokales Optimum*: ein Bereich im Parameterraum, der ein besseres Modell repräsentiert als alle Parameterkombinationen um ihn herum. Ein Algorithmus, der nur kleine Änderungen an den Parametern vornimmt und diese übernimmt, wenn sich das Modell dadurch verbessert, wird nie aus solch einer Region herauskommen.

Die Idee des *Widening* [Ak00] ist es, parallele Recheneinheiten gleichzeitig auf die Suche nach einer optimalen Lösung für ein Problem des maschinellen Lernens einzusetzen. Diese Lösungen sind im Allgemeinen Modelle, die im *Hypothesenraum* gesucht werden. Man möchte im Widening bessere Modelle finden, statt einen existierenden Algorithmus bei gleichem Ergebnis schneller zu machen („Better, not faster“ [AIB12]). Um dieses Ziel zu erreichen muss eine Balance zwischen der Exploration des Hypothesenraums und dem Fokus auf qualitativ hochwertige Modelle gefunden werden. Es handelt sich also um eine Optimierung mit mehreren Zielen, auch Pareto-Optimierung [Eh05] genannt.

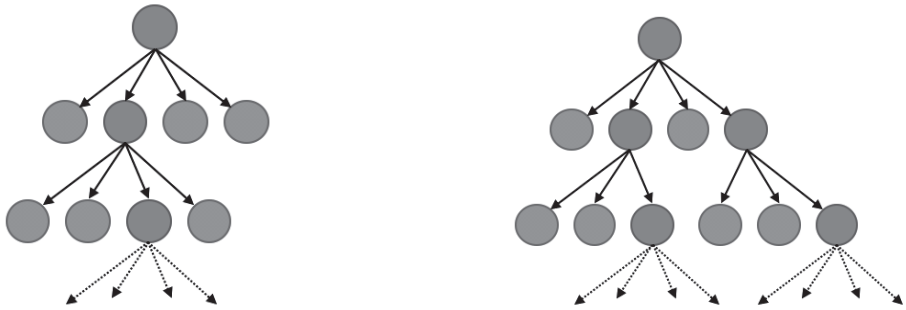
In dieser Zusammenfassung der Hauptarbeit [Fi19] soll zuerst das theoretische Konzept des *Widening* kurz erklärt und Probleme bisheriger Vorgehensweisen in diesem Bereich hervorgehoben werden. Anschließend wird der *Bucket-Selektor* als Hauptbeitrag der Arbeit vorgestellt. Es handelt sich um ein Widening-Verfahren, das schneller als bisherige Ansätze ist und auf zwei untersuchten Anwendungsfällen, dem hierarchischen agglomerativen Clustering und der Join Order Optimierung, gleich gute oder bessere Ergebnisse liefert. Von den beiden Anwendungsfällen wird hier aus Platzgründen nur das Clustering vorgestellt.

2 Widening

Viele Algorithmen im Data Mining und anderen Bereichen, die einen Hypothesenraum nach einem günstigen Modell durchsuchen, gehen dabei gierig, beziehungsweise greedy, vor und führen eine lokale Suche aus. Oft werden Greedy-Algorithmen verwendet, wenn eine vollständige Auflistung aller Modelle des Hypothesenraums zu zeitaufwändig ist. Greedy-Algorithmen sind meist schnell, führen aber nicht zwangsläufig zu einem optimalen Endergebnis. In vielen Fällen haben Greedy-Algorithmen das Problem, dass sie in einem Zwischenschritt eine zu diesem Zeitpunkt optimal erscheinende Entscheidung treffen, von welcher sich aber später herausstellt, dass eine andere Entscheidung längerfristig zu einem besseren Ergebnis geführt hätte. Der Algorithmus verfängt sich also in einem lokalen Optimum.

Im Widening [AIB12] sollen die Auswirkungen eines solchen Greedy-Verhaltens abgemildert werden, indem parallel mehrere Pfade durch den Hypothesenraum traversiert werden, wie es in Abbildung 1 zu sehen ist. Dabei ist es wichtig sicherzustellen, dass parallele Recheneinheiten nicht dieselben und idealerweise sogar möglichst diverse Lösungsansätze verfolgen, um die Gefahr, dass alle in das gleiche lokale Optimum fallen, zu verringern.

Es handelt sich gewissermaßen um einen Kompromiss zwischen der schnellen aber sehr begrenzten Greedy-Suche und der ausführlichen und sehr langsamen vollständigen Suche.



(a) Greedy (b) Widening

Abb. 1: Konzept des Widening, in welchem statt eines Pfades durch den Hypothesenraum parallel mehrere verfolgt werden.

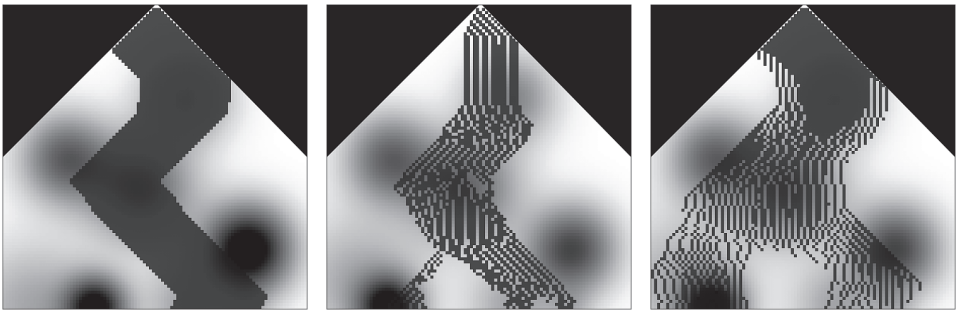
Die beiden Grundbausteine für die Formalisierung des Widening sind die beiden Operatoren *Verfeinerung* $r(\cdot)$ und *Selektion* $s(\cdot)$, deren Zweck in Abbildung 1 zu erkennen ist: Die Verfeinerung erzeugt aus einem Modell entsprechende Nachfolger (zum Beispiel das Hinzufügen eines neuen Asts in einem Entscheidungsbaum oder das Zusammenfügen zweier Cluster im hierarchischen Clustering). Bei der Selektion werden anschließend aus allen Verfeinerungen eine (Greedy) oder mehrere (Widening) ausgewählt. Einen Schritt in einem Greedy-Algorithmus kann mit der Formel $m' = s(r(m))$ beschrieben werden. Hierbei ist m ein Modell und m' dessen Nachfolger, der durch das Anwenden des Selektors auf die Verfeinerungen von m erzeugt wird. Der Greedy-Algorithmus setzt hier bei der Selektion auf eine heuristische Bewertungsfunktion ψ, die jedem Modell einen Qualitätswert zuweist. Das Modell mit der höchsten Bewertung wird in der nächsten Iteration weiterverfolgt. Im Widening dagegen werden mehrere Modelle verfolgt und aus der Greedy-Formel wird die Folgende:

$$\{m'_1, \ldots, m'_k\} = s_{global}(s_{lokal}(r(m_1)) \cup \ldots \cup s_{lokal}(r(m_k))) \; . \tag{1}$$

Hier wird auf Modellmengen gearbeitet und in jedem Schritt k Modelle verfeinert und ausgewählt. k sollte hierbei mindestens so groß wie die Anzahl der verfügbaren parallelen Recheneinheiten sein. Die Selektion wird dabei außerdem in lokale und globale Selektion aufgeteilt, da die Modelle auf die Recheneinheiten verteilt werden und diese bereits lokal vorselektieren können, bevor eventuell an zentraler Stelle die Modelle für die nächste Iteration ausgewählt werden. Benötigt wird die globale Selektion beispielsweise, um sicherzustellen, dass sich die Modelle der einzelnen Recheneinheiten nicht zu ähnlich werden.

Während der Verfeinerungsoperator oft durch den zugrundeliegenden Lernalgorithmus vorgegeben ist, ist die Wahl des Selektors entscheidend für das Erreichen guter Ergebnisse. Im Folgenden sollen einige Selektionsmethoden kurz vorgestellt werden. Zur Veranschaulichung dient hier ein einfaches Beispielproblem, bei dessen Zustandsgraphen es sich um einen endlichen, gerichteten und azyklischen Graphen handelt. Jeder Zustand wird durch

einen Pixel der Visualisierung repräsentiert und je dunkler der Pixel, desto höher die heuristische Bewertung. Jeder Zustand besitzt maximal 3 Nachfolger: die 3 Pixel unter ihm. Der Graph hat genau einen Startzustand und 199 Endzustände und alle Pfade zwischen dem Startzustand und allen Endzuständen haben die Länge 100. Die Algorithmen durchlaufen den durch das Raster repräsentierten Graphen von oben nach unten.



(a) Top-k (b) Diverse Top-k (c) Score erosion mit $\beta = 1$

Abb. 2: Explorationsverhalten verschiedener Selektoren. Dunkle Pixel symbolisieren bessere Zustände und die Algorithmen durchlaufen das Raster von oben nach unten.

Der Top-k-Algorithmus [IB13] ist die naive Erweiterung des einfachen Greedy-Algorithmus, denn statt eines besten Folgemodells werden hier die k besten ausgewählt. Da in vielen Problemen ähnliche Modelle auch ähnliche Bewertungen durch die Bewertungsfunktion erhalten, bedeutet das oft, dass der Top-k-Algorithmus sehr ähnliche Modelle wie der einfache Greedy-Algorithmus exploriert und wenig divers ist. Die Top-k-Selektion ist jedoch sehr schnell und kann in $\mathcal{O}(n \log k)$ mit Hilfe einer Min-Heap-Struktur durchgeführt werden. Wenn die Modelle von parallelen Recheneinheiten berechnet und gespeichert werden, dann kann zuerst jede Einheit lokal die besten k Modelle in ihrer Partition finden, bevor dann k^2 Modelle zentral gesammelt und davon wiederum k Modelle für die nächste Iteration ausgewählt werden können. Weil Modelle zentral gesammelt und wieder an die Recheneinheiten verteilt werden müssen, benötigt das Top-k-Verfahren Kommunikation mit der Komplexität $\mathcal{O}(k^2)$ zwischen den Einheiten.

Diverse-Top-k [IB13] ist eine Erweiterung des Top-k-Algorithmus. Das Verfahren benötigt eine Distanzfunktion δ und einen Schwellenwert θ und wählt die besten Modelle so aus, dass jedes ausgewählte Modell mindestens die Distanz θ von jedem anderen ausgewählten Modell hat. Da lokal und global unterschiedliche Schwellwerte verwendet werden können, sind hier s_{local} und s_{global} nicht zwingend gleich. Die Laufzeitkomplexität des Diverse-Top-k-Selektors entspricht $\mathcal{O}(n \log n + nk)$, da die Modelle sortiert werden und anschließend jedes Modell mit den bereits ausgewählten verglichen werden muss. Durch die Sortierung beträgt die Speicherkomplexität mindestens $\mathcal{O}(n)$ und wie bei Top-k werden in jeder Iteration $(k+1)(k-1) \in \mathcal{O}(k^2)$ Modelle ausgetauscht, weil die Modelle zentral gesammelt und neu verteilt werden müssen.

Der *Score-Erosion*-Algorithmus [Me10] wurde für die Auswahl aktiver und diverser Moleküle im Hochdurchsatz-Screening (*High-Throughput Screening*) entwickelt. Der dort durchsuchte Molekülraum enthält hauptsächlich inaktive Moleküle mit einigen sogenann-

ten *Aktivitätsinseln* (*activity islands*), also Nachbarschaften von Molekülen mit hoher Aktivität. Das Ziel des Score-Erosion-Algorithmus ist es, Moleküle mit hoher Aktivität zu finden und dabei zu vermeiden, mehrere Moleküle einer Aktivitätsinsel auszuwählen. Das Ergebnis ist eine Auswahl aktiver und diverser Moleküle. Um dieses Ziel zu erreichen, wählt der Algorithmus iterativ das Molekül mit der höchsten Aktivität und verringert (oder erodiert) anschließend die Aktivität der anderen Moleküle abhängig von deren Distanz zum gerade ausgewählten. Ein Parameter β kontrolliert die Stärke der Erosion. Der Score-Erosion-Selektor hat eine Laufzeitkomplexität von $\mathcal{O}(nk)$ und eine Speicherkomplexität von $\mathcal{O}(n)$. Parallelisieren lässt sich dieser Selektor nur schwierig, denn nach jeder Auswahl muss die Bewertung aller Modelle angepasst und daraufhin das global beste gefunden werden.

Die Selektoren, die in diesem Abschnitt vorgestellt wurden, haben Nachteile in Bezug auf die Laufzeit, Kommunikation, Parallelisierbarkeit und Speicherverbrauch. Besonders ungünstig sind Methoden, die die Verfeinerungen mehrfach durchlaufen müssen, oder die Zugriff auf alle Verfeinerungen auf einer zentralen Recheneinheit benötigen. Des Weiteren ist das Berechnen von Distanzen zwischen Modellen äußerst kostspielig. Der im nächsten Abschnitt vorgestellte Bucket-Selektor besitzt diese Nachteile nicht.

3 Der Bucket-Selektor

Der Bucket-Selektor unterteilt einerseits den Suchraum in zufällige Partitionen, die von unterschiedlichen Recheneinheiten durchsucht werden, vermeidet aber andererseits das Problem der Nicht-Erreichbarkeit von Modellen an Partitionsgrenzen, indem er einen Synchronisationsschritt zwischen den Einheiten durchführt, bei welchem Modelle ausgetauscht werden. Die Diversität der Auswahl wird bei dem Bucket-Selektor durch eine hashbasierte Partitionierungsfunktion erreicht. Der Selektor garantiert, dass das beste Modell immer und das *i*te Modell mit abnehmender Wahrscheinlichkeit ausgewählt wird. Somit haben auch Modelle, die nicht unter den besten k sind, eine Chance, in die nächste Iteration übernommen zu werden. Wie andere Widening-Selektoren auch geht der Bucket-Selektor level-synchron vor. Das heißt, dass die Iterationen der einzelnen Recheneinheiten durch einen Austauschschritt am Ende jeder Iteration synchronisiert werden. Dies entspricht auch einem möglichen Vorgehen bei der parallelen Breitensuche [BM11].

Der Bucket-Selektor partitioniert den Suchraum mit Hilfe einer Hashfunktion, die jedem Modell einen ganzzahligen Wert zwischen 1 und k zuweist, wobei idealerweise k parallele Recheneinheiten zur Verfügung stehen. Dann ist jede Recheneinheit ist für die Modelle genau einer Partition zuständig. In jeder Iteration verfeinert jede Recheneinheit zuerst ihr aktuelles Modell, dann werden die Verfeinerungen gehasht und für jeden Hashwert das beste Modell ermittelt. Anschließend werden diese besten Modelle an die jeweils verantwortliche Recheneinheit geschickt. Diese kann dann aus allen lokal besten Modellen das global beste auswählen und in der nächsten Iteration weiter verfeinern. Das Verfahren wird in Abbildung 3 dargestellt.

Dadurch, dass aus jeder Partition das beste Modell ausgewählt wird und die Partitionen zufällig mit Hilfe der Hashfunktion ermittelt werden, konkurriert bei N Modellen jedes

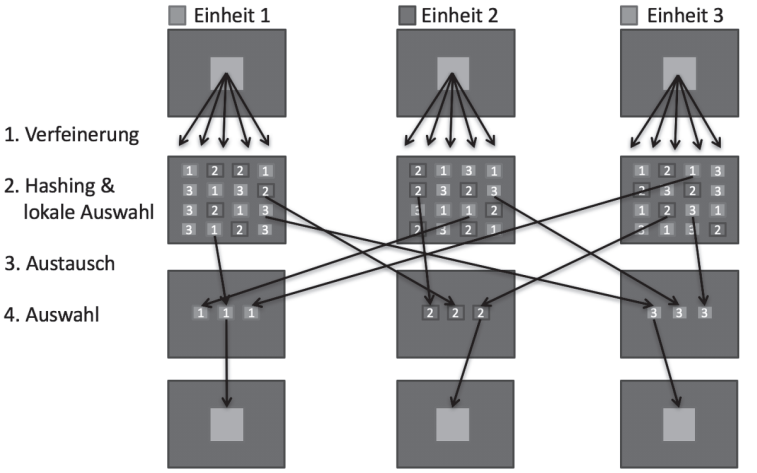


Abb. 3: Das Bucket-Selektionsverfahren. Jede Recheneinheit verfeinert ihr Modell, berechnet Hashwerte für die Verfeinerungen und wählt aus jeder Partition das beste Modell. Anschließend werden die Modelle so ausgetauscht, dass jede Einheit alle lokal besten Modelle ihrer Partition hält. Aus diesen wählt sie das insgesamt beste aus.

Modell mit $\frac{N}{k} - 1$ anderen Modellen um den ersten Platz in der jeweiligen Partition. Wichtig ist deshalb, dass die Werte der Hashfunktion möglichst gleichmäßig verteilt sind, also dass $P(h(m) = i) \approx \frac{1}{k}$, damit jedes Modell die gleiche Chance hat, das beste Modell seiner Partition zu sein. Eigenschaften kryptographischer Hashfunktionen, wie die Einweg-Eigenschaft und das Bestreben, möglichst wenige Kollisionen zuzulassen, spielen bei der Auswahl der Hashfunktion keine Rolle.

Bei N verschiedenen Modellen, aus denen k ausgewählt werden sollen, konkurriert jedes Modell im Mittel mit $\frac{N}{k}$ Modellen, solange die Hashfunktion gleichmäßig verteilte Werte erzeugt. Je besser also ein Modell verglichen mit allen anderen ist, desto wahrscheinlicher wird es ausgewählt. Da aber durch die zufällige Partitionierung auch ein schlechteres Modell das beste in seiner Partition sein kann, werden nicht nur die besten k Modelle gewählt. Für jedes Modell m gibt es genau $\mathrm{rg}_\psi(m) - 1$ Modelle, die eine bessere Bewertung $\psi(m)$ haben. Für jedes dieser Modelle beträgt die Wahrscheinlichkeit, in einer anderen Partition als m zu landen $(1 - \frac{1}{k})$. Folgende Formel kann somit verwendet werden, um die Auswahlwahrscheinlichkeit P_s für m zu berechnen:

$$P_s(m) = \left(1 - \frac{1}{k}\right)^{\left(\mathrm{rg}_\psi(m)-1\right)} . \tag{2}$$

Wie hier ersichtlich ist, beträgt die Auswahlwahrscheinlichkeit für das beste Modell mit $\mathrm{rg}_\psi = 1$ genau $\left(1 - \frac{1}{k}\right)^0 = 1$. Diese Garantie, dass das beste Modell immer ausgewählt wird, unterscheidet den Bucket-Selektor von anderen stochastischen Selektoren.

Im Gegensatz zu den meisten anderen in Abschnitt 2 vorgestellten Selektoren partitioniert der Bucket-Selektor durch seine Hashfunktion den Suchraum in k Partitionen, wobei jede Partition als individuellen Selektionsoperator einen Bucket-Selektor besitzt, der nur das beste Modell aus einem bestimmten Bucket behält und die anderen Modelle an die entsprechende Recheneinheit weiterleitet. Somit ist ein zentrales Sammeln der Modelle nicht nötig. Des Weiteren besitzt die lokale Selektion hier nur eine Laufzeitkomplexität von $\mathcal{O}(n)$, da für jedes Modell nur einmal die Hashfunktion evaluiert werden muss, um für jede Partition das beste Modell zu finden.

4 Hierarchisches Clustering mit dem Bucket-Selektor

Die hierarchische Clusteranalyse [RK90] ist ein Verfahren, in welchem eine Baumstruktur erzeugt wird, die die einzelnen Datenpunkte als Blattknoten und Cluster, also Gruppen von Datenpunkten, als innere Knoten besitzt. Die Evaluierung des Bucket-Selektors befasst sich im Folgenden mit dem agglomerativen hierarchischen Clustering. Hier ist zu Beginn jeder Datenpunkt ein Cluster und mit jeder Iteration des Algorithmus werden die zwei am nächsten beieinander liegenden Cluster zu einem zusammengefasst, bis am Ende ein einzelner Cluster mit allen Datenpunkten entsteht. Anschließend kann das entstandene Dendrogramm an beliebiger Stelle horizontal geteilt werden, um die gewünschte Anzahl an Clustern zu erhalten. Dabei handelt es sich um ein gieriges Vorgehen, da die Entscheidung darüber, welche Cluster miteinander verbunden werden, anhand lokaler Qualitätskriterien getroffen wird.

Die Qualität eines Clusterings ist meist sehr subjektiv und hängt von den gewünschten Eigenschaften der Gruppierung ab. Ist die gewünschte Anzahl von Clustern bekannt, kann man für die Evaluierung des Clusterings Maße wie den Silhouettenkoeffizient [Ro87] und den Davies-Bouldin Index [DB79] verwenden. Bei beiden handelt es sich um interne Qualitätsmaße, das heißt, es gibt keine *Ground Truth* und die Qualität wird allein über die Daten und die ihnen zugewiesenen Cluster bestimmt, ohne externe Informationen miteinzubeziehen.

Soll stattdessen das vom hierachischen agglomerativen Clustering erzeugte Dendrogramm bewertet werden, sind diese Maße nur bedingt geeignet. Stattdessen kann hier auf den *kophenetischen Korrelationskoeffizienten* [SR62] zurückgegriffen werden. Allgemein ist zu erwarten, dass im hierachischen Clustering ähnliche Datenpunkte früh im gleichen Cluster landen. Entsprechend kann man die Höhe des Knotens im Dendrogramm, in welchem zwei Datenpunkte zusammenkommen, als eine durch das Dendrogramm induzierte ultrametrische Distanz betrachten [Mi79]. Der kophenetischen Korrelationskoeffizient berechnet sich als die Korrelation dieser Distanzen mit den ursprünglichen, bekannten, Distanzen zwischen den Datenpunkten.

Bei der Evaluation des Bucket-Selektors wurde das Hauptaugenmerk auf den Vergleich bezüglich Qualität der erzeugten Modelle gelegt, weshalb in dieser Zusammenfassung nur darauf und nicht auf die Laufzeituntersuchungen eingegangen werden soll. Aus Platzgründen wird der Bucket-Selektor im Folgenden lediglich mit dem Greedy-Algorithmus

und mit Top-k verglichen. Für weitere Vergleiche mit explizit diversitätserhaltenden Selektoren wie Diverse Top-k und Score Erosion wird auf die Hauptarbeit [Fi19] verwiesen. Die Experimente wurden auf 7 verschiedenen öffentlich verfügbaren Datensätzen durchgeführt: Seeds, Ruspini, Iris, User Knowledge Modeling (U.K.M.), Breast Cancer (B.C.), SPECTF und Cement. Deren Quellen sind ebenfalls in der Hauptarbeit [Fi19] zu finden.

Alle hier vorgestellten auf hierarchisches Clustering basierenden Vergleiche wurden mit UPGMA-Linkage durchgeführt, da diese mit dem Greedy-Algorithmus von allen Linkage-Kriterien auf den getesteten Datensätzen den besten kophenetischen Korrelationskoeffizienten erzeugte. Als Distanzmetrik für das Clustering wurde die Euklidische Distanz verwendet. Tabelle 1 fasst die Ergebnisse des Vergleichs zusammen. Bis auf den SPECTF-Datensatz kann der Bucket-Selektor in allen Datensätzen bessere Ergebnisse erreichen als Greedy und Top-k. Der SPECTF-Datensatz ist insofern speziell, dass er 44 Dimensionen hat, während die anderen Datensätze maximal 10-dimensional sind. Bekanntermaßen ist die Euklidische Distanz für hochdimensionale Daten nicht gut geeignet, weshalb davon auszugehen ist, dass aus diesem Grund auch das parallele Lernen mehrerer Dendrogramme keine Verbesserung bringt.

k	Datens.	Bucket			Top-k			Greedy		
		μ	σ	Med.	μ	σ	Med.	μ	σ	Med.
100	Seeds	**0,783**	0,025	**0,784**	0,743	0,009	0,744	0,735	0,015	0,739
	Ruspini	**0,883**	0,014	**0,878**	0,876	0,004	0,876	0,875	0,006	0,875
	Iris	**0,855**	0,006	**0,855**	0,852	0,006	0,852	0,853	0,007	0,853
	U.K.M.	**0,583**	0,013	**0,584**	0,578	0,013	0,581	0,574	0,014	0,574
	B.C.	**0,863**	0,007	**0,864**	0,843	0,006	0,844	0,839	0,007	0,84
	SPECTF	0,94	0,004	0,94	0,94	0,004	0,94	**0,943**	0,006	**0,944**
	Cement	**0,706**	0,011	**0,706**	0,703	0,01	0,703	0,704	0,012	0,705
500	Seeds	**0,8**	0,017	**0,801**	0,744	0,008	0,744	0,736	0,012	0,738
	Ruspini	**0,9**	0,019	**0,896**	0,877	0,004	0,876	0,875	0,006	0,875
	Iris	**0,86**	0,006	**0,86**	0,851	0,005	0,852	0,852	0,006	0,852
	U.K.M.	**0,582**	0,01	**0,58**	0,579	0,013	0,579	0,577	0,012	0,576
	B.C.	**0,869**	0,005	**0,869**	0,844	0,006	0,845	0,839	0,008	0,84
	SPECTF	0,94	0,004	0,94	0,94	0,005	0,94	**0,942**	0,006	**0,943**
	Cement	**0,709**	0,011	**0,709**	0,703	0,012	0,702	0,703	0,012	0,702

Tab. 1: Versuchsergebnisse für den Top-k und den Hash-Bucket-Selektor mit $k = 100$ und $k = 500$. Durchschnitt (μ), Standardabweichung (σ) und Median ermittelt über 100 Durchläufe mit je zufälligen 90 % der Daten. Für den Greedy-Selektor gilt immer $k = 1$.

Abbildung 4 zeigt die Ergebnisse für den Breast-Cancer-Datensatz mit feineren Abstufungen für k. Zu sehen ist, dass die Top-k-Selektion zwar leicht besser ist als der Greedy-Algorithmus, von mehr gleichzeitig explorierten Modellen aber kaum profitiert. Der Bucket-Selektor dagegen erzielt mit höherem k auch bessere Ergebnisse.

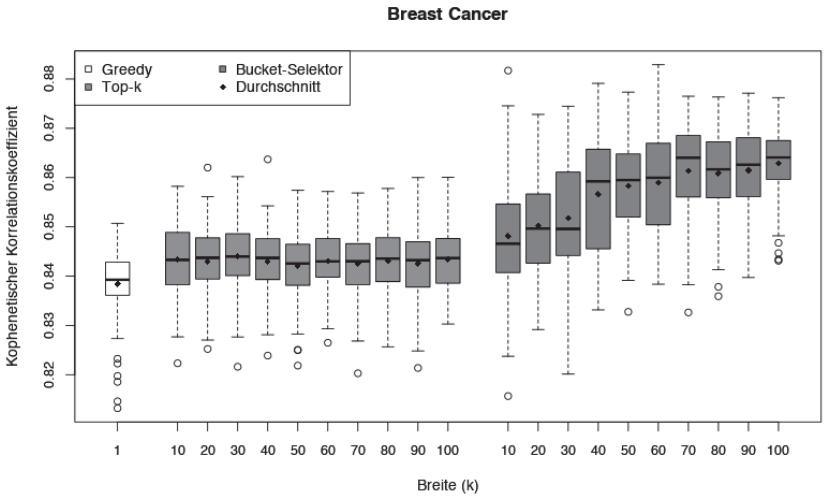


Abb. 4: Ergebnisse für das hierarchische Clustering des Breast-Cancer-Datensatzes und UPGMA-Linkage mit dem Greedy-, Top-*k*- und Bucket-Selektor.

5 Zusammenfassung

Widening betrachtet die lokale Suche als eine Abfolge von Verfeinerungs- und Selektionsschritten. In der Verfeinerung werden alle Nachbarn der aktuell betrachteten Modelle aufgelistet, in der Selektion werden einige dieser Nachbarn dann für die nächste Iteration ausgewählt. Während die Verfeinerungsoperation meist anwendungsabhängig ist und kein Optimierungspotential hat, spielt die Wahl eines passenden Selektors eine große Rolle für die Qualität der erzeugten Lösungen eines Algorithmus. Publizierte Selektoren sind beispielsweise Top-k, Diverse Top-k, K-Medoid und Score Erosion.

Der hier vorgestellte Hash-Bucket-Selektor ist ein Selektor für das Widening, der in jeder Iteration jede Verfeinerung der parallel verarbeiteten Modelle per Hashfunktion einer Partition zuweist und anschließend das beste Modell aus jeder Partition für die nächste Iteration auswählt. Dies garantiert einerseits, dass das global beste Modell immer ausgewählt wird, aber auch Modelle außerhalb der besten k eine Chance haben, weiterverfolgt zu werden. Gegenüber anderen Selektoren hat der Bucket-Selektor den Vorteil, dass Modelle nie zentral gesammelt werden müssen, sondern immer direkt zwischen den Recheneinheiten ausgetauscht werden können. Des Weiteren besitzt er eine geringere theoretische Laufzeitkomplexität als Top-k und benötigt ebensowenig Speicher, da die Verfeinerungen des Vorgängermodells nur einmal durchlaufen werden müssen. Im Gegensatz zu Selektoren wie K-Medoid oder Score Erosion benötigt der Bucket-Selektor keine Distanzmetrik für Modelle, da allein anhand der Hashfunktion bestimmt wird, welche Modelle anhand ihrer Qualität miteinander verglichen werden. Kann gar keine Hashfunktion für die Modelle eines Problems gefunden werden, so ist es auch möglich, die Hashfunktion durch eine Zufallsfunktion zu ersetzen und in vielen Fällen trotzdem gute Ergebnisse zu erzielen.

Literaturverzeichnis

[AIB12] Akbar, Zaenal; Ivanova, Violeta N.; Berthold, Michael R.: Parallel data mining revisited. Better, not faster. In: International Symposium on Intelligent Data Analysis. Springer, S. 23–34, 2012.

[Ak00] Akl, Selim G: Parallel real-time computation: Sometimes quantity means quality. In: Proceedings of the International Symposium on Parallel Architectures. IEEE, S. 2–11, 2000.

[BM11] Buluç, Aydin; Madduri, Kamesh: Parallel breadth-first search on distributed memory systems. In: Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis. ACM, S. 65, 2011.

[DB79] Davies, David L; Bouldin, Donald W: A cluster separation measure. IEEE transactions on pattern analysis and machine intelligence, PAMI-1(2):224–227, 1979.

[Eh05] Ehrgott, Matthias: Multicriteria optimization, Jgg. 491. Springer Science & Business Media, 2005.

[Fi19] Fillbrunn, Alexander: Effektives Widening mit Hashbasierter Partitionierung des Hypothesenraums. in press, University of Konstanz, 2019.

[IB13] Ivanova, Violeta N.; Berthold, Michael R.: Diversity-driven widening. In: International Symposium on Intelligent Data Analysis. Springer, S. 223–236, 2013.

[Me10] Meinl, Thorsten: Maximum-score diversity selection. Dissertation, Universität Konstanz, 2010.

[Mi79] Milligan, Glenn W: Ultrametric hierarchical clustering algorithms. Psychometrika, 44(3):343–346, 1979.

[RK90] Rousseeuw, Peter J; Kaufman, L: Finding groups in data. Series in Probability&Mathematical Statistics, 34(1):111–112, 1990.

[Ro87] Rousseeuw, Peter J: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics, 20:53–65, 1987.

[SR62] Sokal, Robert R.; Rohlf, F. James: The Comparison of Dendrograms by Objective Methods. Taxon, 11(2):33–40, 1962.

Alexander Fillbrunn wurde am 28. Juni 1989 in Mannheim-Neckarau geboren. Nach seinem Abitur und einem einjährigen Freiwilligendienst in Malawi erhielt er seinen BSc in Angewandter Informatik an der Dualen Hochschule Baden-Württemberg in Mannheim. Während einiger Praktika, unter anderem am IBM Almaden Research Center in San Jose, USA und der IBM Forschungsabteilung in Böblingen entdeckte er seine Begeisterung für Data Mining und maschinelles Lernen. Seinen Masterabschluss machte Alexander Fillbrunn 2014 an der Universität Konstanz am Lehrstuhl für Bioinformatik und Information Mining, wo er anschließend für seine Doktorarbeit blieb. Heute arbeitet er als Solution Engineer bei der KNIME GmbH, einer Ausgründung der Universität Konstanz.

Automatisierte Analyse virtueller Prototypen auf der elektronischen Systemebene –Entwurfsverständnis und Anwendungen–[1]

Mehran Goli[2]

Abstract: Die Modellierung elektronischer Systeme mit Hilfe *Virtueller Prototypen* (VP) auf der Ebene elektronischer Systeme ist in den letzten zehn Jahren zu einer branchenweit akzeptierten Lösung geworden. Der Hauptgrund ist, dass VPs viel früher verfügbar sind und ihre Simulation im Vergleich zu Hardwaremodellen, die auf niedrigeren Abstraktionsebenen implementiert sind, um Größenordnungen schneller ist. Daher wird der VP als Referenzmodell im Entwurfsprozess für verschiedene Anwendungen (z. B. Testen, Fehlerbeseitigung, Verifikation, Sicherheitsüberprüfung und Entwurfsraumexploration) verwendet. Es ist jedoch eine genaue Kenntnis der VP-Struktur und des VP-Verhaltens erforderlich, um für die genannten Anwendungen ein geeignetes Referenzmodell erstellen zu können. Dies ist eine sehr herausfordernde Aufgabe. Ziel dieser Dissertation ist es, eine Reihe umfassender und automatisierter Ansätze bereitzustellen, mit denen Designer verschiedene Aufgaben im Entwurfsprozess bewältigen können.

1 Einführung

Moderne eingebettete Systeme bestehen aus vielen verschiedenen Funktionsblöcken. Dazu gehören mehrere IP-Kerne (engl. Intellectual Property) von Drittanbietern sowie verschiedene Chip-interne Verbindungen und Speicher. Gegenwärtig werden diese modernen eingebetteten Systeme (z. B. ein 28-Kern Intel Xeon Platinum-Chip) als Multiprozessorsysteme auf einem einzelnen Chip hergestellt und erreichen bis zu 8 Milliarden Transistoren [Te]. Die exponentielle Erhöhung der Funktionalität von *System-on-Chips* (SoCs) und die Reduzierung der *Time-to-Market*-Einschränkungen haben den Entwurfsprozess erheblich verändert, um der hohen Marktnachfrage gerecht zu werden.

Eine Möglichkeit, mit der steigenden Komplexität elektronischer Systeme umzugehen, besteht darin, die Abstraktionsebene in Richtung der elektronischen Systemebene (ESL) anzuheben [GSD20]. Bei der ESL ist SystemC unter den vorhandenen Hardwaremodellierungssprachen zur De-facto-Standardmodellierungssprache geworden, mit der eingebettete Systeme als Virtual Prototype (VP) beschrieben werden [OS06]. Ein VP ist ein abstraktes und ausführbares Softwaremodell, das normalerweise mit SystemC und seinem TLM-Framework (engl. Transaction Level Modeling) implementiert wird. Auf diese Weise kann ein System schnell als Prototyp erstellt und als Referenzmodell für niedrigere Abstraktionsebenen verwendet werden. Darüber hinaus können Entwickler vor der Herstellung der eigentlichen Hardware die Software eines eingebetteten Systems testen oder

[1] Englischer Titel der Dissertation: "Automated Analysis of Virtual Prototypes at the Electronic System Level– Design Understanding and Applications–"

[2] Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI), mehran.goli@dfki.de

CC BY SA

bewerten. Somit fungiert es als Standard-Kommunikationsplattform zwischen Entwicklern eingebetteter Software und Hardware-Ingenieuren während des Entwurfablaufs.

Um dieses Referenzmodell (d. h. den VP) zu generieren, müssen Designer möglicherweise mehrere Schritte während des Entwurfsprozesses ausführen, z. B. das Testen neuer Funktionen und das Validieren der Funktionen von SoCs, das Durchführen einer Entwurfsraumexploration für die nächste Generation von SoCs, das Debuggen, Überprüfen oder Synthetisieren der vorhandenen SoCs. Um die oben genannten Aufgaben zu bewältigen, ist eine genaue Analyse der Struktur und des Verhaltens der VPs erforderlich. Dieser erste Schritt im Entwurfsprozess wird als "Entwurfsverständnis" bezeichnet. Es ist jedoch keine einfache Aufgabe, einen bestimmten SystemC-basierten VP auf das Ziel des Entwurfsverständnisses hin zu analysieren. Der Hauptgrund ist, dass C++ von Natur aus schwer zu analysieren ist, aufgrund 1) des Mangels an geeigneten Analysemethoden für das Laufzeitverhalten von C++, 2) der unzähligen Compiler-spezifischen Dialekte, in denen ein Quellcode geschrieben werden kann, und 3) des ausführbaren Binärformats, das möglicherweise stark optimiert ist.

In dieser Dissertation wird eine neuartige Methode für das Entwurfsverständnis vorgestellt, mit der Designer die Struktur und das Verhalten von SystemC-basierten VPs genau nachvollziehen können. Die vorgeschlagene Methodik besteht aus drei Hauptphasen: Datenextraktion, Datenanalyse und Visualisierung. In der ersten Phase werden die statischen Informationen und Laufzeitinformationen von VPs aus zwei Perspektiven extrahiert: Dem Debugger-basierten und dem Compiler-basierten Ansatz. In der zweiten Phase werden die extrahierten Informationen in eine Reihe von Zwischensprachen (IR-Modelle) übersetzt, die das Verhalten und die Struktur des gesamten Systems in gut strukturierten Formaten darstellen. Um den Designern zu helfen, die Komplexität der VPs besser zu verstehen, werden diese IR-Modelle in der letzten Phase verwendet, um die Struktur des VPs in einem XML-Format und sein Verhalten als UML-Diagramme (engl. Unified Modeling Language, kurz UML) zu visualisieren. Der zweite Teil dieser Arbeit fokussiert sich auf Anwendungen, die die Hauptvorteile einer starken Entwurfsverständnis-Methodik zeigen. Die Ergebnisse des ersten Teils können als Ausgangspunkt für verschiedene Anwendungen im Entwurfsprozess dienen. In diesem Zusammenhang wurden verschiedene Anwendungen implementiert, um zu zeigen, wie die extrahierten Informationen in der Praxis mit Standardtechnologien oder mit minimalem Übersetzungsaufwand in die gewünschten Formate verwendet werden können [Go19a].

Der Rest dieser Arbeit ist wie folgt gegliedert: Zunächst wird in Abschnitt 2 die vorgeschlagene Methode zum Entwurfsverständnis erläutert. Darauf aufbauend werden in den Abschnitten 3, 4 und 5 drei wichtige Anwendungen des Entwurfsverständnisses vorgestellt. Hierbei geht es insbesondere um die Verifikation, Sicherheitsprüfung und Entwurfsraumexploration von VP-basierten SoCs. Abschließend wird die Arbeit in Abschnitt 6 zusammengefasst.

2 Methodik zum Entwurfsverständnis

In diesem Abschnitt werden die drei Phasen der vorgeschlagenen Methodik erläutert.

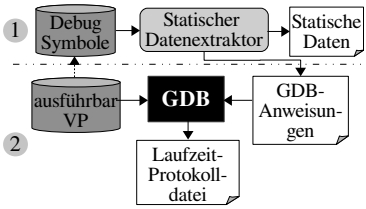


Abb. 1: der Debugger-basierte Ansatz

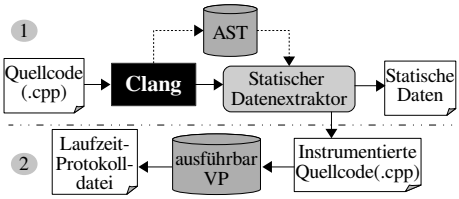


Abb. 2: der Compiler-basierte Ansatz

2.1 Datenextraktion

Die erste Phase der vorgeschlagenen Entwurfsverständnismethodik besteht darin, auf die statischen Informationen und Laufzeitinformationen von VPs zuzugreifen, die ihre *Struktur* und ihr *Verhalten* beschreiben (was in Bezug auf die Transaktion definiert ist). Die Struktur der VPs bezieht sich auf den Aufbau, der im Quellcode der VPs beschrieben ist, z. B. den Modultyp und die Bindungsinformationen der Modul-Sockets. Das Verhalten der VPs bezieht sich auf die Laufzeitinformationen der abstrakten Kommunikation zwischen den verschiedenen IP-Kernen. Hierfür ist der Zugriff auf drei wesentliche Elemente von Transaktionen während der Ausführungszeit notwendig: Fluss, Daten und Typ. Daher schlagen wir zwei Ansätze vor: Die Debugger- und Compiler-basierten Ansätze.

Debugger-basierter Ansatz: Wie in Abb. 1 dargestellt, besteht die Hauptidee des vorgeschlagenen Ansatzes zum Extrahieren der Struktur und des Simulationsverhaltens eines bestimmten SystemC-basierten VPs aus zwei Hauptphasen. Zunächst werden die statischen Informationen des kompilierten Modells abgerufen, indem die Debug-Symbole analysiert werden, um zwei Ziele zu erreichen: 1) Identifizierung aller Komponenten und ihrer Attribute und Elementfunktionen, die zur Beschreibung der Struktur des Modells erforderlich sind. 2) Automatisches Generieren eines Satzes von GDB-Anweisungen, die im nächsten Schritt verwendet werden, um die Struktur des Modells (dynamische Daten) zu extrahieren und sein Verhalten zu analysieren. In der zweiten Phase wird das Modell unter der Kontrolle von GDB unter Verwendung der zuvor erzeugten Anweisungen ausgeführt. Die Ausführung des Modells wird bei bestimmten Ereignissen (z. B. Funktionsaufrufen) angehalten, um die Laufzeitinformationen aufzuzeichnen [GSD16, GSD20, Le18].

Compiler-basierter Ansatz: Abb. 2 gibt einen Überblick über den entwickelten Ansatz, der aus zwei Teilen besteht. Zunächst wird der *Abstract Syntax Tree* (AST) eines gegebenen VPs analysiert, um die statischen Informationen des Modells zu extrahieren, die zur Beschreibung der Struktur des Entwurfs erforderlich sind. Zweitens wird eine instrumentierte Version des VP-Quellcodes unter Verwendung der extrahierten statischen Informationen aus der vorherigen Phase generiert, um die Laufzeitdaten (d. h. das Verhalten) zu extrahieren. Dies wird durchgeführt, indem der instrumentierte Quellcode automatisch mit einem Standard C++ Compiler (z. B. GCC oder Clang) kompiliert und ausgeführt wird, um die Laufzeitinformationen zu protokollieren [Go19b].

2.2 Datenanalyse

Die generierte Laufzeitprotokolldatei aus der vorherigen Phase enthält ungeordnete Informationen zum Verhalten der VPs. Um das Verhalten jeder einzelnen Transaktion darzustellen, muss diese große Datenmenge in kleinere Mengen unterteilt werden, die sich jeweils

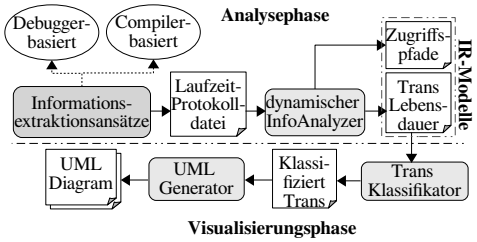


Abb. 3: Datenanalyse und Visualisierung

SQ1 : ([Initiator_A, init_0, gen_frame, 40ns, NULL, initiator], [0x451, 0x09, READ, 4, TLM_INCOMPLETE_RES, 5ns, NULL])

SQ2 : ([LT_AT_BUS, bus0, b_transport, 45ns, 0x6e47f0, interconnect], [0x451, 0x09, READ, 4, TLM_INCOMPLETE_RES, 5ns, NULL])

SQ3 : ([Memory_A, trgA, b_transport, 50ns, 0x6e47f0, target], [0x561, 0x04, READ, 4, TLM_OK_RES, 5ns, TLM_COMPLETED])

Abb. 4: Eine einzelne Transaktionslebensdauer (3 von 5 Sequenzen)

auf eine einzelne Transaktion beziehen. Ziel dieses Abschnitts ist es daher, eine Analyse der extrahierten Laufzeitdaten nach der Ausführung vorzuschlagen, um eine Zwischendarstellung des Simulationsverhaltens des VPs zu erstellen. Daher werden zwei Zwischendarstellungen, die Transaktionslebensdauer und des Transaktionszugriffspfad, vorgeschlagen.

Transaktionslebensdauer: Die Transaktionslebensdauer bezieht sich auf die Zeitspanne vom Erstellen einer Transaktion bis zu ihrer Beendigung. Dies erfordert, dass für jede einzelne Transaktion die entsprechenden Informationen von anderen Transaktionen isoliert sind. Die Grundlage für diese Isolation bildet die Transaktionsreferenzadresse. Die Referenzadresse eines Transaktionsobjektes bleibt während dessen Lebensdauer konstant. Wie in Abb. 3 gezeigt, erhält das *dynamischer Info Analyzer*-Modul das Laufzeitprotokoll als Eingabe. Es extrahiert die erforderlichen Informationen für jede einzelne Transaktion, um die Aktivität innerhalb ihrer Lebensdauer zu beschreiben, und speichert sie in der *Trans-Lifetime*-Datei. Diese Daten erlauben eine genaue Nachverfolgung des Verhaltens jeder Transaktion und decken alle Änderungen der Transaktionsdaten ab, die während der Ausführung des Modells aufgetreten sind. Eine Transaktionslebensdauer besteht aus vielen Sequenzen *SQ*. Jede *SQ* ist wie folgt definiert:

$$SQ = \{(F,D) \mid F = (M, Func, ST, TID, Type),\ D = (data, adrs, cmd, phase, delay)\} \quad (1)$$

Abb. 4 zeigt beispielhaft eine Teiltransaktionslebensdauer, die durch den vorgeschlagenen Ansatz erzeugt wird. Sie enthält drei Sequenzen (Zeitschritte), welche die Transaktionserstellung, die Bearbeitung durch TLM-Module und deren Abschluss veranschaulichen.

Transaktionszugriffspfad: Der Transaktionszugriffspfad ist eine abstrakte Darstellung des gesamten Verhaltens eines bestimmten VPs, mit der Designer die Beziehung verschiedener IP-Kerne in einem bestimmten System in einer Gesamtansicht erkennen können. Ein vollständiges Simulationsverhalten eines bestimmten VPs kann als ein Satz von Zugriffspfaden S_{AP} definiert werden, wobei jeder Pfad *AP* eine Verbindung zwischen einem Initiatormodul *IM* und einem Zielmodul *TM* ist:

$$S_{AP} = \{AP_i \mid AP_i = \{IM \rightarrow TM :: (TID, TT, Tadrs, cmd, TD)\},\ 1 \leq i \leq n_{seq}\} \quad (2)$$

Der Parameter *TT* ist der Transaktionstyp, der veranschaulicht, welches Timing-Modell (*LT* oder *AT*) verwendet wird. *Tadrs* zeigt die Adresse der Transaktion im Zielmodul *TM* an. Der Parameter *cmd* ist das Transaktionsbefehlsattribut (z. B. *Lesen* oder *Schreiben*). Der Parameter *TD* ist die Gesamtverzögerung aller Sequenzen innerhalb der Transaktionslebensdauer. Schließlich ist n_{seq} die Anzahl der Sequenzen in einer Transaktionslebens-

dauer. Zum Beispiel ist die Zugriffspfaddarstellung der Transaktion in Abb. 4:

$$AP = \{Initiator_A : initA \rightarrow Target_A : trgA :: (0x76ab561, T_1, 0x04, READ, 20ns)\} \quad (3)$$

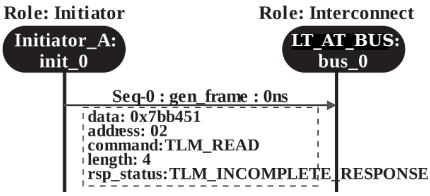


Abb. 5: eine einzelne seq in der Transaktionslebensdauer

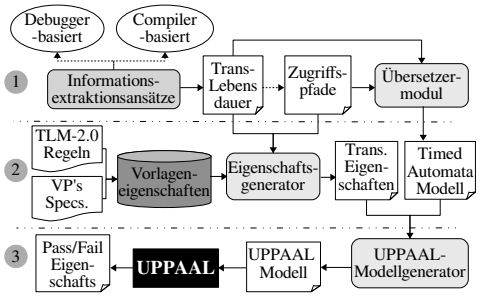


Abb. 6: Überblick über den Verifikationsansatz

2.3 Visualisierung

Da es möglich ist, dass viele der extrahierten Transaktionen in der Trans-Lifetime-Datei denselben Ablauf und denselben Typ haben (sodass sich nur ihre Daten unterscheiden), ist ein weiterer Analyseschritt erforderlich, um nur diejenigen zu visualisieren, die ein eindeutiges Verhalten aufweisen. Dadurch wird die Anzahl der generierten UML-Diagramme effektiv reduziert, sodass Entwickler das Verhalten eines bestimmten VP-basierten eingebetteten Systems schnell verstehen können. Das UML-Diagramm wird vom UML-Generator-Modul generiert (Abb. 3). Das UML-Diagramm enthält eine Reihe von Folgenummern, die sowohl den Transaktionsfluss als auch die Transaktionsdaten innerhalb seiner Lebensdauer angeben. Abb. 5 zeigt das UML-Diagramm einer extrahierten Transaktion. Die schwarzen Formen geben den Stamm- und Instanznamen der Module im Entwurf an.

2.4 Experimentelle Ergebnisse und Diskussion

Die vorgeschlagene Methodik wird auf mehrere von Doulos bereitgestellte Standard-VPs [Ay] und auf das LEON3-basierte VP SoCRocket [Scet] angewendet. Tabelle 1 zeigt die extrahierten Daten, einschließlich der Anzahl der Transaktionen (*#Trans*), der eindeutigen Flüsse (*#UFlow*), eindeutigen Typen (*#UType*) und generierten UML-Diagramme (*#UML*). Die Menge der extrahierten Daten für Debugger- und Compiler-basierte Ansätze ist gleich. Das bedeutet, dass beide Ansätze die gleiche Genauigkeit aufweisen. Tabelle 2 zeigt detailliert die erforderliche Analysezeit für diese Ansätze und vergleicht sie mit der reinen Ausführungs- und Kompilierungszeit (*CET*) jedes VPs.

Der Debugger-basierte Ansatz erfordert nur die ausführbare Version des VPs, sodass der ursprüngliche Quellcode und der Workflow (z. B. die SystemC-Bibliothek oder der Compiler) nicht verändert werden. Bei IPs von Drittanbietern, bei denen der Quellcode möglicherweise überhaupt nicht verfügbar ist, ist dieser Ansatz die einzig anwendbare Lösung. Andererseits ist der Compiler-basierte Ansatz sehr schnell und skaliert auch bei beliebiger Komplexität von VPs gut. Voraussetzung ist jedoch die Verfügbarkeit des originalen VP-Quellcodes. Je nach Bedenken und Anforderungen der Designer besteht jeweils die Möglichkeit zwischen dem Debugger-basierten oder dem Compiler-basierten Ansatz zu wählen.

Tab. 1: Experimentelle Ergebnisse bezogen auf die Menge der extrahierten Daten für alle VPs

VP Modell	LoC	#Comps	TM	#Trans	Compiler-basierter Ansatz			Debugger-basierter Ansatz		
					#UFlow	#UType	#UML	#UFlow	#UType	#UML
Routing-model[1]	456	6	LT	100	4	1	4	4	1	4
AT-example[1]	3,410	19	AT	49	12	9	14	12	9	14
SoCRocket[2]	50,000	20	LT/AT	1,000	19	8	21	19	8	21

LoC: Codezeilen **#Comp:** Anzahl der Komponenten **TM**: Timing-Modell **#Trans**: Anzahl der Transaktionen **#UFlow**: Anzahl der eindeutigen Transaktionsflüsse **#UType**: Anzahl der eindeutigen TransaktionsTypen **#UML**: Anzahl der generieren UML-Diagramm

Tab. 2: Experimentelle Ergebnisse bezogen auf die erforderliche Analysezeit für alle VPs

VP Modell†	LoC	#Comps	TM	#Trans	Compiler-basierter Ansatz (s)				Debugger-basierter Ansatz (s)				CET (s)		
					Phase1	Phase2*	AVT	Total	Phase1*	Phase2	AVT	Total	Cmp	Exe	Total
Routing-model[1]	456	6	LT	100	2.11	0.12	0.31	2.54	2.76	207.29	2.81	212.86	1.42	0.01	1.53
AT-example[1]	3,410	19	AT	49	19.05	0.24	0.39	19.68	22.41	1,591.07	4.96	1,618.44	20.03	0.34	17.19
SoCRocket[2]	50,000	20	LT/AT	1,000	52.82	1.21	1.63	55.66	146.39	17,446.19	29.12	17,621.70	26.72	1.08	27.80

LoC: Codezeilen **#Comp:** Anzahl der Komponenten **TM**: Timing-Modell **#Trans**: Anzahl der Transaktionen **AVT**: Analyse- und Visualisierungszeit

Vorteile: Durch die Einführung dieser Entwurfsverständnismethode können die folgenden Nachteile der vorherigen Methoden [BVL11] wirksam überwunden werden: 1) niedriger Automatisierungsgrad, 2) benutzerdefinierte Code-Annotationen oder Sprachkonstrukte, 3) fehlende Extraktion des Laufzeitverhaltens von VPs, und 4) das Fehlen unterstützender TLM-Konstrukte.

3 Anwendung I: VP-Verifikation

Die wesentlich frühere Verfügbarkeit und die im Vergleich zum RTL-Hardwaremodell deutlich schnellere Simulationsgeschwindigkeit des VPs sind die Hauptgründe, den VP als Referenzmodell für eine frühe Systemverifikation im Entwurfsprozess heranzuziehen. Daher ist die Sicherstellung der Richtigkeit von VPs von größter Bedeutung, da sich unentdeckte Fehler auf niedrigeren Abstraktionsebenen ausbreiten und in der Behebung sehr kostspielig werden können. In diesem Abschnitt wird am Beispiel der Verifikation von SystemC-basierten VPs eine Anwendung des Entwurfsverständnisses schon während des Entwurfsprozesses gezeigt. Ein semi-formaler Verifikationsansatz wird vorgeschlagen, um einen VP anhand seiner Spezifikationen und einer Reihe Protokolle zu verifizieren.

3.1 Verifikationsmethodik

Abb. 6 gibt einen Überblick über den vorgeschlagenen Ansatz, der die folgenden drei Hauptphasen umfasst. Zuerst wird das Laufzeitverhalten (d. h. die Lebensdauer von Transaktionen) des SystemC-VPs extrahiert. Diese extrahierten Informationen werden analysiert, um sie in einen Satz zeitgesteuerter Automatenmodelle umzuwandeln. Die Verhaltensextraktion des VPs in der ersten Phase des Verifikationsansatzes wird unter Verwendung der in Abschnitt 2 vorgeschlagenen Informationsextraktionsansätze durchgeführt. In der zweiten Phase werden in Bezug auf das Funktions- und Zeitverhalten automatisch eine Reihe von Eigenschaften aus den TLM-2.0-basierten Protokollregeln und VP-Spezifikationen generiert. Abschließend wird das Verhalten des VPs anhand der generierten Eigenschaften mit dem UPPAAL-Modellprüfprogramm überprüft [GSD17a, GSD17b, GD19b].

3.2 Experimentelle Ergebnisse und Diskussion

Die experimentellen Ergebnisse der verschiedenen ESL-Benchmarks sind in Tabelle 3 aufgeführt. Die erste Spalte enthält vier Varianten von SystemC-VPs mit den Bezeichnungen

Tab. 3: Experimentelle Ergebnisse des vorgeschlagenen Verifikationsansatzes für alle VPs

Variante	VP Modell	LoC	#Comp	#Trans	#TT	TM	#Eigenschaften			FTrans	R/VMU (MB)
							Total	Pass	Fail		
Original	Routing-model[1]	456	6	10	1	LT	164	164	0	0	6.7/41.5
	AT-example[1]	3,410	19	20	9	AT	1,184	1,184	0	0	7.8/43.2
	SoCRocket[2]	50,000	20	200	8	LT/AT	9,462	9,462	0	0	7.8/43.2
FT1	Routing-model[1]	456	6	10	1	LT	120	112	8	4	6.7/41.5
	AT-example[1]	3,410	19	20	9	AT	1,060	924	142	7	7.8/43.2
	SoCRocket[2]	50,000	20	200	8	LT/AT	8,480	7,806	674	37	7.8/43.2
FT2	Routing-model[1]	456	6	10	1	LT	24	18	6	6	6.7/41.5
	AT-example[1]	3,410	19	20	9	AT	84	69	15	8	7.8/43.2
	SoCRocket[2]	50,000	20	200	8	LT/AT	612	536	76	42	7.8/43.2
FT3	Routing-model[1]	456	6	10	1	LT	20	15	5	5	6.7/41.5
	AT-example[1]	3,410	19	20	9	AT	40	30	10	10	6.7/41.5
	SoCRocket[2]	50,000	20	200	8	LT/AT	370	311	59	59	7.8/43.2

1 und **2** bereitgestellt von [Ay] und [Scet] **LoC:** Codezeilen **#Comp:** Anzahl der Komponenten **#Trans:** Anzahl der Transaktionen **#TT:** Anzahl der Transaktionstypen **TM:** Timing-Modell **FTrans:** Anzahl der fehlerhaften Transaktionen **R/VMU:** Resident/Virtual Memory Usage Peaks

Original, *FT1* (Fehlertyp bezogen auf die Protokollregeln), *FT2* (Fehlertyp bezogen auf das Funktionsverhalten) und *FT3* (Fehlertyp bezogen auf die Zeitbeschränkung), die sich auf das Referenzmodell des VPs bzw. auf drei fehlerhafte Modelle beziehen. Die Spalte *VP Modell* gibt die Namen der VPs an, die Spalten *LoC* und *#Trans* enhalten die Codezeilen und Anzahl der extrahierten Transaktionen für jeden VP. Die Spalten mit der Überschrift *#Queries* zeigen die Anzahl der generierten Abfragen (Eigenschaften) an, um jeden VP gegen die TLM-2.0-Regeln oder seine Spezifikationen zu validieren. Dabei geben *Total*, *Pass*, *Fail* und *FTrans* die Anzahl der generierten, erfüllten und verletzten Abfragen bzw. die Anzahl der fehlerhaften Transaktionen an.

Vorteile: Durch die Einführung dieses semi-formalen Verifizierungsansatzes können die folgenden Nachteile der vorherigen Verfahren wirksam überwunden werden [He18]: 1) fehlende Überprüfung der VPs bezüglich ihrer Spezifikationen, 2) Explosion des Zustandsraums und Einschränkung der Modellierung der VPs in der formalen Semantik (für formale Methoden), 3) fehlende Unterstützung eines breiten Spektrums von SystemC-VPs, und 4) geringer Automatisierungsgrad.

4 Anwendung II: VP-Sicherheitsüberprüfung

Moderne SoCs sind notorisch unsicher. Da unsichere Hardware die Tür für Angriffe auf jede Ebene eines Systems (von der Software bis zur Hardware) öffnet, sollte der Sicherheitsüberprüfungsprozess so früh wie möglich im SoC-Entwurfszyklus beginnen, d. h. auf der ESL. Dieser Abschnitt zeigt eine weitere Anwendung des Entwurfsverständnisses im Entwurfsprozess für die Aufgabe der Sicherheitsüberprüfung von VP-basierten SoCs auf der ESL. Der Schwerpunkt des vorgeschlagenen Ansatzes liegt auf der Erkennung von Sicherheitsverletzungen im Zusammenhang mit den am häufigsten vorkommenden Bedrohungsmodellen Vertraulichkeit und Integrität [Go19c].

4.1 Sicherheitsüberprüfungsmethodik

Abb. 7 gibt einen Überblick über den vorgeschlagenen Ansatz, der aus den folgenden vier Hauptphasen besteht. Zuerst werden Transaktionen eines bestimmten VP-Modells

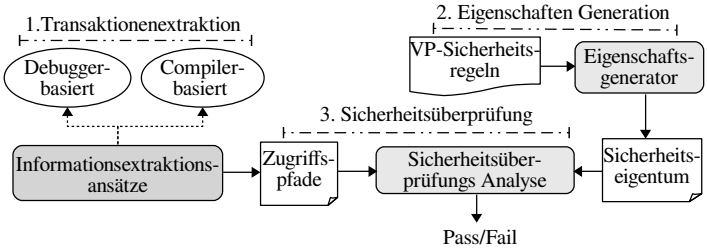


Abb. 7: Überblick über den Ansatz der Sicherheitsüberprüfung

Tab. 4: Experimentelle Ergebnisse zur Qualität des Sicherheitsüberprüfungsansatzes für alle VPs

VP Modell*	LoC	TM	SSc	#IP			#Trans	#ISP			#DSP		
				Total	SI	SM		Total	Pass	Fail	Total	Pass	Fail
AES128-SoC[1]	4,742	AT	S1,S2	10	2	3	350	33	31	2	13	8	5
RISC32-SoC[1,2]	4,850	AT	S1	19	3	3	733	324	304	20	36	34	2
SoCRocket[3,4]	50,000	LT/AT	S2	21	1	3	1,100	180	168	12	30	28	2

*Die VP-Modelle wurden bereit gestellt von [1][Ay], [2]OSCI [OS06] und [3][Scet] [4][SM14]. **LOC**: Codezeilen **TM**: Timing-Modell **SSc**: Sicherheitsszenario **#IP**: Anzahl der Intellectual Properties **SI**: Sicherer Initiator **SM**: Sicherer Speicher **#Trans**: number of Transaktionen **ISP**: Indirekte Sicherheitseigenschaft **DSP**: Direkte Sicherheitseigenschaft

während der Simulationszeit extrahiert und in einen Satz von Zugriffspfaden übersetzt. Die Verhaltensextraktion des VPs in der ersten Phase des Verifikationsansatzes wird unter Verwendung der in Abschnitt 2 vorgeschlagenen Informationsextraktionsansätze durchgeführt. Anschließend werden die Sicherheitsregeln in Eigenschaften übersetzt. Zuletzt kann das übersetzte Modell des VPs gegen die Sicherheitseigenschaften validiert werden.

4.2 Experimentelle Ergebnisse und Diskussion

Die experimentellen Ergebnisse für verschiedene Arten von SoC-Benchmarks sind in Tabelle 4 aufgeführt. Um die Qualität des vorgeschlagenen Ansatzes zu bewerten, betrachten wir zwei mögliche Sicherheitsszenarien (Spalte *SSc*), die sowohl Vertraulichkeits- als auch Integritätsbedrohungsmodelle abdecken. S1: Ändern eines SoC ohne Aktualisierung seiner Sicherheitsrichtlinien. S2: Falsche Initialisierung oder Aktualisierung der SoC-Firmware. Mit dem vorgeschlagenen Ansatz kann die Sicherheit der abstrakten Kommunikation für ein bestimmtes VP-basiertes SoC-Modell überprüft werden. Das Ergebnis unserer Analyse kann den Überprüfungsingenieur effektiv bei der Verbesserung der Sicherheitsrichtlinien des Entwurfs unterstützen, da die anfälligen Pfade genau angezeigt werden.

Vorteile: Der vorgeschlagene Sicherheitsüberprüfungsansatz ist *der erste seiner Art* auf der ESL, der die Nachteile der vorhandenen statischen Sicherheitsüberprüfungsanalyse wirksam beseitigen kann [Ha17]. Wenn die Adressen von Transaktionen zur Laufzeit definiert werden, kann diese Art von Sicherheitsverletzung mit vorhandenen statischen Sicherheitsüberprüfungsanalysen nicht erkannt werden.

5 Anwendung III: KI für VP Entwurfsraumexploration

Als letzte Anwendung wurde die Informationsextraktionsanalyse verwendet, um die approximierbaren Teile des ESL-Designs automatisch zu erkennen. Eine Reihe neuer Ansät-

ze wurde vorgestellt, die auf einer Kombination aus statischen und dynamischen Analysemethoden sowie maschinellen Lerntechniken basieren. Es wurde gezeigt, wie das extrahierte Simulationsverhalten der einzelnen VPs in einen Satz von Trainingsdaten übersetzt und als Eingabe für maschinelle Lerntechniken verwendet werden kann [GSD18, GD19a].

Vorteile: Das erzeugte Vorhersagemodell wurde als Ausgangspunkt für die Bewertung der Fehlertoleranz verwendet, um den Anteil von fälschlicherweise positiven Ergebnissen beim Auffinden der fehlertoleranten Teile zu reduzieren. Die vorgeschlagenen Ansätze können den Designern effektiv helfen, festzustellen, unter welchen Fehlergrenzen unterschiedliche Teile eines gegebenen SystemC-Entwurfs bei unterschiedlichen Granularitätsniveaus (d. h. von einer einzelnen Anweisung zu einem Modul) approximiert werden können. Darüber hinaus wurde eine Approximationsgradanalyse durchgeführt, um die maximale Fehlerrate zu bestimmen, die eine approximierbare Aussage akzeptieren kann. Anschließend wird den Designern die maximale Anzahl von Anweisungen und Modulen mitgeteilt, die gleichzeitig approximiert werden können.

6 Zusammenfassung

Diese Arbeit stellt eine Reihe von SystemC-basierten VPs-Analyseansätzen vor, mit denen Designer verschiedene Aufgaben im Entwurfsprozess auf der ESL ausführen können, wie z.B. Entwurfsverständnis, Verifikation und Sicherheitsüberprüfung. Die vorgeschlagenen Ansätze sind effizient, automatisiert und verbessern den aktuellen Stand der VP-Analyse auf der ESL erheblich. Sie können in Verbindung mit einem vorhandenen Framework oder einem beliebigen SystemC-Setup verwendet werden.

Literaturverzeichnis

[Ay] Aynsley, J.: , TLM-2.0 Base Protocol Checker. https://www.doulos.com/knowhow/systemc/tlm2. Accessed: 2018-01-30.

[BVL11] Broeders, Harry; Van Leuken, René: Extracting behavior and dynamically generated hierarchy from SystemC models. In: Design Automation Conference (DAC). ACM, S. 357–362, 2011.

[GD19a] Goli, Mehran; Drechsler, Rolf: Integrating Hybrid Analysis with Machine Learning Techniques for Portion Resilience Evaluation in Approximating SystemC-based Designs. In: Workshop on Machine Learning for CAD (MLCAD). 2019.

[GD19b] Goli, Mehran; Drechsler, Rolf: Scalable Simulation-based Verification of SystemC-based Virtual Prototypes. In: EUROMICRO Digital System Design Conference (DSD). S. 522–529, 2019.

[Go19a] Goli, Mehran: Automated analysis of virtual prototypes at the electronic system level: design understanding and applications. Dissertation, Universität Bremen, Bremen, 2019. ix, 175 Seiten : Illustrationen, https://suche.suub.uni-bremen.de/peid=B168544980.

[Go19b] Goli, Mehran; Hassan, Muhammad; Große, Daniel; Drechsler, Rolf: Automated Analysis of Virtual Prototypes at Electronic System Level. In: Great Lakes Symposium on VLSI (GLSVLSI). S. 307–310, 2019.

[Go19c] Goli, Mehran; Hassan, Muhammad; Große, Daniel; Drechsler, Rolf: Security validation of VP-based SoCs using dynamic information flow tracking. it - Information Technology, 61(1):45–58, 2019.

[GSD16] Goli, Mehran; Stoppe, Jannis; Drechsler, Rolf: AIBA: an Automated Intra-Cycle Behavioral Analysis for SystemC-based Design Exploration. In: IEEE International Conference on Computer Design (ICCD). S. 360–363, 2016.

[GSD17a] Goli, Mehran; Stoppe, Jannis; Drechsler, Rolf: Automatic Equivalence Checking for SystemC-TLM 2.0 Models Against their Formal Specifications. In: Design, Automation and Test in Europe (DATE). S. 630–633, 2017.

[GSD17b] Goli, Mehran; Stoppe, Jannis; Drechsler, Rolf: Automatic Protocol Compliance Checking of SystemC TLM-2.0 Simulation Behavior Using Timed Automata. In: IEEE International Conference on Computer Design (ICCD). 2017.

[GSD18] Goli, Mehran; Stoppe, Jannis; Drechsler, Rolf: Resilience Evaluation for Approximating SystemC Designs Using Machine Learning Techniques. In: 2018 International Symposium on Rapid System Prototyping (RSP). S. 97–103, Oct 2018.

[GSD20] Goli, Mehran; Stoppe, Jannis; Drechsler, Rolf: Automated Nonintrusive Analysis of Electronic System Level Designs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD), 39(2):492–505, Feb 2020.

[Ha17] Hassan, M.; Herdt, V.; Le, H. M.; Große, D.; Drechsler, R.: Early SoC security validation by VP-based static information flow analysis. In: IEEE/ACM International Conference on Computer Aided Design (ICCAD). S. 400–407, 2017.

[He18] Herdt, V.; Le, H. M.; Große, D.; Drechsler, R.: Verifying SystemC using Intermediate Verification Language and Stateful Symbolic Simulation. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD), 2018.

[Le18] Lemma, David; Goli, Mehran; Große, Daniel; Drechsler, Rolf: Power Intent from Initial ESL Prototypes: Extracting Power Management Parameters. In: IEEE Nordic Circuits and Systems Conference (NORCAS). S. 1–6, 2018.

[OS06] OSCI: IEEE Standard SystemC Language Reference Manual. IEEE Std 1666-2005, S. 1–423, 2006.

[Scet] Schuster, Thomas; Meyer, Rolf; Buchty, Rainer; Fossati, Luca; Berekovic, Mladen: SoCRocket - A virtual platform for the European Space Agency's SoC development. In: ReCoSoC. S. 1–7, 2014, available at http://github.com/socrocket.

[SM14] Schafer, Benjamin Carrion; Mahapatra, Anushree: S2CBench: Synthesizable SystemC Benchmark Suite for High-Level. IEEE Embedded Systems Letters, (3):53–56, 2014.

[Te] TechPowerUp: , https://www.techpowerup.com/cpudb/2055/xeon-platinum-8180. Accessed: 2019-02-01.

Mehran Goli wurde am 21. März 1989 in Sary, Iran, geboren. Er ist wissenschaftlicher Mitarbeiter am Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI) in Bremen. Er erwarb einen B.Sc.-Abschluss in Informatik an der Universität von Shahid Beheshti, Teheran, Iran, im Jahr 2012 und einen M.Sc.-Abschluss in Informatik im Fachbereich Elektrotechnik und Informationstechnik (ECE) der Universität Teheran (UT), Teheran, Iran, im Jahr 2015. Er promovierte 2019 zum Dr.-Ing. in Informatik in der Arbeitsgruppe Rechnerarchitektur (AGRA) bei Prof. Dr. Rolf Drechsler an der Universität Bremen. Seine Forschungsschwerpunkte sind Design, Verifikation, Sicherheitsüberprüfung, Maschinelles Lernen und Approximate Computing auf der Ebene elektronischer Systeme.

Dynamische Graphalgorithmen and Graphsparsifikation: Neue Techniken und Zusammenhänge[1]

Gramoz Goranci[2]

Abstract: Ein dynamischer Graphalgorithmus ist eine Datenstruktur, die einen Graphen speichert, der sich über die Zeit ändert, und die die Lösung für ein zugrunde liegendes Graphproblem nach jeder Änderung effizient aktualisiert. Ein Graph-Sparsifier ist eine komprimierte Darstellung eines großen Eingabegraphen, bei dem einige Grapheigenschaften (annähernd) erhalten bleiben. In dieser Dissertation entwickeln wir neue Techniken sowohl für dynamische Graphalgorithmen als auch für Sparsifikationsalgorithmen. Wir fokussieren uns auf verschiedene graphbasierte Optimierungsprobleme, die in der spektralen Graphentheorie, der Graphpartitionierung und bei metrischen Einbettungen auftreten. Unsere dynamische Algorithmen haben schnellere Laufzeiten als vorherige Ergebnisse. Unsere Sparsifier-Konstruktionen erzeugen kleinere Sparsifier und verbessern gleichzeitig ihre Approximationsqualität. Wir führen zudem neuartige Reduktionstechniken ein, die unerwartete Zusammenhänge zwischen dynamischen Graphalgorithmen und sparsifizierten Graphen aufdecken.

1 Einführung

Die zunehmende Digitalisierung, sowie neue Entwicklungen wie das Internet of Things sorgen jährlich für einen enormen Anstieg des Datenvolumens. Ein beträchtlicher Teil dieser Daten kann mithilfe von *Graphen* modelliert werden. Ein Graph ist eine Menge von Knoten und Kanten, wobei jede Kante ein Knotenpaar verbindet. Graphen sind in der Mathematik und der Informatik allgegenwärtige Strukturen und treten in verschiedenen Zusammenhängen der realen Welt auf, z. B. in sozialen Netzwerken wie Facebook oder Twitter, das Web-Netzwerk und in Telekommunikationsnetzwerken.

Im Vergleich zu anderen Datendarstellungen sind Graphen besonders geeignet, da ihre Visualisierung häufig Möglichkeiten bietet, interessante Muster in den Daten zu identifizieren, z. B. das Erkennen von Communities in sozialen Netzwerken. Graphdarstellungen sind auch nützlich, weil Algorithmen zu ihrer Manipulation und Speicherung seit den Anfängen des Informatik gründlich untersucht wurden. Dennoch arbeitet eine große Anzahl dieser Graphalgorithmen unter der Annahme, dass Graphen statisch sind, d. h. dass sie sich nicht ändern und dass sie im Speicher eines einzelnen Computers gespeichert werden können. Leider sind diese Annahmen für Graphen aus vielen wichtigen realen Szenarien nicht zutreffend.

Betrachten wir etwa einen Kartengraphen, in dem jeder Knoten einer Stadt entspricht und eine Kante zwischen zwei Knoten die Route darstellt, die sie verbindet. Dieser Kartengraph

[1] Englischer Titel der Dissertation:"Dynamic Graph Algorithms and Graph Sparsification: New Techniques and Connections" [Go19]

[2] Department of Computer Science, University of Toronto, ggoranci@cs.toronto.edu

CC BY SA

enthält außerdem die Länge der Routen, indem die Kanten mit der Distanz zwischen den Städten gewichtet ist. Eine grundlegende Frage beim Entwurf von Algorithmen besteht darin, die metrische Struktur des Kartengraphen zu verstehen. Ein konkretes Problem besteht darin, die kürzesten Wege zwischen allen Knotenpaaren im Kartengraphen berechnen. Diese Aufgabe wird von vielen klassischen Algorithmen gelöst, und die Laufzeitkomplexität dieses Problems ist in der Anzahl der Knoten kubisch. In der Realität sind Kartengraphen allerdings nicht statisch, sondern ändern sich mit der Zeit. Beispielsweise kann es aufgrund von Bauarbeiten vorkommen, dass eine Straße, die zwei Städte verbindet, gesperrt wird, was wiederum bedeutet, dass die Kante, die diese Städte verbindet, aus dem Kartengraph gelöscht wird. Das Löschen einer solchen Kante kann sich auf den kürzesten Weg zwischen Städten auswirken, was bedeutet, dass die alte Lösung für den neuen Kartengraph falsch ist. Eine naheliegende Möglichkeit die Lösung zu korrigieren, besteht darin, die kürzesten Wege im veränderten Graphen neu zu berechnen. Diese Lösung hat jedoch einen hohen Rechenaufwand, der für kleine Geräte mit begrenzten Ressourcen, wie z. B. Navigationssysteme, nicht durchführbar ist. Daher ergeben sich folgende Fragen: Können wir Algorithmen entwerfen, die die Lösung nicht jedes Mal komplett neu berechnen müssen, wenn sich der Graph ändert? Wenn ja, was ist die bestmögliche Laufzeit, die wir erreichen können? Müssen wir eine etwas schlechtere Lösungsqualität akzeptieren, um eine bessere Laufzeit zu erzielen?

Andere Herausforderungen beim Umgang mit großen Graphen sind Rechen- und Speicherressourcen. Dies liegt daran, dass die Größe eines Graphen (bei sehr dichten Graphen) quadratisch zur Anzahl der Knoten anwächst. Ein traditioneller Ansatz zur Behebung dieses Problems bestand darin, große Graphen in kleinere zu *komprimieren* und dabei die gewünschten Eigenschaften beizubehalten. Diese komprimierten Versionen von Graphen sind besonders beliebt, da damit Berechnungen für den ursprünglichen Graph auf dem komprimierten Graph ausgeführt werden können, was zu erheblichen Einsparungen bei den Rechen- und Speicherressourcen führt. Die Komprimierung von Graphen wird üblicherweise aus zwei Perspektiven untersucht: (1) Verringerung der Anzahl von Kanten eines Graphen, (2) Verringerung der Anzahl von Knoten. Während der erste Ansatz erfolgreich eingesetzt wurde, um die Laufzeit vieler grundlegender Graphprobleme zu verbessern, ist seine praktische Anwendbarkeit eingeschränkt, da die meisten großen Netzwerke bereits nicht sehr dicht sind und daher wenig Kanten gelöscht werden können. Infolgedessen haben Komprimierungswerkzeuge, die die Anzahl der Knoten reduzieren, in den letzten zehn Jahren immer mehr Beachtung gefunden. Zur Veranschaulichung kehren wir zum Beispiel des Kartengraphen zurück. Nehmen wir an, dass wir unter allen Städten im Graph nur an einer kleinen Teilmenge von Knoten, die für uns wichtig sind, interessiert sind. Dies ist in vielen praktischen Szenarien relevant, z. B. möchte man Entfernungsinformationen nur zwischen großen Städten erhalten, während die kleinen ignoriert werden können. Daher ergeben sich folgende Fragen: Können wir den Kartengraphen zu einem Graphen komprimieren, welcher nur die großen Städte enthält, während die Entfernungen (approximativ) beibehalten werden? Wie gut approximieren die Distanzen im kleinen Graphen die echten Distanzen? Was ist der Kompromiss zwischen der Approximationsqualität und der Größe des komprimierten Graphen?

In dieser Dissertation beschäftigen wir uns mit allen oben genannten Fragen. Wir präsentieren beweisbare algorithmische Werkzeuge sowohl aus dynamischer als auch aus Kompressionsperspektive für verschiedene graphbasierte Optimierungsprobleme, die in der spektralen Graphentheorie, der Graphpartitionierung und bei metrischen Einbettungen auftreten. Darüber hinaus führen wir neuartige Reduktionstechniken ein, die unerwartete Zusammenhänge zwischen sich zeitlich verändernden Graphen und der Graphkomprimierung aufdecken. Im Folgenden werden zunächst die Ergebnisse für dynamische Graphalgorithmen vorstellen und anschließend unsere Beiträge im Bereich der Graphsparsifikation diskutieren.

2 Dynamische Graphalgorithmen

Sei $G = (V, E)$ ein Graph mit n Knoten und m Kanten und $\mathscr{P}$ eine Eigenschaft von G (beispielweise könnte $\mathscr{P}$ der Durchmesser von G sein). Angenommen, G ändert sich durch eine Kanteneinfügung oder -löschung. Können wir die Eigenschaft $\mathscr{P}$ im geänderten Graphen effizient aktualisieren, anstatt sie von Grund auf neu zu berechnen? Diese grundlegende Frage wird seit Jahrzehnten für viele wichtige Grapheneigenschaften gestellt, und der Bereich, der diese Fragen untersucht, heißt *dynamische Graphalgorithmen*. Konkret ist ein dynamischer Graphalgorithmus eine Datenstruktur, die die folgenden Operationen auf einem gegebenen Eingabegraphen G ausführt: (a) INSERT(u,v): füge die Kante (u,v) zu G hinzu, (b) DELETE(u,v): lösche die Kante (u,v) aus G und (c) QUERY$(\mathscr{P})$: frage die Eigenschaft $\mathscr{P}$ ab.

In einigen Varianten dynamischer Graphgorithmen wird die Query-Operation möglicherweise nicht unterstützt, und hier besteht das Ziel darin, zu jedem Zeitpunkt einfach eine korrekte Eigenschaft $\mathscr{P}$ in Bezug auf den aktuellen Graphen zu verwalten. Ein dynamischer Algorithmus hat folgende wichtige Laufzeit-Parameter: (1) *Update-Zeit*, die benötigte Zeit für die Operationen INSERT(u,v) und DELETE(u,v) und (2) *Query-Zeit*, die benötigte Zeit für die Operation QUERY$(\mathscr{P})$. Update- und Query-Zeiten können entweder *worst-case* sein, d. h. die Zeit, die für die Verarbeitung jedes Updates oder Query maximal einzeln aufgewendet wird, oder *amortisiert*, d. h., die durchschnittliche Laufzeit, die über eine Sequenz von Updates amortisiert wird.

Abhängig von den Arten der Update-Operationen werden dynamische Algorithmen in drei Hauptkategorien eingeteilt: (i) *voll-dynamisch*, wenn Updates sowohl aus Einfügungen als auch aus Löschungen bestehen; (ii) *inkrementell*, wenn Updates nur aus Einfügungen bestehen; und (iii) *dekrementell*, wenn Updates nur aus Löschungen bestehen. Bei der Untersuchung der Update-Zeiten in Algorithmen vom Typ (ii) und (iii) ist es üblich, die *Gesamtlaufzeit* zu berücksichtigen, d. h., die Zeit, die für eine Sequenz von $\Theta(m)$ Einfügungen oder Löschungen aufgewendet wird. Dynamische Algorithmen können entweder *deterministisch* oder *randomisiert* sein, und häufig kann man durch Randomisierung Algorithmen mit besseren Laufzeiten erreichen.

Insbesondere in den letzten zwei Jahrzehnten wurden große Fortschritte bei der Entwicklung effizienter dynamischer Graphalgorithmen erzielt. Zu den berücksichtigten Grapheigenschaften zählen Konnektivität, Erreichbarkeit, kürzeste Wege, (globaler) Minimalschnitt,

minimaler Spannbaum, Spanners, Schnitt- und Spektralsparsifier. Trotz all dieser Fortschritte gibt es jedoch weiterhin viele wichtige Probleme, deren dynamische Komplexität nur unzureichend verstanden wird. Zum Beispiel gibt keine nicht-trivialen dynamischen Algorithmen für Probleme wie Graphpartitionierung oder zur Lösung von strukturierten linearen Gleichungssystemen, die zuletzt viel Aufmerksamkeit in verschiedenen Bereichen der Informatik erhalten haben. Aus diesem Grund untersuchen wir in dieser Dissertation dynamische Algorithmen für einige dieser Probleme und zeigen, dass ihre dynamische Komplexität nicht-trivial verbessert werden kann. Wir machen zudem Fortschritte bei grundlegenden Graphproblemen, indem wir ihre seit langem bekannten Laufzeitgarantien verbessern.

Dynamische Algorithmen für spektralbasierte Grapheneigenschaften. Wir untersuchen dynamische Algorithmen zur Verwaltung von Lösungen für Laplace-Systeme und effektive Widerstände. Laplace-Systeme sind eine wichtige Unterklasse von linearen Gleichungssystemen, die in vielen natürlichen Kontexten auftreten, etwa im maschinellen Lernen, in der Computergrafik und in der Bildverarbeitung. Das Lösen von Laplace-Systemen hat nach der mit dem Gödel-Preis ausgezeichneten Arbeit von Spielman und Teng [ST04], die den ersten approximativen Löser mit annähernd linearer Laufzeit entwickelt haben, erhebliche Aufmerksamkeit erhalten.

Bei einem ungerichteten, ungewichteten Graphen $G = (V,E)$ mit n Knoten und m Kanten bezeichnen $\mathbf{L} := \mathbf{D} - \mathbf{A}$ die *Laplace-Matrix* von G, wobei $\mathbf{D}$ und $\mathbf{A}$ die zugehörigen Grad- und die Adjazenzmatrizen von G sind. Die Laplace-Matrix $\mathbf{L}$ bildet zusammen mit einem Vektor $\mathbf{b} \in \mathbb{R}^n$ ein lineares Gleichungssystem $\mathbf{Lx} = \mathbf{b}$, das als *Laplace-System* bezeichnet wird. Sei $\mathbf{L}^\dagger$ die Pseudoinverse von $\mathbf{L}$. Ein *Lösungsvektor* $\tilde{\mathbf{x}} \in \mathbb{R}^n$ ist *ε-approximativ*, wenn er $\left\|\tilde{\mathbf{x}} - \mathbf{L}^\dagger \mathbf{b}\right\|_{\mathbf{L}} \leq \varepsilon \left\|\mathbf{L}^\dagger \mathbf{b}\right\|_{\mathbf{L}}$ erfüllt, wobei $\|\mathbf{x}\|_{\mathbf{L}} := \sqrt{\mathbf{x}^\top \mathbf{L}\mathbf{x}}$ und $\varepsilon > 0$ ist ein Fehlerparameter. Sei $\mathbf{1}_u \in \mathbb{R}^n$ der Indikatorvektor eines Konten u, so dass $\mathbf{1}_u(v) = 1$, wenn $v = u$ und 0 andernfalls.

Wir führen ein dynamisches Modell zum Lösen von Laplace-Systemen ein, das folgende Operationen zulässt: (1) Das Einfügen und Löschen von Kanten im zugrunde liegenden Graphen G (dies entspricht dem Ändern von Einträgen von $\mathbf{L}$, die sich nicht auf der Diagonalen befinden), (2) das Ändern des Vektors $\mathbf{b}$ und (3) Zugriff auf die Koordinaten eines ε-approximativen Lösungsvektors $\tilde{\mathbf{x}}$. Für ungerichtete, ungewichtete Graphen mit beschränktem Grad zeigen wir den ersten nicht-trivialen dynamischen Algorithmus, welcher eine erwartete amortisierte Laufzeit[3] von $\tilde{O}(n^{11/12}\varepsilon^{-5})$ erreicht und damit schneller ist als der $\tilde{O}(n)$-Zeitalgorithmus, der nach jedem Update einen Lösungsvektor von Grund auf neu berechnet. Wenn der Vektor $\mathbf{b}$ nur wenige Nicht-Null-Einträge enthält, erstreckt sich unser Ergebnis auf allgemeine Graphen, wobei eine verbesserte Laufzeit von $\tilde{O}(m^{3/4}\varepsilon^{-5})$ erreicht wird.

Ein spektrales Objekt im Zusammenhang mit Laplace-Systemen ist der *effektive Widerstand*. Die Berechnung von effektiven Widerständen führte zuletzt zu schnelleren Algorithmen zur Berechnung von maximalen Flüssen und gewinnt zunehmen an Popularität in der theoretischen Informatik. Die algebraische Definition des effektiven Widerstands $R^G_{\text{eff}}(u,v)$

[3] Mit der $\tilde{O}(\cdot)$-Notation unterdrücken wir der Übersichtlichkeit halber polylogarithmische Faktoren.

zwischen einem Knotenpaar u und v in einem Graphen G ist gegeben durch

$$R^G_{\text{eff}}(u,v) := (\mathbf{1}_u - \mathbf{1}_v)^\top \mathbf{L}^\dagger (\mathbf{1}_u - \mathbf{1}_v).$$

Physikalisch gesehen, d. h. wenn G als Widerstandsnetzwerk betrachtet wird, ist der effektive Widerstand zwischen u und v die Energie des Flusses, wenn eine Stromeinheit von u nach v in G gesendet wird.

Wir untersuchen voll-dynamische Algorithmen, die den effektiven Widerstand zwischen allen Knotenpaaren verwalten. Wir erhalten einen Algorithmus, der Updates und Querys in $\tilde{O}\left(\min\{m^{3/4}, n^{5/6}\}\varepsilon^{-4}\right)$ Zeit durchführt und effektive Widerstände bis auf einen $(1+\varepsilon)$-Faktor approximiert. Zudem erweitern wir das obige Ergebnis auf zwei Arten: Erstens wird der Algorithmus auf *gewichtete*, ungerichtete Graphen verallgemeinert mit einer Laufzeit von $\tilde{O}(n^{5/6}\varepsilon^{-4})$ für die erwartete amortisierte Update- und Query-Zeit. Wenn zweitens zudem angenommen wird, dass die gewichteten Graphen *kleine* Knoten-Separatoren zulassen (z. B. planare Graphen), verbessert sich unsere *worst-case* Laufzeitgarantie auf $\tilde{O}(\sqrt{n}\varepsilon^{-2})$. Unser Ergebnis für allgemeine Graphen steht in starkem Kontrast zu eng verwandten Grapheigenschaften wie dem kürzesten Weg, für den (bedingte) untere Schranken derartige Laufzeitverbesserungen im voll-dynamischen Setting ausschließen, und dem maximalen Fluss, dessen dynamische Komplexität bisher nur unzureichend verstanden ist.

Die Grundidee unserer Datenstrukturen ist es, dynamisch eine Approximation des Schur-Komplements zu verwalten. Ein *Schur-Komplement* (auch *spektraler Knotensparsifier* genannt) H eines Graphen G in Bezug auf $K \subseteq V(G)$ ist ein Graph mit $V(H) = K$, wobei die effektiven Widerstände zwischen jedem Paar von Knoten in K, die gleichen sind wie ihre effektiven Widerstände in G. Abhängig von der Familie der Graphen, an denen wir interessiert sind, gibt es geringfügige Unterschiede in der Art und Weise, wie wir diesen Sparsifier nutzen. Bei allgemeinen Graphen beruhen unsere Techniken entscheidend darauf, dass das Schur-Komplement als Vereinigung über Random Walks mit Endpunkten in K betrachtet werden kann. Dies ermöglicht es uns, Knoten aus dem Originalgraphen zufällig auszuwählen und dann ein Schur-Komplement in Bezug auf diese zufällig gewählte Knotenmenge zu konstruieren. Das Subsampling stellt sicher, dass die Random Walks kurz sind, und dies ermöglicht es uns, sie unter Verwendung elementarer Datenstrukturen effizient zu samplen und zu verwalten. Für planare Graphen nutzen wir, dass sie Knoten-Separatoren sublinearer Größe haben (daher die Abhängigkeit von $\sqrt{n}$ von der Laufzeit) und dass approximative Schur-Komplemente in nahezu linearer Zeit berechnet werden können [KS16]. Inspiriert von der Arbeit von Lipton, Rose und Tarjan [LRT79] zu *nested dissection*, werden diese beiden Bestandteile dann zusammengeführt, um die Schur-Komplement für diese Graphenfamilie dynamisch zu verwalten.

Alle Resultate, die wir oben präsentiert haben, garantieren $(1+\varepsilon)$-approximative effektive Widerstände. Wir zeigen, dass es nicht möglich ist, exakte effektive Widerstände mit Update-Zeiten von $O(n^{1-\delta})$ und einer Query-Zeit von $O(n^{2-\delta})$ für $\delta > 0$ zu erhalten, außer eine populäre Hypothese der Komplexitätstheorie ist falsch.

Dynamische Low-Stretch Bäume. Wir untersuchen dynamische Algorithmen für Low-Stretch Spannbäume und Spanners. Bäume gehören zu den einfachsten Graphklassen und sind aus algorithmischer Sicht besonders praktisch, da viele Probleme einfache Lösungen

für Bauminstanzen zulassen. Ein gängiger Ansatz zur Ausnutzung dieses Verhaltens von Bäumen besteht darin, allgemeine Graphen als Bäume zu approximieren und dabei die relevanten Grapheigenschaften beizubehalten. Der Begriff des Low-Stretch Spannbaums ist ein prominentes Beispiel für eine solche Graphenapproximation, die bei der Lösung symmetrischer, diagonal dominanter (SDD) linearer Systeme eine zentrale Rolle gespielt hat. Für einem ungewichteten, ungerichteten Graphen $G = (V,E)$ und einem Spannbaum T von G ist der *Stretch* einer Kante $(u,v) \in E$ in Bezug auf T gegeben durch $\text{stretch}_T(u,v) := \text{dist}_T(u,v)$, wobei $\text{dist}_G(u,v)$ die Länge des kürzesten Weges zwischen u und v in G bezeichnet. Das *durchschnittliche Stretch* über alle Kanten von G in Bezug auf T ist $\text{avg-stretch}_T(G) := \frac{1}{|E|}\sum_{(u,v)\in E}\text{stretch}_T(u,v)$. Ein *Low-Stretch* Spannbaum ist ein Baum mit einem subpolynomialen oder polylogarithmischen durchschnittlichen Stretch in der Anzahl der Knoten n.

Motiviert durch die grundlegende Bedeutung von Low-Stretch Spannbäume sowie deren wichtigen Anwendungen betrachten wir dieses Objekt aus dynamischer Sicht. Wir zeigen, dass es für ungewichtete, ungerichtete Graphen einen voll-dynamischen Algorithmus gibt, der einen Spannbaum mit $n^{o(1)}$ erwartetem durchschnittlichen Stretch und $n^{1/2+o(1)}$ amortisierter Update-Zeit verwaltet. Unser Ergebnis beantwortet eine offene Frage von Baswana et al. [BKS12] und kann erweitert werden, um den durchschnittliche Stretch $O(t)$ und die Update-Zeit $n^{1+o(1)}/t$ zu erreichen, wobei $t \geq \sqrt{n}$. Dies zeigt, dass die $\sqrt{n}$ -Barriere in der Laufzeit nicht inhärent ist, zumindest wenn ein sehr großer Strech tolerierbar ist. Eine der wesentlichen Ideen unseres Algorithmus besteht darin, den Graphen in Cluster mit kleinem Durchmesser zu zerlegen und diese Zerlegung dynamisch verwaltet. Dies wird mit dem Random-Shift-Clustering von Miller, Peng und Xu [MPX13] zusammen mit vielen Anpassungen implementiert, damit es im dynamischen Setting funktioniert. Wir verwenden dann eine dynamische Version der Hierarchie von Clustern mit geringem Durchmesser von Alon, Karp, Peleg und West [Al95], was wiederum einen anspruchsvollen Amortisierungsansatz erfordert, um die Verbreitung von Aktualisierungen innerhalb der Hierarchie zu steuern. Zusätzlich verwendet unser Algorithmus *Sparsifier für dynamische Schnitte*, um das Problem auf wenig dichte Graphen zu reduzieren.

Eine direkte Konsequenz unseres dynamisches Random-Shift-Clusterings ist ein verbesserter dynamischer Algorithmus für Spanners. Ein *k-Spanner H* eines Graphen G ist ein *Subgraph* von G, in dem für alle Knoten u und v gilt, dass $\text{dist}_H(u,v) \leq k \cdot \text{dist}_G(u,v)$. Für $t \geq 1$ und jeden ungewichteten, ungerichteten Graphen G entwickeln wir einen voll-dynamischen Algorithmus, der einen $(2t-1)$-Spanner mit $O(n^{1+1/t}\log n)$ Kanten und $O(t\log^2 n)$ amortisierter Update-Zeit verwaltet. Dieses Ergebnis verbessert den Algorithmus von Baswana et al. [BKS12] um Faktor t in der Größe des Spanners und der Update-Zeit. Die Verbesserung ist besonders relevant, da der Trade-Off zwischen Stretch und Graphgröße von $2t-1$ zu $O(n^{1+1/t})$ unter der bekannten Erdös-Girth-Conjecture bereits bestmöglich ist.

Dynamische Graphpartitionierung. Wir untersuchen inkrementelle Algorithmen für den globalen minimalen Schnitt und minimalen Schnitt zwischen Knoten s und t, die beide grundlegende Probleme bei der Graphpartitionierung darstellen. Bei Problemen der Graphpartitionierung wird der Eingabegraph in kleinere Komponenten unterteilt, während die Anzahl der Kanten zwischen diesen Komponenten minimiert wird. Diese Probleme

haben in der Vergangenheit einen zentralen Platz beim Verständnis von Netzwerkflüssen, Paketrouting und VLSI-Layout eingenommen. Sie wurden auch in vielen Divide-and-Conquer-Ansätzen zur Lösung von graphbasierten Clustering-Problemen eingesetzt.

Bei einem ungewichteten, ungerichteten Graphen $G = (V,E)$ ist ein *Schnitt* von G eine Partition von V in zwei nicht-leere Teilmengen A und B. Wenn s und t als verschiedene Knoten in G gegeben sind, ist ein *s-t-Schnitt* von G ein Schnitt (A,B), so dass $s \in A$ und $t \in B$. Für einen Schnitt (A,B) in G ist die *Größe* von (A,B) die Anzahl der Kanten mit einem Endpunkt in A und dem anderen in B. Ein *globaler minimaler Schnitt*, der mit λ_G bezeichnet wird, ist ein Schnitt von minimaler Größe. Ein *s-t-minimaler Schnitt*, bezeichnet mit $\lambda_G(s,t)$, ist ein s-t-Schnitt mit einer minimaler Größe.

Wir entwickeln einen *exakten, deterministischen* inkrementellen Algorithmus, der einen globalen minimalen Schnitt λ_G in $O(\log^3 n \log\log n)$ amortisierter Update-Zeit und $O(1)$ Query-Zeit verwaltet. Unser Ergebnis erzielt eine *exponentielle* Beschleunigung der Update-Zeit des bisher besten bekannten Algorithmus von Henzinger [He97] und beantwortet teilweise eine offene Frage von Thorup [Th07]. Darüber hinaus stehen unsere Laufzeitschranken in starkem Kontrast zu einer bedingten Untergrenze für die dynamische Verwaltung von $\lambda(G)$ in *gewichteten* Graphen, die zeigt, dass es keine Algorithmen gibt, die gleichzeitig eine sublineare Update- und Query-Zeit erreichen können. Die Hauptidee unseres Algorithmus besteht darin, eine Sparsifikationsroutine von Kawarabayshi und Thorup [KT19] und den exakten inkrementellen Algorithmus von Henzinger [He97] zu kombinieren. Die Kombination ist allerdings nicht-trivial, da sie Subroutinen beider Arbeiten deutlich erweitert und diese Erweiterungen zur Erlangung unserer Garantien erfordert.

Obwohl das s-t-Minimalschnitt-Problem eines der zentralen Probleme in der algorithmischen Graphentheorie ist, ist nur sehr wenig über dessen dynamische Komplexität für allgemeine Graphen bekannt. Wir machen die ersten positiven Fortschritte, um dieses Problem zu verstehen, indem wir einen inkrementellen Algorithmus mit $\tilde{O}(n^{2/(\ell+1)})$ Update- und Query-Zeit entwerfen, der eine $O(\log^{4\ell} n)$ Approximation von $\lambda_G(s,t)$ für alle $s,t \in V(G)$ erreicht, wobei $\ell \geq 1$. Unser technischer Beitrag ist ein neue Art von Sparsifiern, genannt *lokale Sparsifier*. Diese Sparsifiern sind eine stärkere Version der gut erforschten Knotensparsifier. Ein *Knotensparsifier* H eines Graphen G in Bezug auf eine Menge von *Terminals* $K \subseteq V(G)$ ist ein Graph mit $V(H) \supseteq K$, in dem (i) $|V(H)|$ "klein" ist und (ii) H eine Grapheigenschaft $\mathscr{P}$ bewahrt, die auf die Terminals K in G beschränkt ist (siehe nächste Abschnitt für eine eingehende Behandlung der Knotensparsifiern). Ein *lokaler Sparsifier* ist eine Datenstruktur, die Knotensparsifier generalisiert. Bei einem gegebenen Graphen $G = (V,E)$ besteht das Ziel darin, eine Datenstruktur zu erstellen, die die folgenden Operationen unterstützt: (1) PREPROCESS(G): verarbeite den Graphen G vor, (2) QUERYSPARSIFIER(G,K): Berechne einen Knotensparsifier H von G, der eine Eigenschaft $\mathscr{P}$ in Bezug auf K beibehält, und gebe diesen aus. Lokale Sparsifier es uns ermöglichen also, Knotensparsifier für eine beliebige Menge von Terminals in K zu extrahieren. In Operation (2) ist eine sehr wichtige Anforderung, dass K Teil der Eingabe ist, da es $\Theta(2^n)$ verschiedene Terminalmengen K gibt.

Wir zeigen, dass eine Variante von *tree cut sparsifiers* von Räcke, Shah und Täubig [RST14] verwendet werden kann, um einen lokalen Sparsifier zu konstruieren, der die Schnittstruktur

des Graphen bis zu polylogarithmische Faktoren bewahrt und $\tilde{O}(m)$ Vorverarbeitungszeit und $\tilde{O}(|K|)$ Query-Zeit erreicht. Wir entwickeln darauf basierend ein Reduktions-Framework, das effiziente lokale Sparsifier in einen inkrementellen Algorithmus umwandelt, was wiederum eine polylogarithmische Approximation des *s-t*-Minimalschnitts impliziert. Zudem impliziert unsere Technik einen inkrementellen Approximationsalgorithmus für das *uniform sparest cut*-Problem mit ähnlichen Garantien.

3 Graphsparsifikation

Ein *Graph Sparsifier* ist eine "komprimierte" Version eines großen Eingabegraphen, der Eigenschaften wie Entfernungs- oder Erreichbarkeitsinformationen, Schnittwert oder das Spektrum des Graphen beibehält. Herkömmlicherweise wurden Graph Sparsifier aus zwei Perspektiven untersucht: (1) *Kantensparsifier*, die die Anzahl der Kanten eines Graphen verringern, und (2) *Knotensparsifier*, die die Anzahl der Knoten verringern. Knotensparsifier wurden erfolgreich angewendet, um die Laufzeit vieler grundlegender graphbasierter Optimierungsprobleme zu verbessern. Die bemerkenswertesten Beispiele sind transitive Reduktionen, Spanners sowie Schnitt- und spektrale Sparsifier.

In dieser Arbeit konzentrieren wir uns auf Knotensparsifier. Ausgehend von einem Graphen $G = (V,E)$ und einer Menge von Knoten $K \subseteq V$, die als *Terminals* bezeichnet werden, besteht das Ziel darin, einen Graphen $H = (V',E')$ zu konstruieren, der die folgenden Eigenschaften erfüllt (i) $V' \supseteq K$ und $|V'|$ ist "klein", idealerweise $|V'| = O(\text{poly}(|K|))$ und (ii) H bewahrt Eigenschaften wie Erreichbarkeit, Distanz oder Schnitte, die zwischen Terminals in K definiert sind (approximativ). Oft ist es wünschenswert, dass H strukturell ähnlich zu G ist, z. B. wenn G planar ist, soll H ebenfalls planar sein. Wenn H eine Eigenschaft approximativ beibehält, wird das Approximationsverhältnis als *Qualität* des Sparsifiers bezeichnet. Dieses Komprimierungswerkzeug ist aus algorithmischer Sicht sehr nützlich: sobald H berechnet ist, kann man algorithmische Aufgaben auch in H anstelle von G ausführen, was wiederum zu Einsparungen bei den Rechen- und Speicherressourcen führt. Neben ihrer praktischen Relevanz haben Knotensparsifier auch Anwendungen in anderen Teilbereichen der theoretischen Informatik gefunden, nämlich in Approximationsalgorithmen und im Netzwerk-Routing.

Im Folgenden werden wir unsere Beiträge zu Knotensparsifiern diskutieren, die strukturell einem Eingabegraphen ähneln und gleichzeitig Distanzen in ungerichteten Graphen oder Informationen zur Erreichbarkeit in gerichteten Graphen beibehalten.

Distance Approximation Minors. Wir untersuchen Knotensparsifier, die Minoren des Eingabegraphen sind und die die Distanzen zwischen den Terminals approximieren. Ein Minor ist ein Graph, der durch das Löschen von Kanten und Knoten und durch Kantenkontraktion erzeugt wurde. Minoren sind besonders vorteilhaft, da sie strukturelle Eigenschaften des Eingabegraphen bewahren, z. B. ist ein Minor eines planaren Graphen ebenfalls ein planarer Graph. Wenn ein Graph $G = (V,E)$ und Terminals $K \subseteq V$ gegeben sind, ist ein α-*Distance Approximating Minor* von G ein Graph $H = (V',E')$, sodass (i) $V' \supseteq K$, (ii) H ein Minor von G ist, der alle Terminals enthält und bei dem keine Kanten von Terminals kontraktieren

wurden, und (iii) der kürzeste Weg zwischen zwei Knoten der Untermenge K liegt innerhalb eines α-Faktors ihres kürzesten Weges in G. Knoten in $V' \setminus K$ werden als *Steiner-Knoten* bezeichnet.

Krauthgamer, Nguyen und Zondiner [KNZ14] untersuchten *distance preserving minors*, d. h. $\alpha = 1$. Sie zeigten, dass allgemeine Graphen distance preserving minors mit $O(|K|^4)$ Steiner-Knoten besitzen. Es stellt sich natürlicherweise die Frage, in welchem Verhältnis sich Qualität der Approximation und die Anzahl der Steiner-Knoten zueinander befinden. Für diese Frage erhalten wir neue unteren und oberen Schranken. Unsere untere Schranke zeigt, dass es für unendlich viele $k \in \mathbb{N}$ und einen Graphen mit k Terminals gibt, die kein $(\alpha - \varepsilon)$-distance preserving minor mit $\Omega(k^\gamma)$ Steiner-Knoten zulässt, für alle $\varepsilon > 0$, wobei $\alpha \geq 2$ und $\gamma = \gamma(\alpha)$. Zum Beispiel folgt aus unseren Resultaten, dass jeder $(2-\varepsilon)$-distance preserving minor mindestens $\Omega(k^2)$ Knoten besitzen muss.

Um die untere Schranke zu beweisen, führen wir eine neue Black-Box-Reduktionstechnik ein, die die untere Schranken für das auf $V' = K$ beschränkte Distance Approximating Minor-Problem umwandelt, d. h. $|V' \setminus K| = 0$, in superlineare untere Schranken für $|V' \setminus K|$ für Distance Approximating Minors mit der gleichen Qualität. Das Herzstück unserer Graphkonstruktionen sind Varianten von *Steiner Systems*, welche vor allem im kombinatiorischen Design untersucht wurden. Unsere obere Schranke zeigt, dass die Klasse der planaren Graphen $(1+\varepsilon)$-distance approximation minors mit $|V'| = O(|K|^2 \varepsilon^{-2} \log^2 |K|)$ zulässt. Durch eine leichte Qualitätssteigerung kann unser Sparsifier daher die Resultate von Krauthgamer et al. [KNZ14] um einen quadratischen Faktor verbessern. Der Schlüssel zu diesem Ergebnis ist das sogenannte *Terminal Path Cover*. Ein solches Cover ist eine Menge von kürzesten Wegen im Graphen, deren Vereinigung (1) die Terminalmenge enthält und (2) kürzeste Distanzen zwischen Terminals approximiert. Wir zeigen, dass *Distance Oracles* für planare Graphen erweitert werden können, um Terminal Path Covers für planare Graphen zu konstruieren. Nun erhalten wir die gewünschten Garantien, indem wir diese Ideen zusammen mit einem Argument zur Anzahl der Verzweigungen auf den kürzesten Wegen kombinieren.

Reachability Preserving Minors. Die Kantensparsifikation von *gerichteten* Graphen ist für viele Probleme deutlich schwieriger als in ungerichteten Graphen. Etwa kann man für Probleme wie Erreichbarkeit zeigen, dass es für gerichtete Graphen keine nicht-trivialen Kantensparsifier geben kann. Wir komplementieren dieses negative Ergebnis für Kantensparsifier, indem wir positive Resultate für Knotensparsifier präsentieren. In diesem Problem ist die Eingabe ein gerichteter Graph $G = (V, E)$ und eine Menge von Terminals $K \subseteq V$. Ein *Reachability Preserving Minor* von G ein gerichteter Graph $H = (V', E')$, der die folgenden Eigenschaften erfüllt: (i) $V' \supseteq K$, (ii) H ist ein Minor von G, und (iii) für jedes Paar von Knoten $u, v \in K$ gibt es einen gerichteten Weg von u nach v in H, wenn es einen gerichteten Weg von u nach v in G gibt.

Wir initiieren die Erforschung solcher Sparsifier und erhalten die ersten nicht-trivialen Garantien für das Problem. Wir zeigen, dass planare Graphen Reachability Preserving Minors mit $|V'| = O(|K|^2 \log |K|)$ besitzen. Für allgemeinen Graphen sind unsere oberen Schranken nur um einen weiteren Faktor $|K|$ schlechter. Weiterhin beweisen wir neue untere Schranken, die zeigen, dass für unendlich viele $k \in \mathbb{N}$ ein gerichteter planarer Graph G mit

k Terminals existiert, so dass jeder Reachability Preserving Minor $\Omega(k^2)$ Steiner-Knoten verwenden muss. Für planare Graphen sind die untere und die obere Schranke bis auf polylogarithmische Faktoren in k scharf.

Literaturverzeichnis

[Al95] Alon, N.; Karp, R.; Peleg, D.; West, D.: A Graph-Theoretic Game and Its Application to the k-Server Problem. SIAM J. Computing, 24(1):78–100, 1995.

[BKS12] Baswana, S.; Khurana, S.; Sarkar, S.: Fully dynamic randomized algorithms for graph spanners. ACM Transactions on Algorithms, 8(4):35:1–35:51, 2012.

[Go19] Goranci, G.: Dynamic Graph Algorithms and Graph Sparsification: New Techniques and Connections. Dissertation, University of Vienna, 2019.

[He97] Henzinger, M. R.: A Static 2-Approximation Algorithm for Vertex Connectivity and Incremental Approximation Algorithms for Edge and Vertex Connectivity. J. Algorithms, 24(1):194–220, 1997.

[KNZ14] Krauthgamer, R.; Nguyen, H. L.; Zondiner, T.: Preserving Terminal Distances Using Minors. SIAM J. Discrete Math., 28(1):127–141, 2014.

[KS16] Kyng, R.; Sachdeva, S.: Approximate Gaussian Elimination for Laplacians - Fast, Sparse, and Simple. In: Symposium on Foundations of Computer Science. S. 573–582, 2016.

[KT19] Kawarabayashi, Ken-ichi; Thorup, Mikkel: Deterministic Edge Connectivity in Near-Linear Time. J. ACM, 66(1):4:1–4:50, 2019.

[LRT79] Lipton, R.; Rose, D.; Tarjan, R.: Generalized Nested Dissection. SIAM Journal on Numerical Analysis, 16(2):346–358, 1979.

[MPX13] Miller, G.; Peng, R.; Xu, S. C.: Parallel Graph Decompositions Using Random Shifts. In: Symposium on Parallelism in Algorithms and Architectures. S. 196–203, 2013.

[RST14] Räcke, Harald; Shah, Chintan; Täubig, Hanjo: Computing Cut-Based Hierarchical Decompositions in Almost Linear Time. In: Symposium on Discrete Algorithms. S. 227–238, 2014.

[ST04] Spielman, D.; Teng, S.: Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In: Symposium on Theory of Computing. S. 81–90, 2004.

[Th07] Thorup, M.: Fully-Dynamic Min-Cut. Combinatorica, 27(1):91–127, 2007.

Gramoz Goranci wurde am 27. Juni 1990 in Gjakova geboren. Er studierte von 2008 bis 2011 im Bachelor-Studiengang Mathematik-Informatik an der Universität Prishtina sowie von 2012 bis 2014 im Master-Studiengang Informatik an der Technischen Universität München, wo er eine Abschlussarbeit im Bereich Graphalgorithmen unter der Betreuung von Harald Räcke schrieb. Von 2015 bis 2019 verfasste er seine Doktorarbeit über dynamische Graphalgorithmen und Graphsparsifikation unter der Betreuung von Monika Henzinger an der Universität Wien. Das Herbstsemester 2017 vebrachte er als Gastwissenschaftler am Georgia Institute of Technoloy. Seit Anfang 2020 forscht er als PostDoc an der Universität Toronto.

Variationsnetzwerke für die medizinische Bildrekonstruktion[1]

Kerstin Hammernik[2]

Abstract: Diese Arbeit befasst sich mit Variationsnetzwerken, die Konzepte aus Variationsmethoden und maschinellem Lernen vereinen, sowie deren Applikation für diverse Inverse Probleme in der Bildrekonstruktion von Computertomographie- oder Magnetresonanztomographieaufnahmen. Obwohl es sich um unterschiedlichste Applikationen handelt, haben diese etwas gemeinsam: Die Art der Artefakte, die bei der Datenakquisition entstehen, sind so strukturiert, dass sie mit gewöhnlichen Algorithmen nicht zufriedenstellend rekonstruiert werden können. Mit den vorgestellten Variationsnetzwerken können diese Artefakte erheblich reduziert und gleichzeitig die Bildqualität verbessert werden. Infolgedessen eröffnen die vorgestellten Variationsnetzwerke das Potential, direkt in den klinischen Arbeitsablauf integriert zu werden, um so Patientenkomfort sowie -sicherheit zu verbessern, sowie die Kosten im Gesundheitswesen zu reduzieren.

1 Einführung

Künstliche Intelligenz (KI) ist aus dem Alltag nicht mehr wegzudenken, jedoch ist KI für medizinische Anwendungen, wie der computergestützten Diagnostik, Quantifizierung von Biomarkern, Datenakquisition, Segmentierung und Bildrekonstruktion, wesentlich herausfordernder. Obwohl die ersten medizinischen Anwendungen von neuronalen Netzen in der Bildrekonstruktion schon in den 90er Jahren präsentiert wurden, erlangten diese erst in den letzten Jahren den Durchbruch, da nun die benötigte Rechenleistung sowie mehr Trainingsdaten für maschinelles Lernen zur Verfügung stehen. Die Resultate von neuronalen Netzen sind beeindruckend, jedoch bereiten diese auch Anlass zur Sorge: Was passiert, wenn die Maschine etwas fehlinterpretiert, Pathologien nicht erkennt oder gar fälschlicherweise Pathologien erfindet, die gar nicht vorhanden sind?

Trotz all dieser Fragen besteht ein großes Potential, wie maschinelles Lernen den medizinischen Sektor beeinflussen und verändern kann. Im Rahmen dieser Arbeit wird der Einsatz von maschinellem Lernen, insbesondere Variationsnetzwerken, im Bereich der medizinischen Bildrekonstruktion präsentiert. Dieser Forschungsbereich hat sich seit 2016 rasant weiterentwickelt und die Entwicklungen wurden innerhalb kürzester Zeit unüberschaubar. Die Verfassung der Dissertation [Ha19a] verlief parallel zu diesen Entwicklungen und umfasst folgende Beiträge zur Wissenschaft:

- Überblick und Klassifikation der Entwicklungen in der medizinischen Bildrekonstruktion basierend auf maschinellem Lernen im Rahmen eines Buchkapitels [HK20]

[1] Englischer Titel der Dissertation: "Variational Networks for Medical Image Reconstruction"

[2] Department of Computing, Imperial College London, k.hammernik@imperial.ac.uk

CC BY SA

- Variationsnetzwerke zur Nachbearbeitung von Computertomographie (CT) Rekonstruktionen mit limitierten Winkeln [Ha17]
- Variationsnetzwerke für die Rekonstruktion von statischen Magnetresonanztomographie (MRT) Daten klinischer Knieuntersuchungen, sowie radiologische Evaluierung der Bildqualität [Ha18, Kn19]
- Variationsnetzwerke in der dynamischen MRT Rekonstruktion des Herzens [Ha19b]

Wie den letzten beiden Punkten zu entnehmen ist, wurde besonderes Augenmerk auf echte, wenn möglich klinische Daten gelegt, um einen potentiellen Einsatz von Variationsnetzwerken im klinischen Alltag zu ermöglichen. In den folgenden Kapiteln werden die genannten Entwicklungen, sowie Variationsnetzwerke und deren Applikation anhand eines Beispiels aus der CT und MRT Bildgebung erörtert. Eine detaillierte Beschreibung findet sich in der Dissertation [Ha19a].

2 Maschinelles Lernen für die medizinische Bildrekonstruktion

Um die Anwendung von maschinellem Lernen in der Bildrekonstruktion zu veranschaulichen betrachtet man zunächst Bildrekonstruktion als Inverses Problem. Ziel von Inversen Problemen ist die Rekonstruktion eines Bildes $x \in \mathbb{K}^{N_x}$ aus Messdaten $y \in \mathbb{K}^{N_y}$, die mit Rauschen $n \in \mathbb{K}^{N_y}$ behaftet sind,

$$y = Ax + n, \tag{1}$$

wobei $\mathbb{K}$ die Menge der komplexen oder reellen Zahlen basierend auf dem Anwendungsbereich beschreibt und N_x sowie N_y die Dimensionen im Bild- und Datenbereich widerspiegeln. Beispiele aus der CT und MRT Bildgebung sind in Abbildung 1a dargestellt. Der Operator $A : \mathbb{K}^{N_x} \to \mathbb{K}^{N_y}$ beschreibt die Transformation zwischen Bild- und Messdatenbereich. Es ist naheliegend, Inverse Probleme mittels linearer Regression zu lösen, jedoch führt dies zur Überanpassung an das Rauschen in den Messdaten. Eine weitere Möglichkeit eine Lösung zu bestimmen ist die Anwendung von Variationsmethoden. Hierbei wird eine zusätzliche Regularisierung $\mathscr{R}[x]$ eingeführt, die Vorwissen über das zu rekonstruierende Bild einbringt. Das Variationsmodell lautet somit

$$\min_x \frac{\lambda}{2} \|Ax - y\|_2^2 + \mathscr{R}[x], \tag{2}$$

wobei der Parameter λ den Einfluss zwischen dem sogenannten Datenterm und dem Regularisierungsterm gewichtet. Jedoch sind die verwendeten Regularisierungsterme oft sehr einfach gestaltet und können die komplexe Struktur von medizinischen Bildern nicht hinreichend genug beschreiben. Des Weiteren werden diese Ansätze mittels iterativen Optimierungsverfahren gelöst, die oftmals rechen- und zeitintensiv sind. Maschinelles Lernen kann hier in mehreren Varianten zum Einsatz kommen, wie Abbildung 1b-d darstellt. Die erste Möglichkeit ist, Rekonstruktion als Nachbearbeitung zu betrachten. Hierzu wird zuerst eine Initiallösung berechnet, die dann mittels eines neuronalen Netzes verbessert wird.

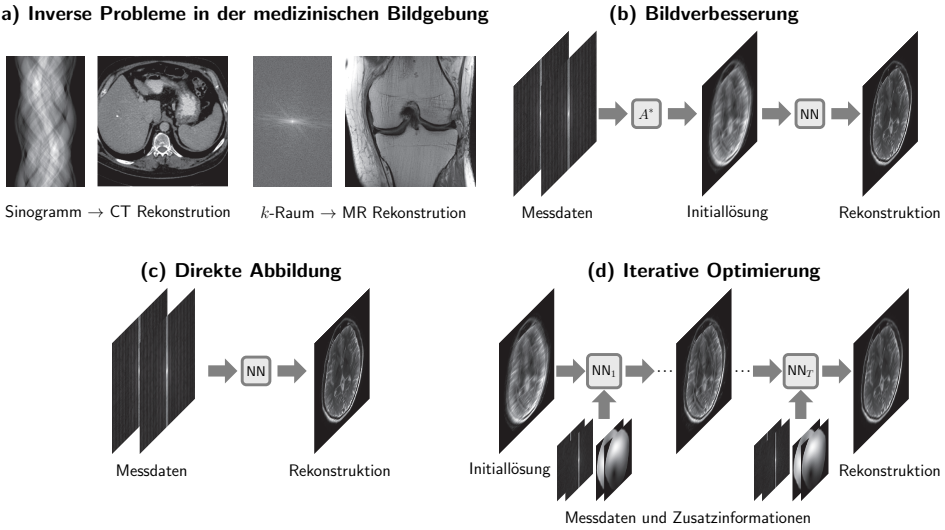


Abb. 1: Inverse Probleme in der medizinischen Bildgebung (a), sowie Möglichkeiten der Anwendung von neuronalen Netzen (NN) in der Bildrekonstruktion als Nachbearbeitung (b), direkte Abbildung (c) und gelernte iterative Optimierung (d).

Diese Methode eignet sich hervorragend zur schnellen und effizienten Artefaktkorrektur, jedoch muss beachtet werden, dass sie keine Ähnlichkeit zu den gemessenen Daten garantiert. Hier sei erwähnt, dass es in einigen Fällen auch keinen Zugang zu den Messdaten gibt. Eine zweite Möglichkeit befasst sich mit der direkten Rekonstruktion eines Bildes aus den Messdaten. Diese Methode ist durch einen enormen Ressourcenaufwand charakterisiert und kann ebenfalls keine Ähnlichkeit zu den gemessenen Daten garantieren. Die dritte Möglichkeit, die hier vorgestellt wird, setzt neuronale Netze in einem iterativen Optimierungsschema ein, das vollständig gelernt wird. Die im nächsten Teilabschnitt präsentierten Variationsnetzwerke stellen eine Schnittstelle zwischen klassischen Variationsmethoden sowie maschinellem Lernen dar und finden in der gelernten iterativen Optimierung Anwendung, sowohl als iterative Nachbearbeitung wie auch als iterative Rekonstruktion.

3 Variationsnetzwerke

Um Gleichung (1) zu lösen, kann eine Optimierung mittels Gradientenabstieges erfolgen. Um eine Überanpassung des Rauschens zu vermeiden, kann die Optimierung nach einer fixen Anzahl von Schritten abgebrochen werden [La51]. Dies kann als eine Form von Regularisierung gesehen werden. Eine weitere Form von Regularisierung wurde in Gleichung (2) eingeführt. In dieser Arbeit betrachten wir als generalisiertes Regularisierungsmodell das Fields-of-Experts (FoE) Modell [RB09, CYP15],

$$\mathscr{R}[x] = \sum_{i=1}^{N_k} \langle \rho_i(K_i x), \mathbf{1} \rangle ,$$

wobei ρ Potentialfunktionen, K lineare Filteroperatoren und N_k die Anzahl der Funktions-/Filter-Paare beschreibt. Wenn nun das fixierte Gradientenschema mit dem generalisierten Regulariserer kombiniert wird, erhält man ein Variationsnetzwerk [Ko17, Ha18]

$$x^t = x^{t-1} - \left(\sum_{i=1}^{N_k} (K_i)^\top \phi_i(K_i x^{t-1}) + \lambda A^*(Ax^{t-1} - y) \right), \quad 0 < t \leq T$$

das nun den Gradienten aus Gleichung (2) beinhaltet. Durch die Berechnung des Gradienten erhält man Aktivierungsfunktionen ϕ, die der ersten Ableitung von Potentialfunktionen ρ entsprechen. Der Operator A^* beschreibt den adjungierten Operator zu A. Der Parameter T gibt die Anzahl von Iterationen vor.

Netzwerkparameter Das Lernen der Parameter basiert auf Minimierung einer Verlustfunktion, die ein artefaktfreies Referenzbild mit dem vom Variationsnetzwerk rekonstruierten Bild vergleicht. Nach dem Training können neue Daten mit den gelernten Parametern, die die mit radialen Basisfunktionen parametrisierten Aktivierungsfunktionen, Filteroperatoren und den Parameter λ umfassen, höchst effizient rekonstruiert werden. Des Weiteren ist es möglich, die gelernten Parameter zu visualisieren. Dies gibt einen ersten Einblick darin, was gelernt wurde. Beispiele für gelernte Filteroperatoren, die als Filterkerne dargestellt werden können, und den dazugehörigen Aktivierungs- und Potentialfunktionen sind in Abbildung 2 ersichtlich. Es können diverse Ableitungsfilter sowie aus der Literatur bekannte Potentialfunktionen, wie zum Beispiel die L_1 Norm, beobachtet werden.

Regularisierungsmodelle Variationsnetzerke sind dadurch charakterisiert, dass das gesamte Regularisierungsmodell sowie das Optimierungsschema aus Trainingsdaten gelernt wird. Im Unterschied zu anderen Algorithmen, die neuronale Netze verwenden, erhalten Variationsnetzwerke durch die spezielle Struktur des Regularisierungstermes die Verbindung zur Energieminimierung aufrecht, wie dies in Abbildung 3 dargestellt ist. Der Regularisierungsterm kann somit jegliche Art von Energiefunktional beschreiben. In der vorliegenden Arbeit werden auch tiefe Regularisierungsterme vorgestellt, die markante Merkmale aus Bildern auf unterschiedlichen Ebenen extrahieren, während versucht wird, die Struktur so zu verschachteln, dass Informationen auf anderen Ebenen wiederverwendet werden können und somit Parameter gespart werden. Ein Beispiel dieses Regularisierungstermes liest sich für zwei Ebenen

$$\begin{aligned} \mathscr{R}_1[x] &= \langle \rho_1(K_1 x), \mathbf{1} \rangle, & K_1 &: \mathbb{K}^{N_x} \to \mathbb{K}^{N_1}, \rho_1 : \mathbb{K}^{N_1} \to \mathbb{K}^{N_1}, \\ \mathscr{R}_2[x] &= \mathscr{R}_1[x] + \langle \rho_2(K_2 \rho_1(K_1 x)), \mathbf{1} \rangle, & K_2 &: \mathbb{K}^{N_1} \to \mathbb{K}^{N_2}, \rho_2 : \mathbb{K}^{N_2} \to \mathbb{K}^{N_2}. \end{aligned}$$

Die Dimension N_l involviert die Anzahl der Merkmale N_{k_l} auf der l-ten Ebene, die Anzahl der Merkmale der $l-1$-ten Ebene sowie die Dimension des Bildraumes N_x, siehe [Ha19a] für Details. Im folgenden Abschnitt wird die Effizienz von Variationsnetzwerken sowie der präsentierten Regularisierungsmethoden anhand von Beispielen aus der CT und MRT Rekonstruktion dargestellt.

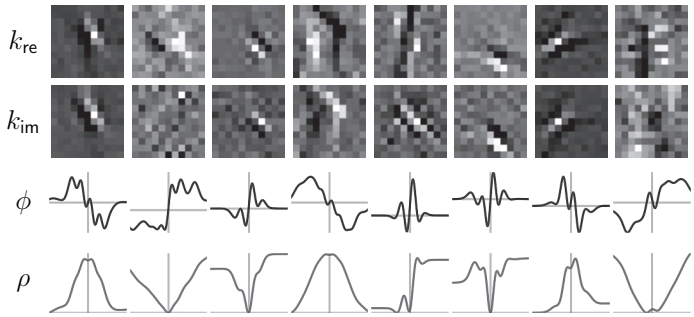


Abb. 2: Gelernte Parameter eines Variationsnetzwerkes aus der MRT Rekonstruktion. Es sind verschiedenste Ableitungsfilter sowie diverse Aktivierungsfunktionen ϕ dargestellt. Potentialfunktionen ρ ergeben sich durch Integration der Aktivierungsfunktionen und weisen Treppenfunktionen, L_1 Norm, konkave Funktionen, oder auch Funktionen mit mehreren lokalen Optima auf.

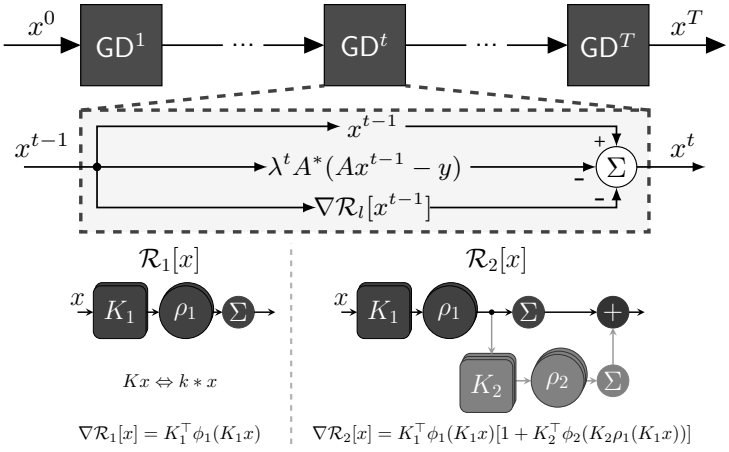


Abb. 3: Graphische Darstellung des Variationsnetzwerkes (oben) und der vorgestellten FoE (unten links) und tiefen Regularisierungsterme (unten rechts). Variationsnetzwerke verwenden den echten Gradienten eines Regularisierers, der mit einem Energiefunktional beschrieben ist.

4 Anwendungsbeispiel aus der Computertomographie

In diesem Abschnitt wird die Anwendung von Variationsnetzwerken für die Nachbearbeitung und Artefaktkorrektur von CT Aufnahmen vorgestellt [Ha17], bei denen die Projektionen nicht über den gesamten Winkelbereich aufgenommen werden können. Diese Art von CT Aufnahmen resultiert in Intensitätsinhomogenitäten, die im Sinogramm-Raum korrigiert werden, sowie charakteristischen Streifenartefakten im Bildraum, die einer nichtlinearen Filterung bedürfen. Jedoch haben viele klassische Methoden Schwierigkeiten, diese Streifenartefakte effizient zu entfernen.

Variationsnetzwerke werden in dieser Arbeit zur nichtlinearen Filterung und somit Artefaktkorrektur im Bildbereich verwendet. Die Initiallösung für die Variationsnetzwerke

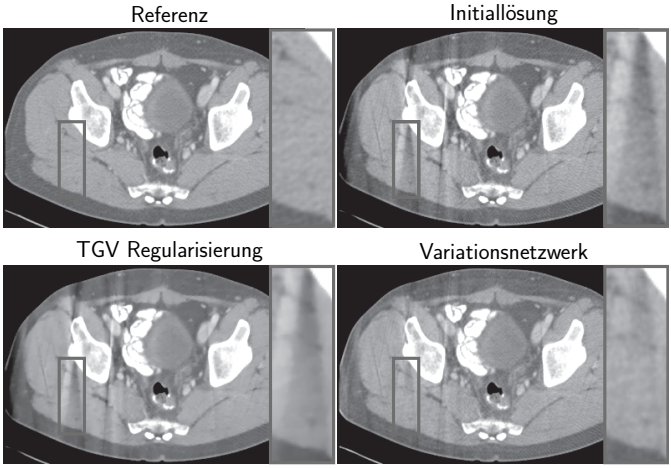


Abb. 4: Artefaktkorrektur im Bildbereich. Eine im Sinogramm-Raum vorkorrigierte Initiallösung wird mit einem klassischen Optimierungsverfahren basierend auf generalisierter Totalvariation (TGV) sowie mit einem Variationsnetzwerk rekonstruiert. Die Rekonstruktion basierend auf dem Variationsnetzwerk zeigt substantiell weniger Streifenartefakte.

sind von Intensitätsinhomogenitäten befreit, da sie bereits im Sinogramm-Raum basierend auf [Wü16] vorkorrigiert wurden. Es werden Variationsnetzwerke mit $T=5$ Gradientenschritten auf einer Trainingsdatenbank basierend auf simulierten Fächerstrahl CT Datensätzen von 10 Probanden trainiert. In jedem dieser Schritte werden $N_k=24$ Filterkerne mit den korrespondierenden Aktivierungsfunktionen gelernt. Ein exemplarisches Ergebnis in Abbildung 4 zeigt, dass das vorgestellte Variationsnetzwerk, hier trainiert mit einer Filterkerngröße von 13, die strukturierten Streifenartefakte effizienter filtern kann, als eine mit generalisierter Totalvariation [BKP10] regularisierte Rekonstruktion. Des Weiteren bedürfen Variationsnetzwerke keiner manuellen Einstellung von Parametern, da die Parameter gelernt werden. Diese Form der sequentiellen Artefaktkorrektur im Messdaten- und Bildbereich ist vielversprechend. Verbesserte Ergebnisse könnten durch gemeinsame Optimierung beider Artefaktkorrekturmethoden erzielt werden, dies bedarf jedoch der Verfügbarkeit der initialen Messdaten.

5 Anwendungsbeispiel aus der Magnetresonanztomographie

MRT Aufnahmen zeichnen sich durch einen exzellenten Weichteilkontrast aus und sind aus der neurologischen Diagnostik sowie der Diagnostik des Bewegungsapparates nicht mehr wegzudenken. Jedoch ist für Patienten die Aufnahme aufgrund des hohen Lärms, der durch die Schaltung magnetischer Gradienten entsteht, sowie der langen Aufnahmezeit, die aufgrund der komplexen physikalischen Prozesse zustande kommt, oft sehr unangenehm und bedarf Methoden, um die Aufnahmezeit zu beschleunigen. Eine Möglichkeit ist die Unterabtastung der Daten im k-Raum. Jedoch wird dadurch das Abtasttheorem verletzt, was im Falle einer kartesischen Abtastung strukturierte Artefakte im Bildbereich zur

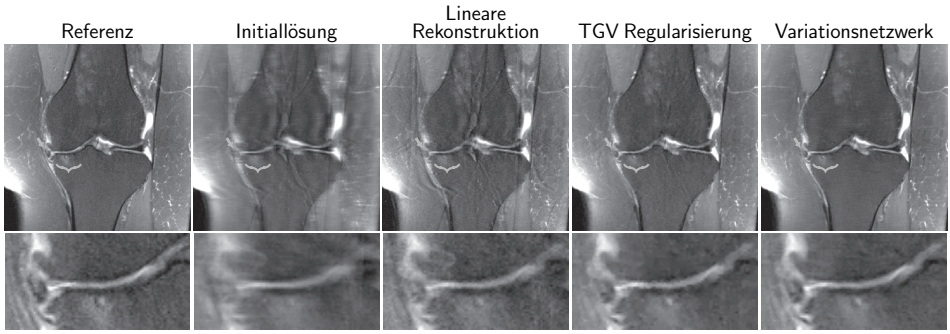


Abb. 5: Protonengewichtetes Knie MRT einer 57 jährigen Patientin, aufgenommen mit Fettsättigung in der Frontalebene. Der grüne Pfeil zeigt einen gerissenen medialen Meniskus, die grüne Klammer deutet auf eine zystische Veränderung im subchondralen Knochen hin. Die Rekonstruktion mit Variationsnetzwerken weist die beste Bildqualität und keine Restartefakte auf, verglichen mit linearer Rekonstruktion [Pr99] und Regularisierung mit generalisierter Totalvariation (TGV) [Kn11].

Folge hat. Auch hier können klassische Optimierungsalgorithmen nicht effizient eingesetzt werden, da sie oft eine hohe Laufzeit aufweisen und der Feineinstellung von Parametern bedürfen. Des Weiteren sind die verwendeten klassischen Regularisierungsterme wie generalisierte Totalvariation [Kn11] nicht für strukturierte Artefakte und somit kartesische Abtastschemata ausgelegt.

Variationsnetzwerke haben das Potential, die angeführten Nachteile von klassischen Optimierungsverfahren zu überwinden. Im Unterschied zur gezeigten CT Anwendung, sind hier sowohl die originalen Messdaten, die mit einem Spulenarray aufgenommen wurden, als auch die geschätzten Spulensensitivitäten [Ue14], die zur gewichteten Kombination der einzelnen Spulenbilder herangezogen werden, im Rekonstruktionsprozess miteinbezogen.

Im ersten Experiment werden Variationsnetzwerke auf 10 klinischen Kniedatensätzen, die eine Protonengewichtung mit Fettsättigung aufweisen und in der Frontalebene aufgenommen wurden, präsentiert. Das Variationsnetzwerk wurde auf 5 dieser Datensätze trainiert und weist $T=10$ Gradientenschritte auf, in denen je $N_k=24$ 11×11 Filter sowie deren Aktivierungsfunktionen gelernt wurden. Die Daten wurden retrospektiv mit einem Beschleunigungsfaktor von 4 abgetastet. Abbildung 5 zeigt, dass Variationsnetzwerke die strukturierten Artefakte im Vergleich zu klassischen Methoden [Pr99, Kn11] unterdrücken.

In dem zweiten hier gezeigten Experiment wird verdeutlicht, dass Regularisierung auf mehreren Ebenen weiter die Rekonstruktionsergebnisse verbessern kann. Hierzu wurde ein Variationsnetzwerk mit demselben FoE Regularisierer wie das erste Experiment gelernt. Für den tiefen Regulariser wurden $N_{k_1}=24$ 5×5 Filter sowie $N_{k_2}=48$ 3×3 Filter mit den dazugehörigen Aktivierungsfunktionen gelernt. Diesmal wurde auf 140 Kniedatensätzen aus dem kürzlich veröffentlichten fastMRI Datensatz [Kn20] trainiert. In Abbildung 6 werden die Ergebnisse dieses Experimentes auf protonengewichteten Knie MRTs gezeigt, die ebenfalls in der Frontalebene aufgenommen wurden. Diese Abbildung ver-

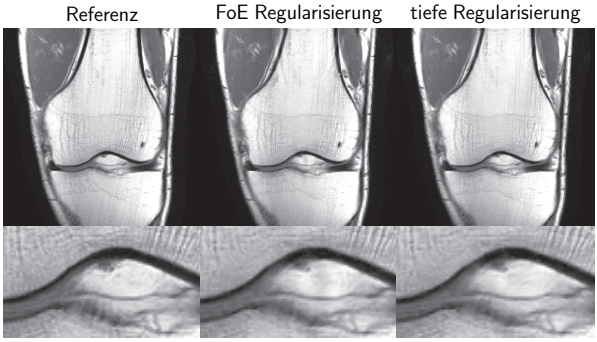


Abb. 6: Einfluss der FoE und tiefen Regularisierung in Variationsnetzwerken anhand eines protonengewichteten Knie MRTs in der Frontalebene. Das Rekonstruktionsergebnis basierend auf tiefer Regularisierung zeigt deutlich reduzierte Artefakte im Vergleich zur FoE Regularisierung.

deutlicht, dass Regularisier, die auf mehreren Ebenen arbeiten, Artefakte noch deutlicher verringern können als die FoE Regularisierung.

Die Qualität der hier präsentierten hochwertigen Ergebnisse wurde auch von Radiologen bestätigt. Die generelle Bildqualität, die Schärfe der Bilder, Restartefakte, sowie Signal-Rausch-Verhältnis wurden untersucht und für die hier gezeigten protonengewichteten MRT Daten in der Frontalebene, mit und ohne Fettsättigung, mit einem einseitigen Wilcoxon Vorzeichen-Rang Test mit Signifikanzniveau α=0.05 geprüft. Die Hypothese, dass Rekonstruktionen basierend auf Variationsnetzwerken den Rekonstruktionen basierend auf generalisierter Totalvariation [Kn11] überlegen sind, ist statistisch signifikant bestätigt [Ha18].

Die hier gezeigten Ergebnisse legen einen Grundstein für die beschleunigte MRT Rekonstruktion basierend auf maschinellem Lernen und zeigen eine Herangehensweise, um mit echten MRT Daten, wie sie im klinischen Alltag verwendet werden, umgehen zu können. Eine der größten Herausforderungen liegt dennoch darin, dass die Variationsnetzwerke basierend auf einer Verlustfunktion trainiert werden, die jedoch die komplexen Wahrnehmungen des menschlichen Auges nicht widerspiegeln können. Dies gestaltet auch die schlussendliche Evaluierung der Rekonstruktionen herausfordernd und bleibt somit weiterhin eine spannende Frage für weiterführende Forschungsarbeiten.

6 Schlussbemerkung

Während maschinelles Lernen für Klassifikation und Segmentierung sich in den letzten Jahren schon etablierte, ist die Anwendung in der medizinischen Bildrekonstruktion noch sehr neu. Betrachtet man die veröffentlichten Rekonstruktionmethoden im Detail, lässt sich erkennen, dass viele Rekonstruktion als Nachbearbeitung sehen, wofür häufig U-Netzwerke, die eine hohe Anzahl von Parametern aufweisen, verwendet werden. Dies hat ein hohes Risiko für Überanpassung der Trainingsdaten zur Folge, wenn nicht genügend Daten vorhanden sind, wie dies oft bei medizinischen Anwendung der Fall ist. Wenn Re-

konstruktion als iterative Optimierung betrachtet wird, besteht die Möglichkeit, Messdatenkonsistenz sowie vorhandenes Vorwissen zu integrieren. Somit können schon mit kleinen, kompakten Netzwerkarchitekturen und wenigen Trainingsdatensätzen hervorragende Resultate erzielt werden.

Ein Grundsatz der Arbeit mit Variationsnetzwerken beruht darauf, Struktur und jegliches Vorwissen in das Modell einzubringen, sowie die enge Verbindung zwischen klassischen Variationsmethoden und maschinellem Lernen näher zu untersuchen. Einen ersten Einblick dazu gibt die Visualisierung und Interpretation der gelernten Parameter wie sie in Abbildung 2 dargestellt sind. Des Weiteren wird angenommen, dass durch geschicktes Einbringen von Vorwissen und Struktur Variationsnetzwerke höchst kompetitiv mit anderen Algorithmen sind, die oftmals eine viel höhere Anzahl von Parametern aufweisen.

Hauptaugenmerk dieser Arbeit lag auf der Entwicklung einer Lösung, die auch im klinischen Alltag eingesetzt werden kann. Im Gegensatz zu vielen anderen Arbeiten, verwendet die vorliegende Dissertation rein klinische Daten für die MRT Rekonstruktion des Knies. Das Potential von Variationsnetzwerken ist aufgrund der qualitativen, quantitativen und radiologischen Bewertung der Bildqualität zu erkennen. Eine jedoch viel wichtigere Frage ist, ob den rekonstruierten Bildern vertraut werden und somit eine korrekte Diagnose gestellt werden kann, was Thema für weitere Forschungsarbeiten ist.

Die Dissertation [Ha19a] verschafft einen Überblick über die Entwicklungen in der medizinischen Bildrekonstruktion und zeigt die vielversprechende Anwendung von Variationsnetzwerken in medizinischen Fragestellungen. Die direkte Übertragung in den klinischen Alltag würden eine geringere Messdauer im MRT sowie eine reduzierte Strahlenbelastung im CT bedeuten, woraus sich unmittelbar ein positiver Einfluss auf Patientenkomfort und -sicherheit sowie eine Verringerung der Kosten im Gesundheitswesen ergeben.

Literaturverzeichnis

[BKP10] Bredies, K; Kunisch, K; Pock, T: Total Generalized Variation. SIAM Journal on Imaging Sciences, 3(3):492–526, 2010.

[CYP15] Chen, Y; Yu, W; Pock, T: On Learning Optimized Reaction Diffusion Processes for Effective Image Restoration. In: IEEE Conference on Computer Vision and Pattern Recognition. S. 5261–5269, 2015.

[Ha17] Hammernik, K; Würfl, T; Pock, T; Maier, A: A Deep Learning Architecture for Limited-Angle Computed Tomography Reconstruction. In: Bildverarbeitung für die Medizin 2017. S. 92–97, 2017.

[Ha18] Hammernik, K; Klatzer, T; Kobler, E; Recht, MP; Sodickson, DK; Pock, T; Knoll, F: Learning a Variational Network for Reconstruction of Accelerated MRI Data. Magnetic Resonance in Medicine, 79(6):3055–3071, 2018.

[Ha19a] Hammernik, K: Variational Networks for Medical Image Reconstruction. Dissertation, Graz University of Technology, 2019.

[Ha19b] Hammernik, K; Schloegl, M; Stollberger, R; Pock, T: Dynamic Multicoil Reconstruction using Variational Networks. In: Proceedings of the International Society of Magnetic Resonance in Medicine. S. 4656, 2019.

[HK20] Hammernik, K; Knoll, F: Machine Learning for Image Reconstruction. In (Rueckert, D; Fichtinger, G; Zhou, SK, Hrsg.): Handbook of Medical Image Computing and Computer Assisted Intervention. Elsevier, 2020.

[Kn11] Knoll, F; Bredies, K; Pock, T; Stollberger, R: Second Order Total Generalized Variation (TGV) for MRI. Magnetic Resonance in Medicine, 65(2):480–491, 2011.

[Kn19] Knoll, F; Hammernik, K; Kobler, E; Pock, T; Recht, MP; Sodickson, DK: Assessment of the Generalization of Learned Image Reconstruction and the Potential for Transfer Learning. Magnetic Resonance in Medicine, 81(1):116–128, 2019.

[Kn20] Knoll, F; Zbontar, J; Sriram, A; Muckley, MJ; Bruno, M; Defazio, A; Parente, M; Geras, KJ; Katsnelson, J; Chandarana, H; Zhang, Z; Drozdzalv, M; Romero, A; Rabbat, M; Vincent, P; Pinkerton, J; Wang, D; Yakubova, N; Owens, E; Zitnick, CL; Recht, MP; Sodickson, DK; Lui, YW: fastMRI: A Publicly Available Raw k-Space and DICOM Dataset of Knee Images for Accelerated MR Image Reconstruction Using Machine Learning. Radiology: Artificial Intelligence, 2(1):e190007, 2020.

[Ko17] Kobler, E; Klatzer, T; Hammernik, K; Pock, T: Variational Networks: Connecting Variational Methods and Deep Learning. In: German Conference on Pattern Recognition. S. 281–293, 2017.

[La51] Landweber, L: An Iteration Formula for Fredholm Integral Equations of the First Kind. American Journal of Mathematics, 73(3):615–624, 1951.

[Pr99] Pruessmann, KP; Weiger, M; Scheidegger, MB; Boesiger, P: SENSE: Sensitivity Encoding for Fast MRI. Magnetic Resonance in Medicine, 42(5):952–962, 1999.

[RB09] Roth, S; Black, MJ: Fields of Experts. International Journal of Computer Vision, 82(2):205–229, 2009.

[Ue14] Uecker, M; Lai, P; Murphy, MJ; Virtue, P; Elad, M; Pauly, JM; Vasanawala, S; Lustig, M: ESPIRiT – An Eigenvalue Approach to Autocalibrating Parallel MRI: Where SENSE meets GRAPPA. Magnetic Resonance in Medicine, 71(3):990–1001, 2014.

[Wü16] Würfl, T; Ghesu, FC; Christlein, V; Maier, A: Deep Learning Computed Tomography. In: International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI). S. 432–440, 2016.

Kerstin Hammernik absolvierte von 2007 bis 2015 das Studium der Biomedizinischen Technik mit dem Schwerpunkt medizinische Bildgebung und Bildverarbeitung an der Technischen Universität Graz. Nach Abschluss des Studiums war sie als wissenschaftliche Mitarbeiterin in der Gruppe von Prof. Thomas Pock am Institut für Maschinelles Sehen und Darstellen an der Technischen Universität Graz tätig, wo sie im Jahr 2019 mit Auszeichnung promovierte. Im Rahmen ihrer Arbeit absolvierte sie im Jahr 2017 einen 4-monatigen Forschungsaufenthalt an der New York University. Seit 2019 ist sie in der Forschungsgruppe von Prof. Daniel Rückert am Imperial College London tätig. Als Pionierin in ihrem Forschungsfeld des maschinellen Lernens für die medizinische Bildrekonstruktion hält sie regelmäßig Vorträge auf renommierten internationalen Kongressen und Institutionen.

Intelligentes, spielorientiertes Crowdsourcing für die Verarbeitung von Audiodaten[1]

Simone Hantke[2]

Abstract: Heutige Sprachassistenzsysteme sind ein fester Bestandteil unseres modernen Lebens. Der Erfolg dieser Technologien ist jedoch maßgeblich auf die Menge und Qualität der annotierten Trainingsdaten zurückzuführen. Dafür ist eine große Anzahl an Sprechern und Annotatoren erforderlich, und zudem erhebliche Investitionen notwendig, um diese Datenressourcen zu strukturieren und zu annotieren. Die Verfahren zur Datenerstellung sind kostspielig, zeitaufwendig und mühsam, so dass derzeit qualitativ hochwertig annotierte Daten knapp sind. In dieser Arbeit wird daher eine intelligente, Crowdsourcing-basierte Plattform mit Spielelementen und innovativen maschinellen Lernalgorithmen für die Datensammlung und Annotation vorgestellt. Es wurden diverse Audiodaten gesammelt und die Plattform in einer Vielzahl von Klassifikations- und Spracherkennungsstudien sowie mittels Perzeptionsstudien evaluiert. Durch die eingeführten Verfahren kann künftig die Audiodatenerfassung in erheblichem Maße beschleunigt sowie kostengünstiger und zuverlässiger durchgeführt werden.

1 Einführung und Motivation

Wir leben in technisch aufregenden und schnelllebigen Zeiten. Heutzutage ist es selbstverständlich, dass wir mit sprachgesteuerten Assistenzsystemen in unserem täglichen Leben in Kontakt kommen, da sie in Alltagstechnologien wie ALEXA, CORTANA und SIRI eingebettet sind.

Zu Beginn der Entwicklung automatischer Sprachanalysesysteme beschäftigte man sich mit relativ kleinen Datensätzen, die hauptsächlich zum Aufbau kleiner automatischer Systeme und zu dessen Validierung verwendet wurden [DBB52]. Im Laufe der Zeit hat sich viel geändert: Es wurden robuste und echtzeitfähige Systeme entwickelt und eine breite Palette an Algorithmen und Methoden des maschinellen Lernens unterstützt die Entwickler nun bei der Datenanalyse [Go16]. Der Erfolg von statistisch basierten Methoden zur automatischen Sprachverarbeitung wie z.B. Deep Learning [Sc15a] – welches im maschinellen Lernen schnell allgegenwärtig wurde – hat den Bedarf an größeren Eingangsdaten dramatisch erhöht. Diese Entwicklungen erfordern eine erhöhte Anzahl an Sprechern und Annotatoren und damit eine erhebliche Investition in Datenressourcen.

Obwohl eine große Menge an Daten frei verfügbarer ist, sind diese meist unstrukturiert und es mangelt ihnen an zuverlässigen Annotationen. Die Strukturierung und Annotation der Daten ist jedoch mit einem erheblichen zeitlichen und finanziellen Aufwand verbunden. Eine mögliche Lösung ist in Form einer Technik namens ‘Crowdsourcing’ [Ho06] gekommen. Hierbei werden Annotationsaufgaben an eine nicht spezifische Gruppe von

[1] Englischer Titel der Dissertation: ”Intelligent Gamified Crowdsourcing for Audio Processing”

[2] MISP Group, Technische Universität München, Germany, simone.hantke@tum.de

cc by sa

Personen ins Internet auslagert, wobei die Annotatoren meisten nicht speziell ausgebildet sind. Auf diese Weise kann Crowdsourcing überall auf der Welt und zu jeder Zeit für viele verschiedene Interessensbereiche genutzt werden und bietet sofortigen Zugang zu einer breiten und vielfältigen Palette an Personen mit unterschiedlichen Hintergründen, Kenntnissen und Fähigkeiten.

Defizite der aktuellen Technik Obwohl heutige intelligente Systeme autonomer arbeiten denn je, erfordern sie immer noch menschliche Interaktion. Nicht alle Aspekte der menschlichen Intelligenz können eigenständig verarbeiten werden. Der weiterhin benötigte Annotationsvorgang ist jedoch eine kostspielige, zeitaufwändige und mühsame Angelegenheit, die nicht jeder investieren möchte [Ra10]. Dies hat zu einer Verknappung an annotierten Daten geführt und dadurch das Wachstum und den Erfolg bei der Entwicklung von intelligenten Systemen verlangsamt. Ein effizienterer Prozess der Datensammlung und Datenannotation wäre hier ein großer Gewinn. Es fehlt ein automatisiertes kombiniertes System, um multimodale Daten zu beschaffen, neue Datensätze aufzuzeichnen und, was noch wichtiger ist, die Daten zeit- und kosteneffizienter zu annotieren.

Zielsetzung und Auflistung der Beiträge In Anbetracht der Defizite modernster Technologien besteht der Hauptbeitrag der Dissertation [Ha19] in der bahnbrechenden intelligenten, spielorientierten Crowdsourcing Plattform iHEARu-PLAY, mit welcher schneller, kosteneffizienter und zuverlässiger Sammlung und Annotation audiovisueller Daten erfolgen können. Als Alternative zu herkömmlichen Crowdsourcing Plattformen geht iHEARu-PLAY über den aktuellen Stand der Technik hinaus und motiviert Teilnehmer, indem es ihnen eine spielerische und unterhaltsame Umgebung bietet, in der die Spieler freiwillig zu wissenschaftlichen Forschungsprojekten beitragen können. Als Kernkomponente enthält die Plattform erweiterte und verbesserte Algorithmen für maschinelles Lernen, um die Genauigkeit der manuellen Annotation mit der Geschwindigkeit und Kosteneffizienz von maschinellem Lernen auf der Basis von Klassifikatoren zu kombinieren. Indem nur dann um manuelle Annotation gebeten wird, wenn das System selbst nicht in der Lage ist, die Daten zu annotieren, kann der manuelle Arbeitsaufwand auf ein Minimum beschränkt werden. Darüber hinaus werden hochqualitativere Annotationen gesammelt, da mehrere Qualitätsbewertungsmerkmale eingeführt werden. Diese berechnen zuverlässig beispielsweise die Intra–Annotator– und Inter–Annotator–Übereinstimmung durch die neu entwickelte Vertrauenswürdigkeitsbewertung eines Annotators.

2 iHEARu-PLAY: Intelligente, spielorientierte Crowdsourcing Plattform

Kern dieser Arbeit ist es die modulare, intelligente, spielorientierte Crowdsourcing Plattform iHEARu-PLAY[3] [Ha15] vorzustellen. Die browserbasierte Plattform läuft auf jedem Standard-PC oder Smartphone und bietet Audio-, Video- und Bildannotation für eine Vielzahl von Annotationsaufgaben sowie audio(-visuelle) Datensammlung und Analysen.

[3] https://www.ihearu-play.eu

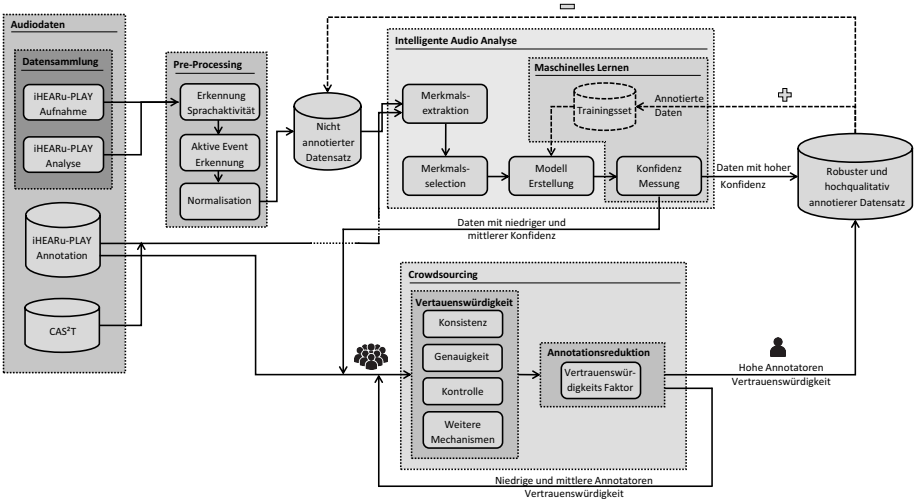


Abb. 1: *In* iHEARu-PLAY *integrierte Interaktion zwischen den Komponenten der intelligenten Audioanalyse, dem aktiven Lernen und Qualitätsmanagement, einschließlich der Vertrauenswürdigkeitsberechnung und der Komponente zur Annotationsreduktion; nach [HZS17, HAS18a].*

Das System hat eine Reihe von Vorteilen gegenüber herkömmlichen Crowdsourcing Plattformen, unter anderem die Anwendung und Integration verschiedener Spielelemente [HAS18b]. In diesem Zusammenhang umfasst die Plattform eine Kombination aus Punkten, Ranglisten, Abzeichen und eine soziale Kommunikationsplattform, um die alltägliche Arbeit des Annotierens zu einer angenehmeren und motivierenden Aufgabe zu machen. Darüber hinaus wurde ein faires, verständliches und offenes Datenschutzkonzept für die Sammlung, Speicherung, Nutzung und Weitergabe von Daten entwickelt, um Missbrauch zu vermeiden und die Anonymität der Annotatoren zu gewährleisten, während gleichzeitig sichergestellt wird, dass unbefugte Dritte keinen Zugang erhalten. Neben diesen Merkmalen ist das Hauptcharakteristikum von iHEARu-PLAY – das es von herkömmlichen Crowdsourcing Plattformen unterscheidet – die Integration mehrerer vertrauenswürdigkeitsbasierter Algorithmen des maschinellen Lernens, die darauf abzielen, die manuelle Annotationsarbeit zu reduzieren. Eine schematische Übersicht über die kombinierten intelligenten Komponenten ist in Abbildung 1 dargestellt.

3 Vertrauenswürdigkeitsbasiertes kooperatives Lernen

Trotz der vielversprechenden Möglichkeiten bedeutet die Online–Rekrutierung von Annotatoren immer noch einen großen Aufwand und es müssen mindestens so viele Annotationen gesammelt werden, wie es nicht annotierte Dateninstanzen gibt. In jüngster Zeit wurden mehrere intelligente Ansätze vorgestellt, um die Belastung der Annotatoren durch nicht annotierte Daten zu reduzieren, wobei einer der vielversprechendsten Ansätze das sogenannte AKTIVE LERNEN (AL) ist [Zh15]. Das Konzept von AL basiert auf der Idee, dass der Algorithmus die Klassifikationsgenauigkeit mit so wenig Trainingsdaten wie möglich

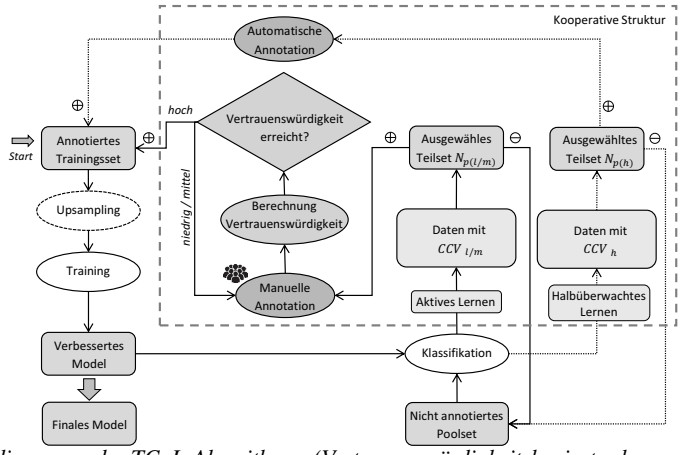


Abb. 2: *Flussdiagramm des TCoL-Algorithmus (Vertrauenswürdigkeitsbasiertes kooperatives Lernen); CCV ist der Vertrauenswert des Klassifikators. Angepasst und erweitert von [Zh15].*

verbessern kann, indem aktiv die Daten ausgewählt werden, bei denen der Algorithmus am sichersten ist [TK11]. Dies bedeutet, dass nur die Daten manuell annotiert werden müssen, für die das trainierte Modell kein Label vorhersagen kann. Dies führt zu einer allgemeinen Reduktion der benötigten Annotationen (CR), wobei eine mindestens gleichwertige Klassifikationsgenauigkeit erzielt werden kann [HZS17].

Daher wird die Anwendbarkeit des entwickelten vertrauenswürdigkeitsbasierten kooperativen Lernalgorithmus (TCoL) untersucht, der AL mit halbüberwachtem Lernen (SSL) auf Basis der Vorhersageunsicherheit bei Support Vector Machines (SVMs) integriert. Dieser ermöglicht eine optimale Aufteilung von manuellen und automatischen Annotationstätigkeiten. Der schematische Ablauf ist in Abbildung 2 dargestellt.

Exemplarische Studie

Der entwickelte TCoL-Algorithmus wurde in einer Reihe von Studien zur Emotionserkennung mit dem *FAU Aibo Emotion Corpus* [St09] evaluiert, wobei sowohl konventionelle als auch auf Vertrauenswürdigkeit basierende Annotationen verwendet wurden. In den Experimenten wurden über 15.000 Audio-Instanzen verwendet mit insgesamt nahezu 7,5 Stunden kontinuierlicher Sprache. 14 Annotatoren (3 weibliche und 11 männliche, im Alter zwischen 20 und 27 Jahren; drei gaben ihr Alter nicht an) annotierten die *FAU Aibo Emotion Corpus* Instanzen in die vorgeschlagenen Emotionsklassen [St09].

Es wurde das IS09-Merkmals-Set [SSB09] angewandt, das auf eine hohe Robustheit für die menschliche Emotionserkennung ausgelegt ist. Die Audiomerkmale wurden mit dem *openSMILE*-Toolkit [Ey13] extrahiert und es wurde die Open-Source-Software für maschinelles Lernen und Data-Mining *WEKA* [Ha09] eingesetzt. Eine SVM mit einem SMO-Algorithmus und einer Komplexitätskonstante C von 0,1 wurde trainiert, um eine Hyperebene zu konstruieren, die die Instanzen verschiedener Klassen trennt. Ein passiver Lernansatz (PL), welcher alle Daten zufällig annotiert, war die Vergleichsbasis.

Die Modelle wurden zunächst mit 200 zufällig ausgewählten Instanzen trainiert, während die restlichen Daten als nicht annotierter Datenpool verwendet wurde. Bei jeder Iteration wählte der Algorithmus 200 Instanzen für die manuelle Annotation aus dem Pool-Set aus. Es wurde ein zufälliges Upsampling angewandt, bei dem Kopien vorhandener Instanzen zu den Klassen hinzugefügt werden, die eine nur geringe Anzahl von Instanzen haben [Pr00]. Je nach Algorithmus wurde nach dem AL-Schritt ein SSL-Schritt durchgeführt, bei dem automatisch 200 Instanzen aus dem Pool-Set annotiert wurden. Der Lernprozess jedes Algorithmus wurde gestoppt, wenn das Leistungsplateau erreicht wurde. Dies wurde mit Hilfe des Unweighted Average Recalls (UAR) bewertet. Um statistische Ungenauigkeiten auszuschließen, wurde jeder Algorithmus außerdem 20 Mal mit zufällig ausgewählten Anfangs-Trainingsinstanzen wiederholt.

Ergebnisauszug und Diskussion

Abbildung 3 zeigt, dass bei allen Ansätzen das sequentielle Hinzufügen von manuell annotierten Instanzen zu einem anfänglichen Trainingsset zu einer kontinuierlichen Verbesserung des Klassifikators führt. Zudem steigt der UAR bei allen Algorithmen zunächst mit der Anzahl der manuellen Annotationen steil an und stagniert anschließend. Die Ergebnisse zeigen, dass alle konventionellen Ansätze PL mit einem UAR von 54,17 % übertreffen. PL erforderte 15.000 Annotationen, um einen maximalen UAR von 57,18 % für die vertrauenswürdigkeitsbasierten Experimente (TCoL) und 54,17 % für die konventionellen zu erreichen. Der TCoL Ansatz reduzierte die Annotationslast erheblich, wobei die relative Kostenreduktion CR bis zu 72 % betrug, während der UAR bei 71,17 % lag. Während die einzelnen (T)AL und SSL Techniken ungefähr den gleichen UAR wie die (T)CoL Algorithmen erreichten, war eine wesentlich höhere Anzahl von Annotationen erforderlich. Daher hat der entwickelte TCoL Algorithmus klare Vorteile gegenüber PL, AL und SSL Ansätzen, da er die Anzahl der erforderlichen manuellen Annotationen effektiv reduziert.

4 Datensammlungen

Aktuelle Technologien bieten die Möglichkeit über das Internet massenhaft neue Daten zu sammeln, wobei in Laptop-PCs, Tablets und Smartphones eingebettete Mikrofone genutzt werden. Aufgrund dessen ist es heutzutage möglich, Sprachdaten unter realen Bedingungen (z.B. mit verschiedenen Mikrofontypen oder Hintergrundgeräuschen) von einer großen Anzahl an Sprechern mit unterschiedlicher geographischer oder kultureller Herkunft, Sprachen oder Altersgruppen zu sammeln. In diesem Zusammenhang wurde in IHEARU-PLAY eine Aufnahmefunktion integriert, die eine multimodale, großflächige Datenerfassung auf ressourcenschonende Weise ermöglicht. Die Anwender können ein breites Spektrum an Aufnahmeaufgaben durchführen und haben die Möglichkeit, das interaktive, webbasierte Sprachklassifizierungs-Framework VOILA zu nutzen [Ha18b]. Dabei können Nutzer den Klassifizierungsprozess verbessern, indem sie qualitativ hochwertig annotierte Sprachdaten zur Verfügung stellen und gleichzeitig ihre Stimme in Bezug auf verschiedene Sprechereigenschaften analysieren lassen. Wann immer Daten jedoch spezielle Anforderungen erfüllen sollen, müssen diese auf konventionelle Weise erhoben werden. In diesem Zusammenhang wurden vier Datensätze gesammelt, von denen zwei nun näher beleuchtet werden.

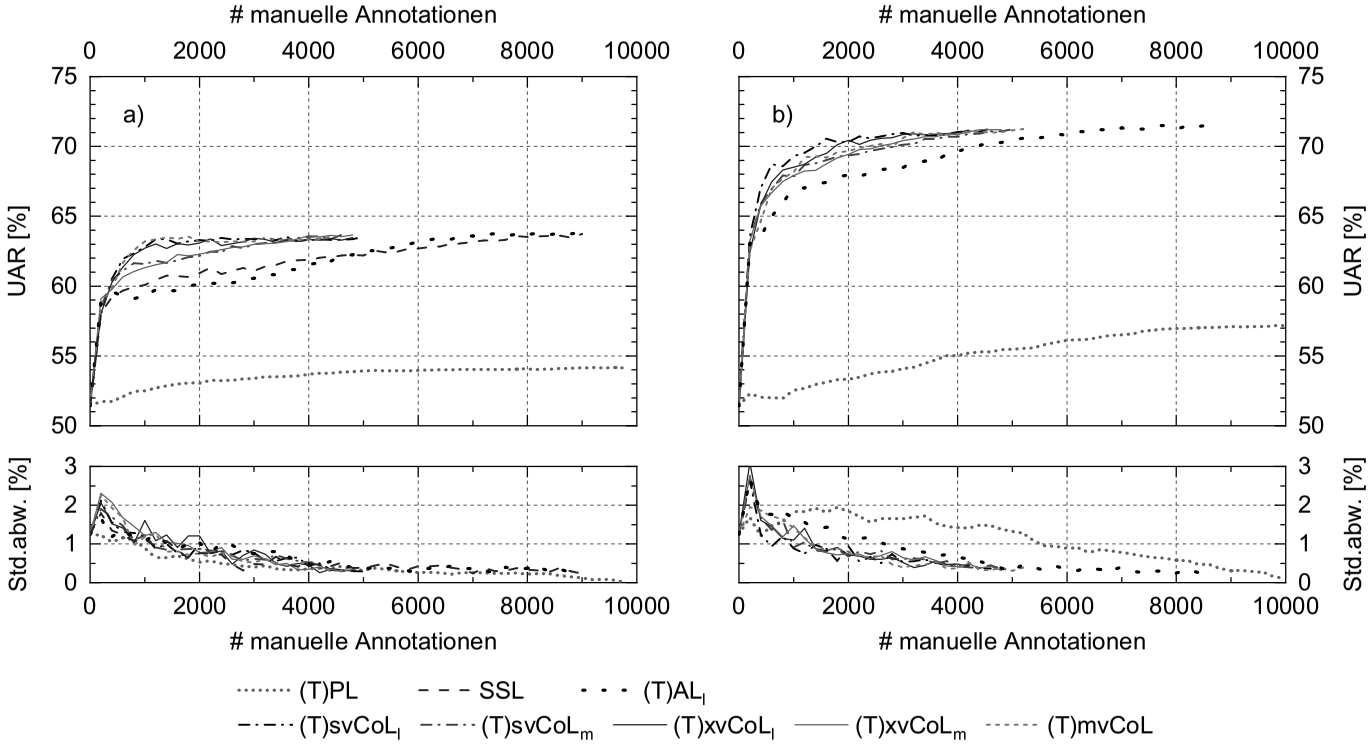


Abb. 3: *Vergleich von Passivem Lernen (PL), halbüberwachtem Lernen (SSL), aktivem Lernen (AL) und den verschiedenen (vertrauenswürdigkeitbasierenden) kooperativen Lernansätzen (T)CoL auf dem FAU Aibo Emotion Corpus mit binären Emotionsklassen [HSS18]. Die Ergebnisse zeigen die über 20 unabhängigen Durchläufe der Algorithmen gemittelten Werte des unweighted average recall (UAR) [%] im Vergleich zur Anzahl der manuell ausgewählten Annotationen. Es werden Ergebnisse mit konventionellen Annotationen (a) und mit auf vertrauenswürdigkeitbasierenden Annotationen (b) gezeigt.*

iHEARu-EAT Der iHEARu-EAT Datensatz ist der erste seiner Art. Es wurden 30 Sprecher beim Verzehr von sechs verschiedenen Lebensmitteln aufgenommen und dabei 1.414 Instanzen und 2,9 Stunden audio-visuelle Daten aufgezeichnet. Es wurde eine 7-stufige Klassifikationsstudie durchgeführt, welche die automatische Klassifizierung der Essbedingungen während des Sprechens demonstrierte. Schließlich ist der Datensatz – mit 14 teilnehmenden Teams – erfolgreich als Eating Condition (EC) Sub-Challenge in die Interspeech 2015 ComParE Challenge [Sc15b] und ist als einziger Datensatz in die ICMI 2018 EAT Challenge [Ha18c] eingeflossen.

EmotAsS Der EmotAsS Datensatz enthält spontan aufgezeichnete emotionale Sprache von 17 geistig, neurologisch und/oder körperlich eingeschränkten Personen [Ha17]. Der Datensatz enthält ca. 12.700 Instanzen mit knapp 11 Stunden Material. Die automatische Analyse der spontanen Emotionen dieser Personen wurde mit dem Ziel durchgeführt, ein sprachgesteuertes, arbeitsplatzintegriertes Assistenzsystem für Menschen mit Einschränkungen zu entwickeln [Ha18a]. Es wurde gezeigt, dass diese atypische Sprachanalyse eine anspruchsvolle Aufgabe ist, die jedoch mit den in der Arbeit vorgestellten Methoden bewältigt werden konnte. Der Datensatz ist schließlich in die renommierten Interspeech 2018 ComParE Challenge [Sc18] eingeflossen.

5 Perzeptionsstudien

Neben den vorgestellten Arbeiten zur Entwicklung und Evaluierung der intelligenten spieleorientierten Crowdsourcing Plattform und zur Sammlung verschiedener Sprachdatensätze wurde iHEARu-PLAY auch für die Durchführung von Perzeptionsstudien genutzt, welche die Vielseitigkeit der Plattform demonstrieren: (i) die Bewertung der automatischen Erkennung des Kontexts und der wahrgenommenen Emotion von Hundebellen [HCS18], (ii) die Untersuchung der menschlichen Wahrnehmung von Stimmzügen bei synthetischen Stimmen [Ba18], (iii) die Bewertung der Wahrnehmung von Emotionen in der Singstimme [Pa18], und (iv) die Untersuchung der menschlichen Wahrnehmung von Sprachemotion in Störgeräuschen [Pa17].

6 Zusammenfassung und Ausblick

Heutige maschinelle Lernalgorithmen sind von der Verfügbarkeit qualitativ hochwertig annotierter Trainingsdaten abhängig. Obwohl es aufgrund moderner Technologien eine Vielzahl verschiedener, frei verfügbarer Daten gibt, können diese häufig nicht in ihrer Rohform übernommen werden, da sie oft unstrukturiert sind und es an zuverlässigen Annotationen fehlt. In Anbetracht der Defizite modernster Technologien zur effizienten Datenannotation waren die wichtigsten Beiträge der Arbeit die bahnbrechenden Einführungen, die mit der vorgestellten intelligenten, spielorientierten Crowdsourcing Plattform iHEARu-PLAY verbunden sind. Die Plattform ermöglicht eine schnellere, kosteneffizientere und zuverlässigere Datenerfassung und Datenannotation, als bisher möglich. In diesem Zusammenhang leistet die vorliegende Forschungsarbeit vier wesentliche Beiträge:

1. Den Entwurf und die Realisierung einer intelligenten Crowdsourcing Plattform insbesondere zur Steigerung der Qualität und Effizienz der Annotation und Sammlung von audiovisuellen Daten für das maschinelle Lernen.

2. Die Einbeziehung und Erweiterung von über den Stand der Technik hinausgehenden Algorithmen des maschinellen Lernens und daraus resultierende neue Verfahren zur Bewertung der Verlässlichkeit geleisteter Annotationen.
3. Die Erstellung und Sammlung von neuen Datensätzen für das Lernen intelligenter audiovisueller Mustererkennungsverfahren, die von Social-Media-Websites übernommen und konventionell in Studien gesammelt werden.
4. Bewertung und Evaluierung der vorgestellten Verfahren, Umsetzungen und Sammlungen von Daten und Annotationen.

Zusammenfassend lässt sich sagen, dass die im Rahmen dieser Arbeit durchgeführte Forschung vielversprechende Ergebnisse und neuartige Methoden geliefert hat, indem ein automatisiertes, kombiniertes und robustes System zur Verfügung gestellt wurde, um neue Datensätze zu beschaffen, aufzuzeichnen und vor allem auf eine sehr effiziente Weise zu annotieren. Als Ergebnis können zuverlässige Trainingsdaten für die Audioverarbeitung schneller, ressourcenschonender und damit kostengünstiger gesammelt werden.

Zukünftige Arbeiten könnten auf den entwickelten vertrauenswürdigkeitsbasierenden Algorithmen des maschinellen Lernens basieren. Durch Modellierung eines Annotators könnte es den intelligenten Algorithmen ermöglicht werden nicht nur selbst zu entscheiden, wann manuelle Annotationsunterstützung erforderlich ist, sondern auch, handverlesene Annotatoren um ihre Meinung zu bitten, abhängig von den bestehenden Vertrauenswürdigkeitsbewertungen jedes Annotators. Schließlich soll iHEARu-PLAY für externe Forscher öffentlich zugänglich gemacht werden, damit diese ihre eigenen Aufgaben und Datensätze über die neue Schnittstelle hochladen, sammeln, erstellen, bearbeiten und löschen können. In dieser Hinsicht werden sich die Ergebnisse dieser Arbeit auf die nächste Generation intelligenter maschineller Lernansätze auswirken. iHEARu-PLAY wird letztlich den langfristigen Mangel eines geeigneten, effizienten und intelligenten Datenerfassungs- und Annotationswerkzeugs im Bereich des überwachten maschinellen Lernens beheben.

Literaturverzeichnis

[Ba18] Baird, Alice; Jørgensen, Stina; Parada-Cabaleiro, Emilia; Hantke, Simone; Cummins, Nicholas; Schuller, Björn: Listener Perception of Vocal Traits in Synthesized Voices: Age, Gender, and Human-Likeness. Journal of the Audio Engineering Society, 66:1–8, 2018.

[DBB52] Davis, KH; Biddulph, R; Balashek, Stephen: Automatic Recognition of Spoken Digits. The Journal of the Acoustical Society of America, 24:637–642, 1952.

[Ey13] Eyben, Florian; Weninger, Felix; Groß, Florian; Schuller, Björn: Recent Developments in OpenSMILE, the Munich Open-Source Multimedia Feature Extractor. In: Proc. of Int. Conference on Multimedia (ACMMM). Barcelona, Spain, S. 835–838, 2013.

[Go16] Goodfellow, Ian; Bengio, Yoshua; Courville, Aaron; Bengio, Yoshua: Deep Learning. MIT press, Cambridge, UK, 2016.

[Ha09] Hall, Mark A.; Frank, Eibe; Holmes, Geoffrey; Pfahringer, Bernhard; Reutemann, Peter; Witten, Ian H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations, 11:10–18, 2009.

[Ha15] Hantke, Simone; Eyben, Florian; Appel, Tobias; Schuller, Björn: iHEARu-PLAY: Introducing a Game for Crowdsourced Data Collection for Affective Computing. In: Proc. of Workshop on Automatic Sentiment Analysis in the Wild, Satellite of Int. Conference on Affective Computing and Intelligent Interaction. Xi'an, P. R. China, S. 891–897, 2015.

[Ha17] Hantke, Simone; Sagha, Hesam; Cummins, Nicholas; Schuller, Björn: Emotional Speech of Mentally and Physically Disabled Individuals: Introducing the EmotAsS Database and First Findings. In: Proc. of INTERSPEECH. Stockholm, Sweden, S. 3137–3141, 2017.

[Ha18a] Hantke, Simone; Cohrs, Christian; Schmitt, Maximilian; Tannert, Benjamin; Lütkebohmert, Florian; Detmers, Mathias; Schelhowe, Heidi; Schuller, Björn: Introducing an Emotion-driven Assistance System for Cognitively Impaired Individuals. In: Proc. of Int. Conference on Computers Helping People with Special Needs (ICCHP). Linz, Austria, S. 486–494, 2018.

[Ha18b] Hantke, Simone; Olenyi, Tobias; Hausner, Christoph; Schuller, Björn: VoiLA: An Online Intelligent Speech Analysis and Collection Platform. In: Proc. of Asian Conference on Affective Computing and Intelligent Interaction. Beijing, China, S. 1–5, 2018.

[Ha18c] Hantke, Simone; Schmitt, Maximilian; Tzirakis, Panagiotis; Schuller, Björn: EAT — The ICMI 2018 Eating Analysis and Tracking Challenge. In: Proc. of Int. Conference on Multimodal Interaction (ICMI). Boulder, USA, S. 1–5, 2018.

[Ha19] Hantke, Simone: Intelligent Gamified Crowdsourcing for Audio Processing. Dissertation, Dissertation, Technische Universität München, 2019.

[HAS18a] Hantke, Simone; Abstreiter, Alexander; Schuller, Björn: Trustability-based Dynamic Active Learning for Crowdsourced Labelling of Emotional Audio Data. IEEE Access, S. 13, 2018. submitted, under review.

[HAS18b] Hantke, Simone; Appel, Tobias; Schuller, Björn: The Inclusion of Gamification Solutions to Enhance User Enjoyment on Crowdsourcing Platforms. In: Proc. of Asian Conference on Affective Computing and Intelligent Interaction. Beijing, China, S. 1–6, 2018.

[HCS18] Hantke, Simone; Cummins, Nicholas; Schuller, Björn: What is my Dog Trying to Tell me? The Automatic Recognition of the Context and Perceived Emotion of Dog Barks. In: Proc. of Int. Conference on Acoustics, Speech and Signal Processing (ICASSP). Calgary, Canada, S. 1–5, 2018.

[Ho06] Howe, Jeff: The Rise of Crowdsourcing. Wired Magazine, 14:1–4, 2006.

[HSS18] Hantke, Simone; Stemp, Christoph; Schuller, Björn: Annotator Trustability-based Cooperative Learning Solutions for Intelligent Audio Analysis. In: Proc. of INTERSPEECH. Hyderabad, India, S. 3504–3508, 2018.

[HZS17] Hantke, Simone; Zhang, Zixing; Schuller, Björn: Towards Intelligent Crowdsourcing for Audio Data Annotation: Integrating Active Learning in the Real World. In: Proc. of INTERSPEECH. Stockholm, Sweden, S. 3951–3955, 2017.

[Pa17] Parada-Cabaleiro, Emilia; Baird, Alice; Batliner, Anton; Cummins, Nicholas; Hantke, Simone; Schuller, Björn: The Perception of Emotions in Noisified Non-Sense Speech. In: Proc. of INTERSPEECH. Stockholm, Sweden, S. 3246–3250, 2017.

[Pa18] Parada-Cabaleiro, Emilia; Schmitt, Maximilian; Batliner, Anton; Hantke, Simone; Costantini, Giovanni; Scherer, Klaus; Schuller, Björn: Identifying Emotions in Opera Singing: Implications of Adverse Acoustic Conditions. In: Proc. of Int. Conference on Society for Music Information Retrieval (ISMIR). Paris, France, 2018.

[Pr00] Provost, Foster: Machine Learning from Imbalanced Data Sets. In: Working Notes of the AAAI Workshop on Learning from Imbalanced Data Sets. Austin, USA, S. 1–3, 2000.

[Ra10] Raykar, Vikas; Yu, Shipeng; Zhao, Linda H; Valadez, Gerardo Hermosillo; Florin, Charles; Bogoni, Luca; Moy, Linda: Learning from Crowds. Journal of Machine Learning Research, 11:1297–1322, 2010.

[Sc15a] Schmidhuber, Jürgen: Deep Learning in Neural Networks: An Overview. Neural Networks, 61:85–117, 2015.

[Sc15b] Schuller, Björn; Steidl, Stefan; Batliner, Anton; Hantke, Simone; Hönig, Florian; Orozco-Arroyave, Juan Rafael; Nöth, Elmar; Zhang, Yue; Weninger, Felix: The INTERSPEECH 2015 Computational Paralinguistics Challenge: Degree of Nativeness, Parkinson's and Eating Condition. In: Proc. of INTERSPEECH. Dresden, Germany, S. 478–482, 2015.

[Sc18] Schuller, Björn; Steidl, Stefan; Batliner, Anton; Marschik, Peter B.; Baumeister, Harald; Dong, Fengquan; Hantke, Simone; Pokorny, Florian; Rathner, Eva-Maria; Bartl-Pokorny, Katrin D.; Einspieler, Christa; Zhang, Dajie; Baird, Alice; Amiriparian, Shahin; Qian, Kun; Ren, Zhao; Schmitt, Maximilian; Tzirakis, Panagiotis; Zafeiriou, Stefanos: The INTERSPEECH 2018 Computational Paralinguistics Challenge: Atypical & Self-Assessed Affect, Crying & Heart Beats. In: Proc. of INTERSPEECH. Hyderabad, India, S. 122–126, 2018.

[SSB09] Schuller, Björn; Steidl, Stefan; Batliner, Anton: The Interspeech 2009 Emotion Challenge. In: Proc. of INTERSPEECH. Brighton, UK, S. 312–315, 2009.

[St09] Steidl, Stefan: Automatic Classification of Emotion Related User States in Spontaneous Children's Speech. Dissertation, Dissertation, University of Erlangen-Nuremberg, 2009.

[TK11] Tong, Simon; Koller, Daphne: Support Vector Machine Active Learning with Applications to Text Classification. Journal of Machine Learning Research, 2:45–66, 2011.

[Zh15] Zhang, Zixing: Semi-Autonomous Data Enrichment and Optimisation for Intelligent Speech Analysis. Dissertation, Dissertation, Technische Universität München, 2015.

Simone Hantke erhielt 2011 ihr Diplom in Medientechnik von der Technischen Hochschule Deggendorf, 2014 ihren Master of Science und 2019 ihren Doktortitel jeweils von der Technischen Universität München, einer der deutschen Exzellenz-Universitäten. Während ihrer Promotionszeit war sie zudem als wissenschaftliche Mitarbeiterin am Lehrstuhl für Complex and Intelligent Systems an der Universität Passau und am ZD.B Lehrstuhl für Embedded Intelligence for Health Care and Wellbeing an der Universität Augsburg tätig. Sie arbeitete auf dem Gebiet der Datenverarbeitung und der Spracherkennung, wobei sich ihr Forschungsschwerpunkt auf der Datenerfassung und neuen maschinellen Lernansätzen für robuste automatische Spracherkennung und Sprechercharakterisierung konzentrierte. Simone Hantke hat im Rahmen ihrer Promotion die vorgestellten Verfahren und Ergebnisse erfolgreich in 29 begutachteten wissenschaftlichen Beiträgen veröffentlichen können.

Erweiterte Kalküle schwächster Vorbedingungen für probabilistische Programme[1]

Benjamin Lucien Kaminski[2]

Abstract: Ich habe mich mit der *quantitativen Analyse probabilistischer Programme* beschäftigt und dabei vornehmlich zwei Aspekte untersucht: Die Analysetechniken selbst sowie die Komplexität der Analyseprobleme. Bezüglich Analysetechniken gibt meine Dissertation zunächst eine Einführung in den *Kalkül der schwächsten Vorerwartungen* à la McIver & Morgan. Es folgen selbst entwickelte, erweiterte Kalküle zur Analyse *erwarteter Laufzeiten*, *bedingter Erwartungswerte* und der *Erwartungswerte vorzeichenbehafteter Zufallsvariablen*. Bezüglich der Komplexität der Analyse probabilistischer Programme – unabhängig von der verwendeten Analysetechnik – habe ich die *Approximation von Erwartungswerten und Kovarianzen*, die *Entscheidbarkeit der Terminierung probabilistischer Programme* und die *Sinnhaftigkeit unterschiedlicher probabilistischer Terminierungsbegriffe* untersucht.

1 Einleitung

Bereits 1976 publizierte Michael Rabin *Probabilistic Algorithms* [Ra76] – eine Arbeit, die heute als *die* bahnbrechende Arbeit über randomisierte Algorithmen angesehen wird [Sm00]. Rabin beschrieb damals ein randomisiertes Lösungsverfahren für das Problem des dichtesten Punktpaares, bei dem es darum geht, aus einer gegebenen Punktmenge diejenigen zwei Punkte zu finden, die den geringsten Abstand zueinander haben. Während ein naiver, deterministischer Algorithmus dafür *quadratisch viel Zeit* bräuchte (für jeden Punkt: vergleiche den Abstand zu allen anderen Punkten; halte stets das jeweils dichteste gefundene Punktpaar vor), benötigt Rabins randomisierter Algorithmus *im Erwartungswert* nur *linear viel Zeit*, sprich: Der Erwartungswert der Laufzeit auf Probleminstanzen mit n Punkten liegt in $O(n)$.

Eine Reduktion der Laufzeiten von Algorithmen ist eine der Hauptmotivationen, Randomisierung in das Rechnen mit einzubringen. Ein anderes, rasant wachsendes Anwendungsfeld probabilistischer Programme ist die Beschreibung komplexer statistischer Modelle im Bereich des maschinellen Lernens. Beispielsweise entwickelt die Forschungssparte *Uber AI Labs* des weltweit größten Personenbeförderungsdienstleisters *Uber* die probabilistische Programmiersprache Pyro [Bi19], welche *Uber* in naher Zukunft für *autonomes Fahren* einsetzen will [El18].

In meiner Dissertation [Ka19] habe ich mich mit der Analyse verschiedenster Aspekte probabilistischer Programme beschäftigt. Im Folgenden gebe ich einen kurzen Überblick über meine Dissertation und gehe dabei auf ausgewählte Themen näher ein.

[1] Englischer Titel der Dissertation: „Advanced Weakest Precondition Calculi for Probabilistic Programs“

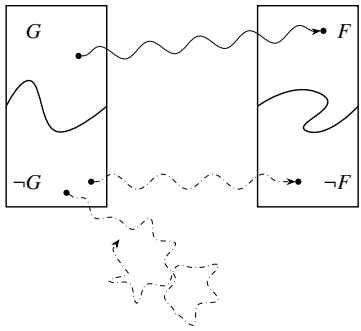
[2] University College London, b.kaminski@ucl.ac.uk

CC BY SA

2 Von Dijkstra zu McIver & Morgan

Kapitel 2 bis 4 meiner Dissertation befassen sich mit einem selbst entwickelten didaktischen Zugang zum *Kalkül schwächster Vorerwartungen* (kurz wp-Kalkül, von *weakest preexpectation calculus*) von Annabelle McIver & Carroll Morgan zur quantitativen Verifikation probabilistischer Programme [MM05]. Das Verständnis dieses Kalküls ist auch für das Verständnis der von mir und meinen Kollegen entwickelten erweiterten Kalküle hilfreich.

Boole'sche Verifikation. Grundlage sämtlicher von uns betrachteter Kalküle bildet Edsger Dijkstras *Kalkül schwächster Vorbedingungen* (*weakest precondition* calculus) zur Verifikation *nicht*-probabilistischer Programme [Di75]. Für ein gegebenes Programm C und eine *Nachbedingung* F will man hierbei die allgemeinste (sprich: die schwächste) *Vorbedingung* G finden, sodass Folgendes gilt: Immer wenn C in einem Zustand gestartet wird, der G erfüllt, dann *terminiert* C in einem Zustand, der F erfüllt. Schematisch ergibt sich rechtsstehendes Bild.



Von G aus startend, trägt uns die Programmausführung also garantiert nach F. Von $\neg G$ aus terminiert C entweder in $\neg F$ oder überhaupt nicht. Somit ist G tatsächlich die *schwächste Vorbedingung*, sodass von dort aus startend die Ausführung von C garantiert in F terminiert.

Betrachten wir zum Beispiel die beiden einfachen Programme

$$C_1 \rhd \quad x := 5 \qquad \text{und} \qquad C_2 \rhd \quad x := x + 1$$

sowie die Nachbedingung $[x \leq 5]$. Dijkstras Kalkül gibt uns nun genaue Regeln vor, wie wir die Nachbedingung „*rückwärts durchs Programm schiebend*" in die schwächste Vorbedingung transformieren können. Für eine Zuweisung $x := expr$ beispielsweise ist in der Nachbedingung jedes Vorkommen von x syntaktisch durch $expr$ zu ersetzen. Als schwächste Vorbedingungen für die Programme C_1 und C_2 ergeben sich bezüglich der Nachbedingung $[x \leq 5]$ dann

$$\begin{array}{l} /\!/\!/\ \mathbf{[5 = 5]} \ = \ \mathbf{[wahr]} \\ x := 5 \\ /\!/\!/\ \mathbf{[x \leq 5]} \end{array} \qquad \text{und} \qquad \begin{array}{l} /\!/\!/\ \mathbf{[x + 1 \leq 5]} \ = \ \mathbf{[x \leq 4]} \\ x := x + 1 \\ /\!/\!/\ \mathbf{[x \leq 5]}\,. \end{array}$$

Sprich: *Immer*, wenn man $x := 5$ ausführt, dann ist anschließend $x \leq 5$; und immer dann, wenn man $x := x + 1$ auf einem Zustand mit $x \leq 4$ ausführt, dann ist anschließend $x \leq 5$.

Quantitative Verifikation. Dijkstras Kalkül operiert auf Prädikaten, also Objekten vom Typ $F\colon \Sigma \to \{0, 1\}$, die jedem Zustand in Σ einen Wahrheitswert in $\{0, 1\}$ zuweisen. Als erste Verallgemeinerung dieses Kalküls lassen wir nun – anstelle logischer Prädikate – als Vor- und Nachbedingungen Funktionen vom Typ $f\colon \Sigma \to \mathbb{R}^{\infty}_{\geq 0}$ zu[3], die jedem Zustand $\sigma \in \Sigma$ eine positive reelle *Zahl* $f(\sigma)$, also eine *Quantität*, zuweisen. Beispielsweise könnte ein solches f immer den *Wert der Programmvariable* x zurückgeben.

Dijkstras Kalkül kann nun so verallgemeinert werden, dass er *Quantitäten statt logischer Prädikate durch das Programm schiebt*. Der Mehrwert: Statt zu fragen, ob x am Ende höchstens 5 ist, können wir mit dem verallgemeinerten Kalkül direkt nach dem Wert von x zu Programmende fragen. Wiederum ersetzen wir für eine Zuweisung $x := expr$ in der (verallgemeinerten) *Nachbedingung* $f = x$ jedes Vorkommen von x durch *expr*, sprich:

$$
\begin{array}{ll}
/\!/\!/\ \mathbf{5} & /\!/\!/\ \boldsymbol{x+1} \\
x := 5 & x := x + 1 \\
/\!/\!/\ \boldsymbol{x} & /\!/\!/\ \boldsymbol{x}
\end{array}
$$

Wenn wir also C_1 ausführen, dann ist der Wert von x am Ende immer 5, unabhängig davon, welchen Initialwert x hatte. Etwas komplizierter ist die Interpretation für C_2. Der resultierende Ausdruck $x + 1$ wird in den Initialzuständen ausgewertet und bedeutet: Der Wert von x *nach* der Ausführung von C_2 ist um eins höher als der initiale Wert von x.

Ein großer Vorteil dieses Kalküls ist, dass der Kalkül *kompositionelle Verifikation* ermöglicht. Wir können also beispielsweise die beiden möglichen Hintereinanderausführungen $C_1 \fatsemi C_2$ und $C_2 \fatsemi C_1$ durch Hintereinanderanwenden der Regel für Zuweisungen verifizieren:

$$
\begin{array}{ll}
/\!/\!/\ \mathbf{5} & /\!/\!/\ \mathbf{5+1} = \mathbf{6} \\
x := x + 1 \fatsemi & x := 5 \fatsemi \\
/\!/\!/\ \mathbf{5} & /\!/\!/\ \boldsymbol{x+1} \\
x := 5 & x := x + 1 \\
/\!/\!/\ \boldsymbol{x} & /\!/\!/\ \boldsymbol{x}
\end{array}
$$

Verifikation probabilistischer Programme. Wir erweitern nun unsere Modellprogrammiersprache um eine *probabilistische Auswahl* $\{ C_1 \} \ [p] \ \{ C_2 \}$ mit folgender Semantik: C_1 wird mit Wahrscheinlichkeit p ausgeführt und C_2 wird mit der Gegenwahrscheinlichkeit $1 - p$ ausgeführt.

[3] Selbstverständlich handelt es sich dann hierbei um *verallgemeinerte* „Bedingungen“, da wir den Bereich rein binärer Wahrheitswerte verlassen haben.

Die Frage nach *dem Wert* der Programmvariablen x am Ende des Programmes erscheint nun weniger sinnvoll, da die Programmausführung – und damit auch der Wert von x zu Programmende – nicht mehr allein vom Initialzustand, sondern auch vom Ausgang der internen Münzwürfe abhängig ist. Eine sinnvolle Frage wäre aber:

Was ist der Erwartungswert von x am Programmende?

Die Frage nach dem Erwartungswert einer beliebigen Quantität $f\colon \Sigma \to \mathbb{R}^{\infty}_{\geq 0}$ kann nun in systematischer Weise durch McIver & Morgans wp-Kalkül beantwortet werden. Es gilt:

$$\mathsf{wp}\,[\![C]\!]\,(f) \;=\; \text{„Erwartungswert von } f \text{ nach Ausführung von } C\text{“}\,.$$

Hierbei ist $\mathsf{wp}\,[\![C]\!]\,(f)$ wiederum vom Typ $\Sigma \to \mathbb{R}^{\infty}_{\geq 0}$, denn es wird *jedem Initialzustand* $\sigma \in \Sigma$ der *jeweilige Erwartungswert von f nach Ausführung von C auf σ* zugewiesen.

Die Regel für probabilistische Auswahlen lautet

$$\begin{aligned}&\mathsf{wp}\,[\![\{\,C_1\,\}\;[p]\;\{\,C_2\,\}]\!]\,(f)\\ &\quad=\; p\cdot \mathsf{wp}\,[\![C_1]\!]\,(f) + (1-p)\cdot \mathsf{wp}\,[\![C_2]\!]\,(f)\;.\end{aligned}$$

Der Erwartungswert von f nach der Ausführung von $\{\,C_1\,\}\;[p]\;\{\,C_2\,\}$ ist also eine gewichtete Summe, nämlich p mal der Erwartungswert von f nach Ausführung von C_1 plus $1-p$ mal der Erwartungswert von f nach Ausführung von C_2. Für das Programm

$$\{\,x := 5\,\}\;[1/2]\;\{\,x := x+1\,\}$$

ergeben sich bspw. die rechtsstehenden Annotationen (zu lesen von unten nach oben).

$$\begin{array}{l}
/\!/\!/\ \tfrac{1}{2}\cdot 5 + \left(1-\tfrac{1}{2}\right)\cdot(x+1) \;=\; \tfrac{x}{2}+3\\
\}\\
\quad /\!/\!/\ 5\\
\quad x := 5\\
\quad /\!/\!/\ x\\
\}\ [1/2]\ \{\\
\quad /\!/\!/\ x+1\\
\quad x := x+1\\
\quad /\!/\!/\ x\\
\}\\
/\!/\!/\ x
\end{array}$$

Der resultierende Ausdruck $\frac{x}{2}+3$ wird wieder in den Initialzuständen ausgewertet und bedeutet: Der Erwartungswert der Variable x nach der Ausführung von $\{\,x := 5\,\}\;[1/2]\;\{\,x := x+1\,\}$ beträgt $\frac{x}{2}+3$ vom Initialwert von x.

3 Verifikation von Schleifen

In Kapitel 5 meiner Dissertation gebe ich einen Überblick über teils eigene, teils fremde, teils Folklorewissen-artige Beweisregeln zur Verifikation von Erwartungswerten für Schleifen mit Randomisierung. Dabei habe ich alle Regeln sozusagen *in einer Sprache*, nämlich dem wp-Kalkül, formuliert. Dies hat den Vorteil, dass man die Regeln – und insbesondere deren Ausdrucksstärke und *praktische Anwendbarkeit* – gut miteinander vergleichen kann.

Eine der wichtigsten Regeln stellt sich wie folgt dar: Für Schleifen der Form $\texttt{while}\,(\varphi)\,\{\,C\,\}$ kann der Erwartungswert von f als *kleinster Fixpunkt* (least fixed point)

$$\mathsf{wp}\,[\![\texttt{while}\,(\varphi)\,\{\,C\,\}]\!]\,(f) \;=\; \mathsf{lfp}\,\Phi_f$$

einer geeigneten Funktion vom Typ $\Phi_f \colon (\Sigma \to \mathbb{R}^\infty_{\geq 0}) \to (\Sigma \to \mathbb{R}^\infty_{\geq 0})$ charakterisiert werden. Eine Charakterisierung als kleinster Fixpunkt ermöglicht eine *konzeptuell einfache* Verifikation von *oberen Schranken an Erwartungswerte*, nämlich nach der Induktionsregel von David Park [Pa69]:

$$\Phi_f(g) \preceq g \qquad \text{impliziert} \qquad \mathsf{wp}\,[\![\texttt{while}(\varphi)\{C\}]\!]\,(f) \preceq g$$

Vermutet man also eine *obere Schranke* g und kann $\Phi_f(g) \preceq g$ zeigen, so hat man nun verifiziert, dass g *tatsächlich* eine obere Schranke an den gesuchten Erwartungswert ist. Ähnliche Regeln für untere Schranken stellen sich als *wesentlich komplizierter* heraus.

4 Fast sichere Terminierung

In Kapitel 6 meiner Dissertation stelle ich eine mit McIver & Morgan entwickelte Verifikationstechnik für *fast sichere Terminierung* (Terminierung mit Wahrscheinlichkeit 1) vor [Mc18]. Diese Regel ist ähnlich zu den klassischen *Schleifenvarianten* von Floyd für nicht-probabilistische Programme [Fl67], bei denen gezeigt werden muss, dass man sich pro Schleifeniteration innerhalb einer geeigneten Wohlordnung *nach unten* bewegt. Wegen der Wohlgeordnetheitseigenschaft ist dies nicht unendlich lange möglich, sodass die Schleife terminieren muss.

Eine ähnliche Technik für den probabilistischen Fall zu entwickeln, ist vergleichsweise schwierig. In Sachen praktische Anwendbarkeit hingegen ist die aus unseren Überlegungen resultierende Beweisregel vergleichsweise simpel. Grob gesprochen muss man lediglich zeigen, dass man sich pro Schleifeniteration mit einer *Mindestwahrscheinlichkeit* um einen *Mindestbetrag* innerhalb der *positiven reellen Zahlen* (welche aber *keine* Wohlordnung darstellen) hinunterbewegen muss. Sobald man die 0 erreicht, terminiert man. Unsere Technik liefert beispielsweise einen sehr simplen Beweis, dass der eindimensionale faire Random Walk fast sicher terminiert.

5 Erwartete Laufzeiten

In Kapitel 7 meiner Dissertation beschäftige ich mich mit der formalen Verifikation erwarteter Laufzeiten von probabilistischen Programmen. Schlussfolgerungen über erwartete Laufzeiten entpuppen sich als überraschend diffizil und voller Nuancen, was einen Bedarf für formale Methoden nochmals unterstreicht. Während zum Beispiel die Laufzeit von Rabins eingangs erwähntem Algorithmus erwartet linear ist, kann sie (mit hinreichend geringer Wahrscheinlichkeit) auch deutlich länger sein. Zur Veranschaulichung: Das Programm

$$\texttt{while}(x > 0)\{ \qquad \{x := x - 1\}\,[1/2]\,\{\texttt{skip}\} \qquad \}$$

terminiert in erwartet $O(x)$ Schritten: Pro Iteration wird entweder x um 1 heruntergezählt, oder nicht – beides jeweils mit Wahrscheinlichkeit $1/2$. Im Schnitt wird x somit *alle zwei Iterationen* um 1 heruntergezählt. Trotz seiner erwartet linearen Laufzeit lässt obiges Programm

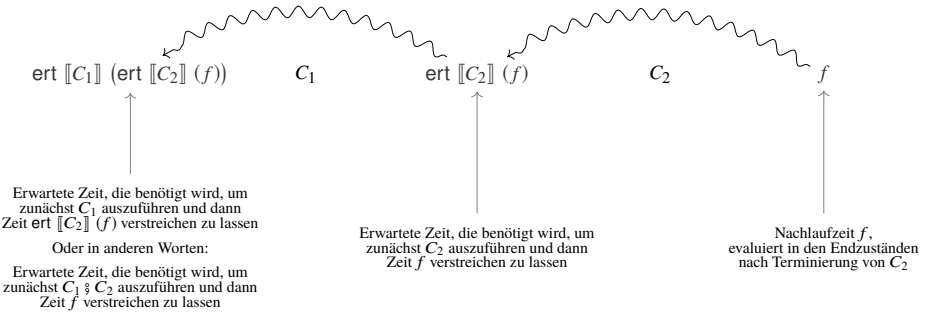


Abb. 1: Prinzipielle Funktionsweise des ert-Kalküls (am besten zu lesen von rechts nach links).

aber sogar *unendliche Läufe* zu, zum Beispiel den, in dem unendlich oft hintereinander der `skip`-Zweig ausgeführt wird. Dies passiert allerdings nur mit Wahrscheinlichkeit 0.

Der ert-Kalkül. Zur Verifikation von Laufzeiten probabilistischer Programme haben wir den *Kalkül erwarteter Laufzeiten* (kurz ert-Kalkül, *expected runtime calculus*) entwickelt [Ka18]. Auch der ert-Kalkül operiert auf Funktionen vom Typ $f\colon \Sigma \to \mathbb{R}^{\infty}_{\geq 0}$, welche hier als *Programmlaufzeiten* interpretiert werden. Auch schiebt der ert-Kalkül die Laufzeit *von hinten nach vorne* durch das zu analysierende Programm, sodass man am Ende der Analyse vorne die erwartete Laufzeit ablesen kann; schematisch dargestellt in Abbildung 1.

Wir veranschaulichen den ert-Kalkül anhand einiger beispielhafter Programme. Angenommen, das Programm $\texttt{tick}(x)$ habe keinen Effekt, außer, x Zeiteinheiten lang für seine Ausführung zu benötigen. Ähnlich dazu habe das Programm $\texttt{tack}(x)$ nur den Effekt, $3x$ Zeiteinheiten zu benötigen. Wir nehmen weiter an, dass wir *nach* der Ausführung von $\texttt{tick}(x)$ bzw. $\texttt{tack}(x)$ noch die Zeit f verstreichen lassen müssen. In unserer Notation vom wp-Kalkül sähen ert-Annotationen zu $\texttt{tick}(x)$ und $\texttt{tack}(x)$ dann wie folgt aus:

$/\!/\!/\ x + f$	$/\!/\!/\ 3x + f$
$\texttt{tick}(x)$	$\texttt{tack}(x)$
$/\!/\!/\ f$	$/\!/\!/\ f$

Für $f = 0$ ergäben sich somit die reinen Laufzeiten von $\texttt{tick}(x)$ und $\texttt{tack}(x)$. Allgemein:

$$\mathsf{ert}\ [\![C]\!]\ (0)\ =\ \text{„Erwartete Laufzeit von } C\text{ “}$$

Wie der wp-Kalkül ermöglicht der ert-Kalkül kompositionelle Verifikation. Packen wir die Programme $\texttt{tick}(x)$ und $\texttt{tack}(x)$ also in einen erweiterten, probabilistischen Kontext:

$$C_\ell \rhd \quad x := x + 1 \fatsemi \{\,\texttt{tick}(x)\,\}\ [1/2]\ \{\,\texttt{tack}(x)\,\}$$

Angenommen, alle Instruktionen außer $\texttt{tick}(x)$ und $\texttt{tack}(x)$ benötigen 0 Zeiteinheiten. Die erwartete Laufzeit von C_ℓ lässt sich dann mit Hilfe des ert-Kalküls wie rechts gezeigt ermitteln (wie auch beim wp-Kalkül sind die Annotationen von unten nach oben zu lesen). Die erwartete Laufzeit von C_ℓ beträgt somit $2x + 2$ vom Initialwert von x.

$/\!/\!/\ \mathbf{2(x+1)} = 2x+2$
$x := x + 1\,;$
$/\!/\!/\ \frac{1}{2}\cdot x + \frac{1}{2}\cdot 3x = 2x$
{
  $/\!/\!/\ \mathbf{0} + x = x$
  $\texttt{tick}(x)$
  $/\!/\!/\ \mathbf{0}$
} [1/2] {
  $/\!/\!/\ \mathbf{0} + 3x = 3x$
  $\texttt{tack}(x)$
  $/\!/\!/\ \mathbf{0}$
}$\,;$
$/\!/\!/\ \mathbf{0}$

Erwartete Laufzeiten von Schleifen. Wie beim wp-Kalkül können erwartete Laufzeiten von Schleifen als *kleinster Fixpunkt* $\mathsf{ert}\,[\![\texttt{while}(\varphi)\{C\}]\!]\,(f) = \mathsf{lfp}\,\Phi_f$ einer geeigneten Funktion Φ_f charakterisiert werden. Wiederum haben wir die Regel

$$\Phi_f(g) \preceq g \qquad \text{impliziert} \qquad \mathsf{ert}\,[\![\texttt{while}(\varphi)\{C\}]\!]\,(f) \preceq g\,.$$

Vermutet man also eine obere Schranke g und kann $\Phi_f(g) \preceq g$ zeigen, so hat man verifiziert, dass g tatsächlich eine obere Schranke an die erwartete Laufzeit der Schleife ist. Ist g endlich, so hat man sogar gezeigt, dass die Schleife fast sicher terminiert, denn eine endliche erwartete Laufzeit impliziert Terminierung mit Wahrscheinlichkeit 1. Ähnliche Regeln für *untere Schranken* stellen sich auch hier wieder als wesentlich komplizierter heraus.

6 Komplexität der Analyse probabilistischer Programme

Teil III meiner Dissertation behandelt die *Komplexität* der Verifikation probabilistischer Programme. Während sofort ersichtlich ist, dass sämtliche für nicht-probabilistische Programme unentscheidbaren Probleme für probabilistische Programme auch unentscheidbar sein müssen, stellt sich jedoch die Frage, ob diese Probleme für probabilistische Programme nicht sogar (in einem mathematisch rigorosen Sinne) *noch unentscheidbarer* sind.

Die arithmetische Hierarchie. Einen mathematischen Rahmen zur Klassifikation der *Grade von Unentscheidbarkeit* bietet die sogenannte *arithmetische Hierarchie* [Mo47] – eine doppelsträngige Hierarchie von *„immer unentscheidbareren"* Problemen. Der linke Strang enthält die Ebenen $\Sigma^0_1, \Sigma^0_2, \Sigma^0_3, \ldots$, der rechte Strang die Ebenen $\Pi^0_1, \Pi^0_2, \Pi^0_3, \ldots$ Vereinfacht gesagt sind Probleme auf höheren Ebenen schwieriger als die auf niedrigeren Ebenen, unabhängig davon, ob sie auf der Σ^0- oder der Π^0-Seite stehen. Außerdem sind *innerhalb einer Ebene* die Σ^0-Probleme in gewissem Sinne *etwas* einfacher als die Π^0-Probleme.

Innerhalb der arithmetische Hierarchie lassen sich wohlbekannte Programmanalyseprobleme einordnen [Od99]. Bekannterweise ist beispielsweise das spezielle Halteproblem $\mathcal{H}$ (terminiert ein nicht-probabilistisches Programm auf einer gegebenen Eingabe) Σ^0_1-vollständig.

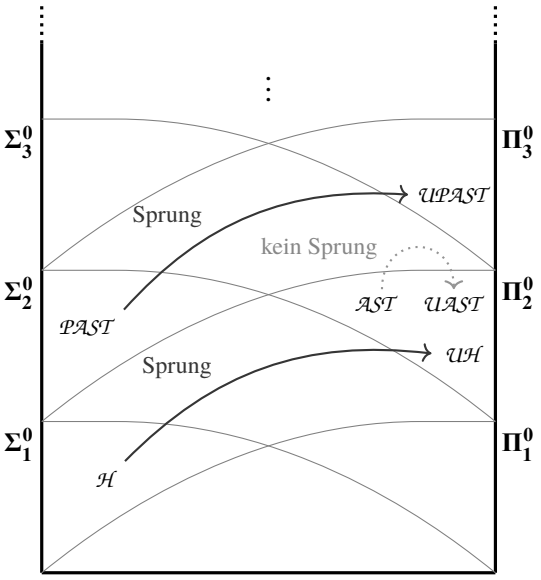


Abb. 2: Die arithmetische Hierarchie mit einigen Programmanalyseproblemen.

Die universelle Variante $\mathcal{UH}$ des Halteproblems (terminiert ein nicht-probabilistisches Programm auf *allen* Eingaben) ist sogar Π_2^0-vollständig, und damit *strikt schwerer* als die spezielle Variante. Es gibt also einen – intuitiv auch zu erwartenden – *Komplexitätssprung* vom speziellen zum universellen Halteproblem, wie in Abbildung 2 dargestellt ist.

Die Komplexität probabilistischer Terminierung. Kapitel 12 meiner Dissertation behandelt die Komplexität von Terminierungsfragen für probabilistische Programme. Allein der Terminierungsbegriff selbst ist hier keine einfache Angelegenheit: *Ein* sinnvoller Terminierungsbegriff wäre, dass ein gegebenes probabilistisches Programm *fast sicher* (also mit Wahrscheinlichkeit 1) terminiert. Ich konnte zeigen, dass für diesen Terminierungsbegriff sowohl die spezielle Variante $\mathcal{AST}$ (von *almost–sure termination*), als auch die universelle Variante $\mathcal{UAST}$, *beide* Π_2^0-vollständig sind [KK15]. Somit gibt es *keinen* Komplexitätssprung von $\mathcal{AST}$ zu $\mathcal{UAST}$, siehe Abbildung 2, was zunächst einigermaßen kontraintuitiv erscheint.

Ein anderer Terminierungsbegriff ist die *positive fast sichere Terminierung* (*positive almost-sure termination*): Terminiert ein gegebenes probabilistisches Programm *innerhalb erwartet endlicher Laufzeit*? Für diesen Terminierungsbegriff ist die spezielle Variante $\mathcal{PAST}$ Σ_2^0-vollständig, während die universelle Variante $\mathcal{UPAST}$ Π_3^0-vollständig ist [KK15]. Wir sehen hier also den vom „gewöhnlichen“ Halteproblem bekannten Komplexitätssprung wieder.

Aus Abbildung 2 entnehmen wir, dass der Terminierungsbegriff für gewöhnliche Programme – aus komplexitätstheoretischer Sicht – eher Analogien zum probabilistischen Begriff der *positiven* fast sichereren Terminierung aufweist. Somit hat uns dieses sehr theoretische – und auf den ersten Anschein weniger praxisorientierte – Resultat ermöglicht, eine Aussage darüber zu treffen, was wohl ein mathematisch robusterer probabilistischer Terminierungsbegriff wäre. Des Weiteren liefert eine Einordnung in die arithmetische Hierarchie ein tieferes (logisches) Verständnis probabilistischer Terminierung, welches bereits durch andere Forschungsarbeiten aufgegriffen und hat beispielsweise Einfluss auf die Entwicklung eines Typsystems zum semi-automatisierten Beweisen fast sicherer Terminierung gehabt [BDL18].

Literaturverzeichnis

[BDL18] Breuvart, Flavien; Dal Lago, Ugo: On Intersection Types and Probabilistic Lambda Calculi. In: Proc. of the International Symposium on Principles and Practice of Declarative Programming (PPDP). ACM, S. 8:1–8:13, 2018.

[Bi19] Bingham, Eli; Chen, Jonathan P.; Jankowiak, Martin; Obermeyer, Fritz; Pradhan, Neeraj; Karaletsos, Theofanis; Singh, Rohit; Szerlip, Paul A.; Horsfall, Paul; Goodman, Noah D.: Pyro: Deep Universal Probabilistic Programming. Journal of Machine Learning Research, 20:28:1–28:6, 2019.

[Di75] Dijkstra, Edsger Wybe: Guarded Commands, Nondeterminacy and Formal Derivation of Programs. Communications of the ACM, 18(8):453–457, 1975.

[El18] Eliot, Lance: , Probabilistic Reasoning for AI Self-Driving Cars, 2018. Zeitungsartikel. Online abgerufen am 10.2.2020.

[Fl67] Floyd, Robert W: Assigning Meanings to Programs. Mathematical Aspects of Computer Science, 19(19-32):1, 1967.

[Ka18] Kaminski, Benjamin Lucien; Katoen, Joost-Pieter; Matheja, Christoph; Olmedo, Federico: Weakest Precondition Reasoning for Expected Runtimes of Randomized Algorithms. Journal of the ACM, 65(5):30:1–30:68, 2018.

[Ka19] Kaminski, Benjamin Lucien: Advanced Weakest Precondition Calculi for Probabilistic Programs. Dissertation, RWTH Aachen, 2019.

[KK15] Kaminski, Benjamin Lucien; Katoen, Joost-Pieter: On the Hardness of Almost–Sure Termination. In: Proc. of the International Symposium on Mathematical Foundations of Computer Science (MFCS). Jgg. 9234 in Lecture Notes in Computer Science. Springer, S. 307–318, 2015.

[KKM18] Kaminski, Benjamin Lucien; Katoen, Joost-Pieter; Matheja, Christoph: On the Hardness of Analyzing Probabilistic Programs. Acta Informatica, 2018.

[Ko83] Kozen, Dexter: A Probabilistic PDL. In: Proc. of the Annual Symposium on Theory of Computing (STOC). S. 291–297, 1983.

[Mc18] McIver, Annabelle; Morgan, Carroll; Kaminski, Benjamin Lucien; Katoen, Joost-Pieter: A New Proof Rule for Almost–sure Termination. Proc. of the Symposium on Principles of Programming Languages (POPL), 2(POPL):33:1–33:28, 2018.

[MM05] McIver, Annabelle; Morgan, Carroll: Abstraction, Refinement and Proof for Probabilistic Systems. Monographs in Computer Science. Springer, 2005.

[Mo47] Mostowski, Andrzej: On Definable Sets of Positive Integers. Fundamenta Mathematicae, 34(1):81–112, 1947.

[Od99] Odifreddi, Piergiorgio: Classical Recursion Theory, Volume II. Elsevier, 1999.

[Ol18] Olmedo, Federico; Gretz, Friedrich; Jansen, Nils; Kaminski, Benjamin Lucien; Katoen, Joost-Pieter; McIver, Annabelle: Conditioning in Probabilistic Programming. Trans. on Programming Languages and Systems, 40(1):4:1–4:50, 2018.

[Pa69] Park, David: Fixpoint Induction and Proofs of Program Properties. Machine intelligence, 5, 1969.

[Ra76] Rabin, Michael Oser: Probabilistic Algorithms. In (Traub, Joseph Frederick, Hrsg.): Algorithms and Complexity: New Directions and Recent Results, S. 21–39. Academic Press, 1976.

[Sm00] Smid, Michiel: Closest–Point Problems in Computational Geometry. In (Sack, Jörg-Rüdiger; Urrutia, Jorge, Hrsg.): Handbook of Computational Geometry, S. 877–935. North–Holland, 2000.

Benjamin Lucien Kaminski wurde am 26. April 1988 in Bonn geboren und absolvierte dort am Kardinal-Frings-Gymnasium 2007 sein Abitur. Nach anschließendem Bachelor- und Masterstudium der Informatik an der RWTH Aachen arbeitete er seit Oktober 2013 als wissenschaftlicher Mitarbeiter in Forschung und Lehre an der Softwaremodellierungs- und Verifikationsgruppe der RWTH Aachen unter der Leitung von Prof. Dr. Ir. Dr. h. c. Joost-Pieter Katoen. Über seine Lehr- und Forschungstätigkeit hinaus hielt Kaminski populärwissenschaftliche Vorträge, hauptsächlich für informatikinteressierte Schülerinnen und Schüler. Außerdem engagierte er sich in der akademischen Selbstverwaltung, beispielsweise in zahlreichen Berufungskommissionen für Professuren oder in der Kommission für Lehre der Fachgruppe Informatik der RWTH Aachen.

Im Februar 2019 wurde Kaminski summa cum laude zum Dr. rer. nat. promoviert und trat im Januar 2020 eine Stelle als Lecturer an der Programming Principles, Logic, and Verification Group des University College London im Vereinigten Königreich an. Kaminskis Dissertation wurde 2020 mit dem *Ackermann Preis* für herausragende Arbeiten im Bereich *Logik in der Informatik* ausgezeichnet.

Zur Verbesserung der Kommunikationskomplexität in der Kryptographie[1]

Lisa Kohl[2]

Abstract:

Das entscheidende Maß der Effizienz eines kryptographischen Protokolls ist oftmals die Kommunikationskomplexität, d.h. die Größe und Anzahl der zu übermittelnden Nachrichten. Diese Dissertation beschäftigt sich sowohl mit der Minimierung der notwendigen Nachrichtengröße in großen Anwendungen, als auch mit der Entwicklung von effizienten Protokollen zur sicheren Mehrparteienberechnung, die mit sehr wenigen Nachrichten auskommen.

Ergebnisse dieser Dissertation umfassen das erste skalierbare Public-Key Verschlüsselungsverfahren mit kompakten Schlüsseln und Chiffraten, das Sicherheit gegen aktive Angriffe bietet (d.h. Angriffe bei denen Nachrichten beliebig verändert werden).

Außerdem wurde im Rahmen dieser Dissertation eine neue Klasse von Protokollen zur sicheren Mehrparteienberechnung entwickelt, die sowohl mit signifikant weniger Kommunikation auskommen, als auch Sicherheit gegen die potentielle Gefahr von Quantencomputern bieten.

1 Einführung

Aufgrund des beispiellosen Aufstiegs des Internets ist Kryptographie aus unserem Alltag nicht mehr wegzudenken. Über klassische Aufgaben wie die Verschlüsselung und Authentifizierung von Nachrichten hinaus, stehen wir heute vor nie dagewesenen Herausforderungen: Wie können wir – in einer Welt, die vernetzt ist wie nie zuvor – die Privatsphäre des Einzelnen schützen?

Ein Grundproblem der Kryptographie ist, dass Protokolle durch eine kryptographische Absicherung weniger effizient werden, und gleichzeitig die Notwendigkeit von Kryptographie oft erst ins Bewusstsein rückt, wenn es zu spät ist. Um Anreize für das Verwenden von kryptographischen Bausteinen zu setzen, besonders über Kontexte hinaus in denen Kryptographie heute schon selbstverständlich ist, ist es daher essenziell, kryptographische Protokolle zu entwickeln, die nur einen minimalen Overhead im Vergleich zu unsicheren Alternativen aufweisen. Da Endgeräte heutzutage mit immer größerer Rechenleistung ausgestattet sind, oft aber nur Netzwerke mit beschränkter Bandbreite zur Verfügung stehen, ist ein besonderer wichtiger Maßstab für die Effizienz eines Protokolls die Kommunikationskomplexität, d.h. die Größe und Anzahl der zu übermittelnden Nachrichten.

Aus dieser Motivation heraus beschäftigt sich diese Dissertation mit der Verbesserung der Kommunikationskomplexität von kryptographischen Bausteinen: Einerseits durch das

[1] Englischer Titel der Dissertation: "On Improving Communication Complexity in Cryptography" [Ko19]

[2] Technion, lisa.kohl@cs.technion.ac.il

cc by sa

Konstruieren von *skalierbaren* kryptographischen Primitiven, d.h. kryptographischen Primitiven, deren Parametergrößen unabhängig von der Größe des Systems (d.h. unabhängig von der Anzahl der Nutzer und Menge der versendeten Nachrichten) gewählt werden können, mit dem Ziel die *Größe der Nachrichten* an sich zu verbessern.

Der Augenmerk liegt hierbau auf sogenannten Public-Key Primitiven, die nicht den Austausch eines geheimen Schlüssel über einen sicheren Kanal voraussetzen: Public-Key Verschlüsselungsverfahren ermöglichen die sichere (d.h. vor Abhörung und Veränderung geschützte) Kommunikation über einen öffentlichen Kanal. *Digitale Signaturen* stellen Authentifizierung und Integrität sicher, indem *öffentlich* überprüft werden kann, von wem eine Nachricht gesendet und ob sie verändert wurde.

Andererseits verbessern wir die Kommunikationskomplexität durch den Entwurf von kryptographischen Protokollen, die grundsätzlich mit *erheblich weniger* Nachrichten auskommen. Der Fokus dieser Arbeit liegt dabei auf Protokollen zur sicheren Mehrparteienberechnung, bei der zwei oder mehr Parteien gemeinsam eine Funktion auswerten wollen, ohne jeweils ihre private Eingabe preiszugeben.

Effiziente Protokolle zur sicheren Mehrparteienberechnung sind essenziell für den Datenschutz in einer digitalen Welt: Während zum Beispiel das Nachschlagen eines Eintrags in einer analogen Enzyklopädie von Dritten (insbesondere dem Hersteller der Enzyklopädie) nicht nachverfolgt werden kann, können aus Suchanfragen an eine Online-Datenbank möglicherweise sensible Informationen abgeleitet werden. Ähnlich zur analogen Welt, erlaubt es sichere Mehrparteienberechnung einem Nutzer eine Online-Datenbanksuchanfrage zu stellen, ohne dass der Server, der die Datenbank unterhält, etwas über die Suchanfrage lernt. Das bemerkenswerte ist, dass der Server hierzu nicht die komplette Datenbank übermitteln muss.

Im Rahmen dieser Dissertation entwickelten wir unter anderem ein Protokoll, dass es einem Benutzer erlaubt, private Datenbanksuchanfragen zu stellen, sodass die Anzahl der zu übermittelnden Nachrichten unabhängig von der Größe der Datenbank ist. Darüber hinaus bietet unser Verfahren Sicherheit gegenüber Quantencomputern. Obgleich Quantencomputer, die mächtig genug sind um traditionelle kryptographische Annahmen (wie zum Beispiel die Faktorisierungsannahme) zu brechen, heute noch Zukunftsmusik sind, ist es für den langfristigen Schutz der Privatsphäre eines Nutzers essentiell, bereits jetzt auf post-quantum-sichere Verfahren umzustellen. Sonst könnten z.B. Datenbanksuchanfragen gespeichert und in einer potentiellen Quantenzukunft entschlüsselt werden.

2 Skalierbare Kryptographie

Besonders für Anwendungen, die mit sensiblen Daten hantieren, ist es von entscheidender Bedeutung, dass die eingesetzten kryptographische Protokolle nicht nur vermutlich nicht zu brechen sind, sondern dass sie *beweisbare* Sicherheitsgarantien bieten. Da nicht bekannt ist, ob $\mathsf{P} \neq \mathsf{NP}$ gilt (d.h. ob es Probleme gibt, deren Lösung zwar effizient nachvollziehbar ist, die aber von keinen effizienten Algorithmus gelöst werden können), können oft

– zum Beispiel im gesamten Bereich der Public-Key Kryptographie – nur *konditionale* Sicherheitsgarantien gegeben werden.

Genauer gesagt wird für die Sicherheit von kryptographische Verfahren üblicherweise bewiesen, dass es *mindestens* so schwer ist das Verfahren zu brechen, wie ein Problem zu lösen, für dessen Lösung trotz intensiver Forschung kein effizienter Algorithmus bekannt ist. Ein Beispiel dafür ist das Faktorisierungsproblem, d.h. das Problem eine große zusammengesetzte Zahl in ihre Primfaktoren zu zerlegen.

Welche Probleme als schwer betrachtet werden, hängt dabei von der aktuellen Rechenleistung ab. Aus diesem Grund wird die Sicherheit eines Verfahrens durch einem *Sicherheitsparameter* λ parametrisiert. Derzeit wird ein Problem als schwierig angesehen, wenn jeder (bekannter) Algorithmus mindestens 2^{100} oder 2^{128} Schritte benötigt, um es zu lösen. Dies wird auch als 100- bzw. 128-Bit Sicherheit bezeichnet. Um mehr Sicherheit zu bieten, können größere Parameter wie z.B. längere Schlüssel gewählt werden. Zur Veranschaulichung: Selbst der stärkste bekannte Supercomputer würde derzeit mehr als hunderttausend Jahre brauchen, um ein Problem zu lösen, das eine Sicherheit von 100-Bit gewährleistet. [3]

Die Reduktion der Sicherheit eines kryptographischen Verfahrens auf die Härte des zugrunde liegenden Problems führt häufig zu einem *Verlust L*. Dies bedeutet, dass ein Angreifer, der die Sicherheit des Schemas in T Schritten bricht, für die Lösung des zugrunde liegende Problems $L \cdot T$ Schritten braucht. Dieser Verlust von L wird im Folgenden als *Sicherheitsverlust* einer Reduktion bezeichnet und wirkt sich entscheidend auf die Parameterwahl des Schemas in der Praxis aus: Typischerweise hängt z.B. der Sicherheitsverlust von digitalen Signaturverfahren linear von der Anzahl der erzeugten Signaturen ab. In typischen Anwendungen kann die Anzahl der Signaturen in der Größenordnung von 2^{30} liegen. Um eine Sicherheit von 128-Bit zu erreichen, müssen die Parameter des Verfahrens dementsprechend so gewählt werden, dass die unterliegende Annahme (z.B. die Faktorisierungsannahme) 158-Bit Sicherheit bietet.

Insbesondere hängt in dem beschriebenen Beispiel die konkrete Größe des öffentlichen Schlüssels und damit der Signaturen entscheidend von der Anzahl der zu erwartenden Signaturen im System ab. Da diese Zahl üblicherweise zum Zeitpunkt der Initialisierung noch nicht bekannt ist, muss sie entweder nach oben abgeschätzt werden (wodurch die Größe der Parameter unnötig erhöht wird) oder es kann dazu kommen, dass keine bewiesene Sicherheit mehr gewährleistet ist.

Aus diesem Grund ist es erstrebenswert, *skalierbare* kryptographische Primitive zu entwickeln, deren Sicherheitsgarantien nicht von der Menge der ausgegebenen Instanzen abhängen. Dies ermöglicht die Wahl der Parameter unabhängig von der Anzahl der erzeugten Signaturen oder Chiffrate und führt dazu, dass in der Praxis kleinere Parameter gewählt werden können. Dadurch wird die Kommunikationskomplexität in Protokollen, die auf jene Schemata aufbauen, verringert. Leider stellen sich solche Verfahren in der Regel als deutlich schwieriger zu konstruieren heraus und sind weisen deshalb trotz Skalierbarkeit oft größere Parameter als entsprechende nicht-skalierbare Verfahren auf.

[3] Stand November 2018, `https://www.top500.org/lists/2018/11/`.

Ergebnis I: Wir entwickeln das erste *skalierbare Verschlüsselungsverfahren* mit kompakten Schlüsseln und kurzen Chiffraten, das Sicherheit gegen aktive Angriffe bietet. Unser Verfahren kommt verglichen mit der besten Vorarbeit mit um den Faktor 30 kleineren öffentlichen Schlüsseln aus. [GHK17]

Wir reduzieren damit den Preis für skalierbare Sicherheit erheblich, und schließen erstmals nahezu die Lücke zu nicht-skalierbaren Konstruktionen: Unser Verschlüsselungsschema hat einen Overhead von nur einem Gruppenelement in den Chiffraten und vier Gruppenelementen im öffentlichen Schlüssel im Vergleich zum nicht-skalierbaren Schema von Kurosawa und Desmedt [KD04].

Ergebnis II: Wir stellen ein *skalierbares strukturerhaltendes* [4] *digitales Signaturverfahren* vor, das im Vergleich zur besten Vorarbeit mit erheblich kleineren öffentlichen Schlüsseln und Signaturen auskommt (von nur etwas 40% bzw. 56%). [Ga18b]

3 Neue Wege zur sicheren Mehrparteienberechnung

Im Kontext der sicheren Mehrparteienberechnung kann ein Wechsel des zugrunde liegenden Paradigmas eines kryptographischen Protokolls zu weitaus drastischeren Verbesserungen der Kommunikationskomplexität führen, als die bloße Reduzierung der Parametergröße. Das Ziel sicherer Berechnung ist, dass zwei (oder mehr) Parteien gemeinsam eine Funktion auswerten können, ohne dabei ihre jeweils privaten Eingaben preisgeben zu müssen.[5] Verglichen mit den Anforderungen an sichere Kommunikation, ist dies ist ein sehr unterschiedliches Ziel: Im Bereich der sicheren Mehrparteienberechnung liegt der Fokus auf dem Schutz gegen Angriffe *innerhalb* des Protokolls (d.h. Angriffe von Teilnehmern an der Berechnung untereinander), wohingegen das Ziel der sicheren Kommunikation der Schutz gegen Angriffe *von außen* ist.

Protokolle zur sicheren Berechnung ermöglichen es einem Nutzer, zum Beispiel online einen Kredit oder eine Krankenversicherung zu beantragen ohne dabei private Informationen an das Kreditinstitut oder den Versicherer herauszugeben. Das Ergebnis der Berechnung sind lediglich die Kreditkonditionen oder die Höhe der Versicherungsbeiträge, basierend auf den privaten Eingaben des Nutzers. Solche Protokolle werden jedoch bislang nur sehr beschränkt in der Praxis eingesetzt. Ein entscheidender Grund dafür ist, dass die meisten Protokolle nicht effizient genug sind. Insbesondere erfordern sie erheblich mehr Kommunikation als vergleichbare nicht-privatsphäreschützende Lösungen.

Der Baustein der sicheren Mehrparteienberechnung wurde 1982 von Yao vorgeschlagen, der vier Jahre später auch das erste effiziente Protokoll entwickelte, das es erlaubt allgemeine Schaltkreise sicher zu berechnen [Ya86]: Die Idee ist, dass eine Partei einen verschlüsselten Schaltkreis zusammen mit den für die Auswertung erforderlichen Schlüsseln erstellt und versendet, so dass die andere Partei den Schaltkreis Gate-für-Gate entschlüsseln kann und dabei nur die Ausgabe lernt. Während dieser Ansatz bezüglich der Rechenleistung sehr effizient ist, skaliert die Kommunikationskomplexität mit der Größe des Schalt-

[5] In dieser Dissertation beschränken wir uns auf den wichtigen Spezialfall der sicheren Zweiparteienberechnung.

kreises, die typischerweise groß und insbesondere viel größer als die Eingangsgröße ist. Dieser Ansatz ist daher nicht praktikabel wenn die Kommunikation über Netzwerke mit geringer Bandbreite erfolgt.

Ein entgegengesetzter Weg zur sichere Mehrparteienberechnung basiert auf *vollhomomorpher Verschlüsselung*. Ein vollhomomorphes Verschlüsselungsverfahren erlaubt die Auswertung von *beliebigen* Schaltkreisen auf verschlüsselten Nachrichten. Dies ermöglicht es Nutzern zum Beispiel ihre Daten in Cloud-Services auszulagern, sodass der Cloud-Provider Berechnungen auf den Daten ausführen kann, ohne selbst Zugriff auf die Klardaten zu erhalten. Insbesondere bietet vollhomomorphe Verschlüsselung eine ideale Lösung zur sicheren Mehrparteienberechnung bezüglich der Kommunikationskomplexität, da die Größe der zu übermittelnden Nachrichten nur von der Eingabe- und Ausgabegröße abhängt, und damit – anders als das Verfahren von Yao – insbesondere vollständig unabhängig von der Komplexität der zu berechnenden Funktion ist.

Dieser Baustein wurde lange gesucht und schließlich 2009 von Gentry entwickelt [Ge09]. Obwohl dieses Resultat einen theoretischen Durchbruch darstellte, wird vollhomomorphe Verschlüsselung bis heute quasi nicht in der Praxis verwendet. Dies liegt darin begründet, dass selbst die aktuellsten optimierten Verfahren zur homomorphen Multiplikation von Chiffraten extrem rechenintensiv sind.

3.1 Sichere Berechnung via Homomorphic Secret-Sharing

Eine Möglichkeit diese Einschränkung zu überkommen, ist es die Anforderung an das Verfahren abzuschwächen. Wie Boyle, Gilboa, and Ishai [BGI16] bemerkten, ist für das Ziel der sicheren Mehrparteienberechnung nicht die volle Funktionalität eines Verschlüsselungsverfahrens notwendig, bei dem eine Nachricht nur mit Hilfe eines geheimen Schlüssels rekonstruiert werden kann. Da nämlich zwei (oder mehr) Parteien an der Berechnung teilnehmen, kann die Nachricht stattdessen in Secret-Shares aufgeteilt werden.

Secret-Sharing bezeichnet die Verteilung einer Nachricht zwischen zwei (oder mehr) Parteien, sodass jede Partei für sich nichts über die Nachricht lernt, aus den Secret-Shares jedoch *gemeinsam* die Nachricht rekonstruiert werden kann. Ein sehr einfaches Beispiel für ein Secret-Sharing Verfahren ist das additive Secret-Sharing: Hierbei wird eine Nachricht $s \in \mathbb{N}$ in Secret-Shares a und b geteilt, so dass $a + b = s \mod p$ (für eine natürliche Zahl p). Wenn a zufällig gewählt wird, verraten wie erfordert die Shares einzeln betrachtet nichts über die Nachricht s.

Für den Einsatz im Bereich der kommunikationsarmen sicheren Mehrparteienberechnung, muss das Verfahren zusätzlich die homomorphe Berechnung von Funktionen auf Shares erlauben. Ein Verfahren, dass diese Eigenschaften erfüllt, wird als Homomorphic Secret-Sharing (HSS) Schema bezeichnet. In Abbildung 1 veranschaulichen wir, wie Homomorphic Secret-Sharing in diesem Kontext genutzt werden kann. Die Idee ist, dass die Parteien zunächst jeweils ihre privaten Eingaben aufteilen. Die Sicherheit des Secret-Sharing Schemas garantiert, dass dies der anderen Partei nichts über die Eingaben verrät.

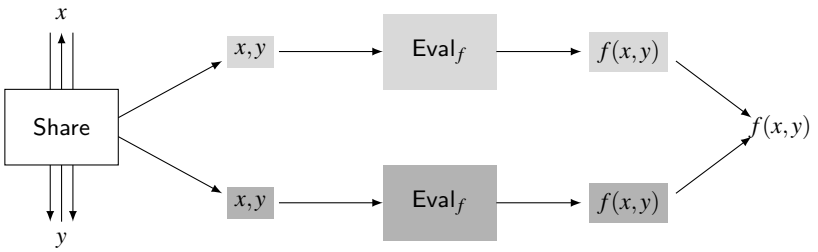


Abb. 1: Sichere Berechnung via Homomorphic Secret-Sharing. Eine Partei wird in hellgrau dargestellt und besitzt Eingabe x, die andere Partei in dunkelgrau besitzt Eingabe y. Mit $\boxed{x,y}$ und $\boxed{x,y}$ sind die jeweiligen Secret-Shares von x und y gemeint. "Share" bezeichnet ein sicheres Protokoll zur Aufteilung der Eingabewerte und Eval_f bezeichnet das Verfahren zur lokalen Evaluierung der Funktion f auf den Shares. Das Besondere an dem resultierenden Protokoll ist, dass die Menge der auszutauschenden Nachrichten nur abhängig von der Eingabe- und Ausgabegröße und insbesondere unabhängig von der zu berechnenden Funktion f ist.

Selbst das einfache additive Secret-Sharing Verfahren unterstützt bereits eingeschränkte homomorphe Berechnungen auf den Shares: Alle *linearen Funktionen*, d.h. Funktionen f, für die $f(a)+f(b)=f(a+b) \mod p$ gilt, können lokal auf den Shares ausgewertet werden. Da lineare Funktionen jedoch nur sehr eingeschränkte Anwendungen unterstützen, konstruierten Boyle et al. [BGI16] ein Homomorphic Secret-Sharing Schema für alle Funktionen, die von Schaltkreisen mit logarithmischer Tiefe berechnet werden können.

Leider erfordert ihr Ansatz eine teure Konvertierung der Shares nach jeder Multiplikation. Dies führt dazu, dass – ähnlich vollhomomorpher Verschlüsselung – Multiplikationen zu viel Rechenleistung erfordern, um praktisch einsetzbar zu sein. Zusätzlich führt die Konvertierung der Shares mit einer gewissen Wahrscheinlichkeit zu einem Fehler. Dies ist nicht nur ein Problem bezüglich der Korrektheit: Ob ein Fehler auftritt oder nicht kann potentiell Informationen über die privaten Eingaben verraten.

Ergebnis III: Wir konstruieren das erste *Homomorphic Secret-Sharing Schema*, das die Auswertung beliebiger Schaltkreise logarithmischer Tiefe unterstützt, bei dem nur mit vernachlässigbarer Wahrscheinlichkeit ($< 1 : 1$ Billion) ein Korrektheitsfehler auftritt und das keine teure homomorphe Multiplikation auf Chiffraten erfordert. Wir zeigen die praktische Anwendbarkeit unseres Schemas für den Anwendungsfall privater Counting-Query-Suchanfragen in verteilten Datenbanken. Darüber hinaus bietet unser Verfahren Sicherheit gegen Quantencomputer. [BKS19]

3.2 Sichere Berechnung über pseudozufällige Korrelationsgeneratoren

Schließlich stellen wir ein neues Paradigma für sichere Mehrparteienberechnung vor, genauer ein neues Paradigma für eine effiziente Vorberechnungsphase, das den großflächigen Einsatz von Protokollen zur sicheren Mehrparteienberechnung in greifbare Nähe rückt.

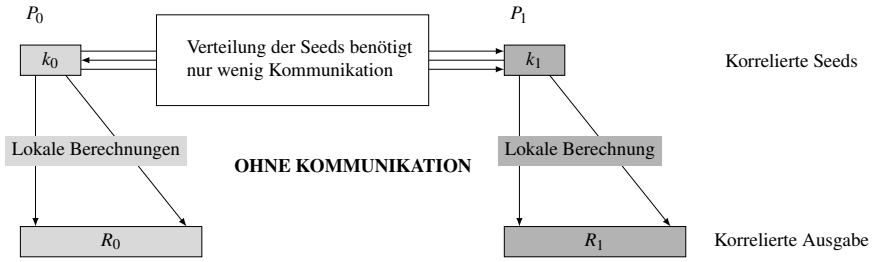


Abb. 2: Pseudozufällige Korrelationsgeneratoren.

Die grundsätzliche Idee von Protokollen zur sicheren Mehrparteienberechnung mit Vorberechnung ist es, das Protokoll in eine langsame eingabeunabhängige Vorberechnungsphase und eine schnelle Online-Phase aufzuteilen [Be92, Da12]. Dieser Ansatz ist besonders attraktiv, wenn Protokolle schnell sein müssen sobald die Eingaben bekannt sind, aber eine vergleichsweise teure Vorberechnungsphase tolerieren, wie zum Beispiel elektronische Wahlverfahren.

Die Idee der Vorberechnungsphase ist es, "korrelierten Zufall" zu erzeugen, der zur Implementierung einer schnellen Online-Phase verwendet werden kann. Ein Beispiel sind sogenannte "Beaver-Tripel" [Be92]: Ein Beaver-Tripel ist ein Tripel $(a,b,c=ab)$, zu dem zwei Parteien jeweils additive Secret-Shares besitzen. Ein solches Tripel kann in der Online-Phase zur schnellen Berechnung von Multiplikationen verwendet werden.

Das Problem an diesem Ansatz ist, dass die Parteien für jede später auszuführende Multiplikation in der Vorberechnungsphase mindestens ein Tripel erzeugen müssen. Darüber hinaus ist die Erzeugung pro Tripel typischerweise vergleichsweise kommunikationsintensiv. Dies erfordert sowohl eine Vorberechnungsphase mit hoher Kommunikationskomplexität, als auch das Speichern einer großen Datenmenge vom Ende der Vorberechnungsphase bis zum Anfang der Online-Phase.

Eine zu Beginn der Dissertation offene Forschungsfrage war, ob es möglich ist Protokolle zur Vorberechnung von Beaver-Tripeln und anderen Korrelationen zu entwickeln, die praktisch effizient sind und deren Kommunikationskomplexität *unabhängig* von der Anzahl der zu berechnenden Multiplikationen ist.

Ergebnis IV: Wir führen den Begriff des *pseudozufälligen Korrelationsgenerators* für allgemeine Korrelationen ein. Ein pseudozufälliger Korrelationsgenerator ermöglicht es zwei Parteien, durch *lokale Berechnungen* kurze korrelierte Seeds in lange korrelierte pseudozufällige Ausgaben zu erweitern.
Wir präsentieren effiziente Instantiierungen unter anderem für (authentifizierte) Beaver-Tripel, die Sicherheit gegen Quantencomputer bieten.
Des weiteren zeigen wir, dass ein Pseudozufallskorrelationsgenerator die Vorberechnungsphase in vielen Protokollen zur sicheren Mehrparteienberechnung ersetzen kann, beispielsweise im Protokoll von Damgård et al. [Da12]. [Bo19b]

Insbesondere können pseudozufällige Korrelationsgeneratoren zu einer erheblichen Verbesserung der Kommunikationskomplexität führen, da das Aufsetzen der kurzen Seeds deutlich weniger Kommunikation erfordert als das direkte Generieren von langen korrelierten zufälligen Ausgaben (siehe auch Abbildung 2). Darüber hinaus müssen die Parteien mit unserem Ansatz nur die kurzen Seeds speichern, die sie lokal (d.h. ohne weitere Kommunikation) erweitern können, direkt bevor sie in der Online-Phase eine Funktion sicher berechnen. Unsere Arbeit stellt die erste Konstruktion einer praktisch effizienten *lokalen* Vorberechnungsphase (auch „*Silent* Preprocessing“ genannt) für allgemeine Protokolle zur sicheren Mehrparteienberechnung dar.

4 Ausblick

Besonders das letzte Ergebnis beeinflusst in großem Maße die Forschung bezüglich praktisch einsetzbarer kryptographischer Protokolle zur sicheren Mehrparteienberechnung. Wie wir in einer Folgearbeit gezeigt haben, kann das neue Paradigma lange offen stehende Fragen positiv beantworten, wie zum Beispiel die Frage ob es ein praktisch effizientes 2-Runden OT-Erweiterungsprotokoll gibt.

Oblivious Transfer (OT) [Ra81, EGL85] ist eine der grundlegendsten Korrelationen, die für die sichere Berechnung von Funktionen nützlich sind: Oblivious Transfer erlaubt es einer Partei eine von zwei Nachrichten m_0, m_1 an eine Partei mit Auswahlbit $b \in \{0,1\}$ so zu übertragen, dass die erste Partei nichts über b lernt, und die zweite Partei nichts über die andere Nachricht m_{1-b}. Wie von Kilian [Ki88] bewiesen, impliziert OT die sichere Berechnung *beliebiger Funktionen*. Dies ist nicht nur ein theoretisches Resultat – OT ist tatsächlich ein grundlegender Baustein vieler Protokolle für effiziente sichere Mehrparteienberechnung. Da diese Protokolle in der Regel viele OT erfordern (im Bereich von mehrerer Millionen), ist die Effizienz dieses Bausteins entscheidend.

OT gehört zum Bereich der Public-Key Kryptographie und ist deshalb vergleichsweise ineffizient. Grundsätzlich sind symmetrische Verfahren, bei denen ein bereits ausgetauschter Schlüssel angenommen werden kann, um ein vielfaches effizienter. Aus diesem Grund folgen die derzeit effizientesten Protokolle zur Erzeugung vieler OT einem hybriden Ansatz namens *OT-Erweiterung*: Unter der Annahme, dass eine Reihe von Basis-OT über eine Black-Box zugänglich sind, werden der Großteil der OT lediglich mit Hilfe von symmetrischen Primitiven und damit vergleichsweise wenig Rechenleistung erzeugt.

Dieser Ansatz wurde von Beaver [Be96] eingeführt, der ein 2-Runden Protokoll zur *OT-Erweiterung* konstruierte, das jedoch eine schlechte konkrete Effizienz aufwies. Später stellten Ishai et al. [Is03] das erste effiziente Protokoll zur OT-Erweiterung vor, das jedoch 3-Runden benötigt. Bis vor kurzem, war nicht bekannt, ob die Vorteile der beiden Protokolle verbunden werden können. Insbesondere wurde in [Ga18a] gezeigt, dass eine solche Konstruktion nicht auf symmetrischen Primitiven basieren kann, ohne strukturelle Eigenschaften dieser Primitive zu nutzen.

Aufbauend auf unserer Arbeit [Bo19b], konstruieren wir in [Bo19a] das erste *effiziente* 2*-Runden OT-Erweiterungsprotokoll*. Für die Erzeugung von zufälligen OT (d.h. OT mit

Zufallsauswahlbit und zufälligen Nachrichten) verbessert unser Ansatz die Kommunikationskomplexität gegenüber dem Protokoll von Ishai et al. um einen Faktor von 1000 (für die Erzeugung von 10 Millionen OT). Unser Protokoll benötigt höhere Rechenleistung, führt aber trotzdem für Anwendungen bei denen nur eine geringe Bandbreite zur Verfügung steht, zu erheblichen Effizienzverbesserungen: Über Netzwerke mit einer Übertragungsrate von 10 MBit/s ist unser Protokoll um Faktor 50 schneller als der bisherige Stand der Technik (für die Erzeugung von 10 Millionen zufälligen OT).

Es bleiben weiterhin viele spannende Fragen in diesem Forschungsfeld offen, insbesondere die folgende: Das Pendant von pseudozufälligen Korrelationsgeneratoren im Bereich der sicheren Kommunikation sind die (lange bekannten) Pseudozufallsgeneratoren, die es erlauben einen kurzen Seed in eine lange Ausgabe zu erweitern. Diese Ausgabe kann zum Beispiel zur (symmetrischen) Verschlüsselung verwendet werden. In [GGM86] wurde gezeigt, dass aus jedem Pseudozufallsgenerator eine pseudozufällige Funktion konstruiert werden kann, die es erlaubt, aus einem kurzen Seed quasi unbeschränkt viel Pseudozufall zu erzeugen. Die Frage ist, ob sich diese Analogie auf den Bereich der sicheren Mehrparteienberechnung übertragen lässt. In anderen Worten, ob es möglich ist, eine pseudozufällige Korrelationsfunktion zu entwerfen, die es den Parteien nach einer einmaligen Vorberechnungsphase erlaubt, aus kurzen korrelierten Seeds nach Bedarf pseudozufällige Korrelationen für *beliebig viele* Berechnungen in der Zukunft zu erzeugen.

Literaturverzeichnis

[Be92] Beaver, Donald: Efficient Multiparty Protocols Using Circuit Randomization. In: CRYPTO'91. Jgg. 576 in LNCS. Springer, Heidelberg, S. 420–432, August 1992.

[Be96] Beaver, Donald: Correlated Pseudorandomness and the Complexity of Private Computations. In: 28th ACM STOC. ACM Press, S. 479–488, Mai 1996.

[BGI16] Boyle, Elette; Gilboa, Niv; Ishai, Yuval: Breaking the Circuit Size Barrier for Secure Computation Under DDH. In: CRYPTO 2016, Part I. Jgg. 9814 in LNCS. Springer, Heidelberg, S. 509–539, August 2016.

[BKS19] Boyle, Elette; Kohl, Lisa; Scholl, Peter: Homomorphic Secret Sharing from Lattices Without FHE. In: EUROCRYPT 2019, Part II. Jgg. 11477 in LNCS. Springer, Heidelberg, S. 3–33, Mai 2019.

[Bo19a] Boyle, Elette; Couteau, Geoffroy; Gilboa, Niv; Ishai, Yuval; Kohl, Lisa; Rindal, Peter; Scholl, Peter: Efficient Two-Round OT Extension and Silent Non-Interactive Secure Computation. In: ACM CCS 19. ACM Press, 2019.

[Bo19b] Boyle, Elette; Couteau, Geoffroy; Gilboa, Niv; Ishai, Yuval; Kohl, Lisa; Scholl, Peter: Efficient Pseudorandom Correlation Generators: Silent OT Extension and More. In: CRYPTO 2019, Part III. Jgg. 11694 in LNCS. Springer, Heidelberg, S. 489–518, August 2019.

[Da12] Damgård, Ivan; Pastro, Valerio; Smart, Nigel P.; Zakarias, Sarah: Multiparty Computation from Somewhat Homomorphic Encryption. In: CRYPTO 2012. Jgg. 7417 in LNCS. Springer, Heidelberg, S. 643–662, August 2012.

[EGL85] Even, Shimon; Goldreich, Oded; Lempel, Abraham: How to exchange secrets with oblivious transfer. In: Communications of the ACM, 28(6). Aiken Computation Lab, Harvard University, S. 637–647, 1985.

[Ga18a] Garg, Sanjam; Mahmoody, Mohammad; Masny, Daniel; Meckler, Izaak: On the Round Complexity of OT Extension. In: CRYPTO 2018, Part III. Jgg. 10993 in LNCS. Springer, Heidelberg, S. 545–574, August 2018.

[Ga18b] Gay, Romain; Hofheinz, Dennis; Kohl, Lisa; Pan, Jiaxin: More Efficient (Almost) Tightly Secure Structure-Preserving Signatures. In: EUROCRYPT 2018, Part II. Jgg. 10821 in LNCS. Springer, Heidelberg, S. 230–258, April / Mai 2018.

[Ge09] Gentry, Craig: Fully homomorphic encryption using ideal lattices. In: 41st ACM STOC. ACM Press, S. 169–178, Mai / Juni 2009.

[GGM86] Goldreich, Oded; Goldwasser, Shafi; Micali, Silvio: How to Construct Random Functions. Journal of the ACM, 33(4):792–807, Oktober 1986.

[GHK17] Gay, Romain; Hofheinz, Dennis; Kohl, Lisa: Kurosawa-Desmedt Meets Tight Security. In: CRYPTO 2017, Part III. Jgg. 10403 in LNCS. Springer, Heidelberg, S. 133–160, August 2017.

[Is03] Ishai, Yuval; Kilian, Joe; Nissim, Kobbi; Petrank, Erez: Extending Oblivious Transfers Efficiently. In: CRYPTO 2003. Jgg. 2729 in LNCS. Springer, Heidelberg, S. 145–161, August 2003.

[KD04] Kurosawa, Kaoru; Desmedt, Yvo: A New Paradigm of Hybrid Encryption Scheme. In: CRYPTO 2004. Jgg. 3152 in LNCS. Springer, Heidelberg, S. 426–442, August 2004.

[Ki88] Kilian, Joe: Founding Cryptography on Oblivious Transfer. In: 20th ACM STOC. ACM Press, S. 20–31, Mai 1988.

[Ko19] Kohl, Lisa Maria: On Improving Communication Complexity in Cryptography. 2019.

[Ra81] Rabin, Michael O.: How to exchange secrets with oblivious transfer. In: TR-81 edition. Aiken Computation Lab, Harvard University, 1981.

[Ya86] Yao, Andrew Chi-Chih: How to Generate and Exchange Secrets (Extended Abstract). In: 27th FOCS. IEEE Computer Society Press, S. 162–167, Oktober 1986.

Lisa Kohl wurde am 8. März 1990 in München geboren. 2009 begann sie ihr Studium am Karlsruher Institut für Technologie. Dort schloss sie einen Bachelor der Mathematik und einen Master in Informatik ab. Im Rahmen ihres Masterstudiums war sie für ein Semester an der National University of Singapore und absolvierte ein Sommerpraktikum am Max-Planck-Institut für Software-Systeme in Saarbrücken. Ihre Masterarbeit schrieb sie zum Thema post-quanten-sichere elektronische Wahlverfahren in der Forschungsgruppe von Ronald Cramer am Centrum Wiskunde & Informatica Amsterdam. Zurück am Karlsruher Institut für Technologie begann sie 2016 ihre Promotion unter der Betreuung von Dennis Hofheinz. 2018 absolvierte sie im Rahmen ihrer Promotion einen achtmonatigen Forschungsaufenthalt bei Elette Boyle am FACT Center des IDC Herzliya. Seit Januar 2020 ist sie Postdoktorandin in der Forschungsgruppe von Yuval Ishai am Technion.

Kryptographische Uhren und Anwendungen[1]

Giulio Malavolta[2]

Abstract: In dieser Dissertation definieren und konstruieren wir kryptographische Objekte, die das *Vergehen von Zeit* simulieren. Im Einzelnen bauen wir unter der Annahme, dass bestimmte mathematische Probleme unlösbar sind, die folgenden Grundelemente auf:

(1) *Inkrementelle Beweise für sequentielle Arbeit* ermöglichen es einem Beweisführer, einen Prüfer davon zu überzeugen, dass eine bestimmte Menge sequentieller Arbeit verrichtet wurde.
(2) *Homomorphe Zeitschlossrätsel* verbergen Geheimnisse für einen bestimmten Zeitraum und ermöglichen es jedem, Funktionen über Klartexte öffentlich zu berechnen, ohne die Rätsel zu lösen.
(3) *Mehrschrittschlösser* ermöglichen es Anwenderpaaren Verträge abzuschließen, bei denen Münzen überwiesen werden, wenn eine bestimmte Bedingung innerhalb einer bestimmten Zeit erfüllt ist.

Einleitung

Wir betrachten das Problem, das Vergehen von Zeit in Gegenwart böswilliger Parteien zu messen. Wir untersuchen jene Szenarien, in denen sich gegenseitig misstrauende Individuen, Bedingungen durchsetzen oder Entscheidungen treffen möchten, die darauf beruhen, ob eine bestimmte Zeitspanne verstrichen ist oder nicht. Man betrachte als ein höchst anschauliches Beispiel den Fall, dass eine Person ein Geheimnis erst nach Ablauf der Zeit T, für ein beliebiges T, veröffentlichen möchte. Wie stellt man sicher, dass das Geheimnis verborgen bleibt, möglicherweise auch nach dem Tod eines solchen Individuums? Das Thema dieser Dissertation ist es, kryptographische Objekte zu identifizieren, zu definieren und zu konstruieren, die als Lösung für diese Klasse von Problemen eingesetzt werden können. Eine Lösung wird als kryptographisch bezeichnet, wenn ihre Sicherheit auf Mathematik beruht und sie keine physischen Eigenschaften ausnutzt oder auf das Vertrauen einer externen Entität angewiesen ist. Diese Dissertation schlägt Lösungskandidaten für die folgenden Fragen vor, welche auf mathematischen Problemen beruhen, bei welchen man davon ausgeht, dass sie sehr schwer zu lösen sind.

Beweis vergangener Zeit. Angenommen, ein Beweisführer möchte einen misstrauischen Verifizierer davon überzeugen, dass (ungefähr) eine bestimmte Zeitspanne T verstrichen ist, um die Echtheit eines bestimmten Zeitstempels zu beweisen. Eine Möglichkeit dies umzusetzen, besteht darin, den Beweisführer vom Zeitpunkt der Erzeugung des Objekts an, eine bestimmte (inhärent sequentielle) Berechnung starten und später nachweisen zu lassen, dass ein angemessener Teil der Berechnung auf seinem Computer ausgeführt wurde.

[1] Englischer Titel der Dissertation: „Cryptographic Clocks and Applications“
[2] giulio.malavolta@hotmail.it

CC BY SA

Im Idealfall sollte der Nachweis aus einer einzigen Nachricht des Beweisführers an den Verifizierer bestehen und öffentlich überprüfbar sein. In der Literatur ist dies als Beweis für sequentielle Arbeit bekannt. In dieser Dissertation erweitern wir dieses Konzept auf inkrementelle Beweise für sequentielle Arbeit, bei denen Rechenthreads ohne zusätzlichen Aufwand aufgeteilt oder an neue Beweisführer weitergegeben werden können. Dies trägt sehr lange Berechnungen oder möglichen Hardwareausfällen Rechnung, die die Lebensdauer eines einzelnen Beweisführers überschreiten. Vor dieser Arbeit waren Beweise für sequentielle Arbeit durch die Zeitkomplexität des Beweisführers von $T \quad \sqrt{T}$ und die Platzkomplexität $\sqrt{T}$ gekennzeichnet. Im Gegensatz dazu sind die hier angegebenen Konstruktionen im Wesentlichen optimal in dem Sinne, dass der Beweisführer eine in T lineare Zeitkomplexität und eine in T polylogarithmische Platzkomplexität aufweist. Unsere Schemata basieren ausschließlich auf symmetrischen Grundelementen und sind in der Praxis effizient.

Zeitschlossrätsel. Ein Zeitschlossrätsel erlaubt es, Geheimnisse für eine bestimmte Zeit T zu verkapseln, d.h. es ist möglich, Nachrichten so zu verschlüsseln, dass der Klartext nach Zeit T (für beliebiges T), aber nicht vorher wiederhergestellt werden kann. Dies kann als ein feinkörnigerer Begriff der probabilistischen Verschlüsselung angesehen werden, bei dem T immer exponentiell groß gesetzt wird. Die Anwendungen von Zeitschlossrätseln reichen von elektronischen Wahlen bis hin zu fairen Verträgen. Unglücklicherweise erfordern Szenarien, in denen mehrere Parteien beteiligt sind, dass man im schlimmsten Fall alle Rätsel, die während des Protokolls generiert werden, lösen muss. Abhängig von der Anzahl der Teilnehmer kann es unsinnig sein, anzunehmen, dass eine solche parallele Leistungsfähigkeit zur Verfügung steht. Um dieses Problem zu lösen, führen wir das Konzept der homomorphen Zeitschlossrätsel ein, bei denen es möglich ist, öffentlich Funktionen auf den Geheimnissen zu berechnen, ohne die entsprechenden Rätsel zu öffnen. Danach muss nur noch das Rätsel, das die Funktionsausgabe kodiert, gelöst werden, wodurch der Rechenaufwand der Parteien stark reduziert wird. In dieser Arbeit werden Zeitschlossrätsel definiert und instanziiert, die die homomorphe Auswertung verschiedener Funktionsfamilien zulassen. Im Anschluss wird gezeigt, wie homomorphe Zeitschlossrätsel als Eckpfeiler für Protokolle für elektronische Wahlen, Auktionen mit versiegelten Geboten, Mehrparteienmünzwurf und Vertragsunterzeichnung dienen.

Atomare Transaktionen. Gegeben sei das Szenario, in welchem eine Partei einige Münzen über eine Kette von Mittelsmännern an eine andere Partei weitergeben möchte. Offensichtlich ist es keine gute Strategie, die nächste Partei in der Kette zu bezahlen und zu hoffen, dass sie die Münzen schließlich vorwärts gibt. In dieser Dissertation wird das Konzept der Mehrschrittschlösser als Lösung für dieses Problem vorgestellt. Ein Mehrschrittschloss ermöglicht es, in einer Kette von Benutzern eine Reihe von Verträgen abzuschließen, die besagen:„Wenn eine bestimmte Bedingung vor der Zeit T erfüllt ist, zahlt A eine Münze an B.“ Der Empfänger (d.h. der letzte Knoten in der Kette) kennt eine Lösung, die er freigibt, wenn die Kette von Verträgen bei ihm ankommt. Dies löst eine Kettenreaktion aus, bei der jeder Knoten die Lösung berechnen kann, um seine linke Verbindung freizugeben, sobald die Bedingung für seine rechte Verbindung erfüllt ist. Mehrschrittschlösser werden in Verbindung mit einer Blockchain verwendet, um indirekte und schnelle Zahlungen zu ermöglichen und die

Atomarität der Transaktion in dem Sinne zu gewährleisten, dass entweder die gesamte Kette aktualisiert wird oder alle Verbindungen unangetastet bleiben. Darüber hinaus garantieren Mehrschrittschlösser ein gewisses Maß an Anonymität aus der Sicht der Zwischenknoten. Es wird gezeigt, wie man Mehrschrittschlösser aus einfachen Annahmen so instanziiert, dass sie mit der großen Mehrheit der Kryptowährungen, einschließlich Bitcoin, kompatibel sind.

Hintergrund und Tragweite dieser Dissertation

Diese Dissertation basiert auf Veröffentlichungen bei führenden Konferenzen der Kryptographie (EUROCRYPT, CRYPTO) sowie der Computersicherheit (NDSS). Des Weiteren wurden die Arbeiten bei diversen weiteren Workshops präsentiert und zur Präsentation bei dem Charles River Crypto Day at Microsoft Research (2019) sowie bei der Konferenz Advances in Financial Technologies (2019) als ein Höhepunkt der Forschung eingeladen. Eine der Veröffentlichungen befand sich unter den Finalisten im CSAW Wettbewerb für angewandte Forschung (2019). Die in dieser Dissertation entwickelte kryptographische Maschinerie wurde anschließend vom Lightning Network sowie KZen Network, zwei auf der Blockchain basierenden Bezahlnetzwerken, implementiert.

Inkrementelle Beweise für sequentielle Arbeit

Man stelle sich vor, jemand habe eine mögliche Lösung für die Goldbachsche Vermutung. Diese Person wäre sich fast sicher, dass ihr Ansatz fundiert ist, jedoch noch nicht bereit für eine Veröffentlichung, da weitere Prüfung notwendig wäre. Auf der anderen Seite bestünde die Gefahr, dass eine andere Partei das Resultat für sich beanspruchte, weshalb die Person einen Zeitstempel für ihre Entdeckung benötigte. Eine Lösung besteht darin, sich auf eine Drittstelle zu verlassen, die die Authentizität des Dokuments bescheinigt, wobei das Vertrauen in einen solchen Nachweis jedoch an das Vertrauen gebunden ist, das man in eine solche Stelle hat. Eine wünschenswertere Alternative ist ein universell verifizierbares kryptographisches Protokoll, das sich auf die Schwierigkeit ausgiebig untersuchter Probleme verlässt.

Ein Beweis für sequentielle Arbeit (engl. Proof of Sequential Work) (PoSW) [MMV13] erlaubt es einem Beweisführer $\mathcal{P}$, bei Eingabe einer bestimmten Aussage χ und eines Zeitparameters T, einen zweifelnden Verifizierer $\mathcal{V}$ davon zu überzeugen, dass seit der Festlegung von χ eine bestimmte Zeit $(1-\alpha)\cdot T$ (für einige feste Konstante α) verstrichen ist. Eine PoSW wird als nicht-interaktiv bezeichnet, wenn die Kommunikation zwischen $\mathcal{P}$ und $\mathcal{V}$ aus einer einzigen Nachricht $\mathcal{P} \xrightarrow{\pi} \mathcal{V}$ besteht. Eine PoSW muss *vollständig* sein, d. h. ein ehrlicher Beweisführer $\mathcal{P}$ muss immer in der Lage sein, den Verifizier $\mathcal{V}$ zu überzeugen, und *zuverlässig*, was bedeutet, dass keine bösartige Maschine in der Lage ist, $\mathcal{V}$ 1 in einer Zeit $< \alpha T$ ausgeben zu lassen. Um Effizienz zu gewährleisten, wird verlangt, dass der Beweis π in (möglicherweise paralleler) Zeit in etwa T und mit wenig Speicher, idealerweise $\log(T)$, berechenbar ist. Die Größe von π und die Zeit für die Verifizierung muss ebenfalls höchstens logarithmisch in T sein.

Inkrementeller PoSW. Eine natürliche Frage ist, ob Beweise für sequentielle Arbeit monolithische Objekte sind oder ob sie eine gewisse Flexibilität zulassen, z.B. die Verlagerung der Arbeitsbelastung auf einen anderen Beweisführer oder die Aufteilung der Berechnung eines Beweises zwischen zwei unabhängigen Parteien. Dies ist von besonderem Interesse, wenn Hardwareausfälle berücksichtigt werden, die bei der Berechnung von langlebigen Beweisen auftreten können. Für wichtige Berechnungen kann es sinnvoll sein, parallele Prozesse zu erzeugen und je nach den Umständen von einem zum anderen zu migrieren.

Die vorliegende Dissertation schließt diese Lücke, indem sie den Begriff der inkrementellen Beweise sequentieller Arbeit (engl. incremental Proofs of Sequential Work) (iPoSW) einführt. Alle diese Ergebnisse basieren auf der in [DLM19] veröffentlichten Arbeit. Vereinfacht gesagt, erlaubt ein iPoSW einem Beweisführer, den Beweis π für eine bestimmte Aussage χ und den Zeitparameter T zu berechnen und später die Berechnung fortzusetzen, um einen Beweis π' für den Zeitparameter $T \quad T'$ zu berechnen. Die Zeitkomplexität des zweiten Beweisführers ist unabhängig von T und seine Speicherkomplexität ist durch eine logarithmische Funktion in der Gesamtzeit begrenzt. Außerdem verlangen wir, dass π und π' identisch verteilt sind.

Aktueller Stand. Wir haben folgende triviale Konstruktion eines iPoSW aus beliebigem PoSW. Ausgehend von einem Beweis π für eine Aussage χ und dem Zeitparameter T, kann man diesen auf die Zeit $T \quad T'$ inkrementieren, indem man eine neue Aussage χ' ableitet (z.B. durch Hashen $\chi' \leftarrow \mathsf{H}(\chi,\pi)$) und einen neuen Beweis π' für χ' mit dem Zeitparameter T' berechnet. Der neue Beweis besteht aus dem Paar (π,π') und kann leicht überprüft werden, indem die Aussage χ' wieder berechnet wird und zwei parallele Instanzen des Verifikationsalgorithmus auf (χ,T,π) und (χ',T',π') ausgeführt werden. Diese naive Lösung erfüllt jedoch keine Kompaktheitsanforderung in dem Sinne, dass die Größe des Beweises linear mit der Anzahl der Inkremente wächst. Insbesondere gibt es keine Hoffnung, dass inkrementierte Beweise die gleiche Verteilung haben wie (nicht inkrementierte) Standardbeweise. Das ist vor allem dann nicht wünschenswert, wenn der Beweis stark fragmentiert ist oder wenn er an viele neue Beweisführer weitergegeben wird.

Außerdem lassen bekannte Instanzen von PoSW keine optimalen Parameter für die Zeit- und Speicherkomplexität des Beweisführers zu. Es sind zwei Hauptkandidaten bekannt:

(1) Das ursprüngliche Schema von Mahmoody, Moran und Vadhan [MMV13] basiert auf tiefenrobuste Graphen und Beweise können in Zeit und Raum proportional zu T berechnet werden.

(2) Die neueste Arbeit von Cohen und Pietrzak [CP18] führt ein neues kombinatorisches Objekt ein und schlägt eine Konstruktion vor, die es dem Beweisführer erlaubt, Speicher gegen zusätzliche Rechenzeit einzutauschen. Bei optimalen Parametern werden die Beweise in der Zeit $T \quad \sqrt{T}$ und mit dem Speicher $\sqrt{T}$ berechnet.

Unsere Ergebnisse. In dieser Dissertation definieren wir inkrementelle Beweise sequentieller Arbeiten und zeigen, wie sie mit im Wesentlichen optimalen Parametern instanziiert werden können, wobei die Berechnungskomplexität des Beweisführers linear in T wächst und seine Raumkomplexität polylogarithmisch in T. Dies schließt die Speicherlücke für den

Beweisführer und löst ein bislang offenes Problem. Wir schlagen zwei nicht-interaktive Konstruktionen vor, die auf der Kombination von Cohen-Pietrzak-Graphen [CP18] mit neuen Techniken für die Auswahl von Herausforderungsknoten basieren.

(1) Unser erstes Schema geht davon aus, dass der Beweisführer Zugang zu einem einzigen Prozessor hat und liefert Beweise der Größe $O(\log(T)^2)$.

(2) Unser zweites Schema geht davon aus, dass der Beweisführer $\log(T)$-viele parallele Prozessoren hat. Die Beweisgröße ist nur um einen Faktor von 9 größer als die von [CP18].

Wir betonen, dass die Annahme, dass $\log(T) \approx 100$, eine sichere Obergrenze für die Anzahl der für unsere zweite Konstruktion benötigten Prozessoren ist. Dies entspricht der Annahme der Existenz von 100 unabhängigen Kernen für die Berechnung einer Hashfunktion, was für moderne GPUs angemessen ist. Beide Schemata sind einfach zu implementieren und machen nur einen Blackbox-Aufruf an eine Hashfunktion (modelliert als ein Zufallsorakel).

Homomorphe Zeitschlossrätsel

Ein Zeitschlossrätsel [RSW96] erlaubt es, Nachrichten für die Zukunft zu verschlüsseln. Das bedeutet, dass ein Rätsel bezüglich eines geheimen Inputs s und eines Zeitparameters T generiert und sichergestellt wird, dass s vor den Augen eines polynomiell großen Angreifers versteckt wird, der in einer Zeit von (deutlich) weniger als T läuft. Das entscheidende Merkmal von Zeitschlossrätseln ist, dass die Anzahl der parallelen Prozesse des Angreifers keinen Einfluss auf die Geschwindigkeit hat, mit der s wiederhergestellt wird, d. h. das Lösen eines Rätsels ist eine inhärent sequentielle Operation. Dies steht im Gegensatz zur Standardauffassung von probabilistischer Verschlüsselung [GM82], bei der eine Senkung des Sicherheitsparameters des Schemas schnellere Algorithmen ermöglichen kann, die eine massive Parallelisierung ausnutzen. Ein weiterer wichtiger Aspekt von Zeitschlossrätseln ist, dass die Zeit, die zum Erstellen eines Rätsels benötigt wird, im Wesentlichen unabhängig von T ist. Das ist besonders dann wünschenswert, wenn ein Geheimnis über einen längeren Zeitraum, z.B. 10 Jahre, versteckt werden muss.

Eines der motivierenden Beispiele für die Nützlichkeit von Zeitschlossrätseln sind elektronischen Wahlen: Der Standardansatz, um (zumindest in der Anfangsphase) die Vertraulichkeit von Stimmen zu gewährleisten, besteht darin, dass sich jeder Benutzer auf seine Wahl committet und alle Commitments gesammelt werden. In einer zweiten Auszählungsphase werden die Benutzer aufgefordert, den Inhalt ihrer Commitments (zusammen mit den privaten Zufallsmünzen) offenzulegen, was die Berechnung und Bekanntgabe des Wahlergebnisses ermöglicht. Dies lässt jedoch die Frage offen, wie mit Benutzern umgegangen werden soll, die in der ersten Phase gültige Commitments senden, aber in der folgenden Auszählungsphase nicht öffnen. Man könnte (i) den gesamten Vorgang wiederholen, wenn dies geschieht, oder (ii) solche Stimmen ignorieren. Die erstere Option ist eindeutig nicht durchführbar, da sie es einem böswilligen Nutzer ermöglicht, die Wahl im Wesentlichen kostenlos zu verhindern. Andererseits könnte letztere Lösung ausgenutzt werden, um das Endergebnis zu manipulieren: Ein Angreifer, der

den Netzwerkverkehr kontrolliert, könnte die Eröffnungen blockieren, die einem unerwünschten Kandidaten entsprechen, und dadurch der bevorzugten Seite einen Vorteil verschaffen.

Zeitschlossrätsel lösen diesen Konflikt auf elegante Weise auf, indem sie die Commitments als Versteckmechanismus für Stimmen ersetzen. Die Stimme von Benutzern, die ihre Zufallsmünzen nicht veröffentlichen, kann einfach durch Knacken des Zeitschlossrätsels bestimmt werden. Wenn der Zeitparameter so gesetzt wird, dass er um einen sicheren Wert länger als die Abstimmungsphase ist, wird sichergestellt, dass die Wahlen geheim gehalten werden, bis eine solche Phase vorbei ist. Leider ist diese Lösung nicht ohne zusätzliche Kosten möglich: Wenn eine große Anzahl von Wählern ihr Rätsel nicht öffnet, so muss die Auszählbehörde vor der Bekanntgabe des Wahlsiegers die gleiche Anzahl von Rätseln bruteforcen, was bedeutet, dass eine massive Anzahl an Parallelberechnungen erforderlich ist, um die Wahl in angemessenem Zeitraum abzuschließen. Bedenkt man, dass es Millionen von Stimmen geben kann, so wird das Problem offensichtlich. Es ist völlig plausibel, dass die für den Zeitschlossansatz benötigten Ressourcen von der Entscheidung für diese Lösung wegführen können. Sicherlich ist das Szenario der elektronischen Wahlen ein gutes Beispiel für die Skalierbarkeit von Zeitschlossrätseln, jedoch nicht der einzige Fall, in dem sie auftauchen. Jede andere Anwendung, an der eine große Anzahl von Benutzern beteiligt ist, wie z.B. versiegelte Auktionen oder Mehrparteienmünzwurf, stößt auf ähnliche Probleme. Wir vermuten, dass solche Einschränkungen eines der Haupthindernisse darstellen, die bisher die umfassende Einführung von Zeitschlossrätseln verhindert haben.

Homomorphe Zeitschlossrätsel. Das Hauptproblem bei auf Zeitschlossrätseln basierenden Lösungen ist, dass eine Partei viele Rätsel öffnen muss und dann eine Funktion über die Geheimnisse berechnen muss. Was wäre, wenn Funktionen zuerst *homomorph* berechnet und dann ein *einziges* Zeitschlossrätsel, das den Funktionsoutput beinhaltet, geknackt werden könnte? Dies würde die Hauptquelle der Ineffizienz der Protokolle, die Zeitschlossrätsel verwenden, auslöschen. In obigem Beispiel könnte die Auszählbehörde einfach den Gewinner der Wahl homomorph berechnen und dann ein einziges Rätsel lösen.

Von diesem Ziel motiviert, schlagen wir den Begriff des homomorphen Zeitschlossrätsels (engl. Homomorphic Time-Lock Puzzle) (HTLP) vor: Ein HTLP ist ein erweitertes Zeitschlossrätsel, das es jedem erlaubt, eine Funktion F über eine Menge an Rätseln homomorph zu berechnen, ohne unbedingt die kodierten Geheimnisse $(s_1,...,s_n)$ zu kennen. Das resultierende Rätsel enthält die Funktionsausgabe $F(s_1,...,s_n)$ und seine Schwere ist nicht von der Größe von F abhängig. Dieser Begriff kann als Gegenstück zur vollständig homomorphen Verschlüsselung [Ge09] für schwache Gegner gesehen werden, bei denen die Schwierigkeit mit abnehmendem Zeitparameter stark abnimmt. Wir betonen, dass die *Kompaktheit* des Algorithmus auch für HTLP eine entscheidende Voraussetzung ist: Wenn man sie ignoriert, so ist die triviale Lösung, die Eingaberätsel zu lösen und dann die Funktion über die Geheimnisse im Klartext zu berechnen, ausreichend.

Unsere Ergebnisse. In dieser Dissertation stellen wir den Begriff des homomorphen Zeitschlossrätsels vor, wir präsentieren formale Definitionen, schlagen verschiedene Schemata für verschiedene Klassen von Funktionen vor und zeigen ihre Nützlichkeit anhand

einiger realer Anwendungen auf. Alle folgenden Ergebnisse basieren auf den vom Autor veröffentlichten und in [MT19] vorgestellten Ergebnissen.

(1) Ein HTLP für lineare Funktionen, das die inhärente Sequentialität der wiederholten Quadrierung in Gruppen zusammengesetzter Ordnung annimmt. Dies ist die gleiche Annahme, die von Rivest, Shamir und Wagner für die Konstruktion des ersten (nicht-homorphen) Zeitschlossrätsels [RSW96] verwendet wurde.

(2) Ein multiplikativ-homomorphes Zeitschlossrätsel aus Standardannahmen in Gruppen zusammengesetzter Ordnung.

(3) Ein vollständig homomorphes Zeitschlossrätsel (oder HTLP für jede Funktion polynomieller Größe F) unter der Annahme, dass es eine ununterscheidbare Verschleierung für probabilistische Funktionen gibt [Ca15].

Die beiden erstgenannten Schemata basieren auf Standardannahmen (in Bezug auf die Faktorisierung großer Zahlen) und sind einfach und effizient genug, um in realen Anwendungen eingesetzt zu werden, während das letztere von einer stärkeren Maschinerie (der Existenz eines Programmverschleierers) ausgeht und als ein Machbarkeitsergebnis von HTLP für jede Funktion interpretiert werden soll. Wir untersuchen verschiedene Erweiterungen unserer grundlegenden Schemata, z. B. die Möglichkeit der Wiederverwendung des Setups für eine unbegrenzte Anzahl von Rätseln und Algorithmen zur Berechnung von Funktionen über Rätsel, die mit unterschiedlichen Härteparametern erzeugt wurden. Wir schlagen auch ein halbkompaktes HTLP für Programme mit unbegrenzter Verzweigungstiefe vor. Schließlich zeigen wir, wie verschiedene Arten von HTLP als Hauptbaustein verwendet werden können, um völlig sichere und *faire* Protokolle zu konstruieren, wie z. B.:

(1) Ein elektronisches Wahlverfahren und ein Protokoll für Auktionen mit versiegelten Geboten, welches eine beliebige Anzahl von Parteien unterstützt, die nach den ersten Phasen offline gehen, ohne dass ein erheblicher Rechenaufwand entsteht oder das Ergebnis des Protokolls verändert wird.

(2) Ein effizientes und skalierbares Mehrparteienmünzwurfprotokoll, das gegen selektive Fehler- und Dienstbehinderungsangriffe resistent ist. Ein solches Protokoll kann auf heutigen Maschinen ohne größeren Rechenaufwand ausgeführt werden.

(3) Eine Mehrparteienvertragsunterzeichnung, bei der ein kürzlich erzieltes Ergebnis zu aggregierten Unterschriften genutzt wird [HW18]. Die Fairness des Protokolls spiegelt sich darin wider, dass bei vorzeitigem Ausstieg einer Partei der Wiederherstellungsaufwand für ehrliche Parteien (unabhängig von deren Anzahl) etwa doppelt so groß ist.

Mehrschrittschlösser

Vollständig dezentralisierte Kryptowährungen wie Bitcoin [Na08] entstanden kürzlich als Alternative zu traditionellen Zahlungssystemen. Anstatt Zahlungen zu protokollieren und Kontostände auf einem zentralen Server zu verwalten, wird eine Datenbank über alle Anwender im

Netzwerk repliziert. Aktualisierungen werden in einer Blockchain gespeichert und durch einen Konsensalgorithmus validiert, an dem alle (möglicherweise misstrauischen) Maschinen im Netzwerk beteiligt sind. Dies begrenzt die Transaktionsrate des Systems auf natürliche Weise und schränkt die Verwendung dezentralisierter Währungen für kleine Transaktionen stark ein.

Zahlungskanalnetzwerke (engl. Payment Channel Networks) (PCNs) [Ma17] haben sich als die praktischste Lösung zur Abmilderung dieser Skalierbarkeitsprobleme erwiesen, da sie es ermöglichen, den Großteil der Zahlungen zwischen zwei Anwendern außerhalb der Blockchain durchzuführen. Vereinfacht gesagt öffnen zwei Anwender einen gemeinsamen Kanal, indem sie einige Münzen an einer gemeinsamen Adresse hinterlegen, die durch einen intelligenten Vertrag abgesichert ist. Spätere Zahlungen zwischen ihnen können dann durch die Vereinbarung eines neuen Saldos durchgeführt werden, ohne die Notwendigkeit, Informationen in die Blockchain hochzuladen. Dies erhöht den Transaktionsdurchsatz der Blockchain drastisch, da die einzige weitere Interaktion, die benötigt wird, das Schließen des Kanals und das Freigeben der entsprechenden Gelder ist. Die Verwendbarkeit des Netzwerkes kann weiter erhöht werden, indem man Transaktionen zwischen Nutzern erlaubt, die nicht direkt einen gemeinsamen Kanal teilen, aber über mehrere Schritte verbunden sind. Es gibt heute mehrere Umsetzungen dieser Idee, wobei der prominenteste Vorschlag der des Lightning Networks ist [PD16]. Jüngste Ergebnisse [Ma19b] haben jedoch die Unzulänglichkeit der derzeitigen Designprinzipien für PCNs offenbart, indem sie einen Angriff auf alle wichtigen Systeme demonstrieren, der es erlaubt, die Gebühren von ehrlichen Mittelsmännern auf einem Zahlungsweg zu stehlen. Dies drängt zu einer fundierten Untersuchung der zugrunde liegenden Mechanismen und zur Erlangung eines tieferen Verständnisses der Sicherheits- und Datenschutzgarantien für ein PCN.

Mehrschrittschlösser. Wir schlagen Mehrschrittschlösser (engl. Multi-Hop Locks) (MHL) vor, ein neues kryptographisches Primitiv, das als Eckpfeiler für die Gestaltung von PCNs dient. Eine MHL ermöglicht es einer Kette von Anwendern, ihre Verbindungen schrittweise zu sperren, bis der letzte Anwender erreicht ist. Ein solcher Anwender kann dann nach seinem Ermessen eine Kaskadenfreigabe der Sperren auslösen, z. B. wenn eine bestimmte Menge von Bedingungen erfüllt ist. Eine MHL sollte drei grundlegende Eigenschaften erfüllen.

(1) Atomarität: Wenn die ausgehende Kante eines Mittelsmannes in der Kette freigegeben wurde, dann kann ein solcher Anwender seine eingehende Kante jederzeit freigeben.
(2) Konsistenz: Jeder Zwischenanwender in der Kette kann seine eingehende Kante nur dann freigeben, wenn seine ausgehende Kante bereits vom vorherigen Anwender freigegeben wurde.
(3) Anonymität: Jede Sperre sollte keine Informationen über die Kette, zu der sie gehört, preisgeben.

MHLs erfassen den Kernmechanismus, der jedem PCN zugrunde liegt, und ermöglichen einen modularen Aufbau und das Verständnis größerer Systeme.

Unsere Ergebnisse. In dieser Dissertation motivieren, definieren und konstruieren wir MHLs. Genauer modellieren wir die Sicherheit der MHL im Rahmen der universellen Kombinierbarkeit (engl. Universal Composability) (UC) [Ca01], was erlaubt, über die

Verwendung von MHLs in einer größeren Umgebung nachzudenken. Dann schlagen wir zwei Instanziierungen für MHLs vor:

(1) Wir zeigen, dass linear homomorphe Einwegfunktionen ausreichen, um MHLs zu konstruieren, und positionieren damit dieses Primitiv in einem Bereich gut studierter Annahmen.

(2) Wir konstruieren MHLs in einer vertragsfreien Umgebung, wobei die Gültigkeit einer Einwegfunktion durch einfache Verifizierung einer ECDSA-Signatur überprüft werden kann. Dieser Ansatz ist von besonderem Interesse, da er die Verwendung von MHLs in Verbindung mit Blockchains ermöglicht, die keine turingmächtige Skriptsprache (wie z.B. Bitcoin) unterstützen. Vor unserer Arbeit wurde dies als ein offenes Problem in diesem Bereich betrachtet.

Schließlich untersuchen und diskutieren wir verschiedene Anwendungen unseres Primitivs: Das MHL kann in einer Blackbox-Art verwendet werden, um effizientere PCNs mit konkreten Datenschutzgarantien zu konstruieren und atomare Münztauschaktionen, sogar über verschiedene Währungen hinweg, zu ermöglichen. Eine Version unseres Schemas wurde durch das Lightning Network [Fr18] implementiert, um der in [Ma19b] aufgedeckten Sicherheitsverletzung entgegenzuwirken. Die oben genannten Ergebnisse basieren auch auf der (vom Autor) in [Ma19b] veröffentlichten Arbeit.

Literatur

[Ca01] Canetti, R.: Universally Composable Security: A New Paradigm for Cryptographic Protocols. In: 42nd FOCS. IEEE Computer Society Press, S. 136–145, Okt. 2001.

[Ca15] Canetti, R.; Lin, H.; Tessaro, S.; Vaikuntanathan, V.: Obfuscation of Probabilistic Circuits and Applications. In (Dodis, Y.; Nielsen, J. B., Hrsg.): TCC 2015, Part II. Bd. 9015. LNCS, Springer, Heidelberg, S. 468–497, März 2015.

[CP18] Cohen, B.; Pietrzak, K.: Simple Proofs of Sequential Work. In (Nielsen, J. B.; Rijmen, V., Hrsg.): EUROCRYPT 2018, Part II. Bd. 10821. LNCS, Springer, Heidelberg, S. 451–467, Apr. 2018.

[DLM19] Döttling, N.; Lai, R. W. F.; Malavolta, G.: Incremental Proofs of Sequential Work. In (Ishai, Y.; Rijmen, V., Hrsg.): EUROCRYPT 2019, Part II. Bd. 11477. LNCS, Springer, Heidelberg, S. 292–323, Mai 2019.

[Fr18] Fromknecht, C.: tpec: 2P-ECDSA Signatures, Github repository, `https://github.com/cfromknecht/tpec`, 2018.

[Ge09] Gentry, C.: Fully homomorphic encryption using ideal lattices. In (Mitzenmacher, M., Hrsg.): 41st ACM STOC. ACM Press, S. 169–178, Mai 2009.

[GM82] Goldwasser, S.; Micali, S.: Probabilistic Encryption and How to Play Mental Poker Keeping Secret All Partial Information. In: 14th ACM STOC. ACM Press, S. 365–377, Mai 1982.

[HW18] Hohenberger, S.; Waters, B.: Synchronized Aggregate Signatures from the RSA Assumption. In (Nielsen, J. B.; Rijmen, V., Hrsg.): EUROCRYPT 2018, Part II. Bd. 10821. LNCS, Springer, Heidelberg, S. 197–229, Apr. 2018.

[Ma17] Malavolta, G.; Moreno-Sanchez, P.; Kate, A.; Maffei, M.; Ravi, S.: Concurrency and Privacy with Payment-Channel Networks. In (Thuraisingham, B. M.; Evans, D.; Malkin, T.; Xu, D., Hrsg.): ACM CCS 2017. ACM Press, S. 455–471, Okt. 2017.

[Ma19a] Malavolta, G.: Cryptographic Clocks and Applications, doctoralthesis, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 2019, S. 108, URL: https://nbn-resolving.org/urn:nbn:de:bvb:29-opus4-112571.

[Ma19b] Malavolta, G.; Moreno-Sanchez, P.; Schneidewind, C.; Kate, A.; Maffei, M.: Anonymous Multi-Hop Locks for Blockchain Scalability and Interoperability. In: NDSS 2019. The Internet Society, 2019.

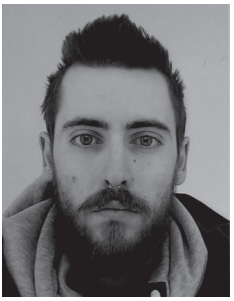
[MMV13] Mahmoody, M.; Moran, T.; Vadhan, S. P.: Publicly verifiable proofs of sequential work. In (Kleinberg, R. D., Hrsg.): ITCS 2013. ACM, S. 373–388, Jan. 2013.

[MT19] Malavolta, G.; Thyagarajan, S. A. K.: Homomorphic Time-Lock Puzzles and Applications. In (Boldyreva, A.; Micciancio, D., Hrsg.): CRYPTO 2019, Part I. Bd. 11692. LNCS, Springer, Heidelberg, S. 620–649, Aug. 2019.

[Na08] Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System, https://bitcoin.org/bitcoin.pdf, 2008.

[PD16] Poon, J.; Dryja, T.: The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments, Technical Report, https://lightning.network/lightning-network-paper.pdf, 2016.

[RSW96] Rivest, R. L.; Shamir, A.; Wagner, D. A.: Time-Lock Puzzles and Timed-Release Crypto./, 1996.



Giulio Malavolta ist derzeit Postdoc an der UC Berkeley und der Carnegie Mellon University. Zuvor war er als wissenschaftlicher Mitarbeiter am Simons Institute for the Theory of Computing tätig. Im Juni 2019 schloss er seine Promotion an der Friedrich-Alexander-Universität Erlangen-Nürnberg ab. Er hat einen Master in Informatik an der Universität des Saarlandes, den er im Dezember 2016 abgeschlossen hat. Sein Forschungsinteresse erstreckt sich auf die Theorie und die Anwendungen der Kryptographie und Computersicherheit. Seine Arbeit wurde auf führenden Konferenzen in den Bereichen Kryptographie (CRYPTO, EUROCRYPT), Berechenbarkeitstheorie (FOCS) und Systemsicherheit (S&P, CCS, NDSS) veröffentlicht.

Visualisierung, Klassifikation, und Interaktion für die Risikoanalyse und Behandlungsplanung zerebraler Aneurysmen[1]

Monique Meuschke[2]

Abstract: Die vorliegende Dissertation präsentiert die Software *Aneulysis* für eine verbesserte Risikoanalyse und Behandlungsplanung zerebraler Aneurysmen. Dies sind lokale Ausbeulungen der Gehirnarterien, deren Ruptur meist tödlich endet. Eine Behandlung muss sorgfältig abgewogen werden, da diese enorme Risiken für den Patienten bürgen. Aneurysma-Wachstum, Ruptur, und Behandlungserfolg hängen von der Gefäßmorphologie und Hämodynamik ab, deren Zusammenhänge bisher schlecht verstanden sind. Mit Hilfe von Simulationen kann die patientenspezifische Hämodynamik modelliert werden. Die Analyse der komplexen Daten ist jedoch zeitaufwendig und fehleranfällig. *Aneulysis* ermöglicht eine interdisziplinäre Nutzung und reproduzierbare Analyse von Aneurysmadaten. Die Konzeption und Evaluierung erfolgte in enger Zusammenarbeit mit Neuradiologen und Simulationsexperten, die den Nutzen und die klinische Notwendigkeit bestätigten.

1 Einführung

Zerebrale Aneurysmen bilden sich aufgrund einer instabilen Gefäßwand. Ihre jährliche Prävalenz liegt bei etwa 5 % in der westlichen Population und betrifft ca. 4,5 Millionen Menschen in Deutschland. Die größte Gefahr besteht in der Ruptur der Gefäßwand, was zu einer inneren Blutung führt. Die jährliche Inzidenz liegt bei 0,5 %, wobei die Folgen einer Ruptur verheerend sind. Etwa die Hälfte der Betroffenen versterben. Zudem tragen 50 % der Überlebenden schwere Defizite (Lähmungen, geistige Beeinträchtigung etc.) davon, ein Drittel bleibt lebenslang abhängig von fremder Hilfe. Damit stellt diese Erkrankung eine erhebliche gesamtgesellschaftliche Belastung dar mit einer jährlichen ökonomischen Belastung von ca. 437 Mio. Euro in Deutschland.

Häufig werden Aneurysmen zufällig durch den verstärkten Einsatz bildgebender Verfahren entdeckt. Derzeit gibt es zwei Behandlungsmöglichkeiten: chirurgisch oder endovaskulär mit dem Ziel das Aneurysma vom Gefäßsystem zu trennen. Beide Therapieansätze bergen jedoch Risiken. Aneurysmen können während der Operation reißen, oder eine vollständige Abtrennung kann nicht erreicht werden. Die geringe Rupturrate und die enormen Operationsrisiken schließen eine prophylaktische Behandlung aus. Daher erfordert die Detektion eines nicht rupturierten Aneurysmas eine patientenspezifische Risikoanalyse [Ce05]. Im Fall einer endovaskulären Behandlung stehen zudem verschiedene Gefäßprothesen zur Verfügung, wobei eine für den Patienten optimale Wahl getroffen werden muss.

[1] Englischer Titel der Dissertation:"'Visualization, Classification, and Interaction for Risk Analysis and Treatment Planning of Cerebral Aneurysms"'

[2] Otto-von-Guericke-Universität Magdeburg, Fakultät für Informatik, Universitätsplatz 2, 39106 Magdeburg, Deutschland, meuschke@isg.cs.uni-magdeburg.de

CC BY SA

Motivation für die vorliegende Arbeit

Die Entstehung und das Wachstum eines Aneurysmas hängen von mehreren Faktoren ab, deren Zusammenhänge nicht gut verstanden sind. Neben geometrischen (morphologischen) Merkmalen hat der Blutfluss (Hämodynamik) einen wichtigen Einfluss auf dessen Stabilität. Quantitative Strömungsattribute wie die tangentiale Wandscherspannung (WSS) und qualitative Merkmale, wie verwirbelnder Blutfluss scheinen wichtige Indikatoren für die Aufweitung und die Ruptur eines Aneurysmas zu sein.

Im klinischen Alltag basiert die Risikobewertung und Behandlungsentscheidung auf Leitlinien, die nur morphologische Merkmale wie Größe und Form des Aneurysmas berücksichtigen. Diese erlauben jedoch keine zuverlässige Abschätzung des Einzelrisikos. Daher besteht ein ausgezeichnetes Interesse daran, Faktoren zu identifizieren, die das Wachstum und die Ruptur beeinflussen, um die Behandlung auf Hochrisikopatienten zu beschränken.

Blutflusssimulationen modellieren die patientenspezifische Hämodynamik und Wandmechanik, da der zerebrale Blutfluss noch nicht gemessen werden kann [Ja15]. Dies führt zu zeitabhängigen Daten, die einen Herzzyklus darstellen. Die Untersuchung hämodynamischer Informationen hat aufgrund verbesserter Bildmodalitäten und höherer Computerleistung zunehmend an Bedeutung gewonnen, was zu einer genaueren Modellierung des Flussverhaltens führt. Basierend auf diesen Daten vergleichen biomedizinische Forscher rupturierte und nicht rupturierte Fälle sowie verschiedene Behandlungen hinsichtlich risikoreicher Korrelationen zwischen morphologischen Eigenschaften und Strömungsinformationen. Die Detektion stark ausgeprägter Korrelationen würde zudem erlauben Wissen über den Blutfluss auf Basis der Morphologie abzuleiten, ohne diesen zu simulieren.

Die visuelle Analyse von Aneurysmadaten ist zeitaufwendig und herausfordernd. Mehrere Probleme werden kaum mit bestehenden Visualisierungstechniken adressiert. Simulierte Blutflussdaten sind hochkomplexe Informationen, die skalare, vektorielle und tensorbasierte Eigenschaften umfassen, deren Visualisierung oft durch visuelle Überladung und Verdeckungen beeinträchtigt wird. Das Gefäß wird meist durch ein 3D-Oberflächennetz dargestellt, was eine Reihe von manuellen Drehungen erfordert, um einen Überblick zu erhalten. Auf der Oberfläche können skalare Größen untersucht werden, wobei standardmäßig nur ein Attribut gleichzeitig dargestellt werden kann. Beim Drehen und Zoomen wechseln die Experten die Attribute, sodass interessante Bereiche gefunden werden. Folglich müssen sie sich die Bereiche merken und hinsichtlich anderer Größen analysieren, was die Erkennung von Zusammenhängen sehr schwierig macht. Die Zeitabhängigkeit macht es praktisch unmöglich, auf diese Weise alle relevanten Teile der Daten darzustellen.

Die Verwendung von Side-by-Side-Ansichten verbessert die Datenexploration nur geringfügig, da nur wenige Zeitschritte und Attribute gleichzeitig dargestellt werden können. Außerdem wird die Analyse der qualitativen Strömungsmerkmale manuell durchgeführt, was ein subjektiver und fehleranfälliger Prozess ist. Kleine Strukturen können leicht übersehen werden, und die Ergebnisse sind nicht reproduzierbar, was ihre Interpretation beeinträchtigt. Schließlich ist die Durchführung großer medizinischer Studien mit verschiedenen Fachleuten eine Herausforderung, da unterschiedliche Softwarelösungen mit inkonsistenten Datenformaten verwendet werden.

2 Wissenschaftliche Beiträge der Dissertation

Eine effiziente und objektive Analyse von Aneurysmen erfordert Visualisierungssysteme, die alle relevanten morphologischen und hämodynamischen Informationen in einer integrierten Ansicht darstellen. Daher haben wir den Forschungsprototyp *Aneulysis* [Me19] entwickelt - ein interaktives Management- und Visualisierungssystem zur Bewertung des Rupturrisikos und der Behandlungsentscheidung zerebraler Aneurysmen. Designentscheidungen und Entwicklungen basieren auf Diskussionen mit Neuroradiologen und CFD-Ingenieuren, über ihre Ziele in der Aneurysmenforschung sowie Probleme mit den derzeit verwendeten Systemen. Demnach haben wir die folgenden Hauptbeiträge geleistet:

- **Konsistentes Datenmanagement und geführte Exploration:** Das Fehlen einer konsistenten Datenverwaltung zwischen verschiedenen Fachleuten sowie einer Explorationssoftware mit geführten Abläufen erschwert die Integration von simulierten Strömungsdaten in klinische Analysen. *Aneulysis* bietet neben einem Konzept zur Datenverwaltung fünf Visualisierungsmodule. Das erste Modul konzentriert sich auf eine automatische Extraktion morphologischer Deskriptoren, wie die Höhe und Breite des Aneurysmas. Dies führt zu reproduzierbaren und weniger fehleranfälligen Ergebnissen. Im Vergleich hat allein die manuelle Messung eines größten Durchmessers eine hohe Interobserver-Variabilität, wobei resultierende Messabweichungen die Behandlungsentscheidungen erheblich beeinflussen können.
- **Oberflächengestützte Datenexploration:** Um Korrelationen zwischen morphologischen und hämodynamischen Eigenschaften zu finden, werden typischerweise verschiedene Attribute auf die Eckpunkte der 3D-Oberfläche abgebildet, wobei verdächtige Bereiche leicht übersehen werden können. Daher entwickelten wir mehrere neuartige Methoden zur gleichzeitigen Exploration dieser Attribute, die in drei weitere Module integriert sind. Das erste konzentriert sich auf eine kombinierte Analyse von Wanddicke und Wanddeformation zusammen mit dem internen Blutfluss. Das zweite Modul ermöglicht die gleichzeitige Analyse multipler Skalarfelder auf der Gefäßwand, wobei beide Module mehrere verknüpfte Ansichten verwenden, die die Daten in verschiedenen Abstraktionsebenen zeigen. Das dritte Modul umfasst verschiedene glyphbasierte Visualisierungen zur Analyse von Tensordaten.
- **Flussbasierte Datenexploration:** Medizinische Studien klassifizieren qualitative Flussmuster nach ihrer Komplexität und zeitlichen Stabilität. Die komplexeren und instabilen Typen korrelieren mit der Ruptur. Manuelle Klassifikationen sind jedoch zeitaufwendig, fehleranfällig und nicht reproduzierbar. Um Erkenntnisse bezüglich der Flussmuster in klinische Diskussionen zu integrieren, müssen aussagekräftige Studien durchgeführt werden. Daher entwickelten wir eine detaillierte Analyse der aneurysmalen Flussmuster. Zunächst werden Fluss-repräsentierende Pfadlinien geclustert, um Gruppen von Linien mit ähnlichem Flussverhalten zu erhalten. Die Gruppierung führt zu räumlich-zeitlich kompakten Clustern, die im Anschluss automatisch klassifiziert werden. Jedes Cluster wird einem von sechs vordefinierten Flusstypen zugeordnet, die leicht erweiterbar sind, um mit der anatomischen Vielfalt umzugehen. Die Klassifikation liefert zuverlässige und reproduzierbare Ergebnisse, wobei verschiedene Visualisierungstechniken eine detaillierte Analyse unterstützen.

3 *Aneulysis* - Aneurysma Analyse Framework

Aneulysis ist eine Windows-Forms-Anwendung und wurde in C# geschrieben. Grundsätzlich besteht Aneulysis aus drei Komponenten: einer Datenbank in Form eines Microsoft SQL Servers, einem FTP-Server und einer Visualisierungskomponente. Für das Rendering der Aneurysmadaten verwenden wir die Open Toolkit Bibliothek (OpenTK), eine C# Grafikbibliothek, die Zugriff auf Funktionen der Open Graphics Library (OpenGL) bietet.

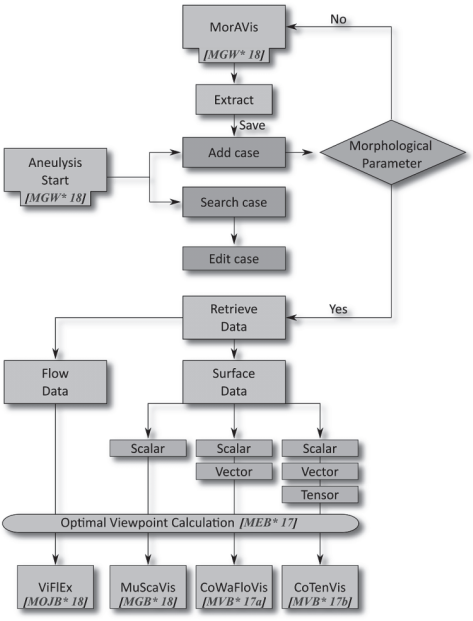


Abb. 1: Überblicksdarstellung des Explorationsprozesses mit Aneulysis.

Die patienten-spezifischen Rohdaten werden auf dem FTP-Server verwaltet. Dafür haben wir eine konsistente und redundanzfreie Dateistruktur entworfen. Für jeden Patienten wird ein neuer Fall auf dem Server angelegt, wobei Dateien zu unterschiedlichen Behandlungszeitpunkten, beispielsweise vor und nach einem Eingriff, in verschiedenen Studien organisiert werden. Pro Studie werden klinische Bilddaten, rekonstruierte Daten wie das Gefäßmesh, Flusssimulationen und charakteristische Bilder gespeichert. Darüber hinaus werden Metainformationen wie z.B. Geschlecht und Alter des Patienten sowie morphologische Parameter wie die Lokalisation und Größe des Aneurysmas in der Datenbank gespeichert.

Ein Datensatz besteht aus einer Gefäßoberfläche, wandbezogenen Daten, die auf der Oberfläche definierte Skalar-, Vektor- und Tensordaten enthalten, und strömungsbezogenen Daten, die den inneren Blutfluss beschreiben, zu sehen in Abbildung 1. Diese Informationen werden als Eingabe für die Visualisierungskomponente verwendet. Sie besteht aus fünf Modulen, die jeweils mit der Datenbank und dem Server kommunizieren, um die Ergebnisse, z.B. Bilder der visuellen Analyse, zu speichern.

3.1 Grundlegender Workflow von *Aneulysis*

Wir stellen einen geführten Explorationsprozess zur Verfügung, wobei bereits erkannte Auffälligkeiten aus vorherigen Analysen dargestellt werden können und somit eine kollaborative Exploration möglich ist. Sobald *Aneulysis* gestartet wird, öffnet sich das Hauptfenster, das eine tabellarische Übersicht aller vorhandenen Datensätze zeigt, zu sehen in Abbildung 2. Neben grundlegenden Informationen, wird ein Vorschaubild des Aneurysmas gezeigt, generiert mit unserer Methode, um optimale Ansichten auf Gefäßoberflächen zu berechnen [Me17a]. Ausgehend vom Hauptfenster hat der Nutzer drei Möglichkeiten

fortzufahren, siehe Abbildung 1. Erstens kann ein Patient hinzugefügt werden, in dem verschiedene Metadaten für neue Fälle, Studien und Aneurysmen bereitgestellt werden können. Zweitens bieten wir konfigurierbare Filtermasken zur Suche nach bestimmten Kohorten an, siehe Abbildung 2. Drittens kann der Benutzer einen Fall bearbeiten, wobei eine Zusammenfassung der Metainformationen und Dateien auf dem Server angezeigt wird. Dateien können auf den/von dem Server hoch- und heruntergeladen werden oder diese können visualisiert werden.

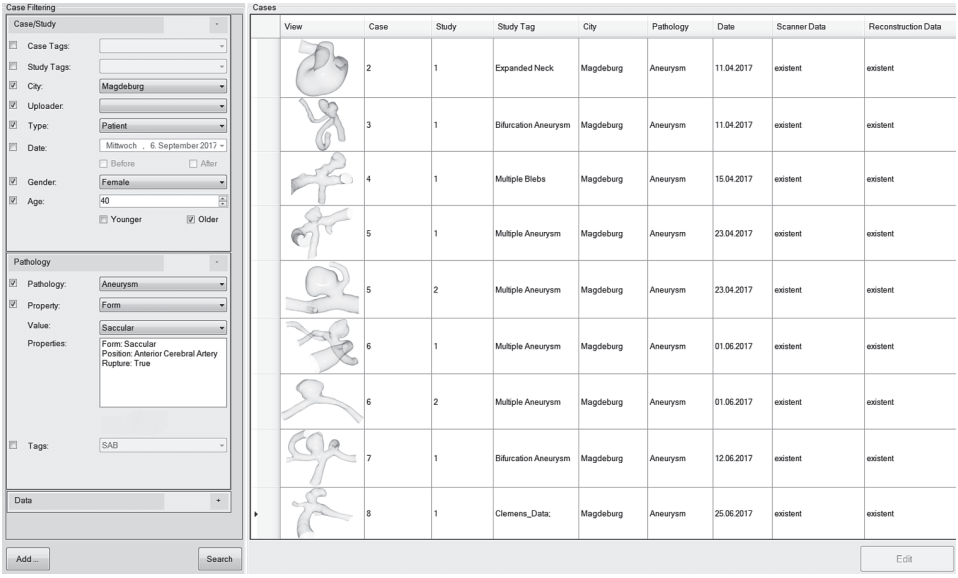


Abb. 2: Eine tabellarische Übersicht aller vorhandenen Datensätze. Masken für das Filtern von spezifischen Kohorten werden bereitgestellt (Aus [Me18b]).

Aneulysis stellt fünf Visualisierungsmodule zur Verfügung. Das Modul *Morphologische Aneurysma-Visualisierung* (MorAVis) dient der Extraktion und Untersuchung morphologischer Deskriptoren, die in der Datenbank gespeichert werden, siehe Abschnitt 3.2. Außerdem unterstützen das *kombinierte Wand- und Strömungsvisualisierungsmodul* (CoWaFloVis), das Modul *Multiple Skalarfeldvisualisierung* (MuScaVis) und das Modul *Vergleichende Tensorvisualisierung* (CoTenVis) die visuelle Analyse von wandbezogenen Daten, siehe Abschnitt 3.3. Mit dem Modul Visuelle Flussexploration (ViFlEx) können Flussmuster innerhalb des Aneurysmas visuell analysiert werden, siehe Abschnitt 3.4.

3.2 Visuelle Exploration der Aneurysmamorphologie

Um Korrelationen zwischen morphologischen Deskriptoren und der Ruptur zu identifizieren, ermöglicht das Modul *MorAVis* eine automatische Extraktion und interaktive Exploration solcher Deskriptoren auf Basis der 3D-Gefäßoberfläche [Me18b]. Ein wichtiger Deskriptor ist das Ostium; die Oberfläche, die das Aneurysma vom Trägergefäß trennt. Daher muss die Gefäßstruktur ohne Aneurysma bestimmt werden. Die meisten Erkennungsalgorithmen erfordern a priori Kenntnisse, z.B. über die Gefäßmittellinie oder die

Benutzerinteraktion. Um solche Abhängigkeiten zu vermeiden, haben wir einen effizienten Algorithmus entwickelt, der automatisch das Ostium berechnet, siehe Abbildung 3(a). Die entstehende Fläche kann bei Bedarf interaktiv korrigiert werden. Kürzlich haben wir die Ostiumdetektion auf Patienten mit mehreren Aneurysmen erweitert, wobei ein klinischer Bericht über die morphologischen Deskriptoren erstellt wird [La19].

Inspiriert durch medizinische Studien berechnen wir mehrere Deskriptoren mit Hilfe des Ostiums, die innerhalb der Gefäßoberfläche angezeigt werden, siehe Abbildung 3(b). Der Anwender kann während der Exploration verschiedene Deskriptoren aktivieren, wobei auch ihre quantitativen Werte aufgeführt werden.

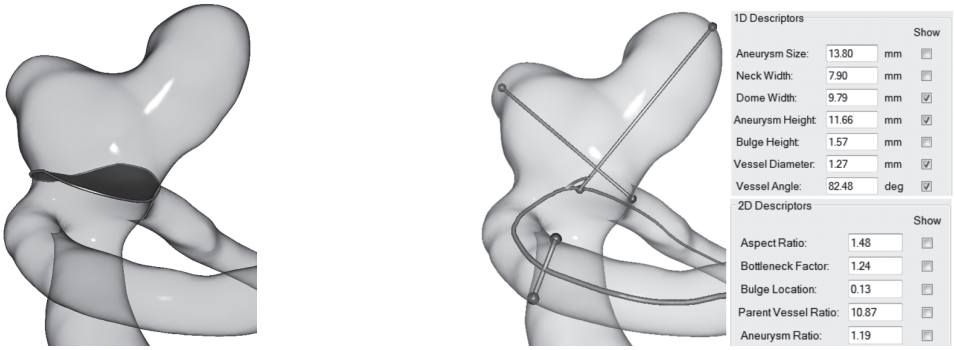


Abb. 3: Automatische Ostium-Extraktion (a) und Berechnung morphologischer Deskriptoren, die zusammen mit ihren quantitativen Werten im Gefäß visualisiert werden (b) (Aus [Me18b]).

3.3 Oberflächen-basierte Exploration von Aneurysmadaten

Je nach Datentyp, d.h. Skalar-, Vektor- oder Tensordaten, ermöglichen drei Module eine interaktive Exploration dieser Informationen, indem sie Einschränkungen bestehender Visualisierungsansätze überwinden. Zudem ist eine leistungsfähige Kamerasteuerung entscheidend für eine effiziente Analyse der zeitabhängigen Daten. Wir haben eine Methode entwickelt, um optimale Ansichten auf Gefäßoberflächen mit einer Kombination aus zwei Attributen zu bestimmen, die potentielle Risikoregionen zeigen [Me17a]. Sie kann innerhalb der Module aktiviert werden, wobei die Ergebnisse in der Datenbank gespeichert werden, siehe Abbildung 1.

3.3.1 Das CoWaFloVis Modul

Die gleichzeitige Untersuchung von zeitabhängigen Wand- und Flusseigenschaften wird derzeit nur eingeschränkt unterstützt. Einige Ansätze illustrieren zwei Attribute mit dem umgebenden Gefäß [Ga11, Ro17] oder zeigen die Wanddicke mit einem hämodynamischen Attribut [La15], aber eine kombinierte Visualisierung war bisher nicht möglich. Außerdem

ist die Wand starr dargestellt, obwohl ihre Verformung für die Risikoabschätzung wesentlich zu sein scheint. Das *CoWaFloVis* Modul ermöglicht eine erweiterte Wandvisualisierung durch die Darstellung der Wanddicke und -deformation [Me17b]. Es besteht aus einer 2,5D-Ansicht, die mit einer 3D-Ansicht der Aneurysmawände verknüpft ist, siehe Abbildung 4. Die 2D-Karte bietet eine übersichtliche Darstellung der Aneurysmageometrie, wobei zwei Attribute durch die Verwendung von Farbe und Schraffur untersucht werden können. Optionale Balkendiagramme unterstützen die Interpretation einer dritten Größe.

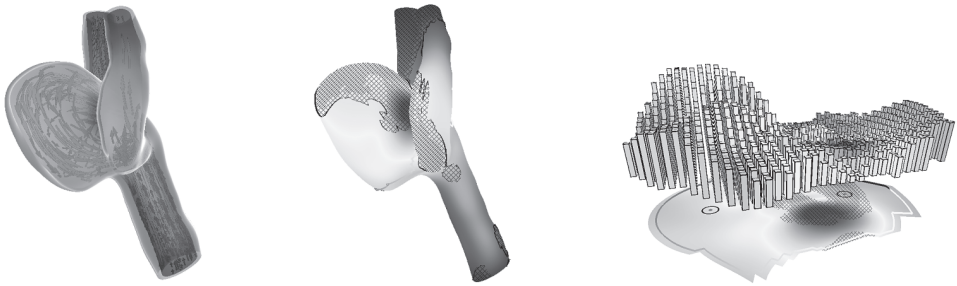


Abb. 4: In 3D werden beide Wände mit dem Blutfluss visualisiert *(a)*. Auf der inneren Wand werden zwei Attribute mit Farbe und Schraffur *(b)* dargestellt. Die 2D-Karte bietet eine verdeckungsfreie Darstellung des Aneurysmas und ein 3D-Balkendiagramm zeigt ein drittes Attribut *(c)* (Aus [Me17b]).

3.3.2 Das MuScaVis Modul

Prinzipiell können Korrelationen zwischen mehr als drei Attributen relevant sein. Das *MuScaVis* Modul kombiniert Techniken aus dem Bereich der Informations- und wissenschaftlichen Visualisierung für eine gleichzeitige Analyse mehrerer Skalarfelder auf der Wand. Brushing and linking helfen dem Anwender, risikoreiche Zusammenhänge zu studieren. Wir verwenden statistische Plots und einen neuartigen 2D-Plot, um einen zeitlichen Überblick über Korrelationen zwischen mehreren Attributen abzubilden [Me16]. Um Korrelationen innerhalb einer bestimmten Oberflächenregion zu analysieren, stellen wir eine Schachbrett-Visualisierung vor [Me17b]. Darüber hinaus zeigen neuartige Glyphen das zeitliche Verhalten von Skalarfeldern auf einen Blick, siehe Abbildung 5.

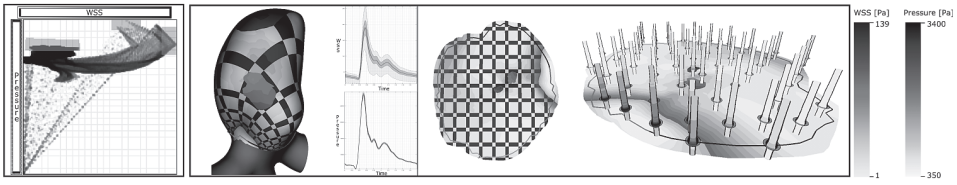
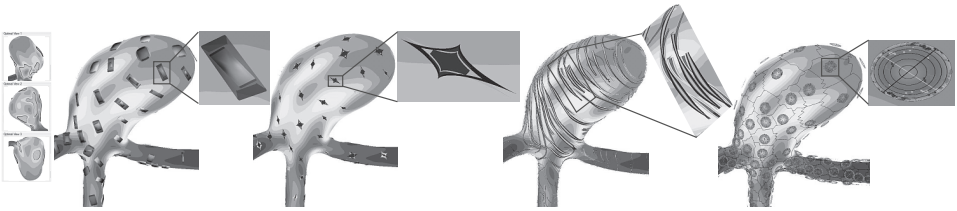


Abb. 5: Ein Aneurysma, das aufgrund der morphologischen Merkmale als gering rupturgefährdet eingeschätzt wurde. Mit Hilfe von *MuScaVis* konnte die Rupturstelle identifiziert werden (Aus [Me18a]).

3.3.3 Das CoTenVis Modul

Tensordaten beschreiben mechanische Spannungen in den Wänden, die durch den Blutfluss induziert werden und die Entwicklung und die Ruptur des Aneurysmas beeinflussen. Das *CoTenVis* Modul bietet die erste integrierte Visualisierung von Spannungstensoren sowie Skalar- und Vektordaten [Me17c]. Wir haben die Anwendbarkeit von vier Glyphendesigns zur Darstellung von Tensoren in einer Nutzerstudie untersucht, siehe Abbildung 6. Sie ermöglichen eine vergleichende Tensorvisualisierung zwischen Innen- und Außenwand, die zur Lokalisierung von rupturgefährdeten Bereichen unerlässlich ist.



(a) (b) (c) (d) (e)

Abb. 6: Vier Glyphendesigns zur Visualisierung von Spannungstensoren (b-e). Optimale Ansichten auf das Aneurysma sind in (a) zu sehen. (Aus [Me17c]).

3.4 Visuelle Exploration von intra-aneurysmalen Flussmustern

Um Korrelationen zwischen Flussmustern und dem Zustand des Aneurysmas aufzudecken, sollten Klassifizierungen von verschiedenen Datensätzen vergleichbar sein. Daher ermöglicht das *ViFlEx* Modul eine objektive und reproduzierbare Klassifikation [Me18c].

Zuerst werden Pfadlinien gruppiert, um Flussmuster zu erhalten, wobei ein von uns entwickelter, automatischer Clustering-Ansatz verwendet wird, der die zeitliche Komponente direkt einbezieht und nicht von vordefinierten Daten abhängig ist [Me18d]. Im Anschluss wird für jedes Cluster ein Repräsentant berechnet. Danach werden die Flussmuster automatisch nach vordefinierten Typen klassifiziert. Dies basiert auf einer neuartigen Parametrisierung der Aneurysmaoberfläche auf eine Hemisphäre durch Berechnung von polar-basierten Koordinaten. Dann werden die polar-basierten Koordinaten den Punkten der Repräsentanten zugeordnet. Jeder Repräsentant wird anhand von sechs vordefinierten Typen, basierend auf der Studie von Cebral et al. [Ce05], klassifiziert, siehe Abbildung 7. Außerdem stellen wir einen Detail-on-Demand-Ansatz vor, der einen visuellen Übergang vom Cluster-Repräsentanten über eine umschließende Fläche zu den zugehörigen Linien schafft, um die räumliche Wahrnehmung von Strömungsmustern zu unterstützen.

4 Schlussfolgerungen

Für die Erforschung zerebraler Aneurysmen wird das Wissen zweier hochspezifischer Fachgebiete, der Strömungsmechanik und der Neuroradiologie, benötigt. Um eine effiziente, kollaborative Datenanalyse zu ermöglichen, ist eine konsistente Verwaltung der

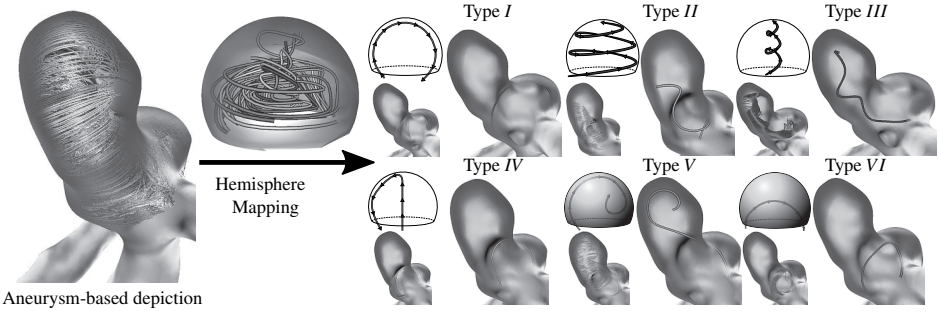


Abb. 7: Automatische Klassifikation von Flussmustern. Jedem Muster wird basierend auf dem Repräsentanten einer von sechs (*I-VI*) vordefinierten Typen zugeordnet. (Aus [Me18c].)

Aneurysmainformationen erforderlich, bei dem Experten beider Fachgebiete Zugriff auf die Daten haben. Um dies zu gewährleisten, haben wir *Aneulysis* entwickelt– ein System zur Verwaltung und visuellen Untersuchung von Aneurysmen, das in enger Zusammenarbeit mit medizinischen und CFD-Experten entwickelt wurde. Wir haben zahlreiche neue Methoden zur Erforschung der dynamischen Interaktion zwischen Gefäßwand und Strömung entwickelt, um ein vertieftes Verständnis für das Wachstum und die Ruptur von Aneurysmen zu bekommen. *Aneulysis* konzentriert sich auf klinisch relevante Aspekte, wie z.B. das Finden einer optimalen Behandlung und unterstützt die Kommunikation zwischen Ingenieuren und Ärzten, z.B. bei der Untersuchung, wie Veränderungen der Strömungsmuster nach der Behandlung mit dem Behandlungserfolg zusammenhängen.

Literaturverzeichnis

[Ce05] Cebral, Juan R.; Castro, Marcelo A.; Burgess, James E.; Pergolizzi, Richard S.; Sheridan, Michael J.; Putman, Christopher M.: Characterization of cerebral aneurysms for assessing risk of rupture by using patient-specific computational hemodynamics models. Am J Neuroradiol, 26(10):2550–59, 2005.

[Ga11] Gasteiger, Rocco; Neugebauer, Mathias; Beuing, Oliver; Preim, Bernhard: The FLOWLENS: A Focus-and-Context Visualization Approach for Exploration of Blood Flow in Cerebral Aneurysms. IEEE Trans Vis Comput Graph, 17(12):2183–92, 2011.

[Ja15] Janiga, Gabor; Berg, Philipp; Sugiyama, Shinichiro; Kono, Kenichi; Steinman, David: The computational fluid dynamics rupture challenge 2013—phase I: Prediction of rupture status in intracranial aneurysms. Am J Neuroradiol, 2015.

[La15] Lawonn, Kai; Glaßer, Sylvia; Vilanova, Anna; Preim, Bernhard; Isenberg, Tobias: Occlusion-free Blood Flow Animation with Wall Thickness Visualization. IEEE Trans Vis Comput Graph, 22(1):728–37, 2015.

[La19] Lawonn, Kai; Meuschke, Monique; Preim, Bernhard; Hildebrnadt, Klaus: A Geometric Optimization Approach for the Detection and Segmentation of Multiple Aneurysms. Comput Graph Forum, 38(3):413–25, 2019.

[Me16] Meuschke, Monique; Köhler, Benjamin; Preim, Uta; Preim, Bernhard; Lawonn, Kai: Semi-automatic Vortex Flow Classification in 4D PC-MRI Data of the Aorta. Comput Graph Forum, 35(3):351–60, 2016.

[Me17a] Meuschke, Monique; Engelke, Wito; Beuing, Oliver; Preim, Bernhard; Lawonn, Kai: Automatic Viewpoint Selection for Exploration of Time-dependent Cerebral Aneurysm Data. In: Proc. of BVM. S. 352–57, 2017.

[Me17b] Meuschke, Monique; Voß, Samuel; Beuing, Oliver; Preim, Bernhard; Lawonn, Kai: Combined Visualization of Vessel Deformation and Hemodynamics in Cerebral Aneurysms. IEEE Trans Vis Comput Graph, 23(1):761–70, 2017.

[Me17c] Meuschke, Monique; Voß, Samuel; Beuing, Oliver; Preim, Bernhard; Lawonn, Kai: Glyph-based Comparative Stress Tensor Visualization in Cerebral Aneurysms. Comput Graph Forum, 36(3):99–108, 2017.

[Me18a] Meuschke, Monique; Günther, Tobias; Berg, Philipp; Wickenhöfer, Ralph; Preim, Bernhard; Lawonn, Kai: Visual Analysis of Aneurysm Data using Statistical Graphics. IEEE Trans Vis Comput Graph, 25(1):997–1007, 2018.

[Me18b] Meuschke, Monique; Günther, Tobias; Wickenhöfer, Ralph; Gross, Markus; Preim, Bernhard; Lawonn, Kai: Management of Cerebral Aneurysm Descriptors based on an Automatic Ostium Extraction. IEEE Comput Graph Appl, 38(3):58–72, 2018.

[Me18c] Meuschke, Monique; Oeltze-Jafra, Steffen; Beuing, Oliver; Preim, Bernhard; Lawonn, Kai: Classification of Blood Flow Patterns in Cerebral Aneurysms. IEEE Trans Vis Comput Graph, 2018.

[Me18d] Meuschke, Monique; Voß, Samuel; Preim, Bernhard; Lawonn, Kai: Exploration of Blood Flow Patterns in Cerebral Aneurysms during the Cardiac Cycle. Computers & Graphics, 72:12–25, 2018.

[Me19] Meuschke, Monique: Visualization, Classification, and Interaction for Risk Analysis and Treatment Planning of Cerebral Aneurysms. Dissertation, Otto-von-Guericke University Magdeburg, 2019.

[Ro17] Rocha, Allan; Alim, Usman; Silva, Julio; Sousa, Mario: Decal-maps: Real-Time Layering of Decals on Surfaces for Multivariate Visualization. IEEE Trans Vis Comput Graph, 23(1):821–30, 2017.

Monique Meuschke, geboren am 3. Juni 1988 in Wippra, Deutschland. Sie studierte von 2009 bis 2015 Computervisualistik mit dem Anwendungsfach Medizin an der Otto-von-Guericke-Universität (OvGU) Magdeburg und promovierte 2019 mit Auszeichnung. Sie erhielt eine Auszeichnung als beste Masterabsolventin der OvGU Magdeburg und bekam den Dissertationspreis für die beste Promotion der OvGU verliehen. Während der Promotionszeit von 3,5 Jahren schrieb sie 11 Publikationen als Erstautor von denen 8 in hochkarätigen Journals im Bereich der Visualisierung erschienen. Nach Abschluss ihres Doktorats begann Monique Meuschke als Postdoc an der Universität Wuppertal und Jena zu forschen. Ihre Forschungsinteressen umfassen verschiedene Themen der medizinischen Visualisierung mit dem Schwerpunkt der visuellen Exploration von Gefäßerkrankungen. Ihre Forschung trägt dazu bei, dass die Diagnose und Therapie schwerwiegender Gefäßerkrankungen signifikant verbessert wird.

Lernen mit Graphen: Kern- und neuronale Methoden[1]

Christopher Morris[2]

Abstract: Die vorliegende Arbeit befasst sich mit (überwachter) Graphklassifikation, d. h. mithilfe einer Menge von bereits klassifizierten Beispielgraphen wird ein Modell so trainiert, dass es die Klassen von bisher ungesehenen Graphen vorhersagen kann. Im ersten Teil dieser Arbeit stellen wir Kernmethoden für Graphen vor. Insbesondere stellen wir skalierbare Kerne vor, die mit kontinuierlichen Knoten- und Kantenbeschriftungen umgehen können. Ferner stellen wir einen Graphkern vor, der globale Grapheigenschaften berücksichtigen kann, die von anderen Graphkernen nicht erfasst werden. Zu diesem Zweck schlagen wir eine lokale Version des k-dimensionalen Weisfeiler-Leman-Algorithmus vor, der eine bekannte Heuristik für das Graph-Isomorphie-Problem ist. Wir zeigen, dass unser lokaler Algorithmus mindestens die gleiche Mächtigkeit wie der ursprüngliche Algorithmus hat, wobei wir gleichzeitig die Spärlichkeit des zugrundeliegenden Graphen berücksichtigen und Overfitting verhindern. Anschließend stellen wir ein theoretisches Framework für die Analyse von Graphkernen vor, und zeigen, dass die meisten Kerne nicht in der Lage sind einfache graphentheoretische Eigenschaften zu unterscheiden. Der zweite Teil beschäftigt sich mit neuronalen Ansätzen zur Graphklassifikation und deren Verbindung zu Kern-Methoden. Wir zeigen, dass die Expressivität sogenannter Graph-Neural-Networks durch den 1-dimensionalen Weisfeiler-Leman-Algorithmus nach oben beschränkt werden kann.

1 Einleitung und Übersicht

Relationale Daten kommen heutzutage in verschiedenen Bereichen der Chemie- und Bioinformatik, in der Analyse sozialer Netzwerke und der Mustererkennung vor. Graphen können solche Daten auf natürliche Weise modellieren. Diese Arbeit befasst sich mit der Entwicklung von Methoden zur (überwachten) Graphklassifikation.

Im ersten Teil konzentrieren wir uns auf Kern-Methoden für Graph, sogenannte *Graphkerne*. Intuitiv gesprochen, messen Graphkerne die Ähnlichkeit eines Paars von Graphen. Genauer gesagt ist ein Graphkern eine positiv-semidefinite Funktion, die ein Paar von Graphen auf eine reelle Zahl abbildet. Daher können Graphkerne zusammen mit etablierten Lernalgorithmen, wie z. B. Support-Vektor-Maschinen, verwendet werden. Alternativ kann jeder Graph auf einen (endlichdimensionalen) Merkmalsvektor abgebildet werden. Der Graphkern ergibt sich dann durch das paarweise innere Produkt zwischen solchen Vektoren. Wir sprechen dann von einem *expliziten* Graphkern. In Anwendungen, z. B. in der Chemieinformatik, sind die Knoten und Kanten eines Graphen üblicherweise mit kontinuierlichen Merkmalen annotiert. Die in der Vergangenheit entwickelten Graphkerne, die mit solchen Eingaben umgehen konnten, skalieren nicht, da sie die Kernfunktion für jedes Paar von Graphen berechnen, was zu einem quadratischen Mehraufwand in der Laufzeit führt. Wir umgehen dieses Problem, indem wir jeden Graphen auf einen endlichdimensionalen, reellen Vektor abbilden. Wir zeigen, dass das innere Produkt zwischen solchen Vektoren bekannte Graphkerne

[1] Orginaltitel: „Learning with Graphs: Kernel and Neural Approaches“

[2] Technische Universität Dortmund, Fakultät für Informatik, christopher.morris@tu-dortmund.de

CC BY SA

approximiert, die mit kontinuierlichen Eingaben umgehen können. Wir evaluieren die von uns vorgeschlagene Methode auf zahlreichen Benchmark-Datensätzen, und zeigen, dass unsere Methode vielfach schneller ist und trotzdem vergleichbar hohe Klassifikationsgüten liefert. Der nächste Abschnitt befasst sich mit Erweiterungen von Kernfunktionen, die auf dem 1-dimensionalen Weisfeiler-Leman-Algorithmus (1-WL) basieren. Dazu leiten wir einen Kern her, der auf dem k-dimensionalen Weisfeiler-Leman-Algorithmus (k-WL) basiert, der in Hinblick auf die Unterscheidung nicht-isomorpher Graphen mächtiger ist als der 1-WL. Um die Anwendbarkeit für das maschinelle Lernen zu gewährleisten, leiten wir eine lokale Variante her, die Overfitting verhindert und die Spärlichkeit des zugrundeliegenden Graphen berücksichtigt. Darüber hinaus zeigen wir, dass die lokale Variante mindestens die gleiche Mächtigkeit wie der ursprüngliche Algorithmus hat. Um den Algorithmus auf große Datenmengen anzuwenden, zeigen wir, wie man ihn approximieren kann. Schließlich evaluieren wir die von uns vorgeschlagenen Kerne empirisch, und zeigen, dass sie auf einer Vielzahl von Benchmark-Datensätzen anderen Methoden bezüglich Klassifikationsgüte überlegen sind. Abschließend entwickeln wir ein theoretisches Framework für die Analyse der Expressivität von Graphkernen, welches sich Konzepten aus dem Graph-Property-Testing bedient. Hier unterscheidet ein Graphkern eine Grapheigenschaft, wenn er einen konstanten Winkel (unabhängig von der Größe des Graphen) zwischen den Merkmalsvektoren zweier beliebiger Graphen garantiert, von denen einer die Eigenschaft hat und der andere weit davon entfernt ist. Wir untersuchen bekannte Graphkerne und zeigen, dass sie nicht in der Lage sind grundlegenden Eigenschaften zu unterscheiden.

Der zweite Teil der Arbeit befasst sich mit neuronalen Methoden zur Graphklassifikation. Insbesondere betrachten wir Graph-Neural-Networks (GNNs), iund zeigen, dass diese nicht mächtiger sein können als der 1-WL, in Bezug auf die Unterscheidung von nicht-isomorphen (Sub-)Graphen. Auf der anderen Seite zeigen wir, dass GNNs bei richtiger Parameterinitialisierung die gleiche Mächtigkeit wie der 1-WL haben. Darauf aufbauend schlagen wir, basierend auf dem k-WL, eine Verallgemeinerung von GNNs vor, sogenannte k-GNNs.

2 Kern-Methoden

Im Folgenden beschreiben wir die Arbeiten im Bereich der Graphkerne, die auf den Veröffentlichungen [Mo16], [MKM17], [MM19] und [Kr18] basieren.

2.1 Skalierbare Graphkerne für Graphen mit kontinuierlichen Merkmalen

Die Grundidee unseres sogenannten Hash-Graph-Kernel-Frameworks ist, dass man kontinuierliche Merkmale mittels einer Familie von Hashfunktionen auf diskrete Label abbildet. Darauffolgend wird ein skalierender Graphkern für Graphen mit diskreten Labeln auf dem so transformierten Graphen angewandt.

Sei $\mathscr{H} = \{h \colon \mathbb{R}^d \to \mathbb{N}\}$ eine Familie von Hashfunktionen und G ein Graph mit kontinuierlichen Knotenmerkmalen $a \colon V(G) \to \mathbb{R}^d$. Wir transformieren den Graphen (G, a) zu einem Graphen

mit diskreten Labeln, indem wir jedes Merkmal $a(v)$ mit einer Hashfunktion h aus $\mathscr{H}$ nach $h(a(v))$ abbilden. Wir schreiben $h(G)$ für den durch diese Prozedur erhaltenen Graphen. Die Funktion h wird hier zufällig und gleichverteilt aus der Familie von Hashfunktionen $\mathscr{H}$ gezogen. Um die Varianz zu kontrollieren, wird die obige Prozedur mehrmals wiederholt. Wir erhalten also eine Sequenz von diskret-gelabelten Graphen $(h_i(G))_{i=1}^I$, hier ist I die Anzahl der Iterationen. Diese Sequenzen von gelabelten Graphen wird nun mit einem beliebigen Graphkern für Graphen mit diskreten Labeln, dem sogenannten diskreten Basiskern, verglichen. Formal können wir den Hash-Graphkern folgendermaßen definieren.

Definition 2.1. Sei $\mathscr{H}$ eine Familie von Hashfunktionen und sei k_b ein diskreter Basiskern. Dann ist der Hash-Graphkern für zwei Graphen G und H mit kontinuierlichen Knotenmerkmalen definiert als

$$k_\mathrm{HGK}(G,H) = \frac{1}{I}\sum_{i=1}^{I} k_\mathrm{b}(h_i(G),h_i(H)),$$

hier ist h_i zufällig und gleichverteilt aus $\mathscr{H}$ gezogen.

Um die Skalierbarkeit des Hash-Graphkerns zu gewährleisten, benötigen wir explizite Merkmalsvektoren für den Kernel k_HGK. Das ist genau dann möglich, wenn wir diese für den diskreten Basiskern effizient berechnen können. Die Merkmalsvektoren können dann über alle Iterationen konkateniert werden und mit dem Faktor $\sqrt{1/I}$ skaliert werden.

Um zu zeigen, dass das obige Verfahren (langsame) Graphkerne für Graphen mit kontinuierlichen Merkmalen approximieren kann, müssen wir einige Anforderungen an die verwendete Familie von Hashfunktionen stellen. Die grundlegende Idee hierbei ist, dass man die Familie so auswählt, dass für zwei zufällige Hashfunktionen h_1 und h_2 die Kollisionswahrscheinlichkeit

$$\Pr[h_1(x) = h_2(y)]$$

einem Kern k gleicht, der die Ähnlichkeit von x und y aus $\mathbb{R}^d$ sinnvoll angibt. Für den Fall $h_1 = h_2$, wurden solche Familie bereits in [RR08] untersucht. Allerdings können diese hier nicht angewendet werden, da viele Graphkerne mittels Multiplikation aus anderen Kernfunktionen zusammengesetzt werden. Daher tritt ein Problem von Abhängigkeit auf. Daher führen wir das Konzept der *unabhängigen k-Hash-Familien* ein.

Definition 2.2. Sei $k\colon \mathscr{X} \times \mathscr{X} \to \mathbb{R}$ eine Kernfunktion und sei $\mathscr{H} = \{h\colon \mathscr{X} \to S\}$ eine Familie von Hashfunktionen, die auf die Menge S abbilden. Dann ist $\mathscr{H}$ eine *unabhängige k-Hash-Familie*, wenn

$$\Pr[h_1(x) = h_2(y)] = k(x,y).$$

Hier sind h_1 und h_2 unabhängig und zufällig aus $\mathscr{H}$ gezogen.

Mithilfe der obigen Hash-Familien konnten wir zeigen, dass das Hash-Graph-Kernel-Framework langsame Graphkerne für Graphen mit kontinuierlichen Merkmalen mit hoher Wahrscheinlichkeit bis auf einen beliebig kleinen, additiven Fehler approximieren kann.

Das obige Verfahren wurde auf zahlreichen, praxisnahen Benchmark-Datensätzen evaluiert. Hier konnte gezeigt werden, dass Instanzen des Hash-Graph-Kernel-Frameworks deutlich schneller sind, dabei aber trotzdem vergleichbar hohe Klassifikationsgüten erzielen.

2.2 Expressive Graphkerne basierend auf dem k-dimensionalen Weisfeiler-Leman-Algorithmus

Die meisten aktuellen Graphkerne berücksichtigen nur *lokale* Grapheigenschaften, d. h. sie werden auf der Basis von Eigenschaften der Nachbarschaft von Knoten oder anderer kleiner Substrukturen berechnet, z. B. der Weisfeiler-Lehman-Subtree-Kern [Sh09], der auf dem 1-WL basiert. Obwohl der 1-WL bereits recht leistungsfähig ist und in anderen Bereichen erfolgreich angewendet wurde, ist er beispielsweise nicht in der Lage unterschiedlich lange Zyklen zu unterscheiden, was ein wichtiges Merkmal für die Analyse sozialer Netzwerke und in der Chemieinformatik ist. Eine mächtigere Erweiterung des 1-WL ist der k-WL.

Formal können wir diesen folgendermaßen beschreiben. Sei G ein Graph und sei $k \geq 2$. In jeder Iteration $i \geq 0$ berechnet der Algorithmus eine *Färbung* $C_i^k \colon V(G)^k \to \mathbb{S}$, wobei $\mathbb{S}$ eine beliebige Kodomäne ist. In der ersten Iteration ($i = 0$) erhalten zwei Tupel $\mathbf{v}$ und $\mathbf{w}$ in $V(G)^k$ die gleiche Farbe, falls die Abbildung $v_i \mapsto w_i$ einen Isomorphismus zwischen den induzierten Graphen $G[\mathbf{v}]$ and $G[\mathbf{w}]$ beschreibt. Für $i \geq 0$ ist C_{i+1}^k folgendermaßen definiert

$$C_{i+1}^k(\mathbf{v}) = (C_i^k(\mathbf{v}), M_i(\mathbf{v})),$$

hierbei ist

$$M_i(\mathbf{v}) = \big(\{\!\{C_i^k(\phi_1(\mathbf{v},w)) \mid w \in V(G)\}\!\}, \ldots, \{\!\{C_i^k(\phi_k(\mathbf{v},w)) \mid w \in V(G)\}\!\}\big). \tag{1}$$

Hier bezeichnet $\{\!\{\ldots\}\!\}$ eine Multimenge. Ferner ist

$$\phi_j(\mathbf{v},w) = (v_1, \ldots, v_{j-1}, w, v_{j+1}, \ldots, v_k).$$

Hier tauscht $\phi_j(\mathbf{v},w)$ also die j-te Komponente des Tupels $\mathbf{v}$ durch den Knoten w aus. Der k-WL berücksichtigt zwar mehr globale Eigenschaften, aber keine lokalen Eigenschaften. Ferner zeigen Experimente, dass er nicht skaliert und zu Overfitting führt, daher schlagen wir eine *lokale Variante* vor. Diese färbt ebenfalls alle k-Tupel, um nun aber lokale Merkmale zu extrahieren, definieren wir die lokale Nachbarschaft eines k-Tupel. Anstatt Gleichung 1 verwenden wir

$$M_i^\delta(\mathbf{v}) = \{\!\{s_\delta(\mathbf{v},w) \mid w \in V(G)\}\!\},$$

wobei

$$s_\delta(\mathbf{v},w) = \{(C_i^{k,\delta}(\phi_j(\mathbf{v},w)), j) \mid j \in [k] \text{ und } w \in \delta(v_j)\},$$

und $\delta(v)$ die Menge der Nachbarn des Konten v bezeichnet. Daher ist die Färbungsfunktion definiert als

$$C_{i+1}^{k,\delta}(\mathbf{v}) = (C_i^{k,\delta}(\mathbf{v}), M_i^\delta(\mathbf{v})).$$

Wir zeigen, dass eine Variante des lokalen Algorithmus mindestens die gleiche Mächtigkeit, in Bezug auf die Unterscheidung von nicht-isomorphen Graphen, wie der k-WL hat. Für spärliche Graphen kann die Laufzeit für jede Iteration des lokalen Algorithmus nach oben durch $|V(G)^k| \cdot kd$ begrenzt werden, wobei d den maximalen oder durchschnittlichen Grad des Graphen bezeichnet. Daher berücksichtigt der lokale Algorithmus die Spärlichkeit des zugrundeliegenden Graphen, was zu (iterationsweise) verbesserten Berechnungszeiten im

Vergleich zu dem nicht-lokalen k-WL führt. Ferner können wir experimentell zeigen, dass er Overfitting verhindert.

Wir berechnen nun einen Kern basieren auf dem lokalen Algorithmus, indem wir ihn für für $h \geq 0$ Iterationen ausführen. Nach jeder Iteration i berechnen wir einen Merkmalsvektor $\phi_{\mathrm{L}}^{i}(G)$ aus $\mathbb{R}^{|\mathbb{S}_i|}$ für jeden Graphen G, wobei $\mathbb{S}_i \subseteq \mathbb{S}$ das Bild von $C_i^{k,\delta}$ bezeichnet. Jede Komponente $\phi_{\mathrm{L}}^{i}(G)_s$ zählt die Anzahl der Vorkommen von k-Tupeln, die mit s in $\mathbb{S}_i$ gefärbt sind. Der finale Merkmalsvektor $\phi_{\mathrm{L}}(G)$ ist dann definiert als die Konkatenation der Merkmalsvektoren aller h Iterationen, d. h

$$\left[\phi_{\mathrm{L}}^{0}(G), \ldots, \phi_{\mathrm{L}}^{h}(G)\right].$$

Seien G und H zwei Graphen dann ist der entsprechende Kernel $k_{\mathrm{L}}(G,H) = \langle \phi_{\mathrm{L}}(G), \phi_{\mathrm{L}}(H) \rangle$. Da der Algorithmus alle möglichen k-Tupel betrachtet skaliert er nicht für sehr große Graphen. Daher haben wir einen Approximationsalgorithmus entwickelt, der den exakten Merkmalsvektor bis auf einen additive Fehler bezüglich der ℓ_1-Norm annähern kann. Für Graphen mit beschränkten Grad konnten wir zeigen, dass die Laufzeit konstant in der Graphgröße ist.

In einer experimentellen Studie konnten wir zeigen, dass der obige Kernel auf einer Vielzahl von Benchmark-Datensätzen aktuelle Graphkerne schlägt, und deutlich schneller als der ursprüngliche k-WL ist.

2.3 Ein theoretisches Framework zur Untersuchung der Expressivität von Graphkernen

In den vergangenen zwei Jahrzehnten wurde eine große Anzahl von Graphkernen vorgeschlagen, siehe z. B. [KJM20]. In Anbetracht der großen Anzahl verfügbarer Graphkerne und der Fülle von Benchmark-Datensätzen wird es zunehmend schwierig einen fairen experimentellen Vergleich der Kerne durchzuführen und ihre Vor- und Nachteile für bestimmte Datensätze und Anwendungsszenarien zu spezifizieren. Tatsächlich können die derzeitigen experimentellen Vergleiche kein vollständiges Bild vermitteln und sind von begrenzter Nützlichkeit für einen Praktiker, der einen Kernel für eine bestimmte Anwendung auswählen muss. Daher führen wir ein theoretisches Framework für die Analyse der Expressivität von Graphkernen ein, welches durch Konzepte aus dem Graph-Property-Testing motiviert ist.

Im Folgenden betrachten wir Graphen mit beschränkten Grad, d. h. der Knotengrad ist unabhängig von der Anzahl der Knoten. Seien G und H zwei Graphen mit n Knoten mit maximalem Knotengrad d. Dann ist die *Editierdistanz* $\Delta(G,H)$ zwischen G und H definiert als die minimale Anzahl von Kantenmodifikation, d. h. das Hinzufügen oder das Entfernen von Kanten, die benötigt werden, um G in eine isomorphe Kopie von H abzuändern. Eine *Grapheigenschaft* ist nun eine Menge $\mathscr{P}$ von Graphen, die unter Isomorphie abgeschlossen ist. Die Menge von Graphen aus $\mathscr{P}$ mit n Knoten bezeichnen wir mit $\mathscr{P}_n$. Ein Graph G auf n Knoten mit maximalem Knotengrad d ist ε-*weit* von $\mathscr{P}_n$ entfernt, falls für alle Graphen H mit maximalem Knotengrad d aus $\mathscr{P}_n$ gilt, dass

$$\Delta(G,H) > \varepsilon dn$$

für $\varepsilon > 0$. Ferner sei $\hat{k}$ der normalisierte Kern eines Kerns k and sei $\hat{\phi}$ der normalisierte Merkmalsvektor, d. h.

$$\hat{k}(x,y) = \langle \hat{\phi}(x), \hat{\phi}(y) \rangle = \left\langle \frac{\phi(x)}{\|\phi(x)\|_2}, \frac{\phi(y)}{\|\phi(y)\|_2} \right\rangle$$
$$= \frac{k(x,y)}{\sqrt{k(x,x) \cdot k(y,y)}} \in [-1,1].$$

Der normalisierte Kern $\hat{k}(x,y)$ entspricht dem Kosinus des Winkels zwischen $\phi(x)$ und $\phi(y)$ im Merkmalsraum. Ein Graphkern *identifiziert* eine Eigenschaft, falls keine zwei Graphen auf den gleichen Merkmalsvektor abgebildet werden, außer beide haben die Eigenschaft oder beide haben sie nicht.

Definition 2.3. Sei $\mathscr{P}$ eine Grapheigenschaft. Falls ein Graphkern $k \colon \mathscr{G} \times \mathscr{G} \to \mathbb{R}$ für jedes n aus $\mathbb{N}$, und jeden G aus $\mathscr{P}_n$ und H nicht in $\mathscr{P}_n$

$$\frac{k(G,H)}{\sqrt{k(G,G) \cdot k(H,H)}} < 1$$

erfüllt, *identifiziert* k die Eigenschaft $\mathscr{P}$.

Damit ein Graphkern eine Grapheigenschaft unterscheiden kann, ist es wünschenswert, dass ein konstanter Winkel, unabhängig von n, zwischen den Merkmalsvektoren existiert. Dies kann aber oft nicht erreicht werden, wie das folgende Resultat für den Shortest-Path-Kern [BK05] zeigt.

Proposition 2.4. Für den Shortest-Path-Kern k und jede Konstante c, $0 < c < 1$, existieren n aus $\mathbb{N}$ und zwei Graphen G und H aus $\mathscr{G}_n$, wobei G zusammenhängend, und H nicht zusammenhängend ist, so dass

$$\frac{k(G,H)}{\sqrt{k(G,G) \cdot k(H,H)}} > 1 - c.$$

Um einen Winkel unabhänging von der Graphgröße zu erhalten, führen wir den Begriff von *Unterscheidbarkeit* ein.

Definition 2.5. Eine Grapheigenschaft $\mathscr{P}$ ist durch den Graphkern $k \colon \mathscr{G} \times \mathscr{G} \to \mathbb{R}$ *unterscheidbar*, falls für jedes $\varepsilon > 0$, und d aus $\mathbb{N}$, ein $\delta = \delta(\varepsilon, d) > 0$ existiert, so dass für jedes n aus $\mathbb{N}$, jeden G in $\mathscr{P}_n$, und jeden H, der ε-weit von $\mathscr{P}_n$ entfernt ist, gilt, dass

$$\frac{k(G,H)}{\sqrt{k(G,G) \cdot k(H,H)}} \leq 1 - \delta.$$

Im Folgenden untersuchen wir Identifizierbarkeit und Unterscheidbarkeit des Random-Walk- [GFW03], des Weisfeiler-Lehman-Subtree- [Sh11], des Shortest-Path- [BK05], und des Graphlet-Kerns [Sh09]. Tabelle 1 fasst die Resultate im Vergleich zum k-Disc-Kern, siehe unten, zusammen.

Eigenschaft	Graphkern				
	WL	RW	SP	GRAPHLET	k-DISC
Zusammenhang	✗	✗	✓	✗	✓
Planarität	✗	✗	✗	✗	✓
Bipartitheit	✗	✗	✗	✗	
Dreiecksfreiheit	✗	✗	✗	•	✓

Tabelle 1: Unterscheidbarkeit von Grapheigenschaften für den Weisfeiler-Lehman-Subtree- (WL), Random-Walk- (RW), Graphlet-, Shortest-Path- (SP), und k-Disc-Kern, Key: ✓Unterscheidbar, •Identifizierbar (aber nicht unterscheidbar), and ✗Nicht identifizierbar.

Die Merkmalsvektoren des Random-Walk- und des Weisfeiler-Lehman-Subtree-Kerns können reguläre Graphen nicht unterscheiden. Wenn also für eine Grapheigenschaft $\mathscr{P}$ and n aus $\mathbb{N}$ ein regulärer Graph mit der Eigenschaft existiert und ein anderer regulärer Graph, der ε-weit von der Eigenschaft entfernt ist, existiert, dann können beide Kerne die Eigenschaft nicht identifizieren. Daher gilt das folgende Resultat.

Theorem 2.6. Der Random-Walk- und der Weisfeiler-Lehman-Subtree-Kern können Zusammenhang, Planarität, Bipartitheit und Dreiecksfreiheit nicht identifizieren.

Für den Shortest-Path-Kern konnten wir zeigen, dass er nur Zusammenhang identifizieren kann. Ferner konnten wir für den Graphlet-Kernel zeigen, dass er nur Dreiecksfreiheit identifizieren kann.

Im Folgenden leiten wir einen Graphkern her, der obige Eigenschaften unterscheiden kann und effizient berechenbar ist. Wir definieren hierzu das Histogramm $\text{hist}_G(k)$, das die Anzahl an unterschiedlichen (d. h. nicht-isomorphen) k-Discs um einen Knoten in G zählt. Hierbei ist eine k-Disk um einen Konten v aus $V(G)$ der induzierte Subgraph aus allen Knoten, die von v auf einem Pfad der Länge kleiner als k erreichbar sind. Ferner benötigen wir das normalisierte Histogramm $\text{freq}_G(k) = \text{hist}_G(k)/n$ für G aus $\mathscr{G}_n$. Wir betrachten den folgenden Graphkern. Seien G and H Graphen, dann ist der *k-Disc-Kern* definiert als

$$k_{\text{KD}}(G,H) = \langle \text{freq}_G(k), \text{freq}_H(k) \rangle.$$

Ein entscheidener Unterscheid zum Graphlet-Kernel ist hier, dass der k-Disc-Kern nur zusammenhängende Subgraphen betrachtet, wohingegen der Graphlet-Kernel alle möglichen Subgraphen betrachet. Für Graphen mit beschränkten Grad kann der k-Disc-Kernel in linearer Zeit berechnet. Das folgende Resultat stellt nun das Hauptresultat dar.

Theorem 2.7. Für den k-Disc-Kernel gilt, dass er
(1) Zusammenhang für $k \geq 4/\varepsilon d$ unterscheiden kann,
(2) Dreieckfreiheit fuer $k \geq 1$ unterscheiden kann, und
(3) für jedes $\varepsilon > 0$, d aus $\mathbb{N}$, existiert ein k aus $\mathbb{N}_{>0}$, sodass Planarität unterschieden werden kann.

Ferner haben wir Lerngarantieren für den Kern-Nächste-Nachbar-Klassifizier hergeleitet. Wir zeigen, dass der Vorhersagefehler für die Klassifikation einer Grapheigenschaft unter der Annahme begrenzt werden kann, dass der verwendete Kernel die Eigenschaft unterscheiden kann. Die Anzahl der benötigen Trainingsbeispiele hängt hier nur von der Dimension des Merkmalsraums ab.

3 Neuronale Methoden

In diesem Abschnitt stellen wir, basierend auf [Mo19], neuronale Methoden für die Graphklassifikation vor. Hier haben sich GNNs in den letzten Jahren als Standardarchitektur herausgebildet. Im Folgenden zeigen wir die Grenzen dieser Architektur auf. Insbesondere zeigen wir, dass jede mögliche GNN-Architektur hinsichtlich der Unterscheidung nicht-isomorpher Graphen nicht mächtiger als der 1-WL sein kann. Des Weiteren schlagen wir, basierend auf dem k-WL, eine beweisbar mächtigere Verallgemeinerung von GNNs vor.

GNNs können als eine neuronale Variante des 1-WL angesehen werden, bei dem Farben durch kontinuierliche Merkmalsvektoren ersetzt werden und neuronale Netze zur Aggregation der Nachbarschaft eines Knotens verwendet werden. Tatsächlich kann das GNN-Framework als Implementierung einer kontinuierlichen Form von graphbasiertem „Message Passing“ betrachtet werden, bei dem lokale Nachbarschafts-Informationen aggregiert und an die Nachbarn weitergegeben werden. Durch den Einsatz eines trainierbaren neuronalen Netzes zur Aggregation von Informationen aus den Knoten-Nachbarschaften können GNNs zusammen mit den Parametern des Klassifikations- oder Regressionsalgorithmus Ende-zu-Ende trainiert werden.

Im Folgenden beschreiben wir GNNs formal. Sei (G,l) ein gelabelter Graph mit initialer Knotenfärbung $f^{(0)}\colon V(G) \to \mathbb{R}^{1\times d}$, die *konsistent* mit l ist. Das heißt, dass jeder Knoten mit einem Merkmalsvektor $f^{(0)}(v)$ aus $\mathbb{R}^{1\times d}$ annotiert, sodass $f^{(0)}(u) = f^{(0)}(v)$ genau dann wenn $l(u) = l(v)$. Alternativ können in $f^{(0)}(v)$ anwendungsspezifische Informationen kodiert werden, z. B. kontinuierliche Eigenschaften von chemischen Molekülen oder Vektorrepräsentation von Text in sozialen Netzwerken. Ein GNN-Modell besteht nun aus einer Komposition von neuronalen Schichten, wobei jede Schicht Informationen aus der lokalen Nachbarschaft eines Knotens aggregiert. Ein einfaches GNN-Modell kann folgendermaßen beschrieben werden. In jeder Schicht $t > 0$, berechnen wir ein neues Merkmal

$$f^{(t)}(v) = \sigma\Big(f^{(t-1)}(v)\cdot W_1^{(t)} + \sum_{w\in N(v)} f^{(t-1)}(w)\cdot W_2^{(t)}\Big) \tag{2}$$

aus $\mathbb{R}^{1\times e}$ für v aus $V(G)$. Hierbei sind $W_1^{(t)}$ and $W_2^{(t)}$ Parametermatrizen aus $\mathbb{R}^{d\times e}$, und σ ist eine punktweise, nicht-lineare Funktion, z. B. eine Sigmoid- oder eine Rectifier-Funktion.

Die innere Summe über die Nachbarschaft kann durch eine Permutations-invariante, differenzierbare Funktion ersetzt werden, und die äußere Summe durch spaltenweise Konkatenation oder LSTM-artige Funktion ersetzt werden. Daher kann eine allgemeine GNN-Schicht als

$$f_{\text{merge}}^{W_2}\Big(f^{(t-1)}(v), f_{\text{aggr}}^{W_1}\big(\{\!\{f^{(t-1)}(w) \mid w\in N(v)\}\!\}\big)\Big), \tag{3}$$

geschrieben werden. Hier aggregiert $f_{\text{aggr}}^{W_1}$ über die Multimenge der Merkmale der Nachbarn und $f_{\text{merge}}^{W_2}$ vereinigt die Knotenrepräsentation des vorherigen Schritts mit den berechneten Merkmal der Nachbarn. Hierbei sind $f_{\text{aggr}}^{W_1}$ und $f_{\text{merge}}^{W_2}$ beliebige differenzierbare, Permutation-invariante Funktionen, z. B. neuronale Netzwerke. Im Folgenden bezeichnen wir obige Architekturen als 1-*dimensionale GNNs* (1-GNNs).

Im Folgendem untersuchen wir die Beziehung zwischen dem 1-WL und 1-GNNs. Sei (G,l) ein gelabelter Graph, und seien $\mathscr{W}^{(t)} = \left(\mathbf{W}_{\mathbf{1}}^{(\mathbf{t}')}, \mathbf{W}_{\mathbf{2}}^{(\mathbf{t}')}\right)_{t' \leq t}$ die Parameter eines 1-GNNs bis Iteration t. Hier kodieren wie initialen Label $l(v)$ als Vektoren $f^{(0)}(v)$ aus $\mathbb{R}^{1\times d}$, z. B. als 1-Hot-Kodierung. Unser erstes theoretisches Ergebnis zeigt, dass jede 1-GNN-Architektur, in Bezug auf die Unterscheidung von nicht-isomorphen Subgraphen, nicht mächtiger sein kann als der 1-WL. Formal, seien $f_{\text{aggr}}^{W_1}$ und $f_{\text{merge}}^{W_2}$ beliebige differenzierbare Funktionen und $\mathscr{W}^{(t)}$ beliebig gewählt, dann gilt, dass die Färbung C_t^1 vom 1-WL immer die Färbung $f^{(t)}$ des 1-GNN $\mathscr{W}^{(t)}$ verfeinert.

Theorem 3.1. Sei (G,l) ein gelabelter Graph. Dann gilt für alle $t \geq 0$, und allen initialen Färbungen $f^{(0)}$, die konsistent mit l sind, und alle Parameter $\mathscr{W}^{(t)}$, dass

$$C_t^1 \sqsubseteq f^{(t)}.$$

Unser zweites Resultat zeigt, dass eine Sequenz von Parametermatrizen $\mathscr{W}^{(t)}$ existiert, sodass das korrespondierende 1-GNN genau die gleiche Mächtigkeit wie der 1-WL hat.

Theorem 3.2. Sei (G,l) ein gelabelter Graph. Sei $t \geq 0$, dann existiert eine Sequenz von Parametermatrizen $\mathscr{W}^{(t)}$ und eine 1-GNN-Architektur, sodass

$$C_t^1 \equiv f^{(t)}.$$

Ferner, wurde, basierend auf dem k-WL, die k-GNN-Architektur entwickelt, und das obige Resultat für allgemeine k hergeleitet. In einer experimentellen Studie wurde diese auf zahlreichen Benchmark-Datensätzen untersucht. Insbesondere konnten für k-GNNs hervorragende Resultate für die Regression von quantenchemischen Eigenschaften gezeigt werden.

4 Zusammenfassung

Die vorliegende Arbeit beschäftige sich mit Methoden für die (überwachte) Graphklassifikation. Im Bereich der Kernmethoden haben wir skalierende Graphkerne für Graphen mit kontinuierlichen Knotenmerkmalen vorgeschlagen. Ferner wurde ein Graphkern, der globale Eigenschaften von Graphen einfangen kann, entwickelt. Zusätzlich wurde ein Framework für die Untersuchung der theoretischen Expressivität von Graphkernen vorgestellt. Hier wurde gezeigt, dass aktuelle Kerne nicht in der Lage sind einfach Grapheigenschaften einzufangen und ein beweisbar mächtiger Graphkern vorgestellt. Im Bereich der neuronalen Methoden wurden die Grenzen von GNNs aufgezeigt und eine beweisbar mächtigere Architektur hergeleitet.

Literaturverzeichnis

[BK05] Borgwardt, K. M.; Kriegel, H.-P.: Shortest-path kernels on graphs. In: IEEE International Conference on Data Mining. IEEE, S. 74–81, 2005.

[GFW03] Gärtner, T.; Flach, P.; Wrobel, S.: On Graph Kernels: Hardness Results and Efficient Alternatives. In: Learning Theory and Kernel Machines, Jgg. 2777 in Lecture Notes in Computer Science, S. 129–143. 2003.

[KJM20] Kriege, N. M.; Johansson, F. D.; Morris, C.: A survey on graph kernels. Applied Network Science, 5(1), Jan 2020.

[Kr18] Kriege, N. M.; Morris, C.; Rey, A.; Sohler, C.: A Property Testing Framework for the Theoretical Expressivity of Graph Kernels. In: International Joint Conference on Artificial Intelligence. IJCAI, S. 2348–2354, 2018.

[MKM17] Morris, C.; Kersting, K.; Mutzel, P.: Glocalized Weisfeiler-Lehman Kernels: Global-Local Feature Maps of Graphs. In: IEEE International Conference on Data Mining. IEEE, S. 327–336, 2017.

[MM19] Morris, C.; Mutzel, P.: Towards a practical *k*-dimensional Weisfeiler-Leman algorithm. CoRR, abs/1904.01543, 2019.

[Mo16] Morris, C.; Kriege, N. M.; Kersting, K.; Mutzel, P.: Faster Kernel for Graphs with Continuous Attributes via Hashing. In: IEEE International Conference on Data Mining. IEEE, S. 1095–1100, 2016.

[Mo19] Morris, C.; Ritzert, M.; Fey, M.; Hamilton, W. L.; Lenssen, Jan Eric; Rattan, G.; Grohe, M.: Weisfeiler and Leman Go Neural: Higher-order Graph Neural Networks. In: AAAI Conference on Artificial Intelligence. S. 4602–4609, 2019. AAAI.

[Mo20] Morris, Christopher: Learning with Graphs: Kernel and Neural Approaches. Dissertation, TU Dortmund University, 2020.

[RR08] Rahimi, A.; Recht, B.: Random features for large-scale kernel machines. In: Advances in Neural Information Processing Systems. NIPS, S. 1177–1184, 2008.

[Sh09] Shervashidze, N.; Vishwanathan, S. V. N.; Petri, T. H.; Mehlhorn, K.; Borgwardt, K. M.: Efficient Graphlet Kernels for Large Graph Comparison. In: International Conference on Artificial Intelligence and Statistics. PMLR, S. 488–495, 2009.

[Sh11] Shervashidze, N.; Schweitzer, P.; van Leeuwen, E. J.; Mehlhorn, K.; Borgwardt, K. M.: Weisfeiler-Lehman Graph Kernels. Journal of Machine Learning Research, 12:2539–2561, 2011. JMLR.

Christopher Morris studierte Informatik an der Technischen Universität Dortmund. Seit Juli 2015 war er wissenschaftlicher Mitarbeit in der Arbeitsgruppe von Prof. Dr. Petra Mutzel und Mitglied des Sonderforschungsbereichs 876 „Verfügbarkeit von Information durch Analyse unter Ressourcenbeschränkung“. Im Jahre 2019 hatte er einen dreimonatigen Gastaufenthalt an der Universität Stanford. Im Jahre 2020 schloss er mit Auszeichnung („summa cum laude“) seine Promotion zu einem Thema im Bereich des maschinellen Lernens mit Graphen ab. Seit März 2020 ist er wissenschaftlicher Mitarbeiter in der Arbeitsgruppe von Prof. Dr. Andrea Lodi an der Universität Montreal, Kanada.

Around-Body Interaction: Über die Nutzung der Bewegungen von Gliedmaßen zur Interaktion in einer digital erweiterten physischen Welt[1]

Florian Müller[2]

Abstract:

Durch den technischen Fortschritt sind Head-Mounted Displays (HMDs) kleiner und kabellos geworden. Sie leisten so einen Beitrag zur Vision von allgegenwärtiger Interaktion mit Informationen in einer digital erweiterten physischen Welt. Zur Interaktion mit solchen Geräten werden heute eingabeseitig – neben wenig intuitiven Fingergesten in der Luft – vor allem dreierlei Techniken verwendet: 1) Toucheingabe auf dem Gehäuse der Geräte oder 2) auf Zubehör (Controller) sowie 3) Spracheingabe. Während diese Techniken, abhängig von der aktuellen Situation des Benutzers, sowohl Vor- als auch Nachteile haben, so ignorieren sie weitgehend die Fähigkeiten und Geschicklichkeit, die wir im Umgang mit der realen Welt zeigen: Während unseres ganzen Lebens haben wir ausgiebig trainiert unsere Gliedmaßen zu benutzen, um mit der physischen Welt um uns herum zu interagieren und sie zu manipulieren. Diese Arbeit entwickelt eine Vision für eine körperlichere Interaktion mit solchen Geräten, welche die Fähigkeiten und Geschicklichkeit, die wir im Umgang mit der physischen Welt zeigen, auf die Interaktion mit HMDs überträgt.

1 Einführung

Unser ganzes Leben lang lernen wir unsere Gliedmaßen zu benutzen, um mit der uns umgebenden physischen Welt zu interagieren und sie zu manipulieren: Wir nutzen unsere Arme und Hände, um Türen zu öffnen oder unsere Beine und Füße, um uns in der Welt zu bewegen. Aufgrund des alltäglichen Umgangs und dem daraus resultierenden Trainings dieser Fertigkeiten führen wir diese Tätigkeiten mit beachtlichem Geschick und trotzdem zu großen Teilen unbewusst aus.

Betrachten wir nun, wie wir heute mit den uns umgebenden Computersystemen interagieren, so lässt sich dabei ein starkes Missverhältnis zwischen diesen hoch entwickelten physischen Möglichkeiten des menschlichen Körpers und den zur Interaktion genutzten Techniken erkennen. Anstatt die Fertigkeiten des menschlichen Körpers zur Interaktion zu nutzen, sitzen wir vor stationären Computern und bedienen diese indirekt durch sekundäre Geräte wie Tastatur und Maus. In den letzten Jahren scheint sich die Interaktion mit Computersystemen noch weiter von den physischen Möglichkeiten des Menschen zu entfernen: Während die indirekte Interaktion mit traditionellen Desktop PCs noch beide Hände (Tastatur) sowie Bewegungen der Hand (Maus) in die Interaktion mit einbezieht, dominieren heute mobile Touchgeräte den Markt. Diese Geräte werden typischerweise nur noch

[1] Englischer Titel der Dissertation: "Around-Body Interaction: Leveraging Limb-movements for Interacting in a Digitally Augmented Physical World" [Mü20]

[2] Technische Universität Darmstadt, mueller@tk.tu-darmstadt.de

CC BY SA

mit einem Daumen bedient, während die restlichen Finger zum Halten und Stabilisieren degradiert werden. Diese Art der Interaktion liegt zu großen Teilen in den inhärenten Limitierungen solcher Geräte begründet: Informationen werden als *Bilder unter Glas* [Vi11] visualisiert und sind an die kleine 2D Oberfläche des Gerätes gebunden.

Als mögliche Lösung können Informationen durch semi-transparente Head-Mounted Displays (HMDs) und Augmented Reality (AR) Technologien das *Glas durchbrechen* und so in die physische Welt einziehen. Solche Geräte bestehen aus einer am Kopf getragenen Kombination von zwei halb-transparenten Displays zur Bereitstellung von stereoskopischer Ausgabe sowie Sensoren zur Verfolgung der Position und Orientierung des Kopfes des Nutzers [Sh02]. Basierend auf den von den Sensoren aufgezeichneten Daten wird ein perspektivisch korrigiertes Bild erzeugt. Dies erlaubt eine natürliche Bewegung in einer (teilweise) virtuellen Umgebung [Su68], in der digitale Informationen mit physischen Objekten visuell gleichberechtigt existieren und so zu einer körperlicheren und physischeren Interaktion einladen. Jedoch besitzen diese digitalen Objekte im Gegensatz zu realen Objekten keinen physischen Körper: Sie sind nicht anfassbar. Daher sind die Interaktionstechniken, welche wir zur Interaktion mit der realen Welt verwenden, nicht direkt übertragbar.

Ein Blick auf die heute verbreiteten Interaktionstechniken für HMDs zeigt hauptsächlich Toucheingabe 1) auf dem Gehäuse des Gerätes oder auf 2) sekundären Geräten oder 3) sprachbasierte Eingabe. Trotz der jeweiligen Vorteile dieser Interaktionsstile ignorieren sie zu großen Teilen die vielen Freiheitsgrade, die unser Körper bietet, und zeigen weitere individuelle Nachteile: Die Toucheingabe auf HMDs und sekundären Geräten unterstützt keine direkte Manipulation von Informationen. Spracheingabe reduziert die Privatheit bei der öffentlichen Verwendung und die Erkennungsqualität ist stark abhängig von Umgebungsgeräuschen. Trotz der vielen technischen Fortschritte, welche HMDs in den letzten Jahren zu vielseitigen Ausgabegeräten gemacht haben, gibt es weiteren Forschungsbedarf, um aus HMDs auch vielseitige Eingabegeräte zu machen.

In den letzten Jahren hat die Forschung physischere Interaktionsstile mit HMDs im Bereich der *körperbasierten (body-based) Interaktion* vorgeschlagen. Als prominentestes Beispiel haben *on-body* Schnittstellen viel Aufmerksamkeit erhalten, welche Toucheingabe - oft in Kombination mit visueller Ausgabe - auf der Oberfläche unseres Körpers bieten [HRH12]. In vielen dieser Systeme dient die Hand oder der Unterarm des Nutzers als zweidimensionaler Interaktionsbereich und empfindet damit die Interaktion mit einem in der Hand oder am Körper getragenen Touchscreengerät nach. Neben allen Vorteilen solcher Systeme ist der Eingabebereich an die zweidimensionale Oberfläche des Körpers gebunden. Weiterhin erfordert diese Art der Interaktion meist beide Hände des Nutzers und kann so viele Situationen nicht unterstützen, in denen die Hände des Nutzers nicht frei sind (z. B. beim Tragen einer Tasche).

Um diese Einschränkungen, welche mit der Interaktion *auf* dem Körper verbunden sind, zu überwinden, beschäftigen sich die Beiträge dieser Arbeit mit der Interaktion im Bereich *um* unseren Körper. Dazu werden Interaktionen unter Verwendung unserer oberen und unteren Gliedmaßen betrachtet, um unsere in der physischen Welt gelernten Fertigkeiten auf die Interaktion mit digitalen Informationen zu übertragen.

Im Folgenden werden die vier Hauptbeiträge der Arbeit beschrieben. Die ersten beiden Beiträge, *Proximity-based Interaction* und *CloudBits*, beschäftigen sich dabei mit Interaktionstechniken unter Verwendung von Armen und Händen. Im Anschluss beschreiben die Beiträge *Mind the Tap* und *Walk the Line*, wie Beine und Füße für die Interaktion mit HMDs verwendet werden können.

2 Proximity-based Interaction

Abb. 1: *Proximity-based Interaction* erlaubt die mobile und einhändige Interaktion mit digitalen Informationen durch Bewegungen der Hand entlang der Sichtlinie des Nutzers.

Besonders in mobilen Situationen ist mindestens eine Hand des Nutzers häufig damit beschäftigt, mit der physischen Welt zu interagieren (bspw. beim Tragen einer Tasche, siehe Abb. 1). Dies hat zur Folge, dass Interaktionstechniken, die beide Hände des Nutzers erfordern, für solche Situationen ungeeignet sind. In diesem Teil der Arbeit wird argumentiert, dass für solche Situationen einhändige Interaktionstechniken die Interferenz zwischen Interaktionen mit einem System und mit der realen Welt reduzieren können. Daher fokussiert sich dieser Beitrag darauf, wie die große Anzahl von Freiheitsgraden, die unsere Hände und Arme bieten, für einhändige Interaktionen genutzt werden können.

Diese Freiheitsgrade der Bewegungen von Arm und Hand werden durch die Freiheitsgrade der beteiligten Gelenke definiert. Dazu gehören insbesondere das Schultergelenk, das Ellenbogengelenk, das Handgelenk sowie die vielen weiteren Gelenke der einzelnen Finger. Da einerseits die Bewegungen von Handgelenk und Fingern relativ ähnlich zu den bereits sorgfältig erforschten (on-body) Touchinteraktionen sind und andererseits vom Schultergelenk gesteuerte Bewegungen Ermüdung verursachen können [Hi14], konzentriert sich diese Arbeit in einem ersten Schritt auf die Freiheitsgrade des Ellenbogengelenks.

In diesem Kapitel wird der durch die Stellung des Ellenbogengelenkes definierte Abstand zwischen Hand und Gesicht des Nutzers als Eingabedimension betrachtet. Dieser Abstand ist durch Beugung (d. h. die Hand wird zum Körper hin bewegt) und Streckung (d. h. die

Hand wird vom Körper wegbewegt) des Gelenkes steuerbar. Der Interaktionsraum entlang der Sichtlinie des Nutzers kann in mehrere parallele Ebenen unterteilt werden. Ähnlich zum Vorgehen von Subramanian et al. [SAL06] enthält dabei jede Ebene individuelle visuelle Inhalte, welche auf der Hand des Nutzers dargestellt werden können. Der Nutzer kann die Hände entlang der Sichtlinie bewegen, um durch aufeinanderfolgende Ebenen zu blättern (siehe Abb. 1).

In dieser Arbeit werden zwei verschiedene Techniken für die Interaktion in diesem eindimensionalen Interaktionsraum untersucht:

Kontinuierliche Interaktionen erlauben es dem Nutzer mittels Streckung und Beugung des Armes durch aufeinanderfolgende Ebenen zu blättern, ähnlich der Bewegung beim Durchblättern eines Aktenregisters. Dies kann z. B. zum Scrollen durch eine Liste verwendet werden.

Diskrete Interaktionen erlauben es dem Nutzer diskrete Aktionen auszulösen, indem der Nutzer seinen Arm in verschiedenen Abständen vom Körper anhebt. Dabei sind den Entfernungen in Relation zum Körper jeweils verschiedene Aktionen zugeordnet. Dies kann beispielsweise ein Shortcut zu einer häufig genutzten Anwendung sein. Aufgrund der Propriozeption - dem Sinn, welcher uns ein natürliches Verständnis über die Position und Orientierung unserer Körperteile gibt, ohne diese zu sehen [TA18] - kann der Nutzer solche Aktionen ohne visuellen Fokus auf die Hand durchführen.

Die beiden Interaktionstechniken wurden in zwei kontrollierten Experimenten hinsichtlich ihrer Effizienz und Genauigkeit sowie der Nutzererfahrung evaluiert. Die Ergebnisse bestätigten die Praktikabilität dieser Eingabemodalität für schnelle und genaue Interaktionen, wobei sich die Entfernung zur Zielebene als primärer Einflussfaktor auf die Genauigkeit und Effizienz erwies.

3 CloudBits

HMDs sind aufgrund ihres speziellen Formfaktors radikal private Geräte: Während sich Informationen bei anderen Systemen leicht gemeinsam mit mehreren Personen betrachten und manipulieren lassen, ist dies bei HMDs nicht möglich, da die visuelle Ausgabe solcher Geräte nur dem tragenden Nutzer zur Verfügung steht.

In diesem Teil der Arbeit wird daher der Fokus auf die Interaktion mit mehreren gleichzeitigen Nutzern gelegt. Dazu nutzt CloudBits die Metaphern von *Wolken* und *Tropfen*, bei der kleine Informationseinheiten – passend zu den Themen des Gespräches – als Tropfen visualisiert werden, die langsam aus einer imaginären Wolke über den Nutzer in Richtung des Bodens fallen. Diese Tropfen stellen zusätzliche Informationen bezüglich der Themen des Gespräches bereit. Nutzer können mittels ihrer oberen Gliedmaßen mit diesen Informationstropfen interagieren (siehe Abb. 2).

Abb. 2: CloudBits unterstützt mehrere Nutzer bei der gemeinsamen Interaktion mit Informationen in einem geteilten Informationsraum.

CloudBits transkribiert dazu die Gespräche und leitet daraus Gesprächsthemen ab. CloudBits nutzt diese Themen als Suchbegriffe, um proaktiv Informationen für die Nutzer aus öffentlichen (z. B. Kartendaten, Webseiten) und privaten (z. B. E-Mail, Kalender) Informationsquellen abzurufen.

Die Verwendung von HMDs erlaubt es sowohl 1) öffentliche Informationen in einem gemeinsamen Interaktionsraum anzuzeigen als auch 2) individuelle private Informationen darzustellen, welche nur für den jeweiligen Nutzer sichtbar sind. Weiterhin bietet CloudBits verschiedene Interaktionstechniken, welche es den Nutzern erlauben, mit Informationen zu interagieren. Nutzer können dabei Informationen räumlich sortieren und gruppieren, um so Informationen implizit mit Metainformationen anzureichern. Die räumliche Position der öffentlichen Informationen ist zwischen allen Nutzern synchronisiert, sodass die Informationen am gleichen physischen Ort erscheinen und somit eine kollaborative Interaktion mit Informationen möglich wird. Weiterhin können Nutzer private Informationen mit anderen Nutzern teilen.

CloudBits bietet einen *Fokus+Kontext* [CMS99] Ansatz für die Interaktion mit Informationen in Gesprächssituationen. Während die Konversation der *Fokus* des Nutzers ist, stellt CloudBits durch kleine Informationstropfen, die den Verlauf der Konversation visualisieren, *Kontext* zur Verfügung. Umgekehrt wird CloudBits bei der Interaktion mit Informationen zum *Fokus* des Nutzers. Im Gegensatz zur Interaktion mit Informationen über ein mobiles Endgerät, bei welcher die visuelle Verbindung zur Konversation verloren geht, erlaubt CloudBits immer noch die audiovisuelle Teilnahme als *Kontext*, da sich die anderen Personen der Konversation noch in der peripheren Sicht befinden. Die enge Integration und Synchronisation von CloudBits mit der Konversation ermöglicht so einen schnellen und reibungslosen Übergang des Fokus zwischen der eigentlichen Konversation und der Interaktion mit Informationen.

Die vorgestellten Konzepte wurden in einer qualitativen Laborstudie untersucht und mit der traditionellen Interaktion mit Smartphones verglichen. Die Ergebnisse der Studie konnten hierbei vielversprechende Fortschritte bei der gemeinsamen Interaktion mit Informationen aufzeigen. So zeigte sich unter anderem im Vergleich mit der Interaktion mit Smartphones eine schnellere und leichtere Wiederaufnahme der Konversation nach Pausen.

4 Mind The Tap

Abb. 3: Mind The Tap erlaubt es Nutzern freihändig mittels Fußberührungen mit Informationen zu interagieren.

Wie in Abschnitt 2 und 3 dargestellt, können (einhändige) Interaktionstechniken für die oberen Gliedmaßen in vielen Situationen eine natürliche, genaue und effiziente Interaktionsform für HMDs darstellen. Es gibt jedoch gerade im mobilen Kontext Situationen, in welchen beide Hände und Arme des Nutzers nicht zur Interaktion zur Verfügung stehen, weil der Nutzer beispielsweise Dinge trägt oder anderweitig mit der realen Welt interagiert.

Zur Unterstützung solcher Situationen wird in diesem Teil der Arbeit untersucht, wie die unteren Gliedmaßen zur Interaktion eingesetzt werden können. Während es eine lange Geschichte der Nutzung der unteren Gliedmaßen für solche fußbasierten Schnittstellen in verschiedenen Bereichen – von Industriemaschinen [Kr71] über Desktop PCs [SB09] zu Smartphones [Fa17] – gibt, ist die Interaktion mittels Füßen noch nicht systematisch für den Umgang mit HMDs evaluiert worden.

Daher wird in diesem Abschnitt die fußbasierte Interaktion mit einer augmentierten Schnittstelle untersucht, welche am dominanten Fuß der Stehposition des Nutzers fixiert ist. Die Schnittstelle besteht aus einem halbkreisförmigen Interaktionsrad, welches durch mehrere Zeilen und Spalten in ein Gitter unterteilt wird. Jede Zelle des Gitters stellt eine Option dar, die der Nutzer durch eine Fußberührung auswählen kann (siehe Abb. 3).

In diesem Abschnitt der Arbeit werden zwei verschiedene Arten der Interaktion mit einem solchen interaktiven halbkreisförmigen Gitter untersucht, die beide durch die beschriebenen Fußberührungen bedient werden, sich aber in ihrer Visualisierung unterscheiden:

Direkte Interaktion Das halbkreisförmige Gitter wird in Beinreichweite auf dem Boden vor dem Teilnehmer visualisiert. Einzelne Bereiche des Gitters werden auf dem Boden vor dem Nutzer dargestellt und können von diesem mittels Fußberührungen ausgewählt werden. Es besteht also eine direkte Verbindung zwischen dem Ort der Eingabe und der Ausgabe.

Indirekte Interaktion Bei der indirekten Interaktion wird die Visualisierung vom Boden in die Luft vor dem Nutzer verschoben, die Bedienung des Systems erfolgt jedoch weiterhin durch Fußberührungen vor dem Nutzer. Der Propriozeptionssinn ermöglicht es Nutzern, trotz der fehlenden visuellen Verbindung zwischen Eingabe und Ausgabe, mit einem solchen System zu interagieren.

Die beiden Interaktionsarten wurden in einem kontrollierten Experiment unter Laborbedingungen miteinander verglichen und hinsichtlich ihrer Genauigkeit und Effizienz bewertet. Die Ergebnisse zeigen die Eignung der Interaktionsarten für eine genaue und angenehme Interaktion durch Fußbewegungen. Der Vergleich der Interaktionsarten zeigte, dass sich die direkte Interaktion besser für kurze und feingranulare Interaktionen eignet, während sich die indirekte Interaktion besser für längerfristige Interaktionen eignet.

5 Walk The Line

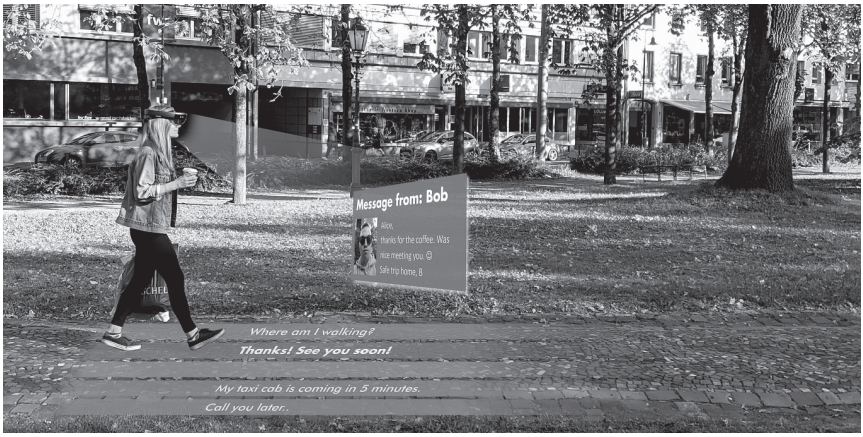


Abb. 4: Walk the Line nutzt seitliche Verschiebungen des Gehweges als Eingabemodalität für HMDs. Optionen werden als Spuren auf dem Boden entlang des Gehweges des Nutzers visualisiert. Der Nutzer wählt Optionen aus, indem er seinen Pfad seitlich verschiebt.

Der letzte Abschnitt beschäftigte sich mit Fußberührungen als Eingabemodalität für HMDs. Neben den dargestellten Vorteilen der freihändigen Nutzung gibt es jedoch auch eine große

Einschränkung bei der Benutzung eines solchen Systems, welche in den inhärenten Grenzen der fußbasierten Bedienung begründet ist: Der Nutzer muss zur Interaktion stehen bleiben, da ein solches System nicht zwischen gewollten Bewegungen zur Interaktion und Bewegungen, wie sie bei der Fortbewegung natürlicherweise auftreten, unterscheiden kann.

Während diese Einschränkung im industriellen Kontext tolerierbar sein mag, wird sie in wirklich mobilen Interaktionssituationen zu einer großen Herausforderung: Mobilität ist stark mit dem Prozess der Fortbewegung und damit insbesondere dem Gehen verbunden. Dies macht die Anwendung traditioneller fußbasierter Interaktionstechniken in derartigen Situationen schwierig, da der Nutzer den Prozess der Fortbewegung stoppen muss, um mit dem System zu interagieren. Wenn jedoch die Vision der ubiquitären Interaktion mit Informationen in einer digital erweiterten physischen Welt Wirklichkeit werden soll, so wird ein wesentlicher Teil der Interaktion mit solchen Geräten auch unterwegs stattfinden müssen. Dies unterstreicht die Notwendigkeit wirklich mobiler Interaktionstechniken, die nicht nur die Interaktion an verschiedenen Orten unterstützen, sondern auch während der Nutzer sich in Bewegung zu diesen Orten befindet.

Heutzutage findet die Interaktion während des Gehens häufig mit Smartphones statt. Der dabei erforderliche visuelle Fokus auf das Gerät lässt Fußgänger während der Interaktion den Kontakt zu ihrer Umwelt verlieren [LH17]. Ähnlich wie das abgelenkte Führen eines Autos führt auch abgelenktes Gehen zu potenziell gefährlichen Situationen: Fußgänger laufen in Hindernisse, kollidieren mit anderen Personen oder gefährden sich und andere Personen auf weitere Arten.

Der Beitrag dieses Abschnitts geht über den aktuellen Stand der Technik hinaus, indem der Prozess der Fortbewegung durch das Gehen aktiv als Inputmodalität genutzt wird. Dazu wird ein System betrachtet, welches mehrere Spuren entlang des Gehweges des Nutzers anzeigt. Dabei stellt jede Spur eine individuelle Option dar, welche der Nutzer durch Änderung seines Gehweges selektieren kann (siehe Abb. 4). Die spezifische Visualisierung der Spuren kann auf die Anwendung zugeschnitten werden. Sie kann z. B. Icons oder Text enthalten oder mit in der Luft schwebenden Elementen verbunden werden, die die zu selektierenden Informationen beschreiben.

Der Nutzer muss lediglich seinen Gehweg seitlich verschieben, bis er auf der jeweiligen Optionsspur läuft, um eine Aktion des Systems auszulösen. Diese Verschiebung geschieht, ohne dass der Nutzer seine Gehgeschwindigkeit ändern muss, ähnlich dem Ausweichen vor einem Hindernis auf dem Bürgersteig. Das System hebt die Spur, auf der der Nutzer gerade läuft, durch eine Änderung der Visualisierung (z. B. der Farbe) hervor. Diese Änderung betrifft die gesamte Spur und ist so auch im peripheren Sichtbereich wahrnehmbar, ohne dass der Nutzer aktiv nach unten schauen muss. Durch das Gehen auf einer der Spuren für eine bestimmte Zeitspanne kann die jeweilige Option ausgewählt werden. Die benötigte Zeit wird dem Nutzer durch eine Änderung der Deckkraft der Spuren visualisiert: Während des Gehens auf einer Spur werden alle anderen Spuren nach und nach ausgeblendet.

Neben den Optionsspuren existiert eine weitere neutrale Spur, welche direkt mittig vor dem Nutzer dargestellt wird und mit keiner Aktion verknüpft ist (vgl. graue Spur in Abb. 4). Geht der Nutzer geradeaus weiter, ohne seinen Gehweg zu verändern, interpretiert das System dies nicht als Interaktion und löst keine Aktionen aus.

Die hier vorgestellte Interaktionstechnik wurde in einem kontrollierten Laborexperiment hinsichtlich ihrer Genauigkeit und Effizienz für die Interaktion mit HMDs untersucht. Die Ergebnisse des kontrollierten Experiments bestätigten die Eignung solcher Schnittstellen für schnelle, genaue und spielerische Interaktionen.

6 Zusammenfassung

Die jüngsten technologischen Entwicklungen deuten darauf hin, dass wir auf dem Weg zu immer kleineren und leistungsfähigeren HMDs sind, welche eines Tages die Rolle übernehmen könnten, die Smartphones heute in unserem Leben spielen. So würde die Interaktion mit solchen Geräten – 50 Jahre nach Sutherlands ersten Schritten [Su68] – endlich aus dem Labor in die reale Welt vordringen.

In dieser Arbeit wurden aus den Anforderungen für die Interaktion mit HMDs unter Verwendung der oberen und unteren Gliedmaßen jeweils zwei Konzepte entwickelt sowie konkrete Interaktionstechniken abgeleitet. Diese Interaktionstechniken wurden im Anschluss eingehend untersucht und evaluiert. Obwohl jede der vorgestellten Techniken auf einen spezifisches Kontext zugeschnitten ist, kann die Kombination dieser Techniken bereits eine Vielzahl von Situationen abdecken, die bei der täglichen Nutzung solcher Geräte auftreten.

Damit hat diese Arbeit einen substanziellen Beitrag zur Vision für die zukünftige Interaktion mit solchen Geräten geleistet, indem sie die Freiheitsgrade, die unser Körper bietet, für natürlichere, angenehmere und spielerischere Interaktionen nutzbar macht.

Literaturverzeichnis

[CMS99] Card, S. K.; Mackinlay, J. D.; Shneiderman, B.: Information Visualization. In: Readings in Information Visualization: Using Vision to Think. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, S. 1–34, 1999, ISBN: 978-1-55860-533-6.

[Fa17] Fan, M.; Ding, Y.; Shen, F.; You, Y.; Yu, Z.: An empirical study of foot gestures for hands-occupied mobile interaction. In: Proceedings of the 2017 ACM International Symposium on Wearable Computers - ISWC '17. ACM Press, New York, New York, USA, S. 172–173, 2017, ISBN: 9781450351881.

[Hi14] Hincapié-Ramos, J. D.; Guo, X.; Moghadasian, P.; Irani, P. P.: Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. In: Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14. ACM Press, New York, New York, USA, S. 1063–1072, 2014, ISBN: 9781450324731.

[HRH12] Harrison, C.; Ramamurthy, S.; Hudson, S. E.: On-body interaction: armed and dangerous. In: Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction - TEI '12. ACM Press, New York, New York, USA, S. 69, Feb. 2012, ISBN: 9781450311748.

[Kr71] Kroemer, K. H. E.: Foot Operation of Controls. Ergonomics 14/3, S. 333–361, Mai 1971, ISSN: 0014-0139.

[LH17] Lin, M.-I. B.; Huang, Y.-P.: The impact of walking while using a smartphone on pedestrians' awareness of roadside events. Accident Analysis & Prevention 101/, S. 87–96, Apr. 2017, ISSN: 00014575.

[Mü20] Müller, F.: Around-Body Interaction: Leveraging Limb-movements for Interacting in a Digitally Augmented Physical World. to appear: TUprints, Darmstadt, 2020.

[SAL06] Subramanian, S.; Aliakseyeu, D.; Lucero, A.: Multi-layer interaction for digital tables. In: Proceedings of the 19th annual ACM symposium on User interface software and technology - UIST '06. ACM Press, New York, New York, USA, S. 269, Okt. 2006, ISBN: 1595933131.

[SB09] Silva, M. G.; Bowman, D. A.: Body-based interaction for desktop games. In: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems - CHI EA '09. ACM Press, New York, New York, USA, S. 4249, 2009, ISBN: 9781605582474.

[Sh02] Shibata, T.: Head mounted display. Displays 23/1-2, S. 57–64, Apr. 2002, ISSN: 01419382.

[Su68] Sutherland, I.: A head-mounted three dimensional display. In: Proceedings of the December 9-11, 1968, fall joint computer conference, part I on - AFIPS '68 (Fall, part I). ACM Press, New York, New York, USA, S. 757, 1968.

[TA18] Tuthill, J. C.; Azim, E.: Proprioception. Current Biology 28/5, R194–R203, März 2018, ISSN: 09609822.

[Vi11] Victor, B.: A Brief Rant on the Future of Interaction Design, Website, unter `http://worrydream.com/ABriefRantOnTheFutureOfInteractionDesign/`; abgerufen am 21.01.2020., Nov. 2011.

Florian Müller ist ein Forscher im Bereich der Human-Computer Interaction. Seine Forschungsinteressen liegen im Bereich Augmented Reality, Virtual Reality und körperbasierter Kozepte zur Interaktion in und mit der physischen Welt. Er hat an der Technischen Universität Darmstadt am Telecooperation Lab unter Leitung von Max Mühlhäuser in diesen Bereichen promoviert und arbeitet dort aktuell als Wissenschaftlicher Mitarbeiter.

Schlussfolgern in Leichtgewichtigen Defeasible Beschreibungslogiken[1]

Maximilian Pensel[2]

Abstract: Klassische Systeme der logischen Wissensrepräsentation und des automatischen Schließens sind nicht in der Lage Standardannahmen von Anwendungen adäquat zu modellieren und auf zu erwartende Ausnahmefälle intuitiv zu reagieren. Trotz ihrer erhöhten Praktikabilität, Popularität und Berechenbarkeit sind auch klassische Beschreibungslogiken durch ihre monotone Semantik nicht in der Lage unvollständiges Wissen individuell durch Standardwerte zu vervollständigen. Um ein derartiges Verhalten für Beschreibungslogiken zu erreichen, werden etablierte *nichtmonotone* Semantiken der Aussagenlogik auf die Ausdrucksstärke der Semantik für Beschreibungslogiken angehoben. Im Rahmen meiner Studien zeigte sich, dass die so entstandene Form der *Rational Closure* die wichtigsten Konsequenzen, die Beschreibungslogiken von Aussagenlogik abheben, vernachlässigt. In meiner Dissertation [Pe19] untersuche und überarbeite ich den bisherigen, unzulänglichen Ansatz zur Rational Closure in Beschreibungslogiken und entwerfe eine Reihe neuartiger Semantiken, welche die fehlenden Konsequenzen nachweislich ermitteln können.

1 Einführung

Computergestützte Assistenzsysteme durchdringen immer mehr gesellschaftliche Bereiche (Mensch-Maschine Interaktion, Robotic Co-Working, Alexa, etc.). Diese Systeme müssen immer mehr leisten und sich als robust behaupten, währen gleichzeitig unsere Abhängigkeit von ihnen steigt. Insbesondere wenn Systeme autonom (Fahren, Fliegen) oder unterstützend (Medizin, Finanzwesen) Entscheidungen treffen, bedeutet ihr Handlungsspielraum unseren technologischen Fortschritt. Neben den modernen stochastischen Methoden (Deep Learning, etc.) spielt im Umgang mit strukturiertem Wissen, formale Wissensrepräsentation und logisches Schließen eine wesentliche Rolle. So kann eine Wissensbasis aus logischen Formeln explizit zu einer gegebenen Anwendungsdomäne, wie z.B. der Medizin, verfasst werden, um aus einer manuell unüberschaubaren Menge an Daten, gültige *implizite* Fakten automatisch zu ermitteln. Welche Logik und welche Semantik dabei zum Einsatz kommen, bestimmt die Art des repräsentierbaren Wissens, sowie der angestrebten logischen Schlussfolgerungen.

Klassische Logiken wie z.B. die Aussagenlogik sind durch monotone Semantiken definiert. D.h. wenn aus einer Menge von Formeln eine Schlussfolgerung als gültig ermittelt wird, so behält diese Folgerung ihre Gültigkeit in jeder Erweiterung dieser Wissensbasis. Bereichen denen eine dynamische Entwicklung von Beobachtungen und Erkenntnissen (Wissen) zugrunde liegt, wird diese Monotonie jedoch zum Verhängnis. Unvollständiges

[1] Englischer Titel der Dissertation: „A Lightweight Defeasible Description Logic in Depth“

[2] Maximilian Pensel, maximilian.pensel@gmx.de

CC BY SA

Wissen durch Standardannahmen zu ergänzen oder stereotypische Eigenschaften für generische Objekte zu folgern sind mächtige Mechanismen, wenn individuelle Ausnahmefälle adäquat behandelt werden können, anstatt zu widersprüchlichen Konsequenzen zu führen. Ein Arzt[3] kann aus einem ersten Patientengespräch eine geeignete Therapie vorschlagen. Wenn weiterführende medizinische Tests allerdings ein Krankheitsbild suggerieren welches der initialen Therapie widerspricht, dann sollte der Arzt seine bisherigen Folgerungen geeignet *revidieren*. Derartiges verhalten mit einem logischen Formalismus zu erreichen, nennt man *nichtmonotones Schließen*. Maschinelles logisches Schließen wird schon heute unterstützend im medizinischen Bereich eingesetzt, allerdings mittels Formalismen denen die Monotonie zugrunde liegt, welche also ein derartiges Szenario nicht abbilden können [Sp].

Seit 1960 untersuchen Theoretiker wie Marvin Minsky, John McCarthy und Ray Reiter die Alternative der *nichtmonotonen* Semantiken, in denen eine logische Schlussfolgerung durch neue Erkenntnisse individuell revidiert werden kann. Da, *a priori*, nicht eindeutig ist, welche Gründe für eine solche Revision von Folgerungen angemessen sind, entstanden über die vergangenen Jahrzehnte eine Vielzahl verschiedener Ansätze zum nichtmonotonen Schließen. Donald Nute sagt „[...] any reasoning system that preserves truth must be monotonic, [...] [but] a reasoning system that preserves justification will *not* be monotonic“ ([Nu01], S. 152). Wenn die Begründung für eine Folgerung erhalten bleibt, kann das Auftreten eines ihr widersprechenden Faktes die Folgerung aufheben, ohne dabei die Begründung anzufechten. Diese Idee liegt den *Defeasible Logiken* zugrunde. Dort werden Aussagen über *typische Situationen* als Begründung zur Folgerung von Wissen verwendet. Aus Situationen, die dem typischen Fall widersprechen, werden somit andere Folgerungen unterstützt. Ein Patient mit Fieber leidet *typischerweise* unter der Grippe. Eine Therapie sollte zunächst entsprechend dieser Diagnose behandeln, wobei weiterführende Tests eine andere Diagnose aufdecken können. Dennoch ist es sinnvoll bei neuen Patienten mit den gleichen Symptomen wieder die gleiche Begründung anzuwenden, um zu folgern, dass auch diese *typischerweise* eine Grippe haben.

Ein Spezialfall für Defeasible Logik wurde 1990 von Kraus, Lehmann und Magidor [KLM90] für Aussagenlogik behandelt. Kraus et al. stellen ein axiomatisches System mittels Postulaten auf. Ein nichtmonotoner Schlussoperator gilt als *rational* wenn er diese Postulate erfüllt. Später isolieren Lehmann und Magidor einen dieser Schlussoperatoren, als *Mindestanforderung* eine Wissensbasis nichtmonoton zu erweitern und nennen ihn *Rational Closure* (RC). Diese These, und somit das Schließen unter RC, ist bis heute weitreichend akzeptiert und wurde seitdem für verschiedene logische Formalismen etabliert. Ein wesentlicher Vorteil der RC ist, dass ihre Folgerungen, mithilfe einer polynomiellen Reduktion, aus klassischen Folgerungen der zugrunde liegenden Logik ermittelt werden können.

Eine Reihe von Formalismen, die sich außerordentlich gut für den praktischen Einsatz und die Konstruktion anwendungsnaher Wissensbasen bewährt haben, sind *Beschreibungslogiken* (BL). BL sind Fragmente der Prädikatenlogik erster Ordnung, mit der wesentlichen

[3] In dieser Arbeit wird aus Gründen der besseren Lesbarkeit das generische Maskulinum verwendet. Weibliche und anderweitige Geschlechteridentitäten werden dabei ausdrücklich mitgemeint.

Einschränkung, dass in den meisten BL ausschließlich unäre und binäre Prädikate sowie Konstanten erlaubt sind. Logische Kompositionen solcher Prädikate werden genutzt um Mengen von Objekten (Konzepte) zu charakterisieren. Der wesentliche Vorteil für die Wissensrepräsentation mit BL gegenüber Aussagenlogik ist die Verwendung von binären Prädikaten und die mit ihnen verknüpften Formen von Quantifikation. Diese erlauben, ergänzend zu der Beschreibung der Eigenschaften einer Klasse von Objekten, die Eigenschaften anderer, in Relation stehender Objekte auszudrücken. Aussagenlogik ermöglicht lediglich die Beschreibung *einer* Situation bzw. *eines* Objektes mittels logischer Komposition von Literalen, wie z.B. „ein Arzt, welcher *nicht* spezialisiert ist“.

$$arzt \wedge \neg spezialisiert \tag{1}$$

BL dagegen können z.B. die Klasse aller Ärzte (unäres Prädikat) charakterisieren, die einen spezialisierten Kollegen (binäres Prädikat) haben.[4]

$$Arzt \sqcap \exists hat_kollege.(Arzt \sqcap Spezialisiert) \tag{2}$$

Dies ermöglicht die getrennte Beschreibung der Eigenschaften von Objekten und den Eigenschaften der zu diesen in Beziehung stehenden Objekte. Eine Wissensbasis in BL setzt sich aus terminologischen Implikationen, wie z.B. „*jeder* Arzt arbeitet in einem Krankenhaus“, sowie Zuweisungen von benannten Individuen (Konstanten) zu komplexen Konzepten, wie z.B. $(Arzt \sqcap Spezialisiert)(charlie)$, zusammen. Für die in (2) beschriebene Klasse von Personen können wir dann folgern, dass sie sowohl in einem Krankenhaus arbeiten, als auch einen Kollegen haben, welcher in einem Krankenhaus arbeitet. Derartige implizite Folgerungen werden *Subsumption* genannt. Die Schlussfolgerung welche prüft, ob weitere Zuordnungen, wie z.B. $(\exists arbeitet_in.Krankenhaus)(charlie)$, implizit gültig sind, wird *Instance Checking* genannt.

Wesentliche Bestandteile der Forschung um BL sind die Untersuchung neuartiger logischer Konsequenzen, sowie die Analyse der BL in Hinsicht auf ihre Ausdrucksstärke und die Komplexität verschiedene logische Konsequenzen (wie Subsumption und Instance Checking) zu berechnen. BL grenzen sich hauptsächlich in der Art der zugelassenen logischen Kombinationen voneinander ab. So lässt die BL $\mathcal{EL}_\bot$, mit der ich mich in meiner Dissertation befasst habe, nur das logische Und ($\sqcap$), existentielle Quantifikation ($\exists$), sowie Wahrheit ($\top$) und Falschheit ($\bot$) zu. Damit ist sie zwar weniger Ausdrucksstark als BL die zusätzlich Negation ($\neg$) erlauben (wie z.B. $\mathcal{ALC}$), allerdings können klassischen logische Konsequenzen in polynomieller Zeit berechnet werden. Des Weiteren reichen in der Praxis die Konstrukte von $\mathcal{EL}_\bot$ häufig aus. So kommt z.B. die Medizin-Ontologie SNOMED [Sp] ohne die Negation von Konzepten aus.

Seit 2010 untersuchen Casini et al. [Ca14, CMS18, CS10, CS12] eine Form der RC für BL. Die *typischen Implikationen* die in Defeasible Logiken zum Einsatz kommen werden in BL als Pendant zu terminologischem Wissen eingeführt, etwa „Ärzte haben *typischerweise* studiert“. Casini et al. erreichen eine syntaktische Übersetzung der algorithmischen

[4] Das in (2) beschriebene Konzept verwendet die Syntax von BL. Es kann äquivalent in die folgende prädikatenlogische Formel mit einer freien Variable x übersetzt werden: $Arzt(x) \wedge \exists y.(hat_kollege(x,y) \wedge Arzt(y) \wedge Spezialisiert(y))$.

Lösung für RC in Aussagenlogik von Lehmann und Magidor, zu BL. Diese Übersetzung ermöglicht allerdings nicht, typische Informationen für quantifizierte Konzepte abzuleiten. Für die Klasse von Ärzten in (2) kann so zwar gefolgert werden, dass sie *typischerweise* studiert haben, nicht jedoch, dass auch ihre spezialisierten Kollegen *typischerweise* studiert haben. Da eine derartige Referenzierung von Elementen den wesentlichen Vorteil von BL über Aussagenlogik, hinsichtlich ihrer Ausdrucksstärke, bildet, ist es für brauchbare Defeasible BL unumgänglich auch typische Eigenschaften für quantifizierte Konzepte ermitteln zu können.

Der Beitrag meiner Dissertation ist zweierlei. Zunächst habe ich eine Reihe von Forschungsbeiträgen zur RC in BL, welche verschiedene Varianten des algorithmischen Ansatzes in unterschiedlicher Formalisierung behandeln, abstrahiert und vereinheitlicht (vgl. Abschnitt 2.1). Mit dem so entstandenen *Framework* konnte ich auch das Spektrum nichtmonotoner Schlussfolgerungen für $\mathcal{EL}_\bot$ substantiell erweitern. Außerdem verdeutlicht meine Abstraktion diejenige Komponente in diesem algorithmischen Verfahren, die für das Vernachlässigen essentieller Konsequenzen verantwortlich ist. Mit dieser Erkenntnis habe ich neuartige Semantiken für die leichtgewichtige Defeasible BL, $\mathcal{EL}_\bot$, entworfen (vgl. Abschnitt 3), mit dem Ziel

1. Die bisherigen Folgerungen der RC konservativ zu ergänzen, und
2. Typische Eigenschaften für quantifizierte Konzepte uniform zu erschließen.

Auch die neuen Semantiken bedienen sich der Abstraktion aus dem ersten Teil meiner Arbeit. Dies hat mir erlaubt einen uniformen Algorithmus und eine einheitliche Komplexitätsanalyse für das Schließen unter den neuen Semantiken zu erreichen. Mit diesen Semantiken werden logikbasierte Assistenzsysteme in Zukunft auch mit den bisher nicht unterstützten Phänomenen adäquat umgehen können.

2 Rational Closure in Beschreibungslogiken

Der Algorithmus den Lehmann und Magidor beschreiben, um Folgerungen in der RC zu ermitteln, verwendet eine Technik die als *Materialisierung* bezeichnet wird. Hierbei werden Defeasible Implikationen (z.B. „Fieber deutet *typischerweise* auf eine Grippe hin“, $f \rightsquigarrow g$) aus der Wissensbasis, in Form einer *materiellen Implikation* ($\neg f \vee g$) genutzt, um das Subjekt einer Folgerung individuell mit dieser Eigenschaft zu erweitern. Wird zum Beispiel gefragt, ob eine Person mit Fieber ($p \wedge f$) unter der Grippe leidet, so würde diese Anfrage erweitert werden zu:

$$(\neg f \vee g) \wedge p \wedge f \rightarrow g, \tag{3}$$

was unter Anwendung der klassischen Semantik gültig ist. Bei dieser Materialisierung von Defeasible Wissen muss allerdings zuerst ermittelt werden welche Defeasible Implikationen eingesetzt werden können, damit Widersprüche vermieden werden. So darf zum

$\mathcal{ALC}$	Rational Closure	Lexicographic Closure	Relevant Closure
Subsumption	✓	✓	✓
Instance Checking	✓	✓/ [Pe19]	[PT18]
$\mathcal{EL}_\bot$			
Subsumption	✓/ [PT17a]	[Pe19]	[PT17b]
Instance Checking	[PT18]	[Pe19]	[PT18]

Tab. 1: Vom Materialisierungs-Framework abgedeckte Methoden des logischen Schließens in [Pe19]. Ein ✓ markiert von anderen Forschungsgruppen eingeführte Methoden [Ca14, CMS18, CS10, CS12]. Felder mit Referenzen bezeichnen unseren ersten Beitrag diese Methode zu konstruieren. Für lexicographic instance checking in $\mathcal{ALC}$, verbessert [Pe19] die Definition gegenüber [CS12], sodass Schlussfolgerungen unter RC vollständig erhalten bleiben. Eine polynomielle Methode für Subsumption in RC für $\mathcal{EL}_\bot$ wurde von Casini et al. [CMS18] unabhängig von uns entdeckt.

Beispiel bei einer Person mit Fieber, für welche die Grippe bereits ausgeschlossen wurde ($p \wedge f \wedge \neg g$), die obige Defeasible Implikation nicht eingesetzt werden.

Die syntaktische Adaption dieses Verfahrens für BL hat zur Folge, dass die Polynomialität der Reduktion, von nichtmonotonen Folgerungen auf klassische Folgerungen, erhalten bleibt. Somit kann die Berechnung einer Folgerung unter der RC die gleiche Komplexität, wie die Berechnung einer klassischen Konsequenz in der jeweiligen BL, haben. Dieses Ergebnis galt zunächst nur trivialerweise für BL, welche Negation und Disjunktion erlauben, insb. $\mathcal{ALC}$. Für eine Logik wie $\mathcal{EL}_\bot$ erhält diese Form der Materialisierung allerdings nicht die Komplexität der Berechnung klassischer Konsequenzen. In [PT17a, PT17b] betrachteten wir explizit die BL $\mathcal{EL}_\bot$ und entwarfen erstmals eine alternative Form der Materialisierung, welche auf den Einsatz von Disjunktion und Negation verzichtet, und somit polynomielle Schlussfolgerungen erhält.

Lehmann und Magidor argumentierten bereits, dass die RC gewisse Nachteile birgt. Der bekannteste Effekt ist das sogenannte *Inheritance Blocking*. Es besagt, dass konsistente *typische Eigenschaften* für eine (aus anderen Gründen) untypische Instanz verworfen, anstatt wie erwartet vererbt zu werden. Lehmann führt eine konservative Erweiterung von RC, die *Lexicographic Closure* (LC), ein um dieses Problem, auf Kosten der polynomiellen Berechnungskomplexität, zu lösen. Konsequenzen unter LC können ebenso mittels Materialisierungen ermittelt werden, lediglich die Auswahl konsistenter typischer Eigenschaften ist komplexer. Auch Casini et al. adaptieren den algorithmischen Ansatz zur LC für Defeasible BL [CS12] und führen ihre eigene Erweiterung, die sogenannte *Relevant Closure* (ReC) ein [Ca14], um Inheritance Blocking zu beheben.

2.1 Ein Framework für Materialisierung

Im ersten Teil meiner Arbeit vereinheitliche ich das algorithmische Vorgehen von Casini et al. um RC, LC und ReC zu reproduzieren, indem ich die beiden elementaren Bestandteile dieses Ansatzes in einem Framework formal abstrahiere. Diese Komponenten sind

1. Die Auswahl von konsistenten Defeasible Implikationen, und
2. Die Art der Materialisierung.

Instanziierungen der ersten Komponente ermöglichen es, unterschiedlich *starke* Schlussoperatoren zu berechnen. Konkrete Instanziierungen hierzu betrachte ich in meiner Arbeit für RC, LC und ReC. Instanziierungen der zweiten Komponente ermöglichen es, verschiedene Materialisierungen für die zugrunde liegende BL anzuwenden. So konnte ich sowohl die ursprüngliche Materialisierung nach Lehmann und Magidor (bzw. Casini et al.) für $\mathcal{ALC}$, als auch unsere Beiträge von 2017 für $\mathcal{EL}_\bot$, mit meinem Framework rekonstruieren.

Die Kombinationen dieser Instanziierungen erlauben es, mit einer einheitlichen Definition, sämtliche Closures, nichtmonotonen Folgerungen (Subsumption/Instance Checking) und BL aus Tabelle 1 nachzubilden, zu ergänzen und in einem Fall zu verbessern. Somit haben wir das Problem des Inheritance Blocking in RC für $\mathcal{EL}_\bot$ erstmals auf zwei verschiedene Arten lösen können. Gleichzeitig bietet das Materialisierungs-Framework eine Grundlage für zukünftige Erweiterungen der Rational (bzw. Relevant/Lexicographic) Closure.

Dennoch leidet jeglicher Einsatz von Materialisierung unter einem enormen Nachteil. Da Materialisierung auf einer syntaktischen Ebene mit den Anfragen interagiert, schauen wir uns diese Interaktion einmal genauer an. Betrachten wir die Menge aller Ärzte welche mindestens einen Patienten mit Fieber versorgen ($Arzt \sqcap \exists versorgt.Fieber$) und formulieren dass Fieber typischerweise auf eine Grippe hindeutet, so wird die beschriebene Menge von Ärzten wie folgt durch Materialisierung der Defeasible Eigenschaft ergänzt:

$$(\neg Fieber \sqcup Grippe) \sqcap Arzt \sqcap \exists versorgt.Fieber. \tag{4}$$

Wir beschreiben hiermit alle Ärzte, welche entweder nicht unter Fieber leiden oder die Grippe haben, und welche einen Patienten mit Fieber versorgen. Die typische Eigenschaft die wir dem Fieber zugeschrieben haben, wird durch diese Form der Materialisierung nicht auf relational verbundene Objekte angewendet und jegliche Defeasible Konsequenzen über diese referenzierten Objekte gehen praktisch verloren. Somit sind bisherige Ansätze zum nichtmonotonen Schließen unter RC in BL unbrauchbar, in Hinsicht auf die Ausdrucksstärke, welche BL von Aussagenlogik abhebt. In meiner Dissertation bezeichne ich dieses Problem als *Quantification Neglect*. Vor unseren ersten Beiträgen in 2017 wurde es ausschließlich in [Bo15] benannt, jedoch nicht explizit untersucht.

3 Eine Neue Semantik für Defeasible Beschreibungslogiken

Um bisherige Ansätze mit meinen neuen Semantiken in Relation zu stellen, ergänze ich die bestehende Abgrenzung der *Stärke* (Rational: `rat`, Relevant: `rel`, Lexicographic: `lex`) um das Attribut der *Reichweite* nichtmonotoner Semantiken. So bezeichne ich die Reichweite von Konsequenzen aus dem Materialisierungs-Algorithmus als *propositional* (`prop`), da die Ausdrucksstärke typischer Konsequenzen, die der Aussagenlogik nicht übersteigt. Alte und neue Schlussoperatoren/Semantiken können dann in einem zweidimensionalen Vektorraum (z.B. $(\texttt{rat}, \texttt{prop})$) identifiziert werden. Außerdem können maßgebende Komponenten, wie die Auswahl konsistenter typischer Eigenschaften für *Stärke*, mit den neuen Semantiken beliebig kombiniert werden (vgl. Abb. 2).

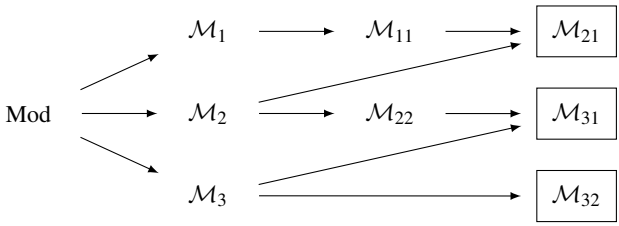


Abb. 1: Beispielhafte Graphenstruktur der Präferenzrelation auf Typicality Modellen. *Mod* bezeichnet die Menge aller Typicality Modelle. Eine Kante bezeichnet, dass der Kindknoten gegenüber dem Elternknoten präferiert ist. Maximal typische Mengen von Modellen (MaxTM) sind umrandet.

Ein Vorteil den leichtgewichtige BL, wie $\mathcal{EL}_\bot$, mit sich bringen, ist, dass ihre klassische Semantik mit endlichen relationalen Strukturen auskommt. Diese relationalen Strukturen sind Prädikatenlogische Interpretationen und können (in BL) als beschriftete, gerichtete Graphen verstanden werden. In klassischer Semantik wird eine Konsequenz aus einer Wissensbasis nur dann gefolgert, wenn sie in *allen* Interpretationen welche die Wissensbasis erfüllen (sog. Modelle) gültig ist. Eine Interpretation erfüllt zum Beispiel eine terminologische Implikation, wie, „Alle Ärzte haben studiert“, wenn jeder Knoten des Graphen mit der Beschriftung *Arzt* auch die Beschriftung *studiert* enthält. Da Defeasible Implikationen aber individuell und konfliktfrei von Objekten erfüllt sein müssen, bedarf es der Definition einer neuen Semantik. Dazu erweitere ich die klassischen Interpretationen für $\mathcal{EL}_\bot$, und ergänze jedes Element einer Interpretation um *typischere Elemente* gleichen Typs. Diese sind individuell mit einer konsistenten Teilmenge der Defeasible Implikationen aus der Wissensbasis assoziiert (vgl. Stärke), dabei gilt, je größer die assoziierte Menge, als umso typischer wird das Element verstanden. Die resultierenden Strukturen nenne ich *Typicality Interpretationen*. Ich definieren eine solche Interpretation genau dann als *Typicality Modell*, wenn jedes Element, *sämtliches* striktes Wissen, und (mindestens) das Defeasible Wissen mit dem es assoziiert ist, erfüllt. Eine *typische Konsequenz* gilt in einer Menge von Typicality Modellen dann, wenn die entsprechenden *typischsten Elemente* in den Interpretationen sie Erfüllen.

Im ersten Schritt zeige ich, dass die typischen Konsequenzen, welche in *allen* Typicality Modellen erfüllt sind, exakt den Folgerungen aus dem Materialisierungs-Framework entsprechen (jeweils für Rational und Relevant Stärke). Dieses Ergebnis ist erwünscht, da es Ziel war eine Semantik zu entwerfen, welche die bisherigen Resultate strikt erweitern kann, um die Mindestanforderung der RC von Lehmann und Magidor zu erfüllen. Folglich ist die Reichweite dieser Semantik weiterhin *propositional* und das Problem des Quantification Neglect bleibt ungelöst. Der Grund dafür dass Quantification Neglect auftritt, ist, dass die Menge *aller* Typicality Modelle auch Interpretationen enthält in denen die Anfrage Objekte (z.B. typische Arzt-Elemente) ausschließlich relationale Verbindungen mit *untypischen* Elementen haben. Für diese untypischen Elemente (bspw. Fieber) ist per Definition nicht erforderlich irgendwelche typischen Eigenschaften (z.B. „Fieber impliziert typischerweise Grippe“) zu erfüllen. Die Menge *aller* Typicality Modelle wird deshalb als *nicht typisch genug* bezeichnet.

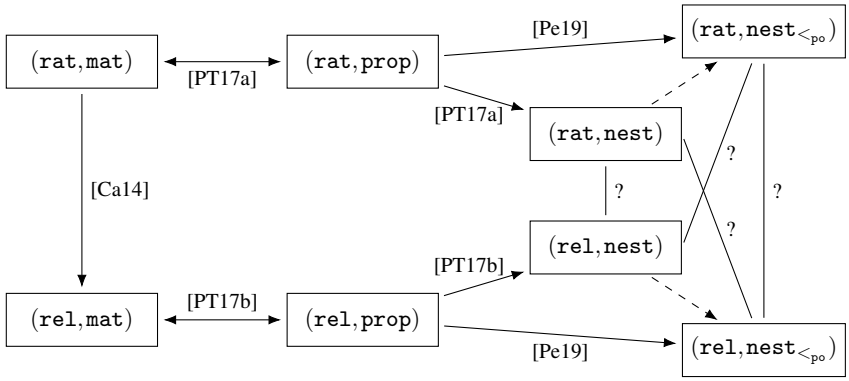


Abb. 2: Implikationen zwischen den Inferenzen der verschiedenen Optionen für das Schließen unter Nested Rational (`rat`) und Relevant (`rel`) Sematiken. Ein Pfeil beschreibt dass die Folgerungen der Semantik seines Ursprungs in der Semantik seines Ziels erhalten bleiben. Ein durchgezogener Pfeil ist in der Dissertation bewiesen, gestrichelte Pfeile sind triviale Konsequenzen, nicht vorhandene Verbindungen sind in meiner Arbeit als ungültig bewiesen (abgesehen von trivialen transitiven Implikationen) und die ungerichteten Verbindungen (markiert mit *?*) sind bislang unbekannt.

Um eine typischere Menge von Modellen zu ermitteln, wollen wir Modelle, wie das zuvor beschriebene, ausschließen. Damit wird indirekt erreicht, dass nur noch Modelle mit relationalen Verbindungen zu *typischeren Elementen* verbleiben und somit typische Eigenschaften für quantifizierte Konzepte gefolgert werden können. Um dies formal zu erreichen definiere ich eine Präferenzrelation, welche Mengen von Modellen, hinsichtlich ihrer unterstützten Konsequenzen typischer Eigenschaften von *relational Verbundenen* Elementen, vergleicht. Diese Präferenzrelation organisiert unterschiedlich typische Mengen von Modellen in einem azyklischen gerichteten Graph, mit der Menge *aller* Typicality Modelle als alleinige Wurzel (s. Abb. 1). Daraus ist ersichtlich, dass die Präferenzrelation, wie gewünscht, einen geeigneten Ausschluss untypischer Modelle erreicht, sowie, dass eine Vielzahl an möglichen *maximal typischen Mengen von Modellen* (MaxTM) existiert.

Ich nutze die Vielzahl an MaxTM um zwei verschiedenartige Semantiken zu definieren, welche beide eine größere Reichweite haben, als wenn man alle Modelle verwendet (`prop`). Diese erhöhte Reichweite von typischer Information, nenne ich *Nested Coverage*, da typische Eigenschaften für, hinter Quantifikation verschachtelte (*nested*) Konzepte gefolgert werden können. Die sogenannte *skeptische Semantik* (identifiziert durch `nest`) ergibt eine Schlussfolgerung nur dann, wenn sie von allen MaxTM erfüllt ist. Dadurch dass einer Verzweigung in der Präferenzrelation von Modellen (potentiell) jeweils widersprüchliche typische Eigenschaften zugrunde liegen, ist die Vereinigung aller MaxTM selbst nicht mehr maximal typisch[5] (bzgl. der Präferenzrelation). Daher führe ich außerdem die *selektive Semantik* ($\texttt{nest}_{<_{po}}$) ein. Sie erlaubt es durch einen ergänzenden Parameter ($<_{po}$) eine der MaxTM einzigartig auszuwählen und somit kohärente maximal typische Konsequenzen zu ermitteln.

[5] Es sei denn es gibt eine einzigartige MaxTM.

	Rational	Relevant
Propositional	P-complete	in Δ_2^P
Selective nested	P-complete	in EXP
Sceptical nested	CO-NP-complete	in CO-NEXP

Tab. 2: Komplexität der Berechnung nichtmonotoner Subsumption und Instance Checking in $\mathcal{EL}_\perp$.

3.1 Ergebnisse

Die neuen Semantiken sind nachweislich in der Lage typische Informationen für referenzierte Objekte zu ermitteln und erweitern das Spektrum bekannter Semantiken in bedeutsamer Weise. In einem einfachen Beispiel kann, aus der typischen Ursache von Fieber, nun für einen Arzt mit einem fiebrigen Patienten gefolgert werden, dass er im typischen Fall einen Patienten mit einer Grippe zu behandeln hat. Die Präferenzrelation erreicht allerdings eine uniforme Steigerung typischer Konsequenzen, sodass auch aus beliebig tief verschachtelten quantifizierten Konzepten, typische Eigenschaften ermittelt werden können. Im allgemeinen ermöglichen beide Nested Semantiken brauchbarere nichtmonotone Schlussfolgerungen, für jede Anwendungsdomäne die auf relationale Strukturen aufbaut. Die Kombination von Semantiken mit erhöhter Reichweite und der Auswahl von konsistenten Defeasible Implikationen mit Stärke der ReC, löst dabei gleichzeitig das bekannte Problem des Inheritance Blocking. Die Erweiterung der Konsequenzen aus Rational und Relevant Closure mithilfe der neuen Semantiken ist in Abbildung 2 verdeutlicht.

Die Trennung der bedingenden Komponenten für Stärke und Reichweite meiner Semantiken bauen erneut ein erweiterbares Framework auf. Mit dieser Separierung war es mir möglich einen uniformen Algorithmus zu entwerfen, welcher Konsequenzen unter jeder der insgesamt 4 (neuen) Semantiken ermitteln kann. Dieser Algorithmus bietet die Grundlage für eine Modular erweiterbare Implementierung meines Ansatzes, sowie ihrer praktischen Evaluation, in weiterführenden Forschungsprojekten. In meiner Arbeit hat er bereits ermöglicht, für die Berechnungskomplexität jeder der neuen Semantiken eine obere Schranke zu bestimmen (s. Tab. 2).

4 Bedeutung

Bereiche wie die Medizin machen schon heute Gebrauch von logischem Schließen mit BL. Unvollständiges Wissen durch geeignete Standardannahmen zu ergänzen, erweitert den Handlungsspielraum für nicht vollständig klassifizierte/klassifizierbare Systeme und autonome Agenten. Dennoch werden nichtmonotone Semantiken in der Praxis bislang kaum eingesetzt. Darüber hinaus sind die in der Forschung etablierten Ansätze, wie Materialisierung, unzulänglich und erreichen nicht die gleiche Ausdrucksstärke von typischen und klassischen Konsequenzen. Mit meinen neuen Semantiken, die das gesamte Spektrum der Ausdrucksstärke von BL mit nichtmonotonen Konsequenzen bereichert, besteht eine solide Grundlage um diese Form des logischen Schließens erfolgreich in die Praxis zu tragen.

Literaturverzeichnis

[Bo15] Bonatti, Piero A.; Faella, Marco; Petrova, Iliana M.; Sauro, Luigi: A New Semantics for Overriding in Description Logics. Artificial Intelligence, AIJ, 222:1–48, 2015.

[Ca14] Casini, Giovanni; Meyer, Thomas; Moodley, Kodylan; Nortje, Riku: Relevant Closure: A New Form of Defeasible Reasoning for Description Logics. In: Logics in Artificial Intelligence - 14th European Conference, JELIA. S. 92–106, 2014.

[CMS18] Casini, Giovanni; Meyer, Thomas; Straccia, Umberto: A polynomial Time Subsumption Algorithm for Nominal Safe $\mathcal{ELO}_\bot$ under Rational Closure. Information Sciences, 2018.

[CS10] Casini, Giovanni; Straccia, Umberto: Rational Closure for Defeasible Description Logics. In: Logics in Artificial Intelligence - 12th European Conference, JELIA. S. 77–90, 2010.

[CS12] Casini, Giovanni; Straccia, Umberto: Lexicographic Closure for Defeasible Description Logics. In: Proceedings of the 8th Australasian Ontology Workshop. S. 28–39, 2012.

[KLM90] Kraus, Sarit; Lehmann, Daniel J.; Magidor, Menachem: Nonmonotonic Reasoning, Preferential Models and Cumulative Logics. Artificial Intelligence, AIJ, 44(1-2):167–207, 1990.

[Nu01] Nute, Donald: Defeasible Logic. In: Web Knowledge Management and Decision Support, 14th International Conference on Applications of Prolog, INAP. S. 151–169, 2001.

[Pe19] Pensel, Maximilian: A Lightweight Defeasible Description Logic in Depth - Quantification in Rational Reasoning and Beyond. Dissertation, Dresden University of Technology, Germany, 2019.

[PT17a] Pensel, Maximilian; Turhan, Anni-Yasmin: Including Quantification in Defeasible Reasoning for the Description Logic $\mathcal{EL}_\bot$. In: Logic Programming and Nonmonotonic Reasoning, 14th Int. Conference, LPNMR. Springer, S. 78–84, 2017.

[PT17b] Pensel, Maximilian; Turhan, Anni-Yasmin: Making Quantification Relevant again—the Case of Defeasible $\mathcal{EL}_\bot$. In: Proceedings of the 4th Int. Workshop on Defeasible and Ampliative Reasoning - DARe. CEUR-WS.org, 2017.

[PT18] Pensel, Maximilian; Turhan, Anni-Yasmin: Reasoning in the Defeasible Description Logic $\mathcal{EL}_\bot$—Computing Standard Inferences under Rational and Relevant Semantics. Int. Journal of Approximate Reasoning, 103:28 – 70, 2018.

[Sp] Spackman, Kent: Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT. Journal of the American Medical Informatics Association. Fall Symposium Special Issue.

Maximilian Pensel, geboren am 3. März 1990, begann 2009 sein Informatikstudium an der Technischen Universität Dresden. Im Masterstudium vertiefte er sich in dem Gebiet der formalen Wissensrepräsentation, insbesondere mit Beschreibungslogiken. Mit seiner, für den *Outstanding Master Thesis Award* des VCLA nominierten Masterarbeit lieferte er seinen ersten theoretischen Forschungsbeitrag. Mit Beginn seiner Promotion 2015, wechselte er thematisch in das Gebiet des nichtmonotonen Schließens. Nach mehreren publizierten Weiterentwicklungen und Verfeinerungen seines Ansatzes zu dieser Form des logischen Schließen, inklusive einer Journal Publikation im *International Journal for Approximate Reasoning* (2018), schloss er seine Promotion im Oktober 2019 mit Auszeichnung ab.

Synchronisierte Pushdown Systeme für Pointer- und Datenfluss-Beziehungen[1]

Johannes Späth[2]

Abstract: Statische Datenflussanalysen untersuchen das Verhalten von Software ohne die Software dabei auszuführen. Eine statische Analyse analysiert alle möglichen Programmpfade und eignet sich damit um Fehler im Programm oder Sicherheitslücken aufzudecken. Allerdings ist es eine Herausforderung, die Analysen präzise und effizient zu designen. Präzise Analysen sind notwendig, um unnötige falsche Warnungen zu vermeiden. Gleichzeitig soll die Analyse effizient sein, sodass sie am Besten direkt in der Entwicklungsumgebung eingesetzt werden kann. Diese Arbeit präsentiert Synchronisierte Pushdown Systeme (SPDS) und auf SPDS aufbauende Entwicklungen, die präzise und effiziente statische Datenflussanalysen ermöglichen.

1 Einführung

Software bestimmt unserem Alltag. Jährlich werden weltweit geschätzt mehr als 111 Millionen Zeilen Quellcode geschrieben. Bei der großen Menge an Code ist es essentiell, dass Qualitäts- und Sicherheitsanalysen der Software automatisiert durchgeführt werden.

Statische Codeanalysen eignen sich dazu. Sie führen den Code nicht aus, sondern modellieren das Verhalten der Software anhand des Quell- oder Bytecodes. Klassische statische Analysen sind Analysen zum Auffinden von SQL Injection (Taint Analysen) oder sogenannte Typestate-Analysen. Typestate-Analysen werden genutzt, um Fehlbenutzungen von Application Programming Interfaces (APIs) zu erkennen.

Damit die Werkzeuge zur Anwendung kommen, ist es notwendig, dass die Analysen (a) präzise sind, also keine unnötigen falschen Warnungen produzieren, (b) nicht-triviale Fehler finden, und (c) Ergenisse am Besten effizient [Jo13] liefern. Falsche Warnungen lenken Entwickler unnötig ab und führen dazu, dass die Werkzeuge langfristig nicht eingesetzt werden. Zum anderen möchten Entwickler/innen Integrationen der Analyse direkt in der Entwicklungsumgebung.

Eine präzise Datenflussanalyse transformiert das Programm in eine *kontext-*, *fluss-* und *feld-sensitive* Approximation der Software. Eine Datenflussanalyse mit dieser Präzision ist unentscheidbar [Ri53] und jeder Algorithmus muss die Lösung approximieren. Wenn der Algorithmus überapproximiert, dann enthalten die Ergebnisse potentiell Falsch Positive. Nutzt der Algorithmus hingegen eine Unterapproximation, dann werden nicht alle Fehler im Programm aufgedeckt (Falsch Negative).

[1] Englischer Titel der Dissertation: "Synchronized Pushdown Systems for Pointer- and Data-Flow Analysis"

[2] Universität Paderborn, johannes.spaeth@uni-paderborn.de

cc by sa

Kontext-Sensitivität bedeutet, dass die statische Analyse Aufrufkontexte modelliert und den Datenfluss am Ende einer Methode an der ursprünglichen Aufrufinstruktion weiterpropagiert. Eine fluss-sensitive Analyse beachtet die Reihenfolge des Kontrollflusses. Eine feld-sensitive Analyse ist bei der Analyse von objekt-orientierten Programmiersprachen relevant, in denen Objekte (transitiv) in Feldern[3] von anderen Objekten gespeichert werden können.

Die Arbeit Synchronized Pushdown System [Sp19, **?**] präsentiert eine neue Approximation für kontext-, fluss- und feld-sensitive Datenflussanalysen. Die Lösung, die wir mit synchronisierte Pushdown Systeme (SPDS) bezeichnen, berechnet Ergebnisse von Datenflussanalysen präzise und effizient. SPDS stützt sich dazu auf zwei Pushdown Systeme. Ein System modelliert Feld-Sensitivität, das andere Kontext-Sensitivität. SPDS akzeptieren ein Ergebnis nur, wenn beide Systeme das Ergebnis akzeptieren. Dies liefert ein entscheidbares Verfahren mit einer Überapproximation. Die Arbeit zeigt, dass SPDS in der Praxis allerdings nur in seltenen Spezialfällen an Präzision verlieren: in Fällen, in denen beide Systeme das Ergebnis akzeptieren, aber nicht entlang des gleichen Datenflusspfades. Experimente, die SPDS mit dem Standardmodell für Feld-Sensitivität (Access Path der Länge k) vergleichen, zeigen, dass SPDS fast so effizient sind wie Access Path der Länge $k = 1$, obwohl SPDS so präzise sind wie k-limiting mit $k = \infty$.

Eine statische Datenflussanalyse muss außerdem die dynamische Heap-Datenstruktur modellieren und Pointer-Beziehungen auflösen. Pointer-Beziehungen sind nicht überall distributiv[4] und daher ineffizient zu berechnen. Diese Arbeit zeigt, dass Pointer-Beziehungen in mehrere distributive Teilprobleme unterteilt werden können. Jedes einzelne Teilproblem kann effizient mit SPDS gelöst werden. Basierend auf diesem Design stellt die Dissertation die Pointer-Analyse BOOMERANG vor. BOOMERANG minimiert den Rechenaufwand, indem es nur den minimalen Teil der Pointer-Beziehungen auf Anfrage berechnet.

Weiterhin präsentiert diese Arbeit IDE^{al}, ein Rahmenwerk für Datenflussanalysen das angewendet werden kann um Typestate-Analysen zu realisieren. Typestate-Analysen detektieren, wenn ein Programm API-Aufrufe in der falschen Reihenfolge tätig. IDE^{al} muss dabei ebenfalls Pointer-Beziehungen auflösen und benutzt dazu BOOMERANG.

Die neuen Datenflussanalysentechniken SPDS, BOOMERANG und IDE^{al} wurden im Rahmen der Arbeit vollständig implementiert und sind Open-Source verfügbar[5]. In die Anwendung kommen die Analysen in CogniCrypt, einem Eclipse-Plugin, das Fehlbenutzungen von Kryptographie direkt in der Entwicklungsumgebung darstellt. In einem großen Experiment auf Maven Central, ein Software Repository mit mehr als 4.5 Millionen Softwareartefakten, wird die Praxistauglichkeit und Effizienz der Analysen zusätzlich nachgewiesen.

[3] häufig auch als Objektvariable, Attribut oder Instanzvariable bezeichnet.
[4] Der Effekt einer einzelnen Instruktion auf den Datenfluss hängt von weiteren Instruktionen im Programm ab.
[5] `https://github.com/CROSSINGTUD/WPDS/`

2 Beispiel

In diesem Abschnitt präsentieren wir ein Beispiel und demonstrieren die Komplexität von statischen Codeanalysen. Die Java Cryptographic Architecture[6] (JCA) ist die Standardbibliothek für Kryptographie in Java. Es ist leicht die API falsch zu benutzen, unter anderem, da Konfiguration über String-Parameter vorgenommen werden.

```
1  public class Encrypter{
2    SecretKey key;
3    Cipher cipher;
4
5    public Encrypter(int keyLength){
6      KeyGenerator keygen = KeyGenerator.getInstance("Blowfish");
7      keygen.init(keyLength);
8      this.key = keygen.generateKey();
9    }
10
11   public byte[] encrypt(String plainText){
12     this.cipher = Cipher.getInstance("AES");
13     //this.cipher.init(Cipher.ENCRYPT_MODE, this.key);
14     byte[] encText = this.cipher.doFinal(plainText.getBytes());
15     return encText;
16   }
17
18   public static void main(String....args){
19     Encrypter enc = new Encrypter(448);
20     enc.encrypt(args[0]);
21   }
22 }
```

Abb. 1: Fehlbenutzung der Java Cryptographic Architecture API anhand eines Beispiels.

Das Beispiel in Abb. 1 präsentiert die Klasse `Encrypter`. Die Klasse verwendet die JCA. Der Konstruktor generiert in Zeile 6 einen `SecretKey` für den Algorithmus `"Blowfish"`. Die Schlüssellänge wird auf 448 festgelegt (Zeile 7 und 19). In Zeile 8 wird der Schlüssel in das Feld `key` des instanziierten Objektes geschrieben. Die `encrypt()` Methode erstellt ein `Cipher` Objekt in Zeile 12. Das Objekt wird über eine Factory-Methode mit dem `"AES"` Algorithm instanziiert (Zeile 12). Das Beispiel beinhaltet weiterhin eine auskommentierte Zeile, die (a) den Modus der Verschlüsselung initiiert und (b) das `SecretKey` Objekt aus dem Feld `key` lädt und an das `Cipher` Objekt übergibt. Der Aufruf `doFinal()` in Zeile 14 führt die Verschlüsselung der Daten durch und verschlüsselt den Inhalt von `plainText` und gibt das Ergebnis zurück.

Das Codebeispiel enthält drei Fehlbenutzungen der API. Erstens, ein notwendiger Methodenaufruf in Zeile 13 ist auskommentiert. Zweitens, selbst wenn die Zeile einkommentiert ist, stimmen die Algorithmen des Schlüssels (`"Blowfish"`) und der Verschlüsselung (`"AES"`) nicht überein. Drittens ist die Schlüssellänge von 448 für AES nicht zulässig.

[6] `https://docs.oracle.com/javase/8/docs/technotes/guides/security/crypto/CryptoSpec.html`

In dem Beispiel sind drei herausfordernde Programmmuster für statische Analysen enthalten.

1. Datenfluss entlang von Variablenzuweisungen: Die Schlüssellänge wird als Parameter an den Konstruktor übergeben und durch das Programm gereicht.
2. Datenfluss entlang von Feldzugriffen: Der Schlüssel wird als Rückgabewert in Zeile 8 in Feld `key` geschrieben und in Zeile 13 (sofern einkommentiert) geladen. Das Feld ist nicht als `final` deklariert und kann innerhalb der Klasse überschrieben und modifiziert werden.
3. Aliasing: Um den fehlenden Methodenaufruf zu detektieren muss die statische Analyse zunächst erkennen, dass das Feld `cipher` an den Aufrufen in Zeile 12 und 14 das selbe Objekt referenziert.

Während das Beispiel den Fokus auf die Fehlbenutzung von Kryptographie legt, treten die gleichen Herausforderungen bei weiteren statischen Datenflussanalysen auf. Zum Beispiel bei Analysen um Null-Pointer-Dereference, Ressourcenlecks oder SQL Injections zu erkennen.

3 Synchronisierte Pushdown Systeme

SPDS lösen die beiden erstgenannten Herausforderungen. Die Arbeit zu SPDS wurde auf der Konferenz POPL 2019 vorgestellt [SAB19] und wurde mit einen Distinguished Paper Award ausgezeichnet.

3.1 Pushdown System für Aufrufkontexte

Das Pushdown System für Aufrufkontexte modelliert den Datenfluss entlang Variablenzuweisungen und Methodenaufrufen.

Definition 1. *Ein* Pushdown System *ist ein Triple* $\mathcal{P} = (P, \Gamma, \Delta)$ *mit endlichen Mengen* P *und* Γ. P *sind die* Kontrollpositionen *und* Γ *ist das* Stackalphabet. *Eine* Konfiguration *ist ein Paar* $\langle\!\langle p, w \rangle\!\rangle$, *mit* $p \in P$ *und* $w \in \Gamma^*$, *also eine Kontrollposition mit einer Sequenz von Stackelementen. Die endliche Menge* Δ *beschreibt die* Regeln *des Pushdown Systems. Eine Regel hat die Form* $\langle\!\langle p, \gamma \rangle\!\rangle \to \langle\!\langle p', w \rangle\!\rangle$, *mit* $p, p' \in P$, $\gamma \in \Gamma$ *und* $w \in \Gamma^*$. *Abhängig von der Länge von w unterscheidet ein Pushdown System zwischen Push-Regeln (*$|w| = 2$*), Pop-Regeln (*$|w| = 0$, *i.e.* $w = \varepsilon$*) und normalen Regeln (*$|w| = 1$*).*

Die Regeln eines Pushdown Systems definieren eine Relation $\Rightarrow$ zwischen den Konfigurationen von $\mathcal{P}$: Existiert eine Regel $\langle\!\langle p, \gamma \rangle\!\rangle \to \langle\!\langle p', w \rangle\!\rangle$, dann gilt $\langle\!\langle p, \gamma w' \rangle\!\rangle \Rightarrow \langle\!\langle p', w w' \rangle\!\rangle$ für alle $w' \in \Gamma^*$. Ausgehend von einer Startkonfiguration c ist die transitive Hülle ($\Rightarrow^*$) eine Menge an erreichbaren Konfigurationen, die mit $post^*(c) = \{c' \mid c \Rightarrow^* c'\}$ bezeichnet wird.

Diese Menge lässt sich in einem endlichen Automaten beschreiben. Ist ein Pushdown System gegeben, lässt sich durch den Algorithmus post* ein Automat konstruieren, der die Menge repräsentiert.

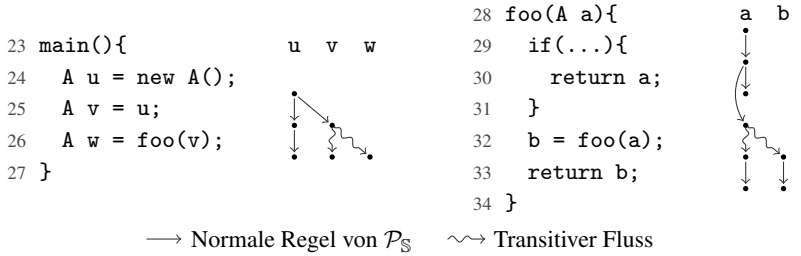


Abb. 2: Beispiel eines Datenfluss anhand eines rekursiven Programms.

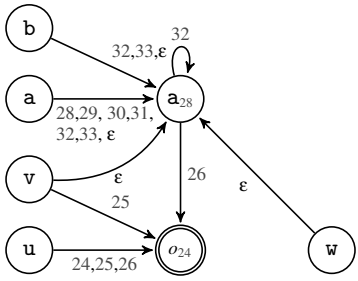


Abb. 3: Der Automat $\mathcal{A}_\mathbb{S}$ nach post*-Sättigung mit $\mathcal{P}_\mathbb{S}$.

Beispiel 1. Abb. 2 zeigt ein Programm, das ein Objekt (o_{24}) erstellt und per Variablenzuweisung und Methodenparameter durch das Programm reicht. Neben dem Programm ist das Pushdown System $\mathcal{P}_\mathbb{S}$ als gerichteter Graph visualisiert. Die Kanten im Graph repräsentieren die normalen Regeln in $\mathcal{P}_\mathbb{S}$. Zusätzlich gibt es noch weitere Regeln, beispielsweise die Push-Regel $\langle\!\langle \mathtt{v}, 25 \rangle\!\rangle \to \langle\!\langle \mathtt{a}, 28 \cdot 26 \rangle\!\rangle$ oder die Pop-Regel $\langle\!\langle \mathtt{b}, 33 \rangle\!\rangle \to \langle\!\langle \mathtt{w}, \varepsilon \rangle\!\rangle$.

Abb. 3 zeigt den post*-gesättigten Automaten $\mathcal{A}_\mathbb{S}$, der die Erreichbarkeit des Objektes aus Zeile 24 (o_{24}), basierend auf $\mathcal{P}_\mathbb{S}$, enkodiert. Aus dem Automaten lässt sich ablesen, dass Variable `a` in Zeile 32 unter Callstack 26 auf o_{24} zeigt. Durch die Rekursion im Programm liegen möglicherweise beliebig viele Wiederholungen des Aufrufes in Zeile 32 auf dem Callstack, entsprechend enthält der Automat eine Schleife.

3.2 Pushdown System für Felder

Das Pushdown System $\mathcal{P}_\mathbb{S}$ ist feld-insensitiv und überapproximiert Feldzugriffe.

Mit einem Pushdown System lässt sich ebenfalls der Effekt von Schreibe- und Lesezugriffen auf Feldern modellieren kann. Feldzugriffe ermöglichen es, Objekte hierarchisch (und zyklisch) durch Felder in Beziehung zueinander zu setzen. Um die dynamischen Objektabhängigkeiten zu modellieren, verwenden statische Analyse typischerweise das Access Path-Modell [De94] verwendet. Diese Approximation muss die Länge eines jeden Access

Paths auf k beschränken und anschließend entweder über- oder unterapproximieren. Kleinere Werte von k liefern höhere Effizienz, reduzieren allerdings die Präzision. Typische Werte sind $k = 1, \ldots, 3$. Das Pushdown System für Felder verhindert diese unnötige Approximation und liefert eine kompaktere Darstellung die effizient und präzise zugleich ist. In Experimenten zeigt die Dissertation, dass das Pushdown System für Felder Ergebnisse fast so effizient liefert wie das Access Path Modell $k = 1$ und dabei so präzise ist wie $k = \infty$.

Definition 2. *Das* PDS für Felder *ist das Pushdown System* $\mathcal{P}_\mathbb{F} = (\mathbb{V} \times \mathbb{S}, \mathbb{F} \cup \{\varepsilon\}, \Delta_\mathbb{F})$. *Eine Kontrollposition von* $\mathcal{P}_\mathbb{F}$ *ist ein Paar aus einer Programmvariablen und einer Programminstruktion. Wir benutzen* $x@s$ *für ein Element* $(x, s) \in \mathbb{V} \times \mathbb{S}$. $\mathcal{P}_\mathbb{F}$ *legt und entfernt Elemente auf einen Stack, der von Elementen aus* $\mathbb{F}$ *besteht. Der leere Stack wird durch* ε *repräsentiert.*

Eine Konfiguration von $\mathcal{P}_\mathbb{F}$ schreiben wir als $\langle\!\langle x@s, f_0 \cdot f_1 \cdot \ldots f_n \rangle\!\rangle$. Diese Konfiguration kann wie folgt gelesen werden: An Instruktion s ist der Datenfluss über den Access Path $x.f_0 \cdot f_1 \cdot \ldots f_n$ erreichbar.

Die Regeln von $\mathcal{P}_\mathbb{F}$ sind ähnlich zu den Regeln von $\mathcal{P}_\mathbb{S}$, unterscheiden sich allerdings an Instruktionen, die Felder lesen oder schreiben. Ein Schreibzugriff generiert dabei eine Push-Regel, eine Lesezugriff eine Pop-Regel.

```
35 foo(u, v, w){
36   v.f = u;
37   x = w.f;
38   if(...){
39     w.g = v;
40   } else {
41     w.h = v;
42   }
43 }
```

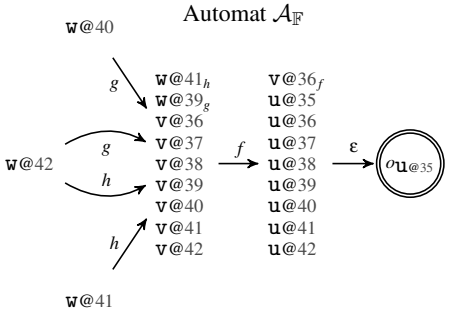


Abb. 4: Programmbeispiel und Automat $\mathcal{A}_\mathbb{F}$. $\mathcal{A}_\mathbb{F}$ wurde ausgehend von der Konfiguration $\langle\!\langle \mathtt{u@36}, \varepsilon \rangle\!\rangle$ mit den Regeln Tabelle 1 konstruiert.

Tab. 1: Die Push- und Pop-Regeln $\Delta_\mathbb{F}$ von $\mathcal{P}_\mathbb{F}$ für den Programmcode in Abb. 4. In den Push-Regeln ist $(*)$ ein Wildcard, für jedes beliebige Feld $g \in \mathbb{F}$ existiert eine entsprechende Regel in $\mathcal{P}_\mathbb{F}$.

Push-Regeln	Pop-Regel
$\langle\!\langle \mathtt{u@35}, * \rangle\!\rangle \to \langle\!\langle \mathtt{v@36}, \mathtt{f} \cdot * \rangle\!\rangle$	$\langle\!\langle \mathtt{w@36}, \mathtt{f} \rangle\!\rangle \to \langle\!\langle \mathtt{x@37}, \varepsilon \rangle\!\rangle$
$\langle\!\langle \mathtt{v@38}, * \rangle\!\rangle \to \langle\!\langle \mathtt{w@39}, \mathtt{g} \cdot * \rangle\!\rangle$	
$\langle\!\langle \mathtt{v@40}, * \rangle\!\rangle \to \langle\!\langle \mathtt{w@41}, \mathtt{h} \cdot * \rangle\!\rangle$	

Beispiel 2. In Abb. 4 ist ein Programmcode mit drei schreibenden Feldzugriffen und einem ladenden Zugriff dargestellt. Die Push- und Pop-Regeln von $\mathcal{P}_\mathbb{F}$ sind in Tabelle 1 gelistet. Ausgehend von der Startkonfiguration $\langle\!\langle \mathtt{u@36}, \varepsilon \rangle\!\rangle$ berechnet der Algorithmus post* den in

Abb. 4 dargestellten Automaten. Der Automat beinhaltet feld- und fluss-sensitive Ergebnisse. Aus dem Automaten lässt sich ablesen, dass das in Variable `u` gespeicherte Objekt sowohl durch Access Path `w.g.f` als auch durch `w.h.f` erreichbar ist. Da der Automat keinen Knoten enthält, der Variable `x` enkodiert lässt sich schlussfolgern, dass `x` nicht auf das gleiche Objekt wie `u` zeigt.

3.3 Synchronisation der beiden Pushdown Systeme

Das Pushdown System für Aufrufkontexte ist feld-*in*sensitiv, das Pushdown System für Felder ist kontext-*in*sensitiv. Durch die Synchronisation der beiden Systeme ist es möglich die Impräzisionen der einzelnen Systeme gegeneinander auszuhebeln.

Definition 3. *Sei* $\mathcal{P}_\mathbb{S} = (\mathbb{V}, \mathbb{S}, \Delta_\mathbb{S})$ *das PDS der Aufrufkontexte und* $\mathcal{P}_\mathbb{F} = (\mathbb{V} \times \mathbb{S}, \mathbb{F} \cup \{\varepsilon\}, \Delta_\mathbb{F})$ *das PDS der Felder, dann ist das Tupel* SPDS $= (\mathbb{V}, \mathbb{S}, \mathbb{F} \cup \{\varepsilon\}, \Delta_\mathbb{F}, \Delta_\mathbb{S})$ *das* Synchronisierte Pushdown System*. Eine Konfiguration von SPDS erweitert die Konfiguration jedes Systems: Eine* synchronisierte Konfiguration *ist ein Tupel* $(v, s, f) \in \mathbb{V} \times \mathbb{S}^+ \times \mathbb{F}^*$*, wobei wir auch* $\langle\!\langle v.f_0 \cdot \ldots \cdot f_m @ s_0^{s_1 \cdot \ldots \cdot s_n} \rangle\!\rangle$ *mit* $s = s_0 \cdot s_1 \cdot \ldots \cdot s_n$ *und* $f = f_0 \cdot \ldots f_m$ *schreiben. Die erreichbaren synchronisierten Konfigurationen basierend einer Startkonfiguration* $c = \langle\!\langle v.f_0 \cdot \ldots \cdot f_m @ s_0^{s_1 \cdot \ldots \cdot s_n} \rangle\!\rangle$ *sind dann gegeben durch*

$$post_{\mathbb{SF}}(c) = \{\langle\!\langle w.g @ t_0^{t_1 \cdot \ldots t_n} \rangle\!\rangle \mid \langle\!\langle w @ t_0, g \rangle\!\rangle \in post^*_\mathbb{F}(\langle\!\langle v @ s_0, f \rangle\!\rangle) \wedge \langle\!\langle w, t \rangle\!\rangle \in post^*_\mathbb{S}(\langle\!\langle v, s \rangle\!\rangle)\}. \quad (1)$$

Intuitiv gesprochen ist eine Konfiguration im synchronisierten Pushdown System akzeptiert, wenn sowohl $\mathcal{A}_\mathbb{F}$ also auch $\mathcal{A}_\mathbb{S}$ die Konfiguration akzeptieren.

```
44 bar(u, v){
45   v.h = u;
46   w = foo(v);
47   x = w.g;
48   y = x.f;
49 }

50 foo(p){
51   q.g = p;
52   return q;
53 }
```

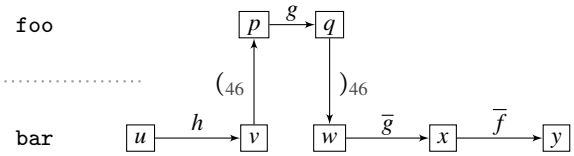


Abb. 5: Ein Codebeispiel und eine Darstellung des Codes als gerichteter Graph.

Beispiel 3. Abb. 5 zeigt ein Codebeispiel in dem Felder und Aufrufkontexte interagieren. Unter dem Code visualisiert ein gerichteter Graph den Datenfluss: Vertikale Kanten sind Aufrufkontexte (Push-Regel ≙ öffnende Klammer / Pop-Regel ≙ schließende Klammer), horizontale Kanten entsprechen schreibenden und ladenden Feldzugriffen. Für ladende

Feldzugriffe hat das Feld einen Strich. SPDS berechnet die Erreichbarkeit in diesem Graphen[7]. SPDS berechnet, dass Variable `y` *nicht* auf den Inhalt von `u`@44 zeigen kann, denn das Wort entlang des Pfades ist $h \cdot (_{46} \cdot g \cdot)_{46} \cdot \overline{g} \cdot \overline{f}$. Die Klammern $(_{46}$ und $)_{46}$ sind korrekt gepaart und der Pfad ist kontext-sensitiv zulässig. Allerdings ist der Pfad nicht feld-sensitiv zulässig. Der schreibende Zugriff g deckt sich mit dem ladenden Zugriff $\overline{g}$, der schreibende Zugriff auf h hingegen nicht mit $\overline{f}$.

SPDS ist eine Überapproximation und es lassen sich Fälle konstruieren, die unpräzise Ergebnisse liefern. Auf echten Codebeispielen konnten wir diese Fälle allerdings nicht finden. Neben den praktischen Erkenntnissen liefert die Dissertation auch eine Worst-Case Abschätzung der Komplexität von SPDS.

4 Boomerang

SPDS berechnet *keine* Points-To Ergebnisse, denn SPDS propagiert Datenflüsse distributiv. Durch Zugriffe auf Felder ist es möglich den Datenfluss an einer Instruktion indirekt, also nicht distributiv, zu beeinflussen. BOOMERANG ist eine Pointer-Analyse und löst dieses Problem. Dabei beantwortet BOOMERANG Anfragen „on-demand“ und kann genutzt werden, um zu überprüfen, ob zwei Variablen aliasen. Um einzelne Anfragen zu beantworten, verknüpft BOOMERANG vorwärts und rückwärts gerichtete Datenflussanalysen und exploriert dabei nur den minimalen notwendigen Datenflusspfad.

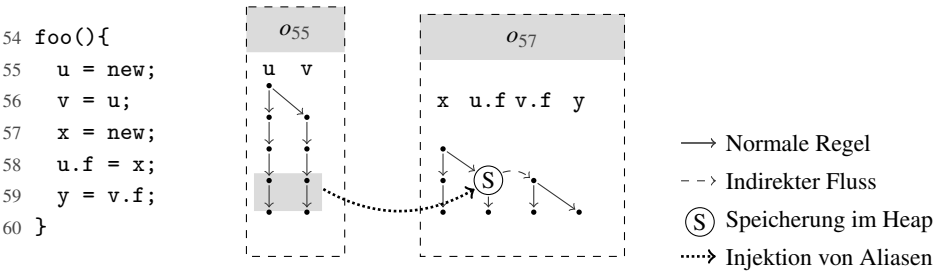


Abb. 6: Nicht-Distributivität von Pointer-Beziehungen.

Beispiel 4. Das Beispiel in Abb. 6 zeigt, dass Pointer-Beziehungen nicht-distributiv sind. Das Programm generiert zwei Objekte o_{55} und o_{57} in den entsprechenden Zeilen 55 und 57. Zur Laufzeit zeigen die Variablen `x` und `y` beide auf das Objekt o_{57}. In Zeile 57 wird das Objekt o_{57} in Access Path `u.f` gespeichert. Zur Laufzeit verändert sich zeitgleich der Access Path `v.f`. SPDS propagiert den Datenfluss in Zeile 57 *nicht* von `x` nach `v.f` und liefert damit keine vollständig korrekten Ergebnisse. BOOMERANG hingegen instanziiert ein SPDS pro Objekt und verbindet indirekte Datenflüsse zwischen verschiedenen SPDS. Dies ist im Beispiel durch den blauen gestrichelten Pfeil angedeutet. Ein Programm erstellt häufig tausende Objekte und ein SPDS pro Objekt skaliert nicht. BOOMERANG ist daher so designed, dass eine Rückwärtsanalyse nur die minimal notwendigen SPDS pro Anfrage berechnet.

[7] Die Automaten $\mathcal{A}_{\mathbb{S}}$ und $\mathcal{A}_{\mathbb{F}}$ sind nicht dargestellt um die Visualisierung zu vereinfachen.

In einem Experiment haben wir die Alias-Strategy der Taint-Analyse FlowDroid [Ar14] durch BOOMERANG ersetzt und konnten die Laufzeit der Taint-Analyse FlowDroid auf 25 Android-Apps im Durchschnitt um den Faktor $0,44\times$ reduzieren.

BOOMERANG wurde auf der ECOOP-Konferenz 2016 in Rom präsentiert [Sp16].

5 Typestate-Analyse mit IDE^{al}

Eine weitere Erweiterung von SPDS ist IDE^{al}. IDE^{al} ist ein Framework zur Datenflussanalyse. IDE^{al} „liftet“ $\mathcal{P}_{\mathbb{S}}$ und ordnet jeder Pushdown-Regel ein Gewicht zu. Die Gewichte müssen Elemente aus einem idempotenten Semiring $(W, \otimes, \oplus)$ sein. Der Operand „*extend*“ $(\otimes : W \times W \to W)$ ist eine Verkettung entlang des Datenflusspfades. Der Operand „*combine*“ $(\oplus : W \times W \to W)$ wird angewendet, wenn zwei unterschiedliche Gewichte entlang verschiedener Kontrollflusspfade vereint werden. Beispielsweise lässt sich mit IDE^{al} eine Typestate-Analyse realisieren. Eine *Typestate-Analyse* erkennt die korrekte Reihenfolge von API Aufrufen auf einem abstrakten Objekt (siehe Abb. 1). Für eine Typestate-Analyse enkodieren die Gewichte den Zustand des Objektes an jeder Programminstruktion. Um korrekte Ergebnisse zu liefern, muss IDE^{al} auch Aliasing berechnen. Für eine Typestate-Analyse für das Programm in Abb. 1 muss die Analyse entsprechend berechnen, dass die Feldzugriffe in Zeile 14 und Zeile 12 das gleiche Objekt referenzieren.

Im Vergleich zu existierenden Typestate-Analysen erhöht die Analyse basierend auf IDE^{al} die Geschwindigkeit um einen Faktor von $3,9\times$ bis $99\times$. Die Präzision bleibt dabei vollständig erhalten. Die Ergebnisse zu IDE^{al} wurden auf der OOPSLA 2017 vorgestellt [SAB17].

6 CogniCrypt – Eine Studie auf Maven Central

In Abb. 1 sind Fehler in der Benutzung der JCA dargestellt. Im Rahmen der Dissertation wurden Analysen entwickelt, die solche und ähnliche Fehler in Java Bytecode entdecken können. Dazu wurde die domänenspezifische Sprache CrySL entwickelt [Kr18]. CrySL eignet sich um Typestate-Analysen und die Interaktionen von Objekten zu spezifizieren. Basierend auf den CrySL-Spezifikation haben wir einen Analyse-Compiler entwickelt. Der Compiler konvertiert eine Spezifikation in eine statische Analyse. Wendet man die Analyse auf ein Program an, warnt sie, wenn das Programm von der spezifizierten Benutzung abweicht.

Die Analyse ist im Werkzeug CogniCrypt[8] integriert. Mit dem Werkzeug haben wir eine Studie auf Maven Central durchgeführt. Mit über 4,5 Millionen Artefakte ist Maven Central das weltweit größte Repository für Java-Bibliotheken. Wir haben die neuesten Versionen der Artefakte (insgesamt 152.996) analysiert. Von den analysierten Artefakten benutzen 8.193 Kryptographie. Im Durchschnitt benötigt die hoch-präzise Analyse 80 Sekunden pro Artefakt. Lediglich 68,3% der Artefakte weisen keine falsche Benutzung der API auf, alle anderen Artefakte beinhalten mindestens einen Fehler.

[8] `https://www.eclipse.org/cognicrypt/`

Literaturverzeichnis

[Ar14] Arzt, Steven; Rasthofer, Siegfried; Fritz, Christian; Bodden, Eric; Bartel, Alexandre; Klein, Jacques; Traon, Yves Le; Octeau, Damien; McDaniel, Patrick: FlowDroid: Precise Context, Flow, Field, Object-Sensitive and Lifecycle-Aware Taint Analysis for Android Apps. In: PLDI. 2014.

[De94] Deutsch, Alain: Interprocedural May-Alias Analysis for Pointers: Beyond k-limiting. In: PLDI. 1994.

[Jo13] Johnson, Brittany; Song, Yoonki; Murphy-Hill, Emerson R.; Bowdidge, Robert W.: Why don't Software Developers use Static Analysis Tools to Find Bugs? In: ICSE. 2013.

[Kr18] Krüger, Stefan; Späth, Johannes; Ali, Karim; Bodden, Eric; Mezini, Mira: CrySL: An Extensible Approach to Validating the Correct Usage of Cryptographic APIs. In: ECOOP. 2018.

[Ri53] Rice, H. G.: Classes of Recursively Enumerable Sets and Their Decision Problems. Transactions of the American Mathematical Society, 1953.

[SAB17] Späth, Johannes; Ali, Karim; Bodden, Eric: IDE^{al}: Efficient and Precise Alias-Aware Dataflow Analysis. In: OOPSLA. 2017.

[SAB19] Späth, Johannes; Ali, Karim; Bodden, Eric: Context-, Flow- and Field-Sensitive Data-Flow Analysis using Synchronized Pushdown Systems. In: POPL. 2019.

[Sp16] Späth, Johannes; Do, Lisa Nguyen Quang; Ali, Karim; Bodden, Eric: Boomerang: Demand-Driven Flow- and Context-Sensitive Pointer Analysis for Java. In: ECOOP. 2016.

[Sp19] Späth, Johannes: Synchronized pushdown systems for pointer and data-flow analysis. Dissertation, University of Paderborn, Germany, 2019.

Johannes Späth wurde am am 2. Februar 1988 geboren und hat Mathematik an der Johannes-Gutenberg Universität in Mainz studiert und anschließend für seinen Master an die Technischen Universität Darmstadt gewechselt. Zwischen März 2014 und Juli 2016 hat Johannes Späth als wissenschaftlicher Mitarbeiter in der Abteilung Secure Software Engineering am Fraunhofer SIT in Darmstadt gearbeitet. Im Anschluss hat Johannes Späth ein Forschungspraktikum bei Oracle Labs in Brisbane absolviert und hat im März 2017 seine Promotion am Fraunhofer IEM in Paderborn fortgeführt. Die Dissertation hat Johannes Späth im Januar 2019 verteidigt. Johannes Späth befindet sich aktuell in der Vorbereitung der Ausgründung CodeShield (www.codeshield.de). Die Ausgründung bringt die neuen Forschungstechnologien in einem Werkzeug für Softwareentwickler in die Praxis.

Neue Graphen-Algorithmen mittels polyedrischer Methoden[1]

Jakub Tarnawski[2]

Abstract: In dieser Arbeit werden neue Algorithmen für zwei grundlegende Graphenprobleme vorgestellt. Mit Hilfe von linearen Programmen – darunter auch Programme exponentieller Größe – werden Struktureigenschaften ermittelt die vom Algorithmus verwendet werden. Etwas überraschend können ähnliche polyedrische Methoden in beiden Graphenproblemen angewendet werden. Der erste Teil der Dissertation widmet sich dem *asymmetrischen Handlungsreisendenproblem* (Asymmetric Traveling Salesman Problem – ATSP), einem Benchmark-Problem der kombinatorischen Optimierung. Bei diesem Problem geht es darum, die kürzeste Tour für einen gerichteten und kantengewichteten Graphen zu finden, die alle Knoten besucht. Seit langem galt es als offen, ob es einen Approximationsalgorithmus für dieses Problem mit einer konstanten Güte gibt. Ein Ergebnis dieser Arbeit ist ein solcher Algorithmus. Der zweite Teil der Dissertation widmet sich dem *perfekten Matching-Problem*. Zudem wurde in den Achtzigerjahren gezeigt, dass es effiziente parallele Algorithmen für das Matching-Problem gibt, sofern die Verwendung von Zufälligkeit zulässig ist. Allerdings ist es noch offen ob das Matching-Problem in der Komplexitätsklasse NC liegt, also ob Zufälligkeit notwending ist. Diese Arbeit zeigt, dass das Matching-Problem in quasi-NC liegt.

1 Einführung

Die Entwicklung und Analyse von Algorithmen ist seit jeher eine zentrale Aufgabe der Informatik. Mit steigenden Datenvolumina wird die rechnerische Effizienz immer wichtiger. Gleichzeitig werden Entscheidungen von immer größerer Tragweite zunehmend von Algorithmen oder anhand der Ergebnisse von Algorithmen getroffen. Unsere Algorithmen müssen daher Lösungen von hoher Qualität liefern. Außerdem benötigen wir ein gutes Verständnis dafür, was diese Algorithmen leisten und wo ihre Grenzen liegen. Eine strikte theoretische Untersuchung effizienten Rechnens wird daher wichtiger als je zuvor. In der vorliegenden Dissertation legen wir den Schwerpunkt auf die Entwicklung von Algorithmen mit beweisbaren Worst-Case-Gütegarantien.

Die Aufgaben, die sich mit Hilfe eines Computers lösen lassen, sind durch die rechnerischen Ressourcen beschränkt, die zur Verfügung stehen. Davon ist Zeit die wichtigste. In der klassischen Sicht auf rechnerische Komplexität gilt ein Algorithmus dann als effizient, wenn sich die Anzahl der Rechenoperationen, die bei einer beliebigen Eingabe durchgeführt werden, durch eine Polynomialfunktion der Größe der Eingabe eingrenzen lässt. Die grundlegende Komplexitätsklasse P umfasst Probleme mit zugehörigen Algorithmen, die im obigen Sinne effizient sind (wir nennen sie „polynomiale Algorithmen“) und immer die richtige Lösung liefern. Solche Probleme gelten als effizient lösbar. Ein Beispiel hierfür ist das *Problem des kürzesten Wegs*: Gegeben seien ein kantengewichteter Graph,

[1] Englischer Titel der Dissertation: ”New Graph Algorithms via Polyhedral Techniques”

[2] Microsoft Research, Zürich, Schweiz. jakub.tarnawski@microsoft.com

CC BY SA

d. h. ein Graph mit Gewichten (Distanzen) entlang seiner Kanten, sowie zwei ausgezeichnete Knoten. Was ist die Länge des kürzesten Wegs, der die beiden Knoten miteinander verbindet?

Bedauerlicherweise geht man heute davon aus, dass eine große und relevante Klasse grundlegender Probleme nicht effizient lösbar ist. Diese Probleme werden *NP-schwere Probleme* genannt. Ein bekanntes Beispiel ist das *Problem des Handlungsreisenden (Traveling Salesman Problem – TSP)*: Gegeben sei ein kantengewichteter Graph. Was ist die Länge des kürzesten Rundwegs, der über alle Knoten des Graphen führt?

Es sind zahlreiche Ansätze zum Umgang mit NP-schweren Problemen entwickelt worden. Zum Beispiel kann man die Anforderung aufgeben, dass der Algorithmus den kürzestmöglichen Rundweg liefern muss, und sich stattdessen mit einem Rundweg zufriedengeben, der nicht viel länger ist als der kürzestmögliche. Dann hätten wir aber gern eine beweisbare Garantie dafür, wie lang dieser Weg höchstens werden kann. Das bringt uns zu dem Konzept der *Approximationsalgorithmen*: Für ein Minimierungs-Problem, wie etwa das TSP, wird ein Algorithmus als *Approximationsalgorithmus der Güte* α bezeichnet, wenn er in Polynomialzeit abläuft und die Länge des zurückgegebenen Rundwegs immer höchstens um einen Faktor α länger als der optimale (kürzeste) Rundweg ist. Die Größe α wird als *Approximationsverhältnis* (oder *Approximationsfaktor*) bezeichnet. Für ein gegebenes Optimierungsproblem lautet dann die Frage: Was ist das bestmögliche Approximationsverhältnis?

Eine besondere Schwierigkeit bei der Analyse von Approximationsalgorithmen besteht darin, für einen gegebenen Algorithmus zu beweisen, dass jede von ihm gelieferte Lösung mit Kosten behaftet ist, die nicht mehr als um einen Faktor α über den optimalen Kosten liegen – obwohl die letzteren unbekannt sind und sich wahrscheinlich weder effizient berechnen lassen, noch einfache Überlegungen dazu möglich sind. Aus diesem Grund suchen wir nach starken unteren Schranken für die Kosten einer beliebigen Lösung. Diese unteren Schranken können dann anstelle der optimalen Werte benutzt werden. Dies lässt sich anhand des Problems des Handlungsreisenden auf ungerichteten Graphen (das so genannte *symmetrische TSP*) mit einem Approximationsalgorithmus veranschaulichen, bei dem der minimale Spannbaum (MST) des Graphen berechnet und jede seiner Kanten doppelt genommen wird. Man sieht leicht, dass dies einen Rundweg ergibt. Zudem enthält der optimale Rundweg, da er den gesamten Graphen erfasst, ebenfalls einen Spannbaum, der mindestens so kostspielig ist wie der MST. Somit sind die Kosten des MST eine untere Grenze für die Kosten des optimalen Rundwegs. Da bei dem gewählten Approximationsalgorithmus die doppelten Kosten des MST anfallen, ist er *a fortiori* eine Näherung der Güte 2 für das symmetrische TSP.

Ein kanonischer Weg, belastbare untere Schranken zu erhalten, ist die *LP-Relaxation*. Dabei ermittelt man zunächst eine äquivalente Formulierung des Problems als ganzzahliges Programm. Das lineare Programm (LP) erhält man sodann durch Relaxieren, das heißt Weglassen, der Ganzzahligkeitsbedingungen für die Variablen (daher der Name). Entscheidend dabei ist, dass das LP in Polynomialzeit gelöst werden kann. Die LP-Lösung kann als Orientierung für den Algorithmus bei der Entscheidungsfindung dienen, und ihr Wert stellt eine untere Schranke dar, anhand derer sich die Güte der Näherung analysieren

lässt. Die Qualität (Festigkeit bzw. Tightness) dieser unteren Schranke wird durch eine Größe namens *Integritätslücke* angegeben.

Der Nutzen von Methoden zur linearen Programmierung liegt jedoch nicht allein bei Approximationsalgorithmen für NP-schwere Probleme. Vielmehr kann man, wenn man das Problem aus polyedrischer Perspektive betrachtet, beim Entwickeln eines Algorithmus aus einem reichhaltigen Fundus an Werkzeugen und Ergebnissen aus der Polyedertheorie, der Dualitätstheorie und der polyedrischen Kombinatorik schöpfen. Diese Methoden können in unterschiedlichen Szenarien benutzt werden, um die Struktur des jeweiligen Problems besser zu verstehen.

In der vorliegenden Dissertation [Ta19] werden solche Methoden benutzt, um der Lösung zweier grundlegender Optimierungsprobleme näher zu kommen: der asymmetrischen Version des Handlungsreisenden-Problems sowie des Problems der perfekten Paarung (Perfect-Matching-Problem). Beides sind grundlegende Probleme, die Graphen betreffen – die vielleicht häufigsten und wichtigsten Strukturen in der Informatik. In vielerlei anderer Hinsicht dagegen scheinen beide Probleme sehr unterschiedlich zu sein: Während das asymmetrische Handlungsreisenden-Problem (ATSP) NP-schwer ist, liegt das perfekte Matching-Problem in P, und wir sind (obgleich in einem *parallelen* Szenario) an exakten Algorithmen interessiert und nicht an Approximationen. Es überrascht ein wenig, dass sich mit gleichartigen Methoden entscheidende strukturelle Einblicke in beide Probleme gewinnen lassen. Tatsächlich ist in beiden Fällen eine LP-Formulierung von exponentieller Größe mit *laminarer* Struktur der Ausgangspunkt der Lösung.

2 Unsere Beiträge

Im ersten Teil dieser Dissertation betrachten wir das asymmetrische Handlungsreisenden-Problem (ATSP) im Kontext von Approximationsalgorithmen und LP-Relaxationen. Im Speziellen geben wir den ersten Algorithmus für dieses Problem an, dessen Approximationsverhältnis beweisbar nicht durch eine unbeschränkte Funktion der Eingabegröße, sondern durch eine Konstante beschränkt ist. Es wurde lange vermutet, dass ein derartiger Algorithmus existiert; diese Vermutung können wir nun bestätigen. Da die Kosten der von unserem Algorithmus gelieferten Lösungen in Bezug auf die Standard-LP-Relaxation für das ATSP analysiert werden, bestätigt unser Ergebnis auch die vermutete konstante Integritätslücke dieser Relaxation.

Der erste Teil dieser Dissertation beruht auf einer Zusammenarbeit mit Ola Svensson und László A. Végh, die in STOC 2018 [STV18a] veröffentlicht wurde. Eine kurze Abhandlung dieses Teils wird in Abschnitt 3 gegeben.

Danach wenden wir uns dem Perfect-Matching-Problem zu, das im parallelen Szenario untersucht wird. Im Speziellen betrachten wir NC: die Komplexitätsklasse mit Problemen, deren Algorithmen parallel auf einer polynomialen Anzahl von Prozessoren und in polylogarithmischer Zeit ausgeführt werden können. Vom Perfect-Matching-Problem – zu bestimmen, ob sich alle Knoten eines gegebenen Graphen mit Kanten zu Paaren verbinden lassen – ist nicht bekannt, dass es in der NC-Komplexitätsklasse liegt. Es ist eine

bedeutende und ungelöste Frage, einen solchen Algorithmus anzugeben. Nun stellt sich heraus, dass dies eine Frage der Derandomisierung ist. In Szenarien, bei denen Randomisierung erlaubt ist, sind nämlich entspreche Algorithmen schon seit 40 Jahren bekannt. Die Frage, ob alle effizienten Berechnungen deterministisch durchgeführt werden können, ist von grundlegender Bedeutung in der Informatik, und das parallele Berechnen von Paarungen ist eines von mehreren viel beachteten, konkreten Problemen, für die ein (einfacher) randomisierter Algorithmus bekannt ist, jedoch kein deterministischer.

Der zweite Teil dieser Dissertation beschreibt wesentliche Fortschritte auf dem Weg zu einer Lösung dieser Frage: Wir geben einen deterministischen Algorithmus für das Perfect-Matching-Problem an, der quasi-polynomial viele (im Gegensatz zu polynomial vielen) Prozessoren benutzt und in polylogarithmischer Zeit läuft. Dies zeigt, dass das Matching-Problem in der Komplexitätsklasse quasi-NC liegt. Der zweite Teil dieser Dissertation beruht auf einer Zusammenarbeit mit Ola Svensson, die in FOCS 2017 [ST17] veröffentlicht wurde.

3 Das asymmetrische Problem des Handlungsreisenden

Das Problem des Handlungsreisenden – den kürzesten Rundweg zu finden, der durch n gegebene Städte führt – ist eines der bekanntesten NP-schweren Optimierungsprobleme.

Ohne irgendwelche Annahmen über die Distanzen zeigt eine einfache Reduzierung von dem Problem, zu entscheiden, ob ein Graph hamiltonsch ist, dass es NP-schwer ist, den kürzesten Rundweg bis auf einen gegebenen Gütefaktor zu approximieren. Daher wird das Problem üblicherweise dahingehend relaxiert, dass der Rundweg mehr als einmal durch eine jeweilige Stadt führen darf. Dies ist äquivalent zu der Annahme, dass die Distanzen die Dreiecksungleichung erfüllen: Die Distanz von Stadt i nach Stadt k ist nicht länger als die Distanz von i nach j plus die Distanz von j nach k. Alle hier erwähnten Ergebnisse beziehen sich auf dieses Szenario.

Wird außerdem angenommen, dass die Distanzen symmetrisch sind, dann liefert der klassische Algorithmus von Christofides aus dem Jahr 1976 [Ch76] garantiert einen Rundweg mit einer Länge von höchstens $3/2$ der optimalen Länge. (An dieser Stelle sei daran erinnert, dass ein einfacher Approximationsalgorithmus der Güte 2 in Abschnitt 1 behandelt wurde.) Die Verbesserung dieser garantierten Approximationsgüte ist eine notorisch offene Frage bei Approximationsalgorithmen. Es gab einigen Wirbel um neuere Fortschritte für den Sonderfall, dass die Distanzen als ungewichtete Kürzeste-Pfad-Metriken gegeben sind [GSS11, MS16, Mu12, SV14]. Doch obgleich man mutmaßt, dass die Standard-LP-Relaxation das Optimum mit einem Gütefaktor von $4/3$ approximiert, so bleibt es doch eine schwer lösbare Aufgabe, den Algorithmus von Christofides zu verbessern.

Wenn wir uns nicht auf symmetrische Distanzen beschränken, erhalten wir das allgemeinere *asymmetrische* Handlungsreisenden-Problem (ATSP). Im Vergleich zum symmetrischen Szenario ist unserem Verständnis nach die Lücke hier viel größer, und die derzeitigen algorithmischen Methoden sind nicht in der Lage, eine konstante Approximationsgüte zu garantieren. Das ist insbesondere deswegen faszinierend, weil man davon ausgeht, dass

die Standard-LP-Relaxation, auch als untere *Held-Karp-Schranke* bekannt, sich bis auf eine kleine Konstante an das Optimum annähert. Tatsächlich ist lediglich bekannt, dass ihre Integritätslücke[3] mindestens 2 beträgt [CGK06].

Der erste Approximationsalgorithmus für das ATSP wurde von Frieze, Galbiati und Maffioli [FGM82] vorgeschlagen. Er erreicht eine garantierte Approximationsgüte von $\log_2(n)$. Ihr eleganter Ansatz der „wiederholten Zyklus-Überdeckung“ wurde in mehreren Veröffentlichungen verfeinert [Bl08, Ka05, FS07], es gab aber keine *asymptotische* Verbesserung der garantierten Approximationsgüte. Dies änderte sich erst mit dem neueren Ergebnis von Asadpour et al. [As10], das eine Approximation der Güte $O(\log n \,/\, \log\log n)$ liefert. Asadpour et al. haben einen neuen und richtungsweisenden Ansatz zur Lösung des ATSP vorgestellt, der auf einer Verbindung mit dem graphentheoretischen Konzept dünner Spannbäume beruht. Dies hat im weiteren Verlauf zu verbesserten Algorithmen für Sonderfälle des ATSP geführt, wie etwa für Graphen mit beschränktem Genus [GS11]. Zudem haben Anari und Oveis Gharan unlängst diese Verbindung dazu benutzt, die beste bekannte obere Schranke für die Integritätslücke der Standard-LP-Relaxation signifikant auf $O(\text{poly}\log\log n)$ zu verbessern [AG15]. Dies impliziert einen effizienten Algorithmus zum Schätzen des Optimalwerts eines Rundwegs innerhalb eines Gütefaktors von $O(\text{poly}\log\log n)$, aber – da ihre Argumente nicht-konstruktiv sind – keinen Approximationsalgorithmus zum Finden eines Rundwegs mit entsprechender Gütegarantie.

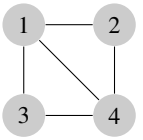
Ungefähr zur gleichen Zeit stellte Svensson [Sv15] einen alternativen Ansatz vor. Er reduziert die Aufgabe, das ATSP zu approximieren, auf ein anscheinend einfacheres Problem namens „Local-Connectivity ATSP“ (asymmetrisches TSP mit lokaler Konnektivität). Die Veröffentlichung [Sv15] gibt außerdem einen Algorithmus für Local-Connectivity ATSP an, der auf den Sonderfall knotengewichteter Metriken beschränkt ist, was einen Approximationsalgorithmus mit konstanter Güte für diesen Sonderfall impliziert. Svensson, Tarnawski und Végh haben dies in einer darauf aufbauenden Arbeit [STV18b] auf Graphen mit höchstens zwei verschiedenen Kantengewichten verallgemeinert. In diesem Teil der Dissertation knüpfen wir an diese beiden Ergebnisse an, verallgemeinern sie und geben einen Approximationsalgorithmus mit konstanter Güte für alle Metriken an.

Theorem 1. *Es gibt einen Algorithmus für das ATSP, der in Polynomialzeit einen Rundweg mit einem Wert von höchstens einem konstanten Vielfachen der unteren Held-Karp-Schranke liefert.*

4 Matching liegt in quasi-NC

Das Perfect-Matching-Problem ist eine grundlegende Frage der Graphentheorie. Die Arbeit an diesem Problem hat zu der Entwicklung vieler zentraler Konzepte der modernen Informatik beigetragen, darunter linear-algebraische, probabilistische und parallele Algorithmen. Edmonds [Ed65] war der Erste, der einen in Polynomialzeit laufenden Algorithmus dafür angegeben hat. Doch auch ein halbes Jahrhundert später verstehen wir die

[3] Die Integritätslücke ist definiert als das maximale Verhältnis zwischen den optimalen Werten der exakten (ganzzahligen) Formulierung und deren Relaxation.



$$T(G) = \begin{pmatrix} 0 & X_{12} & X_{13} & X_{14} \\ -X_{12} & 0 & 0 & X_{24} \\ -X_{13} & 0 & 0 & X_{34} \\ -X_{14} & -X_{24} & -X_{34} & 0 \end{pmatrix}$$

Abb. 1: Beispiel für eine Tutte-Matrix eines ungerichteten Graphen.

deterministische parallele Komplexität des Perfect-Matching-Problems immer noch nicht vollständig. In dieser Dissertation erzielen wir Fortschritte in dieser Richtung. Wir betrachten ein Problem als parallel effizient lösbar, wenn es einen Algorithmus dafür gibt, der polylogarithmische Zeit benötigt und polynomial viele Prozessoren benutzt. Formeller gesprochen:

Ein Problem gehört der Komplexitätsklasse NC an, wenn es uniforme Kreise von polynomialer Größe und polylogarithmischer Tiefe aufweist. Die Komplexitätsklasse RNC ergibt sich, wenn wir außerdem Randomisierung zulassen.

Wir untersuchen die Entscheidungsversion und die Suchversion des Problems. Bei der Entscheidungsversion ist für einen gegebenen ungerichteten einfachen Graphen zu bestimmen, ob er eine perfekte Paarung aufweist. Bei der Suchversion wird eine perfekte Paarung gefunden und zurückgegeben, falls eine solche existiert. Dass die Entscheidungsversion in RNC liegt, hat zuerst Lovász [Lo79] gezeigt. Die Suchversion hat sich als schwieriger erwiesen, und erst mehrere Jahre später stellten Karp, Upfal und Wigderson [KUW86] sowie Mulmuley, Vazirani und Vazirani [MVV87] fest, dass sie in RNC liegt. Alle diese Algorithmen sind randomisiert, und es bleibt ein bedeutendes ungelöstes Problem, zu bestimmen, ob Zufälligkeit erforderlich ist, d. h., ob eine der Versionen in NC liegt.

Als erfolgreich erwiesen hat sich der linear-algebraische Ansatz zur Lösung des Perfect-Matching-Problems. Er nutzt die Tutte-Matrix, verknüpft mit einem Graphen $G = (V,E)$, wobei es sich um eine $|V| \times |V|$-Matrix handelt, die wie folgt definiert ist (für ein Beispiel siehe Abb. 1):

$$T(G)_{u,v} = \begin{cases} X_{(u,v)} & \text{if } (u,v) \in E \text{ and } u < v, \\ -X_{(v,u)} & \text{if } (u,v) \in E \text{ and } u > v, \\ 0 & \text{if } (u,v) \notin E, \end{cases}$$

wobei $X_{(u,v)}$ für $(u,v) \in E$ Variablen sind. Tuttes Theorem [Tu47] sagt aus, dass $\det T(G) \neq 0$ genau dann gilt, wenn G eine perfekte Paarung aufweist. Das sind großartige Neuigkeiten für die Parallelisierung, da das Berechnen von Determinanten in NC ist [Cs76, Be84]. Die Matrix ist allerdings über einen Ring aus Unbestimmten definiert, so dass normalerweise Zufälligkeit benutzt wird, um zu testen, ob die Determinante von null verschieden ist. Ein Ansatz besteht darin, jede Unbestimmte durch einen zufälligen Wert aus einem großen Feld zu ersetzen. Dies führt, unter anderem, zu den schnellsten bekannten Einprozessor-Laufzeiten für dichte Graphen [MS04].

Ein zweiter, von Mulmuley, Vazirani und Vazirani [MVV87] für die Suchversion gewählter Ansatz, besteht darin, die Unbestimmten durch zufällig ausgewählte Zweierpotenzen zu ersetzen. Genauer wird für jede Kante (u,v), ein zufälliges Gewicht $w(u,v) \in \{1,2,...,2|E|\}$ ausgewählt, und wir substituieren $X_{(u,v)} := 2^{w(u,v)}$. Nun treffen wir die entscheidende Annahme, dass eine perfekte Paarung (perfektes Matching) M *isoliert* ist, in dem Sinne, dass sie die *einzige perfekte Paarung mit minimalem Gewicht* ist (also $w(M)$ minimiert wird). Dann bleibt $\det T(G)$ nach der Substitution von null verschieden: Es lässt sich zeigen, dass M einen Term $\pm 2^{2w(M)}$ zu $\det T(G)$ beiträgt, wohingegen alle anderen Terme Vielfache von $2^{2w(M)+1}$ sind und daher $2^{2w(M)}$ nicht aufheben können. Die Determinante kann weiterhin in NC berechnet werden, da alle Einträge $2^{w(u,v)}$ der Matrix eine polynomiale Bitlänge aufweisen, und somit ein paralleler Algorithmus für die Entscheidungsversion vorliegt. Es folgt auch ein Algorithmus für die Suchversion: Für jede Kante wird parallel geprüft, ob die niederwertigste Stelle $2^{2w(M)}$ in der Determinante verschwindet, wenn die Kante entfernt wird; diejenigen Kanten, bei denen dies der Fall ist, werden ausgegeben.

Die grundlegende Behauptung in [MVV87] ist, dass sich durch Zuweisen von zufälligen Gewichten an Kanten in der Tat mit hoher Wahrscheinlichkeit eine Paarung isolieren lässt. Dies ist als das Isolations-Lemma bekannt, und es erweist sich in dem wesentlich allgemeineren Szenario von wahlfreien Mengenfamilien als zutreffend:

Theorem 2 (Isolations-Lemma)**.** *Sei $\mathscr{M} \subseteq 2^E$ eine beliebige Familie von Untermengen eines Universums $E = \{1,2,...,|E|\}$ und eine Gewichtsfunktion $w : E \to \{1,2,...,2|E|\}$ dadurch definiert, dass zufällig jedes $w(e)$ für $e \in E$ unabhängig und uniform gewählt ist. Dann gibt es mit einer Wahrscheinlichkeit von mindestens $1/2$ eine* einzige *Menge $M \in \mathscr{M}$, welche das Gewicht $w(M) = \sum_{e \in M} w(e)$ minimiert.*

Eine solche Gewichtsfunktion w nennen wir *isolierend*. Wir nehmen $\mathscr{M}$ in Theorem 2 als die Menge aller perfekten Paarungen.

Da Theorem 2 die einzige randomisierte Zutat zu dem RNC-Algorithmus ist, besteht ein natürlicher Ansatz zu zeigen, dass das Perfect-Matching-Problem in NC liegt, darin, das Isolations-Lemma zu derandomisieren. Das heißt, wir hätten gerne eine Menge aus polynomial vielen Gewichtsfunktionen (mit polynomial beschränkten Werten), für die garantiert ist, dass sie eine isolierende Gewichtsfunktion enthält. Um einen NC-Algorithmus zu erhalten, müssen wir diese Menge effizient parallel erzeugen können. Dann können wir alle Gewichtsfunktionen gleichzeitig ausprobieren.

Das Derandomisieren des Isolations-Lemmas erweist sich jedoch als eine anspruchsvolle offene Frage. Gelungen ist dies bisher für bestimmte Klassen von Graphen: stark chordale Graphen, planare bipartite Graphen sowie Graphen mit einer kleinen Anzahl von perfekten Paarungen. Allgemeiner gab es viel Interesse an NC-Algorithmen für das Perfect-Matching-Problem auf eingeschränkten Graphen-Klassen (nicht notwendigerweise unter Verwendung des Isolations-Lemmas), z. B.: regulär bipartit, P_4-tidy, dicht, konvex bipartit, klauenfrei, Unvergleichbarkeitsgraphen. Das allgemeine Mengen-Familien-Szenario des Isolations-Lemmas steht auch in Beziehung mit unteren Schranken für die Kreiskomplexität und polynomialen Identitätstests. Für Referenzen siehe [ST17].

Kürzlich haben Fenner, Gurjar und Thierauf [FGT16] einen wesentlichen Beitrag geleistet. Ihnen ist es gelungen, das Isolations-Lemma für bipartite Graphen fast vollständig zu derandomisieren. Konkret definieren sie eine Familie von Gewichtsfunktionen, die sich „vergesslich“ berechnen lassen (lediglich unter Verwendung der Anzahl von Knoten n), und beweisen, dass für jeden bipartiten Graphen eine dieser Funktionen isolierend ist. Da die solchermaßen definierte Familie von quasi-polynomialer Größe ist und die Gewichte quasi-polynomial groß sind, hat dies das perfekte bipartite Matching-Problem in die Komplexitätsklasse quasi-NC versetzt.

Nichtsdestotrotz blieb das Szenario des Derandomisierungsproblems mit allgemeinen Graphen (sei es unter Verwendung des Isolations-Lemmas oder nicht) ungelöst. Selbst im planaren Fall, wo NC-Algorithmen für bipartite planare Graphen und Graphen mit kleinem Genus seit Langem bekannt sind [MN89, MV00], kannte man bisher keinen quasi-NC-Algorithmus für nicht-bipartite Graphen. Im Allgemeinen war die beste bekannte obere Schranke für die Größe uniformer Kreise mit polylogarithmischer Tiefe exponentiell.

Wir sind in der Lage, diese Verständnislücke nahezu zu überbrücken. Das Hauptergebnis des zweiten Teils der dieser Dissertation lautet:

Theorem 3. *Für jede Zahl n ist es möglich, in quasi-NC $n^{O(\log^2 n)}$ Gewichtsfunktionen auf $\{1,2,\ldots,\binom{n}{2}\}$ mit durch $n^{O(\log^2 n)}$ beschränkten Gewichten derart zu konstruieren, dass für jeden Graphen auf n Knoten eine der Gewichtsfunktionen eine perfekte Paarung isoliert (sofern eine solche existiert).*

Zusammengenommen implizieren die Ergebnisse aus [MVV87] und Theorem 3, dass das Perfect-Matching-Problem (sowohl die Entscheidungs- als auch die Suchvariante) bei allgemeinen Graphen in quasi-NC liegt. Wir merken an, dass der implizite Algorithmus sehr einfach ist. Die Komplexität liegt in der Analyse, d. h., im Beweis, dass eine der Gewichtsfunktionen isolierend ist.

Literaturverzeichnis

[AG15] Anari, Nima; Gharan, Shayan Oveis: Effective-Resistance-Reducing Flows, Spectrally Thin Trees, and Asymmetric TSP. In: IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS). S. 20–39, 2015.

[As10] Asadpour, Arash; Goemans, Michel X.; Madry, Aleksander; Gharan, Shayan Oveis; Saberi, Amin: An O(log n/ log log n)-approximation Algorithm for the Asymmetric Traveling Salesman Problem. In: Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010. S. 379–389, 2010.

[Be84] Berkowitz, Stuart J.: On Computing the Determinant in Small Parallel Time Using a Small Number of Processors. Information Processing Letters, 18(3):147–150, 1984.

[Bl08] Bläser, Markus: A new approximation algorithm for the asymmetric TSP with triangle inequality. ACM Transactions on Algorithms, 4(4), 2008.

[CGK06] Charikar, Moses; Goemans, Michel X.; Karloff, Howard J.: On the Integrality Ratio for the Asymmetric Traveling Salesman Problem. Math. Oper. Res., 31(2):245–252, 2006.

[Ch76] Christofides, Nicos: Worst-case analysis of a new heuristic for the travelling salesman problem. Bericht, Graduate School of Industrial Administration, CMU, 1976.

[Cs76] Csanky, L.: Fast parallel inversion algorithm. SIAM Journal of Computing, 5:618–623, 1976.

[Ed65] Edmonds, Jack: Paths, trees, and flowers. Canadian Journal of Mathematics, 17:449–467, 1965.

[FGM82] Frieze, Alan M.; Galbiati, Giulia; Maffioli, Francesco: On the worst-case performance of some algorithms for the asymmetric traveling salesman problem. Networks, 12(1):23–39, 1982.

[FGT16] Fenner, Stephen A.; Gurjar, Rohit; Thierauf, Thomas: Bipartite perfect matching is in quasi-NC. In: Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC. S. 754–763, 2016.

[FS07] Feige, Uriel; Singh, Mohit: Improved Approximation Ratios for Traveling Salesperson Tours and Paths in Directed Graphs. In: 10th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems APPROX 2007. S. 104–118, 2007.

[GS11] Gharan, Shayan Oveis; Saberi, Amin: The Asymmetric Traveling Salesman Problem on Graphs with Bounded Genus. In: Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011. S. 967–975, 2011.

[GSS11] Gharan, Shayan Oveis; Saberi, Amin; Singh, Mohit: A Randomized Rounding Approach to the Traveling Salesman Problem. In: IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011. S. 550–559, 2011.

[Ka05] Kaplan, Haim; Lewenstein, Moshe; Shafrir, Nira; Sviridenko, Maxim: Approximation algorithms for asymmetric TSP by decomposing directed regular multigraphs. J. ACM, 52(4):602–626, 2005.

[KUW86] Karp, Richard M.; Upfal, Eli; Wigderson, Avi: Constructing a perfect matching is in random NC. Combinatorica, 6(1):35–48, 1986.

[Lo79] Lovász, László: On determinants, matchings, and random algorithms. In: FCT. S. 565–574, 1979.

[MN89] Miller, G. L.; Naor, J.: Flow in planar graphs with multiple sources and sinks. In: 30th Annual Symposium on Foundations of Computer Science. S. 112–117, 1989.

[MS04] Mucha, Marcin; Sankowski, Piotr: Maximum Matchings via Gaussian Elimination. In: 45th Symposium on Foundations of Computer Science (FOCS). S. 248–255, 2004.

[MS16] Mömke, Tobias; Svensson, Ola: Removing and Adding Edges for the Traveling Salesman Problem. J. ACM, 63(1):2:1–2:28, 2016.

[Mu12] Mucha, Marcin: 13/9-approximation for Graphic TSP. In: 29th International Symposium on Theoretical Aspects of Computer Science, STACS 2012. S. 30–41, 2012.

[MV00] Mahajan, Meena; Varadarajan, Kasturi R.: A New NC-algorithm for Finding a Perfect Matching in Bipartite Planar and Small Genus Graphs. In: Proceedings of the Thirty-second Annual ACM Symposium on Theory of Computing, STOC. S. 351–357, 2000.

[MVV87] Mulmuley, Ketan; Vazirani, Umesh V.; Vazirani, Vijay V.: Matching is as easy as matrix inversion. Combinatorica, 7(1):105–113, 1987.

[ST17] Svensson, O.; Tarnawski, J.: The Matching Problem in General Graphs Is in Quasi-NC. In: 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS). S. 696–707, 2017.

[STV18a] Svensson, Ola; Tarnawski, Jakub; Végh, László A.: A Constant-factor Approximation Algorithm for the Asymmetric Traveling Salesman Problem. In: Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing. STOC 2018, New York, NY, USA, S. 204–213, 2018.

[STV18b] Svensson, Ola; Tarnawski, Jakub; Végh, László A.: Constant Factor Approximation for ATSP with Two Edge Weights. Mathematical Programming, 172(1-2):371–397, 2018.

[SV14] Sebő, András; Vygen, Jens: Shorter tours by nicer ears: 7/5-Approximation for the graph-TSP, 3/2 for the path version, and 4/3 for two-edge-connected subgraphs. Combinatorica, 34(5):597–629, 2014.

[Sv15] Svensson, Ola: Approximating ATSP by Relaxing Connectivity. In: FOCS 2015: Proceedings of the 56th Annual IEEE Symposium on Foundations of Computer Science. 2015.

[Ta19] Tarnawski, Jakub: New Graph Algorithms via Polyhedral Techniques. EPFL, Lausanne, 2019.

[Tu47] Tutte, W. T.: The Factorization of Linear Graphs. Journal of the London Mathematical Society, 22:107–111, 1947.

Jakub Tarnawski ist Algorithmenforscher bei Microsoft Research. Er interessiert sich umfassend für theoretische Informatik und kombinatorische Optimierung, insbesondere für Graphenalgorithmen und Approximationsalgorithmen. Bei seiner Arbeit widmet er sich grundlegenden Problemen auf diesen Gebieten, wie etwa dem Handlungsreisenden-Problem, der submodularen Maximierung, dem Matching, und verschiedenen Scheduling-Varianten. Er ist Träger des Best Paper Award des STOC 2018 für seine Arbeit zum Handlungsreisenden-Problem sowie des Best Paper Award des FOCS 2017 für seine Arbeit zu Paarungen. 2019 promovierte Jakub Tarnawski an der EPFL unter Anleitung von Ola Svensson zum Doktor der Informatik. Seine Dissertation wurde mit dem Chorafas Foundation Prize und der EPFL Thesis Distinction ausgezeichnet. Seine Master-Abschlüsse in Mathematik und Informatik erhielt er von der Universität Wrocław, Polen.

Methoden zum Monitoring des zirkadianen Rhythmus im Alltag[1]

Florian Wahl[2]

Abstract: Unsere innere Uhr, die zirkadiane Uhr, regelt wann Aktivitäten im Körper stattfinden und bestimmt so z.B. wann wir die höchste Konzentrationsfähigkeit haben. Die zirkadiane Phase wird durch den Zeitpunkt und die Intensität von Lichtexposition verändert. Im Alltag stellt die Synchronisation von interner und externer Uhr eine zunehmende Herausforderung dar. Ein Leben gegen die innere Uhr birgt negative gesundheitliche Konsequenzen. Bisher kann die zirkadiane Phase nur mit erheblichem Aufwand gemessen werden. Ziel der Dissertation ist es Methoden zur Beobachtung der inneren Uhr im Alltag zu erforschen. Wir entwickeln dafür personalisierte Datenbrillen, welche Alltagsaktivitäten des Trägers erkennen und so z.B. vor Bildschirmbenutzung bei Nacht warnen können. Des Weiteren schätzen wir die Lichtexposition mit Smartphones, um so die Verschiebung der inneren Uhr vorherzusagen. Wir verwenden Daten von Smartphones, um Schlaf-Wach Zeiten zu erkennen. Expertenmodelle werden mit datengetriebenen Modellen kombiniert, um die Ergebnisse zu verbessern und Vorhersagen in relevante Domänen übersetzen.

1 Einleitung

1.1 Zirkadiane Rhythmen

Unser Tagesablauf wurde bereits in der Steinzeit vom Tag-Nacht-Rhythmus beeinflusst. Menschen sind tagaktiv, da die Chance zu überleben und Fressfeinde zu besiegen bei Tage höher ist als in der Nacht. Die Nacht wird dann genutzt, um sich zurückzuziehen und zu erholen. Da die Tageszeit so wichtig für unser Überleben war, haben sich unsere Körper angewöhnt, sich dem Tag-Nacht-Rhythmus anzupassen. Zirkadiane Rhythmen haben eine Periode von ungefähr einem Tag (vom lateinischen circa diem). Im Menschen passt sich die zirkadiane Phase dem natürlichen Tag-Nacht-Rhythmus an. Dabei wird die zirkadiane Phase durch das Timing und die Intensität von Licht angelernt. Dabei wird die zirkadiane Phase komprimiert oder expandiert und so die innere Uhr vor- oder rückwärts gestellt. Die Hauptuhr sitzt dabei im Nucleus suprachiasmaticus (SCN), welcher hell/dunkel Informationen über einen direkten Kommunikationskanal zu den Augen erhält. Zeitinformationen werden dann vom SCN an Zellen und Organe im Körper weitergegeben.

Die Phase der inneren Uhr im Vergleich zur Uhrzeit der Außenwelt variiert von Person zu Person. Diese persönliche Präferenz wird im Chronotyp ausgedrückt. Der Chronotyp variiert von früh (Lerche) bis spät (Eule) und kann geschätzt werden, z.B. mit dem Munich Chronotype Questionnaire [Ro07]. Der Chronotyp verändert sich während eines

[1] Englischer Titel der Dissertation: "Methods for monitoring the human circadian rhythm in free-living"

[2] Technologie Campus Grafenau, Technische Hochschule Deggendorf, florian.wahl@gmail.com

cc by sa

Menschenlebens. So haben Kinder im Mittel eine frühere zirkadiane Phase, Heranwachsende eine spätere und mit fortschreitendem Erwachsenenalter verschiebt sich die Phase im Mittel wieder mehr nach vorne.

Fast alle Funktionen des Körpers hängen von der inneren Uhr ab. So haben z.B. physische Stärke oder fokussiertes Denken jeweils einen täglichen Höhepunkt. Wenn die zirkadiane Phase bekannt ist, könnte für jede dieser Aktivitäten die optimale Uhrzeit berechnet werden und so die eigene Leistung maximiert werden. Jedoch ist es aufgrund der vielen verschiedenen Verbindlichkeiten unseres Alltags oftmals schwierig innere und soziale Uhren zu synchronisieren.

Als „sozialen Jetlag“ beschreibt man die Zeitverschiebung zwischen interner (zirkadianer) und externer (Uhrzeit der Umgebung) Uhr. Wer regelmäßig gegen seine innere Uhr lebt, nimmt Gesundheitsrisiken auf sich, z.B. eine höhere Chance übergewichtig zu sein oder an Krebs zu erkanken. Künstliches Licht ist ein wesentlicher Ursprung für sozialen Jetlag, besonders künstliches Licht aus LED-Lichtquellen. Diese haben einen erhöhten Lichtanteil im blauen Spektralbereich, welcher die Phase der inneren Uhr besonders stark beeinflusst. Die Messung der zirkadianen Phase ist notwendig, um sozialen Jetlag zu minimieren.

Zur Bestimmung der zirkadianen Phase gibt es einerseits Labormethoden, welche invasiv und kostenintensiv, aber genau sind, und andererseits Aktigraphie und Fragebögen, welche zwar eine Schätzung der zirkadianen Phase im Alltag erlauben, aber kein detailliertes und genaues Ergebnis liefern. Ziel der Dissertation [Wa19] ist es, mit neuen Messmethoden und Algorithmen Ansätze zu schaffen, welche ein kontinuierliches Monitoring der zirkadianen Phase im Alltag erlauben.

Die Dissertation liefert folgende Beiträge: 1) Eine personalisierte, 3D-gedruckte Brille, welche mit einem multi-modalen Sensorsystem ausgestattet ist und Alltagsaktivitäten erkennen sowie Lichtexposition im Alltag messen kann. 2) Die Erkennung von Schlaf-Wach-Mustern sowie Schätzung von Lichtexposition mit Smartphones. 3) Neue Algorithmen zur Schätzung von Schlaf-Wach-Mustern und der Verschiebung der zirkadianen Phase aufgrund von Lichtexposition. Dabei werden Methoden des maschinellen Lernens mit Expertenmodellen kombiniert. Die Beiträge können in Kombination verwendet werden, um die innere Uhr des Menschen im Alltag zu beobachten. Die vorgeschlagenen Methoden sind dabei genauer und kosteneffizienter als die bisherigen Methoden.

2 Personalisierte intelligente Brillen

Der Kopf ist eine hervorragende Messposition, da viele physiologische Größen am Kopf messbar sind. Zahlreiche Aktivitäten lassen sich am charakteristischne Bewegungsmuster des Kopfes erkennen (z.B. Laufen). Die Schwierigkeit besteht darin, die Sensoren so am Kopf anzubringen, dass eine Stigmatisierung des Trägers vermieden wird. Gleichzeitig stellen Brillen ein bereits gesellschaftlich akzeptiertes, am Kopf getragenes Accessoire dar. Dank der fortschreitenden Miniaturisierung von Elektronik ist es heute möglich, Sensorik in ein Brillengestell zu integrieren. Jedoch muss der Tragekomfort und die Anpassung an den Kopf des Trägers weiterhin gewährleistet sein. Reguläre Brillen werden durch Erhitzen

und Biegen an den Kopf des Trägers angepasst. Die integrierte Sensorik würde der Hitze und mechanischen Kräften eines solchen Verfahrens nicht standhalten. Jedoch erlauben neuartige additive Produktionsmethoden eine günstige Individualisierung bereits in der Produktion. Da im Gegensatz zum Spritzgussverfahren beim 3D-Druck keine teure Form benötigt wird, entstehen durch die kleinen Veränderungen keine zusätzlichen Kosten. Um diesen Vorteil des 3D-Drucks zu nutzen, haben wir einen digitalen Anpassungsprozess für Brillen entwickelt [Wa17]. Dieser passt die Brillenform bereits vor der Herstellung an den Kopf des Trägers an. So können verschiedene Sensoren im Brillengestell untergebracht werden ohne von Außenstehenden wahrgenommen zu werden. Dafür passen wir ein parametrisiertes CAD-Modell an den Kopf des Trägers an.

Unser digitaler Entwicklungsprozess besteht aus vier Schritten: 1) Messung von drei anthropometrischen Parametern am Kopf des Trägers. 2) Erstellung eines digitalen Kopfmodells mit der Open-Source Software MakeHuman. 3) Anpassung des parametrisierten CAD-Modells an das digitale Kopfmodell. 4) Druck des Brillengestells und Integration der Sensortechnologie. Mit der so erstellten Sensorbrille lassen sich eine ganze Reihe an Anwendungen, besonders im Bereich Gesundheit und Sport, realisieren. So ist es möglich Aktivitäten des täglichen Lebens [WFA15], Bildschirmnutzung [WKA17] und Kaubewegungen [ZBA16] zu erkennen. Dabei ist es von Vorteil, dass die von uns entwickelte Brille, im Gegensatz zu anderen Datenbrillen wie z.B. Google Glass, für Außenstehende wie eine gewöhnliche Brille aussieht. So kann das Risiko der Stigmatisierung des Trägers minimiert werden.

Der Prozess wurde mit Hilfe einer Simulation und eines Realversuchs evaluiert. Für die Simulation wurden neun Kopfmodelle erstellt, welche jeweils auf das 1., 50. und 99. Perzentil der drei Kopfparameter gesetzt sind. Nach der Anpassung des CAD-Modells an die Kopfmodelle wurden die Brillengestelle an einem Punkt an der Nase des Kopfmodells ausgerichtet. Mit einer Interferenzanalyse wurde verifiziert, dass sich Kopf- und Brillenmodelle nicht überlappen. Für den Realversuch wurden vier Probanden rekrutiert, welche bereits Brillenträger sind. Für jeden Probanden wurde ein personalisiertes Brillengestell angefertigt. Die Probanden trugen die Brille für einen Tag und füllten anschließend einen Fragebogen zu Passform und Tragekomfort, auch im Vergleich zur ihrer eigenen Brille, aus. Die Probanden bewerteten den Tragekomfort der gedruckten Brillen als angenehm (4,25 von 5 Punkten). Im Vergleich zur eigenen Brille empfanden die Probanden den Tragekomfort der gedruckten Brillengestelle als vergleichbar.

3 Erkennung von Aktivitäten des täglichen Lebens

Um zu wissen, welche Aktivitäten zu einer ungewünschten Verschiebung der zirkadianen Phase führen, muss der tägliche Ablauf bekannt sein. Eine Erkennung von Aktivitäten des täglichen Lebens (ADLs) ist mit Hilfe von Bewegungssensoren möglich. Ein Kapitel der Dissertation [WFA15] zeigt die Integration solcher Sensoren in ein Brillengestell und beschreibt Methoden zur Erkennung von ADLs.

Zur Integration in die Brille entwickelten wir eine Hardwarearchitektur, welche aus einem Mikrocontroller mit integrierter Bluetooth-Kommunikation, Flashspeicher, Batterie, Lade-

controller, inertialer Messeinheit, Lichtsensor und Herzratensensor besteht. Für den ersten Prototyp wurde die gesamte Elektronik an einem regulären Brillengestell angebracht. In einer späteren Iteration wurde die Elektronik dann in einem 3D-gedruckten Brillengestell untergebracht, so dass von außen nicht mehr erkennbar war, dass es sich um eine mit Sensorik ausgestattete Brille handelt.

Zur Klassifizierung von ADLs verwendeten wir Gaußsche Mixtur-Modelle (GMM). Unser Modell ist in der Lage neun Gruppen von ADLs zu unterscheiden. Diese bestehen aus Essen, Laufen, Zähneputzen, Treppensteigen, Joggen, Radfahren, Lesen, Bildschirmarbeit und Putzen. Zur Erkennung werden die Daten standardisiert und normiert und anschließend mit einem gleitenden Fenster (Größe 30 s, Schrittweite 1 s) segmentiert. Danach werden die L^2 Normen von Beschleunigung und Drehrate ermittelt. Nun werden für jede Achse (x, y, z, und L^2 Norm) der Beschleunigung und Drehrate 25 Merkmale (z.B. Mittelwert, Maximum, etc.) berechnet. Aus den resultierenden 175 Merkmalen werden mit Hilfe der Hauptkomponentenanalyse 78 Merkmale selektiert, sodass 99,9% der Varianz in den Daten erklärbar bleibt. Für jede ADL Gruppe wird ein GMM verwendet, um die Wahrscheinlichkeit der Zugehörigkeit zu dieser Aktivität zu bestimmen. Alle Modelle berechnen die Wahrscheinlichkeit wobei sich das Modell mit der höchsten Wahrscheinlichkeit durchsetzt. Da es aber auch möglich ist, dass keine der modellierten Aktivitäten, sondern eine ganz andere Aktivität, ausgeführt wird, haben wir den Klassifizierer um eine Minimalgrenze erweitert. Erreicht keine der modellierten Aktivitäten die geforderte Minimalwahrscheinlichkeit, so wird die sogenannte Nullklasse als Aktivität erkannt.

Zur Evaluierung wurde ein Versuch mit neun Probanden durchgeführt. Diese wurden mit einer Brille und einem tragbaren EKG Referenzmessgerät ausgestattet. Jeder Proband führte einen vorgeschriebenen Tagesablauf aus. Dabei annotierte ein Beobachter die jeweilige Aktivität. Die Aufzeichnungen starteten mit einem Frühstück und endeten mit einem Spaziergang nach dem Abendessen. Es wurden verschiedene Aktivitäten aus allen genannten ADL-Gruppen aufgezeichnet. So entstand eine Datenmenge von insgesamt 66,08 h. Für die gesamte Evaluierung wurde Leave-One-Participant-Out Kreuzvalidierung verwendet.

Ein GMM mit drei Komponenten je Klasse und diagonaler Kovarianzmatrix stellte für uns den besten Kompromiss zwischen Rechenaufwand und Erkennungsleistung dar. Für alle ADL-Gruppen außer Radfahren lag die Erkennungsleistung über 80% mit einem Mittel von 77% über alle Klassen. Mit einer solchen kontinuierlichen Aktivitätserkennung wäre es problemlos möglich, den Alltag des Brillenträgers zu rekonstruieren und so für die zirkadiane Phasenverschiebung kritische Aktivitäten zu identifizieren.

4 Erkennung von Bildschirmbenutzung

Da viele Menschen heute jederzeit und überall arbeiten können, ist Bildschirmbenutzung bei Nacht weit verbreitet. Aufgrund des hohen prozentualen Blauanteils ist Bildschirmbenutzung für das Monitoring der zirkadianen Phase von besonderem Interesse. Die Hintergrundbeleuchtung heutiger Bildschirme verwendet in der Regel Leuchtdioden (LEDs).

Der blaue Spektralanteil von LED Licht ist besonders stark ausgeprägt. Blaues Licht am Abend hemmt die Produktion von Melatonin und sorgt so für eine Verzögerung der zirkadianen Phase. Neben dem Abschalten des Bildschirms gibt es die Möglichkeit, das Farbprofil anzupassen und so die Auswirkungen auf die zirkadiane Phase zu verringern.

Unsere Brille verfügt über einen Lichtsensor, welche neben der Lichtintensität über das gesamte sichtbare Spektrum auch die anteilige Intensität im blauen, roten und grünen Teilspektrum misst. Umgebungslicht und Licht von Bildschirmen lassen sich durch die Verhältnisse zwischen den Teilspektren unterscheiden. Um das Erkennungsproblem möglichst gut zu isolieren, entwickelten wir einen geschlossenen Messaufbau, in welchem Umgebungslicht verschiedener Lampentechnologien zuschaltbar ist. Innen befindet sich ein 24 Zoll Monitor, ein Spektrometer und vier Lichtsensoren vom Typ TCS34725 der Firma ams, welche Lichtintensität des gesamten Spektrums sowie die Intensitäten der blauen, roten und grünen Spektralanteile separat messen.

Die Bildschirmerkennung wurde folgendermaßen realisiert: Aus den einzelnen Kanälen des Lichtsensors (Gesamtspektrum, blaue, rote und grüne Teilspektra) werden 42 verschiedene virtuelle Sensorkanäle berechnet. Diese virtuellen Sensorkanäle werden durch ein gleitendes Fenster mit 50% Überlappung segmentiert. Im nächsten Schritt werden für jedes Fenster 18 Merkmale in der Zeitdomäne (Mittelwert, Varianz, Median, etc.) berechnet. Schlussendlich verwenden wir eine Support Vector Maschine mit linearem Kernel zur Erkennung von Bildschirmnutzung.

Zusätzlich zur Evaluierung im Messaufbau wurde eine Messreihe im Labor mit 14 Probanden aufgezeichnet. Dabei las jeder Proband einen Artikel, schaute eine Dokumentation über Kaffee auf einem 27 Zoll Fernseher in einem Abstand von 140 cm und surfte im Internet auf einem 24 Zoll Bildschirm bei 70 cm Entfernung. Jede Aktivität wurde dabei für 20 Minuten ausgeführt. Zudem wurde die Intensität des Umgebungslichts mit einem Handmessgerät bestimmt. Für die Evaluierung wurde außerdem der Datensatz aus der Studie zur Erkennung von Alltagsaktivitäten verwendet. Für die Evaluierung aller Datensätze wurde Leave-One-Participant-Out Kreuzvalidierung angewendet.

Für die Labormessreihe war es möglich, Bildschirmbenutzung bei Umgebungslichtintensitäten von weniger als 200 lux mit einer Rate von über 0,9 ROC AUC zu erkennen. Je weniger Umgebungslicht vorhanden war, desto besser lies sich Bildschirmnutzung erkennen. Dieser Effekt ist für unsere Anwendung von Vorteil, da bei starkem Umgebungslicht die Bildschirmnutzung im Vergleich zum Umgebungslicht kaum einen Einfluss auf die zirkadiane Phase hat.

5 Erkennung von Schlaf-Wach-Transitionen

Wir verbringen etwa ein Drittel unseres Lebens damit zu schlafen, was Schlaf zu einer der wichtigsten menschlichen Aktivitäten macht. Dennoch ist Schlafmangel ein weit verbreitetes Problem. Während die National Sleep Foundation zu einer täglichen Schlafdauer von 7 bis 9 Stunden rät, schlafen 29% der US-Amerikaner regelmäßig weniger als 7 Stunden. Wenn wir ohne Wecker oder äußere Umstände aufwachen, so bestimmt unser Chrono-

typ den Aufwachzeitpunkt. Somit könnte mit einem kontinuierlichen Schlaftagebuch der Chronotyp aus Schlafzyklen, die ohne Wecker enden, geschätzt werden. Die manuelle Erhebung von Schlafdaten ist jedoch schwierig, da wir morgens oft nicht mehr genau sagen können, wann wir am Vorabend eingeschlafen sind. Rogers et al. [RCA93] fanden heraus, dass handschriftliche Tagebücher mit einer zeitlichen Auflösung von 15 Minuten im Mittel eine Diskrepanz von 39 Minuten aufweisen. Darüber hinaus stellen handschriftliche Schlaftagebücher einen erheblichen Aufwand für Probanden dar. Smartphones sind mit einer Vielzahl verschiedener Sensoren ausgestattet und für viele Menschen ohnehin schon ein täglicher Begleiter. Im Rahmen der Dissertation haben wir untersucht, ob Einschlaf- und Aufwachzeitpunkte mit Smartphonesensoren geschätzt werden können [WA18].

Unser Schlaferkennungsalgorithmus besteht aus drei Hauptkomponenten: 1) Einem Klassifizierer, welcher die präferierten Schlafzeiten des Nutzers lernt und alleine anhand der Uhrzeit eine Vorhersage über den Schlaf-Wach-Zustand des Nutzers macht. 2) Einem weiteren Klassifizierer, welcher Sensordaten des Smartphones zur Vorhersage des Schlaf-Wach-Zustands nutzt. Hierbei werden Nutzeraktivität und die Konfidenz der Aktivitätsschätzung der Google ActivityRecognitionAPI verwendet. Des Weiteren werden Bildschirmnutzung, Umgebungsgeräuschlautstärke und die Varianz des Accelerometers herangezogen. 3) Wir verwenden das Two-Process-Model (TPM) von Daan et al. [DBB84], um die Vorhersagen der beiden Klassifizierer zu fusionieren und auf relevante Schlaf-Wach-Transitionen (SWT) zu filtern. Dabei modellieren wir den Schlafdruck für jeden Klassifizierer separat und filtern für jeden Tag aus allen erkannten SWTs die zwei treffendsten SWTs (je eine Aufwach- und eine Einschlaftransition) heraus.

Da wir unsere Schlaferkennung mit arbiträren Schlafmustern evaluieren wollten, wurde ein Simulator zur Datengenerierung entwickelt. Dabei wird das Schlafverhalten durch fünf Simulationsparameter charakterisiert: Den Mittelwert und die Standardabweichung der Schlafdauer, den Mittelwert und die Standardabweichung des Schlafmittelpunkts, sowie der Anzahl Stunden sozialen Jetlags. Sozialer Jetlag wird an den Wochenenden auf den Schlafmittelpunkt addiert. Mithilfe der Simulationsparameter werden dann eine Vielzahl Schlafepisoden generiert (durch Sampling aus den Normalverteilungen und ggf. Addition des sozialen Jetlags). Die generierten Schlafepisoden werden mit Blöcken von gemessenen Daten durch zufälliges Auswählen ohne Zurücklegen gefüllt. Dabei besteht ein Block aus einem oder mehreren identischen Messwerten (erste Ableitung aller Merkmale ist Null).

Zur Datenaufzeichnung wurde eine Smartphone Applikation an 30 Probanden zur Installation auf ihren eigenen Smartphones verteilt. Zusätzlich füllte jeder Proband ein handgeschriebenes Schlaftagebuch aus. Am Ende des Versuchszeitraums gaben allerdings nur 16 Probanden ein ausgefülltes Schlaftagebuch ab. Zwei weitere Probanden füllten den Abschlussfragebogen nicht aus, ein weiterer war die meiste Zeit krank. Schlussendlich umfasste der Datensatz 196 Nächte von 13 Probanden. Diese wurden mit Hilfe des Simulators zu einem simulierten Datensatz von 98280 Nächten (mit sozialem Jetlag) und 16380 Nächten (ohne sozialen Jetlag) erweitert.

Auf dem simulierten Datensatz ohne sozialen Jetlag erreicht unser Algorithmus im Mittel einen absoluten Erkennungsfehler von 40 bzw. 42 Minuten (relativer Fehler im Mittel -2 bzw. -5 Minuten.) für die Schätzung des Einschlaf- bzw. Aufwachzeitpunkts. Diese Er-

gebnisse befinden sich nah an der von Rogers et al. [RCA93] ermittelten Unschärfe von handgeschriebenen Schlaftagebüchern. Dabei verbessert die Anwendung des Expertenmodells die Erkennungsergebnisse um 47% für den Einschlafzeitpunkt und 58% für den Aufwachzeitpunkt. Die vorgeschlagene Schätzmethode erlaubt es uns, aus Smartphone-Daten die Schlaf-Wach-Transitionen zu schätzen ohne das Verhalten des Nutzers zu beeinflussen (z.B. durch Platzieren des Smartphones unter dem Kissen).

6 Schätzung der zirkadianen Phasenverschiebung

Licht verschiebt die zirkadiane Phase. Dabei ist die Intensität und der Zeitpunkt der Lichtexposition ausschlaggebend. Grob ausgedrückt verschiebt Licht am Morgen die zirkadiane Phase nach vorne, während Licht am Abend die zirkadiane Phase nach hinten verschiebt. Eine kontinuierliche Messung der Lichtexposition im Alltag gestaltet sich jedoch schwierig, da der Sensor im Gesicht angebracht werden muss, um eine belastbare Messwerte zu liefern. Hingegen sind Smartphones mit einer Vielzahl von Sensoren ausgestattet und bereits weit verbreitet. So waren 2016 im Median 68% der Bewohner aller Industriestaaten bereits im Besitz eines Smartphones [Po16]. In diesem Kapitel befassen wir uns mit der Schätzung der Lichtexposition mithilfe der im Smartphone verbauten Sensoren und Wetterinformationen am aktuellen Standort, welche das Smartphone aus dem Internet abruft.

Wir entwickelten eine Smartphone-Anwendung, welche aus Kontextinformationen wie Bewegungsintensität, Anzahl empfangener GPS Satelliten, GPS Positionsgenauigkeit, Lichtsensor und Näherungssensor in Kombination mit Wetterinformation die Lichtexposition des Smartphone-Nutzers schätzt. Dabei wird die Lichtintensität mit einem Regressor auf Basis von Gradienten-geboosteten Entscheidungsbäumen geschätzt. Das Kronauer Modell [JFK99] modelliert die Verschiebung der inneren Uhr. So kann aus der gemessenen oder geschätzten Lichtexposition die zu erwartende Phasenverschiebung ermittelt werden. Die Umwandlung von Lichtintensität in zeitliche Verschiebung der zirkadianen Phase erlaubt uns den Fehler der Schätzung in der Zeitdomäne zu beurteilen.

Zur Evaluierung wurden Daten von 12 Probanden für sechs aufeinanderfolgende Tage aufgezeichnet. Die Probanden trugen ein Lichtmessgerät, welches an einem Brillengestell fixiert war, einen Aktigraphen mit integriertem Lichtsensor am Handgelenk, sowie ein Smartphone mit sich. Dabei diente das Lichtmessgerät am Kopf als Referenz. Die Evaluierung wurde mit einer Leave-One-Participant-Out Kreuzvalidierung durchgeführt. Anschließend wird der Schätzfehler der zu erwartenden Phasenverschiebung für den Tag i als $e_i = t_{m,i} - t_{s,i}$ berechnet, wobei $t_{m,i}$ die errechnete Verschiebung aus der Kopfmessung am Tag i und $t_{s,i}$ die errechnete Verschiebung aus der Smartphone Schätzung am Tag i darstellt. Der Summenfehler über den gesamten Beobachtungszeitraum $e_s = \sum_i e_i$ wird ebenfalls berechnet. Analog dazu wurden die Fehler des Aktigraphen im Vergleich zur Messung am Kopf berechnet.

Unsere Smartphone-basierte Lichtschätzung hat gegenüber der Messung am Kopf einen absoluten Summenfehler $|e_s|$ von 32 ± 16 Minuten über einen Messzeitraum von sechs Tagen. Die Messung des Aktigraphen im Vergleich zur Messung am Kopf zeigte einen absoluten Summenfehler von 57 ± 24 Minuten. Unsere Schätzmethode lieferte dabei geringere

absolute Summenfehler für 11 von 12 Probanden. Dies zeigt, dass unsere Schätzmethode gegenüber einer Messung an einem ungeeigneten Ort, dem Handgelenk, überlegen ist. Denn eine Lichtmessung am Handgelenk ist fehlerbehaftet, z.B. durch das Verdecken des Sensors mit langärmliger Kleidung. Ein Vergleich in der Lichtdomäne wäre für unsere Anwendung wenig aussagekräftig gewesen. Die Kombination von Methoden des maschinellen Lernens und einem Expertenmodell, was es ermöglicht den Schätzfehler in eine relevante Domäne (hier zeitliche Phasenverschiebung) umzuwandeln, hat sich als vielversprechend erwiesen. Mit der von uns vorgestellten Methode ist eine Vorhersage der zirkadianen Phasenverschiebung im Alltag anhand von Smartphonedaten möglich.

7 Schlußfolgerungen

In der Dissertation werden neue Geräte, Methoden und Algorithmen aufgezeigt, um die innere Uhr im Alltag zu beobachten. Die geleisteten wissenschaftlichen Beiträge teilen sich in drei Kategorien: 1) Die Integration von Sensoren in ein personalisiertes Brillengestell, um Daten im Alltag aufzuzeichnen. 2) Methoden zur Kontext- und Aktivitätserkennung von Alltagsaktivitäten, im Besondern die Benutzung von Bildschirmen. 3) Neue Algorithmen, welche Expertenmodelle und Methoden des maschinellen Lernens verknüpfen, um die Erkennung von Schlafzeiten sowie die Schätzung der zirkadianen Phasenverschiebung aus Smartphonedaten zu verbessern.

Soziale Akzeptanz ist der Schlüssel zum Erfolg von körpergetragenen Sensoren. Daher integrierten wir Sensoren in Brillen, die ein bereits sozial akzeptiertes und weit verbreitetes Accessoire sind. Der Kopf ist eine hervorragende Messposition, da sich anhand der Kopf-Bewegungsmuster viele Alltagsaktivitäten erkennen lassen. Außerdem lässt sich mit der Brille ein Lichtsensor zwischen den Augen des Trägers platzieren. Elektrische Komponenten und Stromversorgung verschwinden geschickt hinter den Ohren des Trägers. Brillen werden darüber hinaus selten vergessen, da ihre Träger auf die Sichtkorrektur angewiesen sind. Dies stellt z.B. gegenüber einem Armband einen erheblichen Vorteil dar. Dank unserem digitalen Anpassungsprozess können auf den Träger personalisierte Brillen mit nur drei Messwerten vom Kopf hergestellt werden. In einer Untersuchung mit vier Teilnehmern haben wir den Tragekomfort und die Passform untersucht. Dabei trugen die Probanden eine an sie angepasste Brille für einen Tag. Die Ergebnisse der Evaluierung zeigten, dass die Probanden zu ihren gewöhnlichen Brillen keinen Unterschied im Tragekomfort empfanden.

Unsere Brillen wurden zur Erkennung von Alltagsaktivitäten (ADLs) eingesetzt. Dabei war es uns möglich neun Gruppen von ADLs mit einer durchschnittlichen Genauigkeit von 77% zu erkennen. Kenntnis über die tägliche Routine ist wichtig, um Verhalten zu identifizieren welches zur zirkadianen Phasenverschiebung beiträgt. Bildschirmnutzung bei Nacht hemmt die Melatoninproduktion und führt so zu einer Verschiebung der zirkadianen Phase. Mit dem Lichtsensor der Brille können wir bei Umgebungslichtintensitäten von unter 200 lux die Benutzung von Computerbildschirmen mit einer Genauigkeit von 0.9 ROC AUC erkennen. So kann der Nutzer gewarnt werden oder eine automatische Anpassung des Farbprofils vorgenommen werden.

Expertenmodelle wurden in der Schlaferkennung und der Schätzung der Phasenverschiebung mit Smartphones verwendet. Bei der Schlaferkennung haben wir das Two Process Model von Daan et al. [DBB84] angewandt, um die Ausgänge der Klassifizierer basierend auf Uhrzeit und Smartphone Sensordaten zu fusionieren und die relevanten Schlaf-Wach-Transitionen zu filtern. Dank des Expertenmodells konnte die Erkennung des Einschlafzeitpunkts um 47% und des Aufwachzeitpunkts um 58% verbessert werden. Damit erreichen wir eine ähnliche Genauigkeit wie handgeschriebene Schlaftagebücher aber bereiten dem Probanden kaum Arbeit, da die Smartphone-Anwendung im Hintergrund arbeitet.

Die Anwendung des Kronauer Modells [JFK99] auf die von Smartphones geschätzte Lichtexposition erlaubt es uns, die Schätzung von Lichtintensität in eine Schätzung der zu erwartenden Phasenverschiebung zu transformieren. Somit können Daten von Smartphonesensoren und Wetterdaten aus dem Internet die zu erwartende zirkadiane Phasenverschiebung mit einem akkumulierten Fehler von 32 ± 17 Minuten über sechs Tage Versuchsdauer im Vergleich zu einem am Kopf getragenen Referenz-Messgeräts bestimmen. Dabei war unsere Schätzung einer Messung am Handgelenk bei 11 von 12 Probanden überlegen. Mit dieser Methodik ist eine kontinuierliche Vorhersage der Phasenverschiebung möglich. Dies erlaubt Nutzern den Effekt bestimmter Aktivitäten, z.B. einem Spaziergang nach dem Mittagessen, direkt einzuschätzen.

Die in der Dissertation vorgestellten Methoden erlauben ein kontinuierliches Monitoring der zirkadianen Phase und geben Nutzern und Forschern die Möglichkeit den Einfluss einzelner Aktivitäten auf die innere Uhr besser zu verstehen. Durch die Einbindung von Expertenmodellen wird wertvolles, bereits vorhandenes Domänenwissen in entscheidungsunterstützende Systeme integriert. Im Gegensatz zu neuronalen Netzen, welche die Modellstruktur direkt aus den Daten lernen, führt die Kombination klassischer Verfahren des maschinellen Lernens mit Expertenmodellen zu besser verständlichen Entscheidungen.

Literaturverzeichnis

[DBB84] Daan, S.; Beersma, D. G.; Borbély, Alexander A.: Timing of Human Sleep: Recovery Process Gated by a Circadian Pacemaker. American Journal of Physiology - Regulatory, Integrative and Comparative Physiology, 246(2):R161–R183, Februar 1984.

[JFK99] Jewett, Megan E.; Forger, Daniel B.; Kronauer, Richard E.: Revised Limit Cycle Oscillator Model of Human Circadian Pacemaker. Journal of Biological Rhythms, 14(6):493–500, 1999.

[Po16] Poushter, Jacob: Smartphone Ownership and Internet Usage Continues to Climb in Emerging Economies. Pew Research Center, 22, 2016.

[RCA93] Rogers, Ann E.; Caruso, Claire C.; Aldrich, Michael S.: Reliability of Sleep Diaries for Assessment of Sleep/Wake Patterns. Nursing research, 42(6):368–371, 1993.

[Ro07] Roenneberg, Till; Kuehnle, Tim; Juda, Myriam; Kantermann, Thomas; Allebrandt, Karla; Gordijn, Marijke; Merrow, Martha: Epidemiology of the Human Circadian Clock. Sleep Medicine Reviews, 11(6):429–438, Dezember 2007.

[Wa17] Wahl, Florian; Zhang, Rui; Freund, Martin; Amft, Oliver: Personalizing 3D-Printed Smart Eyeglasses to Augment Daily Life. IEEE Computer, 50(2):26–35, 2017.

[WA18] Wahl, Florian; Amft, Oliver: Data and Expert Models for Sleep Timing and Chronotype Estimation from Smartphone Context Data and Simulations. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol., 2(3):139:1–139:28, September 2018.

[Wa19] Wahl, Florian: Methods for monitoring the human circadian rhythm in free-living. Dissertation, Universität Passau, 2019.

[WFA15] Wahl, Florian; Freund, Martin; Amft, Oliver: WISEglass: Smart Eyeglasses Recognising Context. EAI Endorsed Transactions on Pervasive Health and Technology, 2(5), 2015.

[WKA17] Wahl, Florian; Kasbauer, Jakob; Amft, Oliver: Computer Screen Use Detection Using Smart Eyeglasses. Front. ICT, 4, 2017.

[ZBA16] Zhang, Rui; Bernhart, Severin; Amft, Oliver: Diet Eyeglasses: Recognising Food Chewing Using EMG and Smart Eyeglasses. In: Proceedings of the International Conference on Wearable and Implantable Body Sensor Networks (BSN' 16). IEEE, S. 7–12, 2016.

Florian Wahl wurde am 24. Mai 1985 in Ostfildern-Ruit geboren. Er studierte Technische Informatik an der Fachhochschule Esslingen. Seine Bachelorthesis schrieb er bei Mercedes Benz Research & Development in Palo Alto, Kalifornien. Anschließend studierte er Embedded Systems an der Technischen Universität Eindhoven. In seiner Masterarbeit entwickelte er einen autonomen, solar-betriebenen Sensor mit drahtloser Kommunikation zur Zählung von Menschen in Bürogebäuden. Seine Dissertation begann er an der Technischen Universität Eindhoven im Jahr 2012 in der Forschungsgruppe von Prof. Dr. Oliver Amft. In seiner Dissertation entwickelt er Methoden zum Monitoring zirkadianer Rhythmen im Alltag. In 2014 zog die Forschungsgruppe an die Universität Passau um, wo Florian im Jahr 2019 seine Dissertation mit Auszeichnung verteidigte. Seit 2018 arbeitet er als wissenschaftlicher Mitarbeiter am Technologie Campus Grafenau der Technischen Hochschule Deggendorf im Bereich angewandte künstliche Intelligenz.

Skalierbare & Push-basierte Echtzeitanfragen für Pull-basierte Datenbanken[1]

Wolfram Wingerath[2]

Abstract: Traditionelle Datenbanksysteme sind für Pull-basierte Anfragen optimiert, d.h. sie stellen Informationen als direkte Antwort auf Anfrage eines Klienten zur Verfügung. Dieses Zugriffsmuster ist zwar für überwiegend statische Domänen praktikabel, erfordert allerdings ineffiziente und langsame Workarounds (z.B. periodische Neuauswertung einer Anfrage), wenn die Klienten auf dem neuesten Stand gehalten werden müssen. Moderne Echtzeitdatenbanken beheben diesen Mangel zwar konzeptuell, indem sie Ergebnisaktualisierungen durch Push-basierte Echtzeitanfragen proaktiv an ihre Klienten ausliefern. Die derzeitig auf dem Markt befindlichen Systeme sind jedoch nur von begrenzter praktischer Relevanz, da sie schwer in bestehende Anwendungen zu integrieren sind, mangelhafte Skalierbarkeit aufweisen oder komplexe Anfragen von vornherein nicht unterstützen. Um diese Probleme zu lösen, schlagen wir in dieser Dissertation das Systemdesign InvaliDB vor, welches lineare Lese- und Schreibskalierbarkeit für ausdrucksmächtige Echtzeitanfragen als Opt-in-Feature für Pull-basierte Datenbanksysteme bereitstellt. InvaliDB befindet sich seit Juli 2017 im produktiven Einsatz als Teil der Backend-as-a-Service-Plattform der Firma Baqend.

1 Motivation: Push-basierte Datenbanken braucht das Land

Die hier vorgestellte Arbeit [Wi19] behandelt ein skalierbares Systemdesign, welches traditionelle Pull-basierte Datenbanksysteme um Push-basierte Echtzeitanfragen erweitert.

Heute informieren immer mehr (Web-)Anwendungen ihre Nutzer über Status-Updates und andere Ereignisse in Echtzeit und so gehen Nutzer oft schon davon aus, dass sich die Aktivitäten der anderen im eigenen Benutzungskontext widerspiegeln. Beispiele für derartige Anwendungen sind kollaborative Online-Dokumente und Webseiten, die auf soziale Interaktion abzielen (soziale Medien). Datenzugriff ist hier *Push-basiert*, weil Informationen sofort bei Veröffentlichung proaktiv an den Nutzer herangetragen („gepusht“) werden. Traditionelle **Datenbankmanagementsysteme** (DBMSs oder schlicht Datenbanken) [He07] sind allerdings auf *Pull-basierten* Datenzugriff ausgelegt, bei dem Informationen nur auf Anfrage des Klienten übermittelt werden. Trotz Erweiterung um Trigger und andere Push-orientierte Konzepte werden traditionelle Datenbanken bei Push-basierten Zugriffsmustern von nativ Push-basierten Systemen hinsichtlich der Performance um mehrere Größenordnungen übertroffen [SC05]. Die Unzulänglichkeit traditioneller Datenbanktechnologie für den Umgang mit sich schnell ändernden Daten wird allgemein als eine der grundlegenden Herausforderungen beim Entwurf von Datenbanksystemen angesehen [SCZ05].

Um Aktualisierungen mit geringer Latenz in Domänen mit hoher Änderungsrate zu gewährleisten, brechen **Datenstrommanagementsysteme** (DSMSs) [GZ10] mit der Idee

[1] Englischer Titel der Dissertation: „Scalable Push-Based Real-Time Queries on Top of Pull-Based Databases“

[2] wolle@baqend.com, Baqend GmbH (vorgeschlagen durch die Universität Hamburg)

CC BY SA

eines persistent Datenspeichers. Statt beliebiger ad hoc-Anfragen über Collections (Datensammlungen) werten sie sequentielle, lang laufende Anfragen über Datenströme aus. DSMSs erzeugen neue Ausgaben, sobald neue Daten verfügbar werden, und sind daher nativ Push-basiert. Die Daten können jedoch nur in einem einzigen Durchgang verarbeitet werden, da Datenströme konzeptionell *unbegrenzte* Sequenzen von Datenelementen sind und daher nicht für beliebige Zeit aufbewahrt werden können. Datenstromanfragen können daher nur über Daten ausgewertet werden, die nach ihrer Aktivierung eingehen.

	Datenbankmanagement	**Datenstrommanagement**
Datenzugriff	Pull-basiert	Push-basiert
Datenmodell	persistente Collections	vergängliche Datenströme
Anfrageauswertung	ad hoc, random access	kontinuierlich, sequenziell

Tab. 1: Ein direkter Vergleich der Kernmerkmale von Datenbank- und Datenstrommanagement.

Datenbank- und Datenstrommanagement folgen jeweils grundlegend unterschiedlichen Semantiken bei Datenverarbeitung und -zugriff, wie Tabelle 1 zusammenfasst. Datenbankanfragen über persistente Collections sind für Anwendungen ideal, die eine (konsistente) Sicht auf ihre Domäne erfordern wie beispielsweise Finanzbuchhaltung oder Lagerbestandsverwaltung. Das Konzept eines Datenstroms hingegen adressiert Anwendungen, in denen Ereignisabläufe oder die Beziehung zwischen einzelnen Ereignissen betrachtet werden, z.B. um Aktienkurse zu analysieren oder bösartiges Benutzerverhalten zu identifizieren. Das Zugriffsparadigma – Pull- oder Push-basiert – ist dabei an das Datenmodell gebunden: Traditionelle Datenbanksysteme sind Pull-basiert und bieten keine Unterstützung für kontinuierliche Anfragen mit automatischer Ergebnisaktualisierung, wohingegen nativ Push-basierte Systeme zur Datenstromverwaltung nur begrenzte Möglichkeiten für den Zugriff auf Archivdaten bieten. Die Trennung zwischen diesen beiden Systemklassen bedingt sowohl hohe Komplexität als auch hohe Wartungskosten bei Anwendungen, die gleichzeitig Persistenz und Echtzeitbenachrichtigungen bei Ergebnisänderungen erfordern.

Zum Schließen der Kluft zwischen beiden Paradigmen wurde eine neue Klasse von Informationssystemen gebildet, welche die Collection-basierte Semantik von DBMSs mit dem Push-basierten Zugriffsmodell von DSMSs kombiniert: **Echtzeitdatenbanken** („real-time databases“, RTDBs) halten die vom Klienten angefragten Daten kontinuierlich mit dem Datenbankzustand synchron und übermitteln alle Änderungen proaktiv „in Echtzeit“, d.h. unmittelbar nach jeder Änderung. Wie traditionelle Datenbankanfragen werden **Echtzeitanfragen** in RTDBs über persistenten Collections ausgewertet, welche konsistente Momentaufnahmen der modellierten Domäne repräsentieren. Gleichzeitig liefern Echtzeitanfragen allerdings nicht nur ein einziges Ergebnis zurück, sondern auch kontinuierliche Aktualisierungen, ähnlich wie Anfragen im Datenstrommanagement.

2 Echtzeitdatenbanken für kontinuierliche Ergebnisaktualisierung

In der Vergangenheit wurde der Begriff „Echtzeitdatenbanken“ („real-time databases“) für spezialisierte Pull-basierte DBMSs verwendet, die eine Ausgabe innerhalb strikter Zeitvorgaben produzieren [Pu93] [AH98]. Im Rahmen dieser Arbeit sind **Echtzeitdatenbanken**

jedoch Systeme, die einen Push-basierten Zugriff auf Collections in einer Datenbank ermöglichen. Populäre Vertreter sind beispielsweise Firebase, Meteor und RethinkDB. Entsprechend bezeichnen wir mit **Echtzeitanfragen** Push-basierte Anfragen, welche der gleichen Collection-basierten Anfragesemantik wie gewöhnliche Datenbankanfragen folgen, aber zusätzlich zum initialen Ergebnis auch einen kontinuierlichen Strom an Aktualisierungen liefern. Die in unserem Fokus stehenden **Echtzeitanwendungen** sind also reaktive oder interaktive Applikationen, die dem Benutzer neue Informationen so schnell wie möglich zur Verfügung stellen, nachdem sie in der Datenbank festgeschrieben wurden. Es handelt sich hierbei um Nutzungsszenarien mit weichen zeitlichen Beschränkungen („soft timing constraints“) und explizit nicht um sicherheitskritische oder andere Anwendungen, die strikte Obergrenzen für die Reaktionszeiten festlegen, wie z.B. Flugkontrollsysteme oder Kernkraftwerkskontrollen [St88].

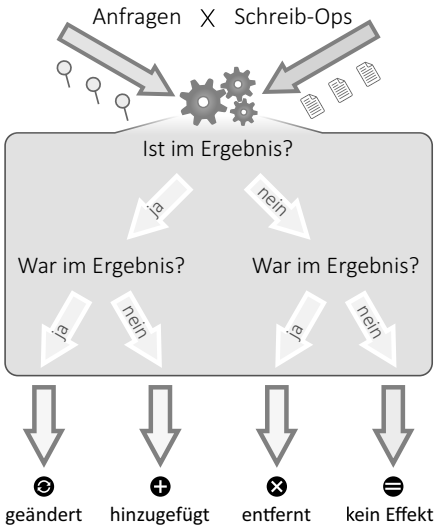


Abb. 1: Eine Echtzeitdatenbank muss jede Echtzeitanfrage (blauer Pfeil) mit jeder eingehenden Schreiboperation (roter Pfeil) abgleichen, um Ergebnisänderungen (grüne Pfeile) zu detektieren.

Konzeptuell haben die Informationen einer Echtzeitanfrage zwei Komponenten: das initiale Ergebnis und einen Strom an nachfolgenden Änderungsbenachrichtigungen. Das **initiale Ergebnis** entspricht den Daten, die von einer herkömmlichen ad hoc-Datenbankabfrage zurückgegeben würden, und erfasst somit die Datenelemente, die bei der Aktivierung der Anfrage mit dem Anfrageprädikat übereinstimmen. Anders als bei einer traditionellen Datenbankanfrage bleibt jedoch die Verbindung zwischen Klient und Datenbank bestehen und es werden **Änderungsbenachrichtigungen** („change events“) für alle Ergebnisänderungen übermittelt. Auf Basis dieser Informationen kann der Klient das Ergebnis auf dem neuesten Stand halten.

Ergebnisänderungen für eine aktive Echtzeitanfrage werden dabei durch einen kontinuierlichen Prozess ermittelt, der für alle eingehenden Schreiboperationen überprüft, ob und wie sie das Ergebnis der Anfrage beeinflussen. Zur Illustration erläutern wir diesen Prozess hier für unsortierte Filteranfragen über einer einzigen Collection und verweisen auf [Wi19, Kapitel 3] für eine Betrachtung komplexerer Echtzeitanfragen mit Ergebnissortierung, Aggregationen und Joins.

Abbildung 1 veranschaulicht, wie dieser Prozess auf zwei einfache Fragen heruntergebrochen werden kann, die für jeden geschriebenen Datensatz beantwortet werden müssen:

1. Wird das Anfrageprädikat nach dem Schreibvorgang erfüllt? („Ist im Ergebnis?“)
2. War das Anfrageprädikat vor dem Schreibvorgang erfüllt? („War im Ergebnis?“)

Wenn der Datensatz nach der Schreiboperation das Prädikat der Anfrage erfüllt (linker Zweig im Entscheidungsbaum), gibt es nur zwei Möglichkeiten: Entweder war der Daten-

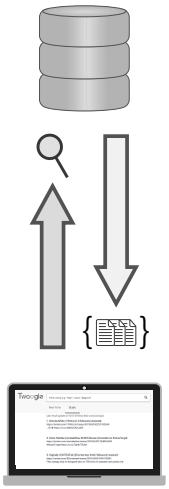
satz bereits Teil des Ergebnisses und wurde lediglich *geändert* oder er wurde durch die Schreiboperation zum Ergebnis *hinzugefügt*. Ähnlich verhält es sich, wenn ein Datensatz nach einer Schreiboperation nicht in das Anfrageergebnis fällt (rechter Zweig): Entweder war der Datensatz vor der Schreiboperation Teil des Ergebnisses und wurde folglich gerade *entfernt* oder die Schreiboperation hatte *keinen Effekt*.

Bei vielen aktiven Echtzeitanfragen oder hohem Schreibdurchsatz wird dieser kontinuierliche Prozess sehr aufwendig. Um den Ressourcenbedarf zu veranschaulichen, betrachten wir eine Applikation mit 1 000 aktiven Benutzern und einem durchschnittlichen Durchsatz von 1 000 Schreiboperationen pro Sekunde. Selbst bei nur einer Echtzeitanfrage pro Benutzer und bei sehr simplen Filteranfragen mit nur einem Prädikat muss die Echtzeitdatenbank bereits 1 Million Vergleichsoperationen durchführen – pro Sekunde. Für komplexe Anfragen ist der **Aufwand** noch größer, beispielsweise wenn ein bestimmter Sortierschlüssel zur Auswertung von Limit- oder Offset-Klauseln beachtet werden muss oder wenn Aggregationen oder Joins durchgeführt werden müssen. Durch gewisse „Optimierungen“ (z.B. Stapelverarbeitung) kann der Durchsatz auf Kosten der Latenz verbessert werden. Ebenso lässt sich durch effiziente Indizierung bei manchen Prädikaten eine bessere als die hier angenommene quadratische Komplexität erreichen. Wenn jedoch minimale Latenz zwingend erforderlich ist, gibt es für ein Echtzeitdatenbanksystem keine Alternative dazu, jede Schreiboperation im Kontext jeder aktiven Echtzeitanfrage zu betrachten. Eine skalierbare Implementierung der Anfrageauswertung ist daher essenziell, um Echtzeitdatenbanken in der Praxis nutzbar zu machen.

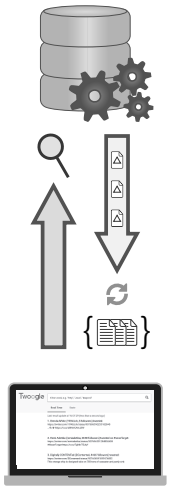
3 Anwendungsentwicklung mit Echtzeitanfragen

Es gibt zwei verschiedene Arten von Echtzeitanfragen, die sich darin unterscheiden, wie Daten der Anwendung zur Verfügung gestellt werden: **Ereignisstromanfragen** („event stream queries“) geben direkt das initiale Ergebnis und danach die unbearbeiteten Änderungsbenachrichtigungen an die Anwendung weiter, so dass diese das Ergebnis aktualisieren oder Geschäftslogik anwenden kann. **Selbsterneuernde Anfragen** („self-maintaining queries“) sind abstrakter, da sie die Ergebnisaktualisierung für die Anwendung transparent machen und bei jeder Änderung das vollständige (aktualisierte) Anfrageergebnis zur Verfügung stellen. Dadurch kann reaktives Verhalten in der Anwendung implementiert werden, ohne dass in der Geschäftslogik Konzepte zur Verarbeitung von Datenströmen oder Änderungsdeltas umgesetzt werden müssen. Durch Einsatz von selbsterneuernden Anfragen können einige statische Anwendungen somit in Echtzeitanwendungen umgewandelt werden, ohne signifikante Änderungen am Anwendungscode vorzunehmen.

Bei Anfrageschnittstellen, die auf Callback-Funktionen basieren, ist der Übergang zwischen statischem und Echtzeitverhalten besonders fließend. Dabei wird eine Anfrage asynchron ausgeführt und der Main-Thread der Anwendung wartet nicht auf das Eintreffen des Ergebnisses. Stattdessen wird mittels Callback-Funktion vorab festgelegt, wie das Ergebnis verarbeitet werden soll, wenn es schließlich eintrifft: Es wird dann nur noch die Callback-Funktion mit dem Ergebnis als Argument aufgerufen. Wenn die Anfrage als statische ad hoc-Anfrage ausgeführt wird (siehe Abbildung 2a), wird die Callback-



(a) Eine statische ad hoc-Anfrage liefert nur ein einziges Ergebnis, welches den Datenbankzustand zum Zeitpunkt der Anfrage repräsentiert.



(b) Eine selbsterneuernde Anfrage liefert eine Sequenz von Ergebnissen, von denen jedes einzelne die jeweils letzte Änderung widerspiegelt.

Abb. 2: Während eine statische ad hoc-Anfrage *Pull-basiert* ist und somit nur einmalig Informationen an den ausführenden Klienten zurückgibt, liefert eine selbsterneuernde Echtzeitanfrage nach jeder Änderung *proaktiv* eine aktualisierte Instanz des Ergebnisses aus.

Funktion nur einmal aufgerufen. Wenn die Anfrage als selbsterneuernde Echtzeitanfrage ausgeführt wird (siehe Abbildung 2b), wird die Callback-Funktion dagegen einmal aufgerufen, wenn das ursprüngliche Ergebnis zurückgegeben wird, und dann wiederum jedes Mal, wenn sich die zugrundeliegenden Daten geändert haben. Die Anwendung verhält sich also effektiv so, als ob eine neue statische Anfrage immer dann ausgeführt würde, wenn sie ein anderes Ergebnis als der letzte Aufruf zurückgeben würde. Als einzige Anforderung der Anwendung muss dazu jede Callback-Funktion idempotent implementiert werden, so dass ein Aufruf alle Auswirkungen des vorherigen Aufrufs überschreibt. Wenn die Callback-Funktion beispielsweise[3], das Ergebnis einer benutzerdefinierten Suche in ein HTML-Element rendert und dabei das zuletzt gerenderte Ergebnis ersetzt, verhält sich die Anwendung sowohl bei Pull-basierten ad hoc-Anfragen als auch bei Push-basierten selbsterneuernden Anfragen semantisch korrekt. Würde das Suchergebnis hingegen lediglich angehängt (ohne vorherige Entfernung des zuletzt gerenderten Ergebnisses), wäre die Darstellung ab dem zweiten Aufruf der Callback-Funktion nicht mehr sinnvoll.

4 Problemstellungen & Forschungsfrage

Konzeptuell können Echtzeitdatenbanken als Erweiterung zu traditionellen Datenbanksystemen verstanden werden, da sie der gleichen Anfragesemantik folgen, aber zusätzlich zu Pull- auch Push-basierten Zugriff bieten. In der Praxis hat sich jedoch keine Architektur für Echtzeitdatenbanken bewährt, die auf einem existierenden System aufsetzt. Da heutige

[3] Für eine Demonstration der im Rahmen dieser Arbeit entwickelten Implementierung siehe [WGR19b].

Vertreter meist von Grund auf neu entwickelt wurden, profitieren sie daher auch nicht vom reichhaltigen Funktionsumfang und der Stabilität, die einige Pull-basierte Systeme über Jahrzehnte erworben haben. Bisher wurde kein Mechanismus für die Bereitstellung von Push-basierten Echtzeitanfragen entwickelt, der jede der folgenden Challenges löst:

- $\mathbf{C_1}$ **Skalierbarkeit.** Die Bereitstellung einer Echtzeitanfrage ist ressourcenintensiv, da sie die kontinuierliche Überwachung aller Schreibvorgänge erfordert, die das Anfrageergebnis beeinflussen könnten. Für erhöhte Skalierbarkeit verteilen heutige Echtzeitdatenbanken dabei die Anfragen oft über mehrere Server, sodass jeder Server nur für eine Teilmenge aller Anfragen verantwortlich ist. Keines der heutigen Systeme verteilt jedoch gleichzeitig auch die Schreiblast und daher muss jeder Server stets alle eingehenden Schreiboperationen verarbeiten: Der maximale Schreibdurchsatz des Gesamtsystems bleibt somit limitiert durch die Kapazität der einzelnen Server.
- $\mathbf{C_2}$ **Ausdrucksmächtigkeit.** Die meisten APIs (Application Programming Interfaces) für Echtzeitanfragen sind im Vergleich zu ihren Pull-basierten Pendants stark limitiert. Aggregationen sind in der Regel nicht verfügbar und sortierte Anfragen werden oft nicht unterstützt oder haben starke Einschränkungen; so gibt es beispielsweise Implementierungen, die nur Sortierschlüssel mit einem einzigen Attribut erlauben oder eine Limit-, aber keine Offset-Klausel anbieten. Das Fehlen derartig grundlegender Funktionalität auf der Datenbankseite macht selbst bei relativ simplen Zugriffsmustern ineffiziente Workarounds im Anwendungscode erforderlich, weil Teile der Anfrageverarbeitung klientenseitig stattfinden müssen.
- $\mathbf{C_3}$ **Rückwärtskompatibilität.** Heutige Echtzeitdatenbanken basieren auf Neuentwicklungen oder NoSQL-Systemen, die keinen Standards bzgl. Datenmodell oder Anfragesprache folgen. Sie implementieren proprietäre Protokolle für Pull- und Push-basierten Datenzugriff gleichermaßen und die Schnittstellen unterschiedlicher Systeme sind untereinander weitgehend inkompatibel. Ebenso lassen sie sich nur schwer mit traditionellen Datenbanken integrieren, sodass bestehende Anwendungen nur mit sehr hohem Aufwand um Push-basierte Anfragen erweitert werden können.
- $\mathbf{C_4}$ **Abstrakte Schnittstellen.** Viele APIs für Echtzeitanfragen exponieren Eigenheiten der zugrundeliegenden Implementierung und bieten so nur eine geringe Datenunabhängigkeit. Viele Echtzeitdatenbanken bieten etwa keine automatische Ergebnisaktualisierung (vgl. selbsterneuernde Anfragen) und erfordern daher ein Verständnis von Systeminterna wie der Struktur und Semantik von Änderungsbenachrichtigungen. Derartige Schnittstellen sind hochgradig ineffizient, da sie Entwickler zum Lösen von Problemen außerhalb der eigentlichen Anwendungsdomäne zwingen.

In dieser Arbeit vertreten wir die Auffassung, dass keine der obigen Einschränkungen inhärent mit der Herausforderung verbunden ist, Push-basierte Echtzeitanfragen über Datenbank-Collections bereitzustellen. Wir gehen daher folgender Forschungsfrage nach:

Forschungsfrage: *Wie können ausdrucksmächtige Push-basierte Echtzeitanfragen für eine existierende Pull-basierte Datenbank skalierbar und generisch bereitgestellt werden?*

Um unseren Standpunkt zu belegen und die obige Forschungsfrage zu beantworten, entwickeln und implementieren wir ein generisches Systemdesign für Echtzeitdatenbanken,

welches Herausforderungen C_1 bis C_3 löst. Anschließend adressieren wir Herausforderung C_4, indem wir den Prototypen in eine bestehende Datenbank-Middleware integrieren und ihre rein Pull-basierte Anfrageschnittstelle um Push-basierte Echtzeitanfragen erweitern.

5 Hauptbeiträge der Arbeit

Wir präsentieren vier Hauptbeiträge. Unser erster Beitrag ist eine **Klassifikation** der Push-basierten Datenzugriffsmechanismen im heutigen Datenmanagement. Durch die Identifikation und Analyse der Mängel am aktuellen Stand der Technik leiten wir wichtige Erkenntnisse für unser Design ab und motivieren unsere Arbeit weiter. Unser zweiter Beitrag ist das **Systemdesign** InvaliDB, welches Push-basierte Echtzeitanfragen auf einer rein Pull-basierten Datenbank ermöglicht. InvaliDB ist hochskalierbar (vgl. C_1), unterstützt eine breite Palette an Anfragetypen (vgl. C_2) und ist als Erweiterung von traditionellen – rein Pull-basierten – Datenbanksystemen konzipiert (vgl. C_3). Unser dritter Beitrag besteht in der **Implementation** und ausführlichen experimentellen Evaluation unseres Designs. Als vierten Beitrag demonstrieren wir die Praktikabilität unseres Ansatzes durch die **Integration** des Prototyps mit der Pull-basierten Datenbank-Middleware Orestes. Das integrierte System bietet Entwicklern eine Schnittstelle für Zugriff auf die unbearbeiteten Änderungsbenachrichtigungen und eine für intuitiv nutzbare selbsterneuernde Anfragen (vgl. C_4).

- **H_1 Kategorisierung der heutigen Systemlandschaft.** Um ein intuitives Verständnis dessen zu vermitteln, was Echtzeitdatenbanken einzigartig macht, stellen wir ein Klassifikationsschema für Datenmanagementsysteme vor, das sich um die Unterstützung von Pull- und Push-basierten Anfragen in den jeweiligen Systemklassen dreht [WRG19]. Zu diesem Zweck vergleichen wir die verschiedenen Push-basierten Anfragemechanismen in traditionellen Datenbanksystemen, Echtzeitdatenbanken und Systemen zur Verwaltung und Verarbeitung von Datenströmen. Dabei heben wir jeweils die konzeptionellen Unterschiede zwischen Echtzeitdatenbanken und den anderen Systemklassen hervor. Darüber hinaus untersuchen wir die Limitationen bestehender Echtzeitdatenbanken und diskutieren, durch welche Designentscheidungen sie verursacht und wie sie vermieden werden können.
- **H_2 Systemdesign für skalierbare Echtzeitanfragen.** Aufbauend auf unseren Erkenntnissen präsentieren wir *InvaliDB* als ein skalierbares Systemdesign, welches Push-basierte Echtzeitanfragen auf einer Pull-basierten Datenbank ermöglicht. InvaliDB unterscheidet sich von bestehenden Systementwürfen durch (1) ein einzigartiges Verfahren zur Lastverteilung für *lineare Skalierbarkeit*, (2) Unterstützung für *ausdrucksmächtige Echtzeitanfragen* mit Ergebnissortierung, Aggregationen und Joins, (3) eine modulare Anfrage-Engine für *Datenbankunabhängigkeit* und (4) die *strikte Trennung* zwischen der Pull-basierten Datenbank und dem Subsystem für Echtzeitanfragen, um ihre Fehlerdomänen effektiv zu entkoppeln und eine unabhängige Skalierung zu ermöglichen.
- **H_3 Implementation auf MongoDB.** Um InvaliDBs Realisierbarkeit zu demonstrieren, entwickeln wir einen InvaliDB-Prototypen auf Basis des verteilten Stream Processors Storm und der Hauptspeicherdatenbank Redis für Push-basierte Echtzeitanfra-

gen auf der Pull-basierten Dokumentendatenbank MongoDB. In einer ausführlichen experimentellen Evaluation zeigen wir, dass der für eine InvaliDB-Instanz tragbare Durchsatz bei der Anfrageverarbeitung („sustainable query matching throughput“) linear mit der Anzahl der für die Anfrageverarbeitung eingesetzten Server skaliert, wobei die Ende-zu-Ende-Latenz über alle InvaliDB-Konfigurationen hinweg konstant niedrig bleibt und selbst bei Spitzenlast selten 100ms übersteigt.

H_4 Integration mit einer kommerziellen Datenbank-Middleware. Um die praktische Anwendbarkeit unseres Ansatzes zu demonstrieren, integrieren wir unseren Prototypen in die Datenbank-Middleware Orestes [Ge19] in der Quaestor-Architektur [Ge17]. Hier erfüllt InvaliDB zwei sehr unterschiedliche Aufgaben. Erstens stellt es für Klienten Push-basierte Echtzeitanfragen bereit, die von der Middleware selbst nicht unterstützt werden: Selbsterneuernde Anfragen sind leicht zu handhaben und eignen sich für Anwendungen, die lediglich eine aktuelle Sicht auf ihre kritischen Daten benötigen, während die hochflexiblen Ereignisstromanfragen für komplexere Zugriffsmuster wie benutzerdefinierte Echtzeit-Joins oder -Aggregationen geeignet sind. Zweitens ermöglicht InvaliDB ein konsistentes Caching der Ergebnisse von Pull-basierten Anfragen, indem es die Anfrageergebnisse kontinuierlich überwacht und die entsprechenden Caches bei einer Änderung sofort invalidiert. In einer weiteren experimentellen Evaluation zeigen wir, dass das durch InvaliDB ermöglichte Caching sowohl Latenz als auch Durchsatz der bestehenden Pull-basierten Anfrageschnittstellen um mehr als eine Größenordnung verbessert.

Unsere Klassifikation des heutigen Datenmanagement (vgl. H_1) bildet die Grundlage für ein Buch [WRG19] und verschiedene Tutorials [GWR17] [Wi18] [WGR19a]. Das Systemdesign InvaliDB (vgl. H_2), sein Prototyp (vgl. H_3) sowie die Quaestor-Architektur (vgl. H_4) wurden auf renommierten Konferenzen präsentiert und veröffentlicht, darunter die VLDB 2017 [Ge17], die BTW 2019 [WGR19b] und die VLDB 2020 [WGR20]. Das integrierte System wird von der Firma Baqend zur Entwicklung dynamischer Webseiten genutzt und seit Juli 2017 als Teil einer Backend-as-a-Service-Plattform kommerziell vertrieben. In diesem Kontext erfolgte weitere Forschung zur Beschleunigung von Webseiten, die u.a. in Buchform und auf der ICDE 2020 veröffentlicht wurde [GWR20] [Wi20].

6 Offene Fragen & Geplante Weiterentwicklung

Wir sind der festen Überzeugung, dass unsere Arbeit wertvolle Erkenntnisse für die Architektur von Echtzeitdatenbanken liefert, aber wir sehen noch viele Möglichkeiten für weitere Forschung und Entwicklung. Abschließend möchten wir einen Ausblick auf offene Herausforderungen und mögliche zukünftige Arbeiten in drei Kernbereichen geben.

Aggregationen & Joins. Unser Prototyp unterstützt bisher lediglich sortierte Filteranfragen über einzelne Collections, da dies der Anfragemächtigkeit vieler dokumentenorientierter NoSQL-Datenbanken entspricht (u.a. MongoDB, welches dem Prototypen zugrundeliegt). Zukünftige Arbeiten könnten unsere Implementierung um Anfragen mit Aggregationen und Joins erweitern oder Unterstützung für weitere Datenbanksprachen hinzufügen. Ebenso könnte eine Konsolidierung von InvaliDBs Konsistenzmodell („eventual

consistency“) mit strikteren Garantien verfolgt werden, z.B. durch die Realisierung transaktionaler Sichtbarkeit für Schreiboperationen.

Performance für Endbenutzer. In unserer Arbeit beschäftigen wir uns hauptsächlich mit der Infrastruktur im Backend, doch man sollte den Ressourcenverbrauch von Echtzeitanfragen für Endnutzer nicht aus den Augen verlieren. So könnte in nachfolgenden Arbeiten beleuchtet werden, welche Ressourcen eine Echtzeitanfrage beispielsweise für einen mobilen Nutzer tatsächlich erfordert und wie sie sich minimieren lassen. Eine Betrachtung aus der Verbraucherperspektive erscheint uns insbesondere deswegen relevant, weil Smartphones und andere Geräte mit eingeschränkter Rechenleistung, schwachen Netzwerkverbindungen und/oder begrenztem (monatlichem) Datenvolumen in bestimmten Nutzungsszenarien eher einen Flaschenhals darstellen können als Anwendungs- und Datenbankserver.

Anwendungsentwicklung. Wir sehen ein enormes Innovationspotenzial in der Aufwertung bestehender *Pull-basierter Schnittstellen* mit Push-basierten Funktionalitäten, da sich InvaliDB als zusätzliches Modul gut in bereits existierende Architekturen integrieren lässt. So könnten zum Beispiel bestimmte Teile einer Website interaktiv gemacht werden, um die Benutzererfahrung zu verbessern. Auch könnte InvaliDB eingesetzt werden, um ähnlich zu seiner Rolle innerhalb der Quaestor-Architektur veraltete Cache-Einträge mit geringer Latenz zu invalidieren und so die Datenaktualität zu verbessern.

7 Schlussgedanken

Zu Recht wurden Echtzeitdatenbanken in der Vergangenheit stets mit Vorsicht eingesetzt, da sie sich nur schwer in bestehende Anwendungen integrieren ließen und aufgrund mangelnder Skalierbarkeit nur für kleine Projekte genutzt werden konnten. Das in dieser Dissertation entwickelte Systemdesign InvaliDB adressiert diese Bedenken mit einer einzigartigen Kombination aus Merkmalen: Erstens abstrahiert es vom zugrundeliegenden Datenbanksystem und kann daher praktisch auf jeder Datenbank implementiert werden. Zweitens skaliert InvaliDB sowohl mit der Anzahl der nebenläufigen Anfragen als auch mit dem Schreibdurchsatz. Drittens – und wohl am wichtigsten für produktive Einsatzzwecke – ist InvaliDB als modulare Komponente mit einer isolierten Fehlerdomäne konzipiert und beeinträchtigt daher nicht die Ausfallsicherheit anderer Komponenten. Wir hoffen, dass unsere Arbeit neues Vertrauen in die Praxistauglichkeit von Echtzeitdatenbanken schafft und zu weiterer Forschung an diesem Thema innerhalb der Datenbank-Community anregt.

Literaturverzeichnis

[AH98] Andler, Sten F.; Hansson, Jörgen, Hrsg. Active, Real-Time, and Temporal Database Systems, Lecture Notes in Computer Science 1553. Springer Berlin Heidelberg, 1998.

[Ge17] Gessert, F.; Schaarschmidt, M.; Wingerath, W.; Witt, E.; Yoneki, E.; Ritter, N.: Quaestor: Query Web Caching for Database-as-a-Service Providers. VLDB, 2017.

[Ge19] Gessert, F.: Low Latency for Cloud Data Management. Diss., Univ. of Hamburg, 2019.

[GWR17] Gessert, F.; Wingerath, W.; Ritter, N.: Scalable Data Management: An In-Depth Tutorial on NoSQL Data Stores. In: BTW. 2017.

[GWR20] Gessert, F.; Wingerath, W.; Ritter, N.: Fast & Scalable Cloud Data Management. Springer Int. Publishing, 2020.

[GZ10] Golab, L.; Zsu, M. T.: Data Stream Management. Morgan & Claypool Publishers, 2010.

[He07] Hellerstein, J. M.; Stonebraker, M.; Hamilton, J. et al.: Architecture of a Database System. Foundations and Trends in Databases, 1(2), 2007.

[Pu93] Purimetla, B.; Sivasankaran, R.; Stankovic, J. et al.: A Study of Distributed Real-Time Active Database Applications. Bericht, University of Viginia, 1993.

[SC05] Stonebraker, M.; Cetintemel, U.: „One Size Fits All“: An Idea Whose Time Has Come and Gone. ICDE, 2005.

[SCZ05] Stonebraker, M.; Cetintemel, U.; Zdonik, S.: The 8 Requirements of Real-time Stream Processing. SIGMOD, 2005.

[St88] Stankovic, J. A.: Misconceptions About Real-Time Computing: A Serious Problem for Next-Generation Systems. Computer, 21(10), 1988.

[WGR19a] Wingerath, W.; Gessert, F.; Ritter, N.: NoSQL & Real-Time Data Management in Research & Practice. In: BTW. 2019.

[WGR19b] Wingerath, W.; Gessert, F.; Ritter, N.: Twoogle: Searching Twitter With MongoDB Queries. In: BTW. 2019.

[WGR20] Wingerath, W.; Gessert, F.; Ritter, N.: InvaliDB: Scalable Push-Based Real-Time Queries on Top of Pull-Based Databases (Extended). In: VLDB. 2020.

[Wi18] Wingerath, W.; Gessert, F.; Witt, E.; Friedrich, S.; Ritter, N.: Real-Time Data Management for Big Data. In: EDBT. 2018.

[Wi19] Wingerath, W.: Scalable Push-Based Real-Time Queries on Top of Pull-Based Databases. Dissertation, University of Hamburg, 2019.

[Wi20] Wingerath, W.; Gessert, F.; Witt, E.; Kuhlmann, H. et al.: Speed Kit: A Polyglot & GDPR-Compliant Approach For Caching Personalized Content. In: ICDE. 2020.

[WRG19] Wingerath, W.; Ritter, N.; Gessert, F.: Real-Time & Stream Data Management: Push-Based Data in Research & Practice. Springer Int. Publishing, 2019.

Wolfram „Wolle“ Wingerath wurde am 26. Januar 1988 in der wohl schönsten Ecke Deutschlands geboren (im Norden). Nach seinem Abitur 2007 hatte er Glück und bekam ein Vollstipendium der Studienstiftung des deutschen Volkes, sodass er sich bis zu seinem Master-Abschluss 2012 aufs Studium konzentrieren konnte. Während seiner anschließenden Promotion an der Uni Hamburg konzipierte er das skalierbare Design hinter der Echtzeitanfrage-Engine der Backend-as-a-Service-Plattform Baqend. Dabei entwickelte Wolle nicht nur einen soliden Hintergrund in Echtzeitdatenbanken, sondern auch den damit verbundenen Technologien wie Datenstromverarbeitung, NoSQL-Systemen, Cloud Computing und Big Data Analytics. Seit Januar 2018 arbeitet Wolle in Vollzeit bei Baqend und verantwortet dort heute als Data Engineer alle Belange rund um Datenanalyse und Echtzeitdatenverarbeitung. 2020 wurde Wolle für seine Arbeit als Junior-Fellow der Gesellschaft für Informatik ausgezeichnet. Weil Wolle eigentlich immer Lust auf neue Gesichter hat und weil er sich gerne mit anderen austauscht, referiert er regelmäßig auf Entwickler- und Forschungskonferenzen über spannende Erkenntnisse und Entwicklungen aus seinem Arbeitsumfeld.

Charakterisierung der Lösbarkeit von Consensus unter Nachrichtengegnern[1]

Kyrill Winkler[2]

Abstract: Motiviert durch die zunehmende Allgegenwärtigkeit von kleinen, mobilen, drahtlos kommunizierenden Endgeräten und den daraus resultierenden dynamischen Netzen, beschäftigt sich diese Arbeit mit den theoretischen Grundlagen von verteilten Algorithmen in solchen Systemen. Unser Hauptaugenmerk liegt hier auf der Charakterisierung der Lösbarkeit des klassischen Consensus-Problems in dynamischen Netzen, in denen beliebige Nachrichten verloren gehen können. Dafür verwenden wir das Modell der Nachrichtengegner, welches es uns erlaubt, einen beinahe beliebigen dynamischen Verlust von Nachrichten in einem synchronen System darzustellen. Wir betrachten verschiedene Aspekte dieser Fragestellung, beginnend mit dem verhältnismäßig simplen Fall der grenzwertmäßig abgeschlossenen Nachrichtengegner. Wir gehen dann über zu stabilisierenden Nachrichtengegnern, die diese Eigenschaft nicht besitzen, bevor wir schließlich, mit der Hilfe von Punktmengentopologie, eine Charakterisierung von allgemeinen Nachrichtengegnern präsentieren.

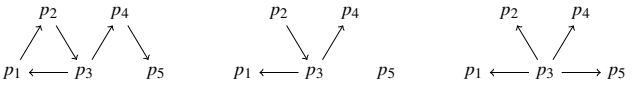
1 Grundlagen

Bei drahtloser Kommunikation kann es, aufgrund eingeschränkter Ressourcen (wie z.B. schwacher Energiespeicher oder Übertragungsleistung), sowie wegen lokalen Interferenzphänomenen (wie z.B. dem Capture Effekt) zum Verlust einzelner Nachrichten kommen. Zumindest konzeptionell wäre das so, als würde ein Angreifer ("Nachrichtengegner") bestimmte Nachrichten unterdrücken um die kommunizierenden Teilnehmer zu stören. Ein mathematisches Modell, das es uns erlaubt solche Systeme abzubilden, ist das Nachrichtengegner-Modell von Afek und Gafni. Eng verwandte Modelle wurden bereits von Santoro und Widmayer [SW89], sowie von Charron-Bost und Schiper [CBS09] für ähnliche Fragestellungen untersucht. Das Nachrichtengegner-Modell ist jedoch allgemeiner und enthält diese Modelle als Spezialfall. Es beschreibt ein Ensemble von n Zustandsautomaten, die *Prozesse* genannt werden, und durch den Austausch von Nachrichten miteinander kommunizieren. Ein Nachrichtengegner sorgt dafür, dass einzelne Nachrichten verloren gehen. Wir nehmen an, dass die Prozesse einzigartige Identifikationsnummern haben und n kennen. Unter einem bestimmten Nachrichtengegner ist das Lösen eines verteilten Problems genau dann möglich, wenn die Prozesse eine Strategie bzw. einen Algorithmus haben um den Nachrichtengegner zu überwinden. Wenn dagegen der Nachrichtengegner eine Strategie hat, welche es den Prozessen unmöglich macht, das Problem zu lösen, dann ist das Problem unter diesem Nachrichtengegner unlösbar.

[1] Englischer Titel der Dissertation [Wi19]: "Characterization of Consensus Solvability under Message Adversaries". Diese Kurzfassung beruht auf dem eingeladenen Übersichtsartikel [WS19].

[2] Technische Universität Wien, kwinkler@ecs.tuwien.ac.at

cc by sa



G_1 für Runde 1 G_2 für Runde 2 G_3 für Runde 3

Abb. 1: Kommunikationsgraphen G_1, G_2, G_3 für 3 Runden.

Formal agiert ein Nachrichtengegner in einem synchronen System: Zu Beginn einer Runde r werden Nachrichten anhand eines sogenannten *Kommunikationsgraphen* G_r ausgetauscht. Dieser Graph ist gerichtet und die Anwesenheit einer Kante (p,q) in G_r bedeutet dass die entsprechende Nachricht von p nach q in Runde r erfolgreich zugestellt werden kann, während ihre Abwesenheit bedeutet dass der Nachrichtengegner diese Nachricht unterdrückt. Anschließend berechnet jeder Prozess, abhängig von den empfangenen Nachrichten, einen Folgezustand, woraufhin die nächste Runde beginnt (siehe Abb. 1). In einem deterministischen Algorithmus ist die Abfolge der Zustände der Prozesse durch deren Initialzustand sowie der Sequenz der Kommunikationsgraphen eindeutig bestimmt. Wir nennen eine solche Sequenz von Kommunikationsgraphen ein *Kommunikationsmuster* und definieren einen konkreten Nachrichtengegner über die Menge an *zulässigen* Kommunikationsmustern, also jenen die er generieren bzw. gegen die Prozesse spielen kann. Ein *Präfix* (die ersten paar Kommunikationsgraphen) eines Kommunikationsmusters wird zulässig genannt, wenn er der Präfix eines zulässigen Kommunikationsmusters ist.

In dieser Zusammenfassung interessieren wir uns für die Lösbarkeit des Consensus Problems [FLP85] im Nachrichtengegner-Modell. Bei dieser klassischen Problemstellung starten alle Prozesse mit einem Eingangswert, (üblicherweise) aus der Menge $\{0,1\}$. Ihre Aufgabe ist es, sich irgendwann auf einen Ausgangswert zu einigen, welcher der Eingangswert mindestens eines Prozesses war. Dieses fundamentale Problem abstrahiert die Fähigkeit dass mehrere Prozesse das selbe Ergebnis liefern, wie es insbesondere der Fall ist wenn man Fehlertoleranz mit der Hilfe von Replikation erreichen möchte. Hier kann Consensus Replikadeterminismus, also ein identes Verhalten der Replikate, sicherstellen.

Für verschiedene Spezialfälle wurde diese Frage bereits früher beantwortet: Für zwei Prozesse wurde diese Frage von Fevat und Godard in [FG11] studiert und beantwortet. Santoro und Widmayer beschreiben im Hauptresultat von [SW89] eine Lösbarkeitscharakterisierung für Consensus mit mobilen Verbindungsfehlern. Hier wird der Nachrichtengegner über die Anzahl der Nachrichten definiert, die jede Runde unterdrückt werden dürfen. Dieses Resultat wurde später von Couloma, Godard und Peters in [CGP15] verallgemeinert, für die sogenannten *vergesslichen* Nachrichtengegner. Ein Nachrichtengegner dieser Klasse ist definiert über eine Menge an erlaubten Kommunikationsgraphen, wobei er jede Runde einen beliebigen Graphen aus dieser Menge auswählen darf und stellt wiederum einen Spezialfall eines abgeschlossenen Nachrichtengegners dar.

Diese Zusammenfassung bietet einen Überblick über die in [Wi19] etablierten Bedingungen, die ein Nachrichtengegner erfüllen muss, um Consensus lösbar zu machen. Zunächst präsentieren wir eine rein kombinatorische Bedingung für die Lösbarkeit von Consensus unter der Klasse der abgeschlossenen Nachrichtengegner. Anschließend erklären wir kurz die praktisch relevanten stabilisierenden Nachrichtengegner und geben exempla-

risch eine Charakterisierung für einen generischen Nachrichtengegner aus dieser Klasse. Schließlich zeigen wir, wie man Punktmengentopologie benutzen kann, um die Lösbarkeit von Consensus unter allgemeinen Nachrichtengegnern zu charakterisieren. Die Ergebnisse der Dissertation wurden in den Tagungsbändern mehrerer internationaler Konferenzen [Sc14, Bi15, SWS16, SSW18, WSM19] und Journalen [Bi18, WSS19] publiziert[3]; ein eingeladener Übersichtsartikel ist im EATCS Bulletin [WS19] erschienen.

2 Abgeschlossene Nachrichtengegner

Eine besonders nützliche fundamentale Eigenschaft von Nachrichtengegnern ist ihre Abgeschlossenheit (vgl. auch [LM95]). Intuitiv verständlich wird sie, wenn wir die zulässigen Kommunikationsmuster eines Nachrichtengegners als Baum darstellen, wie in Abb. 2.

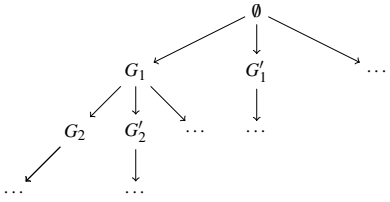


Abb. 2: Ein Nachrichtengegner, dargestellt als Baum. Jeder unendliche Pfad (z.B. $G_1, G_2', \ldots$) von der Wurzel aus korrespondiert zu einem zulässigen Kommunikationsmuster und umgekehrt.

Hier repräsentieren die Pfade von der Wurzel bis zu einer Ebene r die ersten r Runden eines zulässigen Kommunikationsmusters. Ein Nachrichtengegner ist nun abgeschlossen wenn jeder infinite Pfad in diesem Baum zu einem zulässigen Kommunikationsmuster korrespondiert. Dies ist keineswegs selbstverständlich: Der Nachrichtengegner etwa, der garantiert dass ein bestimmter Graph G nach einer unbestimmten, jedoch finiten Zeit gespielt wird, erfüllt diese Eigenschaft nicht. Hier gibt es nämlich für jede Runde einen Präfix, bzw. einen entsprechenden Pfad, auf dem G noch nicht gespielt wurde und darum beinhaltet der Baum den infiniten Pfad, in dem der Graph nie gespielt wird (der unter diesem Nachrichtengegner zu keinem zulässigen Kommunikationsmuster korrespondiert). Etwas präziser definiert man diese Eigenschaft folgendermaßen.

Definition 1. *Sei (σ_i) eine beliebige Sequenz von zulässigen Präfixen mit der der Eigenschaft dass σ_i ein Präfix von σ_{i+1} ist. Ein Nachrichtengegner ist abgeschlossen wenn der Grenzwert $\lim_{i\to\infty}(\sigma_i)$ ein zulässiges Kommunikationsmuster ist.*

Diese Eigenschaft schränkt den Nachrichtengegner stark ein, denn sie erlaubt es nicht, Dinge wie "irgendwann wird etwas Gutes passieren" auszudrücken (vgl. [AS85]). So ist etwa der relativ einfache Nachrichtengegner, der garantiert, dass irgendwann eine vollständig verbundener Kommunikationsgraph auftaucht, nicht abgeschlossen, obwohl ein Lösungsalgorithmus auf der Hand liegt: Ein Prozess wartet einfach bis er von allen anderen Prozessen gehört hat und entscheidet dann z.B. das Maximum aller Eingangswerte.

[3] Eine detaillierte Darstellung der Zusammenhänge dieser Publikationen mit der Dissertation befindet sich in Abschnitt 1.3 von [Wi19].

Nichtsdestotrotz wurden alle bisherigen Lösbarkeitscharakterisierungen von Consensus mit mehr als zwei Prozessen für abgeschlossene Nachrichtengegner formuliert (insbesondere sind die Nachrichtengegner von [CGP15] und [SW89] abgeschlossen). In Wahrheit sind abgeschlossene Nachrichtengegner sogar noch allgemeiner als die bisherigen Klassen, denn sie erlauben insbesondere Formulierungen der Form "innerhalb von x Runden wird etwas Gutes passieren", wobei x den Prozessen bekannt ist.

Interessanterweise gibt es eine relativ einfache Lösbarkeitscharakterisierung für Consensus unter abgeschlossenen Nachrichtengegnern, die wir im folgenden kurz vorstellen. Sie basiert auf dem sogenannten *Kern* eines Kommunikationsmusters bzw. dessen Präfix, geschrieben als $\mathrm{Ker}(\sigma)$, welcher jene Prozesse beinhaltet, von denen alle Prozesse gehört haben. In Abb. 1 sind zum Beispiel p_1, p_2 und p_3 im Kern des abgebildeten Präfixes. Desweiteren benutzen wir die transitive Hülle der Ununterscheidbarkeitsrelation, einer Äquivalenzrelation, die alle jene Paare von (Präfixen von) Kommunikationsgraphen beinhaltet, welche für irgendeinen Prozess gleich ausschauen. Formal definiert werden kann diese Relation mit Hilfe der *Sicht* eines Prozesses (die im Wesentlichem dem kausalen Kegel aus [BZM14] entspricht), die in Abb. 3 grün dargestellt ist.

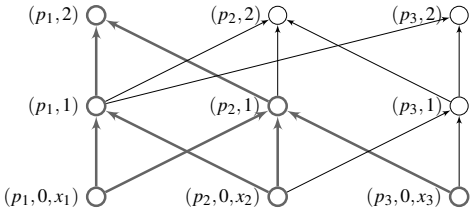


Abb. 3: Der Prozess-Zeit Graph PT^t für Runde $t = 2$ und die $n = 3$ Prozesse p_1, p_2, p_3 mit den Eingangswerten x_1, x_2, x_3. Er stellt den Eingangswert x von Prozess p als Knoten $(p, 0, x)$ dar und enthält eine Kante $((p,r,x),(q,r+1))$ für $r = 0$ bzw. $((p,r),(q,r+1))$ für $r > 0$ genau dann wenn $(p,q) \in G_{r+1}$. Die Sicht von p_1 am Ende von Runde 2 ist der grün markierte Teilgraph.

Die grundlegende Idee der Charakterisierung basiert auf der eher banalen Erkenntnis, dass der Kern eines zulässiges Kommunikationsmuster nicht für immer leer bleiben kann. Diese Erkenntnis folgt leicht aus der Definition des Consensus Problem mit Hilfe der Techniken von [FLP85] bzw. [SW89]. Nun ist die Existenz eines nicht trivialen Kerns zwar notwendig, aber keineswegs hinreichend, damit ein Prozess eine Entscheidung treffen kann. Klarerweise muss ein Prozess auch die Möglichkeit haben, festzustellen, ob der Kern tatsächlich nicht leer ist. Eine zusätzliche Komplikation ergibt sich durch die Eigenheit, dass es durchaus Präfixe von Kommunikationsmustern σ, ρ geben kann, sodass $\mathrm{Ker}(\sigma) = K$, während $\mathrm{Ker}(\rho) = K'$ und ρ ist für einen Prozess p ununterscheidbar von σ. Auf welchen Wert soll nun p entscheiden, wenn nun $K \cap K' = \emptyset$ ist und alle Prozesse aus K Eingangswert 0 haben, während alle Prozesse aus K' Eingangswert 1 haben? Man kann dieses Argument weiterführen und beweisen dass jedes Element der Äquivalenzklasse der transitiven Hülle der Ununterscheidbarkeitsrelation einen gemeinsamen Prozess im Kern haben muss.

Angenommen, wir haben nun einen Nachrichtengegner, der diese Bedingung erfüllt, also jedes zulässige Kommunikationsmuster irgendwann eine nicht triviale Durchschnittsmenge der Kerne der Elemente seiner Äquivalenzklasse besitzt, können die Prozesse hier

Consensus lösen? Die Antwort ist, dass es sogar einen Algorithmus gibt, der hier simultanen Consensus löst, bei dem alle Prozesse gleichzeitig entscheiden. Alles was ein Prozess tun muss, ist die Äquivalenzklasse des aktuellen Präfixes zu berechnen und zu überprüfen, ob deren Kern eine nicht triviale Durchschnittsmenge besitzt. Sobald das eintritt, kann der Prozess auf den Eingangswert eines bestimmten Prozesses (z.B. jenen mit der größten Identifikationsnummer) aus dieser Durchschnittsmenge entscheiden. Die Konstruktion der Äquivalenzklasse stellt sicher, dass alle Prozesse diese Durchschnittsmenge finden.

Das folgende Theorem gibt eine detailliertere, vollständige Charakterisierung der Lösbarkeit von Consensus für abgeschlossene Nachrichtengegner. Wir benutzen hier die Konvention $\sigma|_r$ für den Präfix, bestehend aus den ersten r Runden von σ. Weiters bezeichnen wir mit $[\sigma|_r]$ die Äquivalenzklasse von $\sigma|_r$ bezüglich der transitiven Hülle der Ununterscheidbarkeitsrelation.

Theorem 2 (vgl. [Wi19], Theorem 5.0.1)**.** *Consensus ist lösbar unter einem abgeschlossenen Nachrichtengegner dann und nur dann wenn es für jedes zulässige Kommunikationsmuster σ eine Runde r gibt sodass $\bigcap_{\rho\in[\sigma|_r]} \mathrm{Ker}(\rho) \neq \emptyset$.*

3 Stabilisierende Nachrichtengegner

Wie bereits angedeutet, ist eine der gravierendsten Limitationen der abgeschlossenen Nachrichtengegner dass sie keine Garantien der Form “irgendwann passiert etwas Gutes”, also Garantien, die zu einem finiten aber unbekannten Zeitpunkt eintreten, ausdrücken können. Für Systeme mit chaotischen Startsequenzen, bei der es eine unbekannte Zeit dauert, bis sich eine verlässliche synchrone Kommunikation etabliert hat, sowie in Systemen, in denen massive transiente Fehler von unbekannter Dauer auftreten können, wäre eine Modellierung solcher Eigenschaften jedoch äußerst nützlich. Aus diesem Grund beschreiben wir nun einen Nachrichtengegner, der eine generische Form von Stabilisierung ausdrücken kann und führen eine präzise Charakterisierung der Lösbarkeit von Consensus bezüglich der Dauer dieser Stabilität ein.

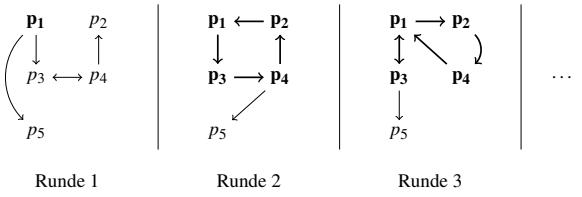


Abb. 4: Die Wurzelkomponente von Runde 1 besteht aus dem Knoten p_1. In Runde 2 und 3 existiert eine knotenstabile Wurzelkomponente, bestehend aus der Knotenmenge $\{p_1, p_2, p_3, p_4\}$.

Er basiert auf dem Konzept einer sogenannten *Wurzelkomponente* (siehe Abb. 4), einer starken Zusammenhangskomponente, bei der kein Knoten eine eingehende Kante von außerhalb der Komponente hat. Eine Wurzelkomponente wird *knotenstabil* genannt, wenn sie für mehrere aufeinanderfolgende Runden eines Kommunikationsmusters aus der selben Knotenmenge besteht. Dabei ist es erlaubt, dass sich die Konnektivität innerhalb der

Komponente stark ändert, sie muss jedoch stets eine Wurzelkomponente des entsprechenden Kommunikationsgraphen bleiben.

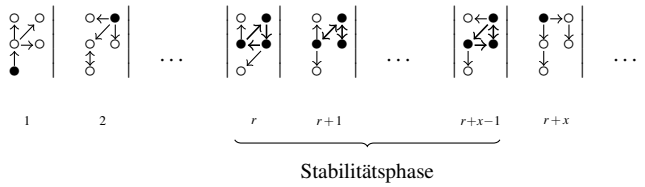


Abb. 5: Ein zulässiges Kommunikationsmuster des Nachrichtengegners $\Diamond\text{STABIL}(x)$. Die knotenstabile Wurzelkomponente (schwarz markiert) taucht in Runde r auf und dauert bis Runde $r+x-1$.

Vermutlich der grundlegendste Nachrichtengegner, basierend auf knotenstabilen Wurzelkomponenten, ist $\Diamond\text{STABIL}(x)$ und wird exemplarisch dargestellt in Abb. 5. Er unterliegt zwei Einschränkungen: 1. Jeder Kommunikationsgraph besitzt genau eine Wurzelkomponente und 2. Jedes zulässige Kommunikationsmuster besitzt eine Wurzelkomponente, die für mindestens x Runden knotenstabil bleibt.

Die zentrale Frage bezüglich dem Nachrichtengegner $\Diamond\text{STABIL}(x)$ ist, wie sich der Parameter x auf die Lösbarkeit von Consensus auswirkt. Wir fassen die Antwort auf diese Frage in Theorem 3 zusammen und geben im Folgenden eine intuitive Begründung. Wie in Abb. 5 skizziert, drückt der Parameter x die Dauer der Stabilitätsphase aus.

Theorem 3 (vgl. [Wi19], Theorem 6.3.6 und 8.4.4)**.** *In einem System aus n Prozessen ist Consensus lösbar unter* $\Diamond\text{STABIL}(x)$ *dann und nur dann wenn* $x \geq n$.

Der Grund, warum Consensus unlösbar ist, wenn $x < n$, ist im Wesentlichen, dass nach $n-1$ Runden Stabilitätsphase der Ausgangswert noch nicht feststehen kann. Die Ursache hierfür wird mit Hilfe von Abb. 6 im Folgenden illustriert:

G_1: $p_1 \longrightarrow p_2 \longrightarrow \cdots \longrightarrow p_n$ $\quad$ G_2: $p_1 \longleftrightarrow p_2 \longrightarrow \cdots \longrightarrow p_n$ $\quad$ G_3: $p_2 \longleftrightarrow p_1 \longrightarrow \cdots \longrightarrow p_n$ $\quad$ G_4: $p_2 \longrightarrow p_1 \longrightarrow \cdots \longrightarrow p_n$

Abb. 6: Kommunikationsgraphen für die Unmöglichkeit von Consensus mit $x < n$.

Nehmen wir an, dass p_1 mit Eingangswert 0 und alle anderen Prozesse mit Eingangswert 1 starten und, um einen Widerspruch abzuleiten, nehmen wir an, dass nach $n-1$ Runden Stabilitätsphase der Ausgangswert bereits feststünde. Wenn wir nun als Kommunikationsmuster G_1^{n-1}, also den Kommunikationsgraph G_1 aus Abb. 6, $n-1$ mal wiederholt, wählen, ist klar dass der Ausgangswert hier auf 1 festgelegt werden muss. Der Grund dafür ist schlicht dass der Prozess p_1 nur seinen eigenen Eingangswert gehört hat und deswegen, wenn er gezwungen wird, sich festzulegen, sich auf 1 festlegen muss (es könnte ja sein, dass alle anderen Prozesse auch 1 als Eingangswert hatten und hier wäre, aufgrund der Consensus Spezifikation, nur 1 ein zulässiger Ausgangswert).

Wenn wir uns nun weiters ansehen, was p_n über das Kommunikationsmuster gelernt hat, ist klar dass G_1^{n-1} für p_n ununterscheidbar ist von G_2^{n-1}. Deshalb muss auch in G_2^{n-1} der Ausgangswert auf 1 festgelegt sein. Ein ganz analoges Argument für G_4^{n-1} und G_3^{n-1} zeigt,

dass der Ausgangswert in G_3^{n-1} auf 0 festgelegt ist. Dies ist aber ein Widerspruch da G_2^{n-1} und G_3^{n-1} ununterscheidbar für p_1 und p_2 ist, wie man leicht überprüfen kann.

Ein klassisches Bivalenz-Argument, wie es bereits in [FLP85, SW89] eingeführt wurde, kann benutzt werden um zu zeigen, dass auch später die Entscheidungen nicht mehr festgelegt werden können.

Um zu zeigen, dass n Runden Stabilitätsphase hinreichend für die Lösbarkeit von Consensus sind, skizzieren wir im Folgenden einen Lösungsalgorithmus. Eine der größten Herausforderungen beim Entwickeln eines solchen Algorithmus ist, die kurze Dauer der Stabilitätsphase von nur n Runden effizient auszunutzen. Die Grundoperation unseres vorgeschlagenen Algorithmus ist es, in Runde r jedem möglichen Präfix σ einen Wert $\Delta(\sigma)$ zuzuordnen. Sobald in einer Runde r alle Präfixe, die für einen Prozess p in Betracht kommen (also jene, die für p vom tatsächlichen Präfix ununterscheidbar sind), denselben Wert Δ zugewiesen bekommen haben, entscheidet p auf den Wert von Δ. Die Berechnung von $\Delta(\sigma)$ selbst in Runde r erfolgt nach folgenden Regeln:

- Falls σ bereits einen Wert zugewiesen bekommen hat, übernimm diesen Wert.
- Ansonsten, weise $\Delta(\sigma)$ einen neuen Wert v zu wenn
  - es einen Präfix τ mit zugewiesenem Wert $\Delta(\tau) = v$ gibt, sodass die Stabilitätsphase von τ mindestens $2n-2$ Runden zurückliegt oder
  - die Stabilitätsphase von σ selbst $2n-2$ Runden zurückliegt und v der Eingangswert des Prozesses mit der größten Identifikationsnummer in der knotenstabilen Wurzelkomponente während dieser Stabilitätsphase ist.

4 Allgemeine Nachrichtengegner

Nach den bisher präsentierten Lösbarkeitscharakterisierungen für abgeschlossene Nachrichtengegner und den Nachrichtengegner $\Diamond\text{STABIL}(x)$, stellt sich die Frage, ob es auch eine solche Charakterisierung für allgemeine Nachrichtengegner gibt, die also sowohl abgeschlossene als auch nicht abgeschlossene Nachrichtengegner beinhaltet. Ein Weg, der zu einer solchen Charakterisierung führt, ist es, den topologischen Raum der Kommunikationsmuster zu betrachten, ganz analog zur Herangehensweise im klassischen Resultat von Alpern und Schneider [AS85]. Als Punkte im topologischen Raum eines Nachrichtengegners wählen wir im Wesentlichen[4] die Prozess-Zeit Graphen aus Abb. 3, die die Eingangswerte zusammen mit den Kommunikationsmustern des Nachrichtengegners beinhalten.

Wir definieren die Topologie der Punktmenge eines Nachrichtengegners mit Hilfe einer Distanzfunktion, die die "Nähe" von zwei Prozess-Zeit Graphen α, β quantifizieren soll. Intuitiv gesprochen bringen wir damit zum Ausdruck, wie ähnlich sich α, β sind im Bezug auf die Sicht einzelner Prozesse, also wie leicht bzw. schwer es die Prozesse haben α von β zu unterscheiden. Das macht Sinn, da der Extremfall der ewigen Ununterscheidbarkeit

[4] Aus technischen Gründen wählen wir genaugenommen die infiniten Produkte $PT^1 \times PT^2 \times \cdots$.

von α, β bedeutet dass in beiden Prozess-Zeit Graphen irgendwann dieselbe Entscheidung getroffen wird. Die folgenden Distanzfunktionen (vgl. Abb. 7), geben dieser nützlichen Einsicht gewissermaßen eine noch feinere Granularität, die es uns erlauben wird, eine Lösbarkeitscharaktscharakterisierung zu formulieren. Sei $d_p(\alpha, \beta) = 2^{-t}$ wobei t die letzte Runde ist sodass α und β für den Prozess p noch ununterscheidbar sind und sei $d_{\min}(\alpha, \beta) = \min_{p \in \Pi} d_p(\alpha, \beta)$, wobei Π für die Menge aller Prozesse im System steht.

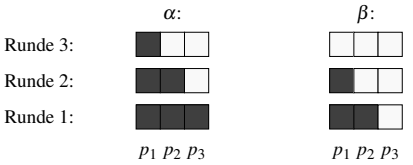


Abb. 7: Distanzfunktion für ein System aus den drei Prozessen $\{p_1, p_2, p_3\}$ mit den Prozess-Zeit Graphen α, β. Wir benutzen blau bzw. gelb für zwei verschiedene Sichten eines Prozesses. So hat p_3 bereits in Runde 1 eine unterschiedliche Sicht, weshalb $d_{p_3}(\alpha, \beta) = 2^0 = 1$. Analog sieht p_2 keinen Unterschied bis Runde 1 und $d_{p_2}(\alpha, \beta) = 2^{-1} = 1/2$ und p_1 sieht keinen Unterschied bis Runde 2, weshalb $d_{p_1}(\alpha, \beta) = 2^{-2} = 1/4$. Die minimale Distanzfunktion ist also $d_{\min}(\alpha, \beta) = \min_{p \in \{p_1, p_2, p_3\}} d_p = d_{p_1} = 1/4$.

Die Lösbarkeitscharakterisierung von Consensus für allgemeine Nachrichtengegner ist im folgenden Theorem 4 formuliert. Sie benutzt eine Handvoll grundlegender topologischer Begriffe, wie den topologischen Raum, der durch $d_{\min}$ induziert wird: Wenn X die Menge der Punkte ist, ist damit X gemeint, zusammen mit der Familie von *offenen* Teilmengen $U \subseteq X$, definiert durch $\forall x \in U \; \exists \varepsilon > 0 : B_\varepsilon(\alpha) \subseteq U$. Hierin bezeichnet $B_\varepsilon(\alpha)$ den offenen ε-Ball um α bzgl. der Distanzfunktion $d_{\min}$, der definiert ist als $B_\varepsilon(\alpha) = \{\beta : d_{\min}(\alpha, \beta) < \varepsilon\}$. Ein topologischer Raum ist *nicht zusammenhängend* wenn er in mehrere (nicht leere) offene Mengen partitioniert werden kann. Eine *Zusammenhangskomponente* eines topologischen Raumes ist eine maximale offene zusammenhängende Menge. Sei $PT(v)$ die Menge der Prozess-Zeit Graphen, in denen alle Prozesse Eingangswert v haben. Dann gilt:

Theorem 4. *Consensus ist lösbar unter einem Nachrichtengegner dann und nur dann wenn keine Zusammenhangskomponente des Raumes der Prozess-Zeit Graphen, der durch $d_{\min}$ induziert wird, sowohl $PT(0)$ als auch $PT(1)$ schneidet.*

Diese Charakterisierung sagt aus, dass, um eine Lösung von Consensus zu ermöglichen, der topologische Raum, der durch $d_{\min}$ induziert wird, in keiner Zusammenhangskomponente sowohl einen Prozess-Zeit Graph aus $PT(0)$ (also mit allen Eingangswerten auf 0 gesetzt) als auch einer aus $PT(1)$ auftauchen kann (siehe auch Abb. 8).

Wir erklären nun kurz, warum die Bedingung aus Theorem 4 nötig ist, damit Consensus gelöst werden kann. Um einen Widerspruch herzuleiten, nehmen wir also an, dass es eine Komponente S gibt, die sowohl $PT(0)$ als auch $PT(1)$ schneidet. Zunächst statten wir die Menge der Ausgangswerte $\{0, 1\}$ mit der diskreten Topologie aus, in der sowohl 0 als auch 1 offen sind. Man kann aus der Consensus Spezifikation folgern, dass die Funktion Δ, die jeden Prozess-Zeit Graphen aus S auf ein Element dieses topologischen Raums über

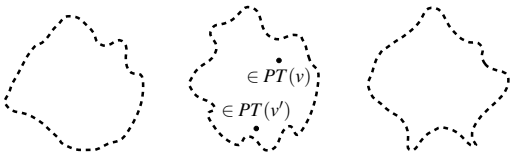


Abb. 8: Wenn eine Zusammenhangskomponente sowohl $PT(v)$ als auch $PT(v')$ schneidet, dann ist Consensus lösbar genau dann wenn $v = v'$.

$0,1$ abbildet, lokal konstant und deshalb stetig ist. Aufgrund der Stetigkeit von Δ ist das Bild von Δ zusammenhängend, jedoch folgt aus der Consensus Spezifikation, dass der Bildraum von Δ sowohl 0, als auch 1 beinhaltet. Dies ist in der diskreten Topologie jedoch nicht zusammenhängend, was obiger Aussage widerspricht.

Um zu sehen, dass die Erfüllung der Bedingung aus Theorem 4 ausreicht, um Consensus zu lösen, betrachten wir folgenden Lösungsalgorithmus. Wir nehmen also an, es existiert eine Partition P des topologischen Raumes, welche die Bedingung aus Theorem 4 erfüllt. Basierend auf jenen Prozess-Zeit Graphen, die ein Prozess p in Betracht zieht, berechnet er in jeder Runde t den ε-Ball bezüglich der d_p-Metrik, also $B_{\varepsilon,d_p}(\alpha) = \{\beta : d_p(\alpha,\beta) < \varepsilon = 2^{-t}\}$. Sobald $B_{\varepsilon,d_p}(\alpha)$ ganz in einer Menge der Partition P liegt, kann p auf einen Wert entscheiden: Wenn diese Menge $PT(v)$ beinhaltet, entscheidet p auf v, sonst auf 0.

5 Schlussfolgerungen

Wir haben hier kurz die wichtigsten der in [Wi19] vorgestellten Consensus-Charakterisierungen präsentiert. Wir haben gesehen, dass es für abgeschlossene Nachrichtengegner eine relativ einfache Charakterisierung gibt, während eine allgemeine Charakterisierung nur mit Hilfe topologischer Methoden erlangt werden konnte. Dazwischen liegt die Charakterisierung für stabilisierende Nachrichtengegner, die zwar nicht so allgemein ist wie letztere, jedoch ohne topologische Methoden auskommt. Wir hoffen, dass diese Resultate nützlich sind, um zu erklären wann bzw. warum Consensus in einem System, das durch einen Nachrichtengegner modelliert wird, (nicht) gelöst werden kann.

Literaturverzeichnis

[AS85] Alpern, Bowen; Schneider, Fred B.: Defining Liveness. Information Processing Letters, 21(4):181–185, 1985.

[Bi15] Biely, Martin; Robinson, Peter; Schmid, Ulrich; Schwarz, Manfred; Winkler, Kyrill: Gracefully Degrading Consensus and *k*-Set Agreement in Directed Dynamic Networks. In: NETYS'15. LNCS 9466. Springer, S. 109–124, 2015.

[Bi18] Biely, Martin; Robinson, Peter; Schmid, Ulrich; Schwarz, Manfred; Winkler, Kyrill: Gracefully degrading consensus and k-set agreement in directed dynamic networks. Theoretical Computer Science, 726:41–77, 2018.

[BZM14] Ben-Zvi, Ido; Moses, Yoram: Beyond Lamport's Happened-before: On Time Bounds and the Ordering of Events in Distributed Systems. J. ACM, 61(2):13:1–13:26, April 2014.

[CBS09] Charron-Bost, Bernadette; Schiper, André: The Heard-Of model: computing in distributed systems with benign faults. Distributed Computing, 22(1):49–71, April 2009.

[CGP15] Coulouma, Étienne; Godard, Emmanuel; Peters, Joseph G.: A characterization of oblivious message adversaries for which Consensus is solvable. Theoretical Computer Science, 584:80–90, 2015.

[FG11] Fevat, Tristan; Godard, Emmanuel: Minimal Obstructions for the Coordinated Attack Problem and Beyond. In: IPDPS'11. S. 1001–1011, 2011.

[FLP85] Fischer, Michael J.; Lynch, Nancy A.; Paterson, M. S.: Impossibility of Distributed Consensus with one Faulty Process. J. ACM, 32(2):374–382, April 1985.

[LM95] Lubitch, Ronit; Moran, Shlomo: Closed Schedulers: A Novel Technique for Analyzing Asynchronous Protocols. Distributed Computing, 8(4):203–210, Juni 1995.

[NSW19] Nowak, Thomas; Schmid, Ulrich; Winkler, Kyrill: Topological Characterization of Consensus Under General Message Adversaries. PODC'19. ACM, S. 218–227, 2019.

[Sc14] Schwarz, Manfred; Winkler, Kyrill; Schmid, Ulrich; Biely, Martin; Robinson, Peter: Brief Announcement: Gracefully Degrading Consensus and k-set Agreement under Dynamic Link Failures. PODC'14. ACM, S. 341–343, 2014.

[SSW18] Schmid, Ulrich; Schwarz, Manfred; Winkler, Kyrill: On the Strongest Message Adversary for Consensus in Directed Dynamic Networks. SIROCCO'18. Springer, S. 102–120, 2018.

[SW89] Santoro, Nicola; Widmayer, Peter: Time is not a healer. In: STACS'89. LNCS 349. Springer, S. 304–313, Februar 1989.

[SWS16] Schwarz, Manfred; Winkler, Kyrill; Schmid, Ulrich: Fast Consensus Under Eventually Stabilizing Message Adversaries. ICDCN'16. ACM, S. 7:1–7:10, 2016.

[Wi19] Winkler, Kyrill: Characterization of Consensus Solvability under Message Adversaries. Dissertation, TU Wien, 2019. `http://katalog.ub.tuwien.ac.at/AC15506161`.

[WS19] Winkler, Kyrill; Schmid, Ulrich: An Overview of Recent Results for Consensus in Directed Dynamic Networks. Bulletin of the EATCS, 128, 2019.

[WSM19] Winkler, Kyrill; Schmid, Ulrich; Moses, Yoram: A Characterization of Consensus Solvability for Closed Message Adversaries. OPODIS'19, 2019. (to appear).

[WSS19] Winkler, Kyrill; Schwarz, Manfred; Schmid, Ulrich: Consensus in rooted dynamic networks with short-lived stability. Distributed Computing, 32(5):443–458, Oct 2019.

Kyrill Winkler studierte Informatik an der technischen Universität Wien und promovierte im Jahr 2019 bei Prof. Ulrich Schmid mit Auszeichnung. Er absolvierte einen mehrmonatigen Forschungsaufenthalt an der Ecole Polytechnique in Paris und ist aktuell Universitätsassistent am Institut für Computer Engineering der TU Wien. Sein Forschungsschwerpunkt sind verteilte Algorithmen, insbesondere die Analyse von Consensus und verwandten Problemen in dynamischen Netzen.

Entwurfsautomatisierung für Quantencomputer[1]

Alwin Zulehner[2]

Abstract: Quantencomputer erlauben es, bestimmte Probleme exponentiell schneller als konventionelle Rechner zu lösen. Obwohl ihre Entwicklung aktuell noch in den Kinderschuhen steckt, scheinen Quantencomputer in greifbarer Nähe. Um deren Potential voll auszuschöpfen, müssen jedoch einige Entwurfsaufgaben effizient bewältigt werden. Die Dissertation stellt hierfür Ansätze vor, welche auf geschickte Weise Expertise aus dem Gebiet der Entwurfsautomatisierung nutzt. Dadurch konnten zahlreiche Methoden für den Entwurf von Quantenalgorithmen realisiert werden, die den derzeitigen Stand der Technik (sowohl hinsichtlich der Laufzeit als auch der Qualität der Ergebnisse) weit übertreffen. Die entwickelten Lösungen erzielten auch über den akademischen Bereich hinaus eine hohe Resonanz, was Auszeichnungen von Google und IBM, sowie die Integration in die Werkzeuge von IBM und Atos belegen.

1 Einführung

In den 1970er Jahren begannen Forscher, Fragen der Informatik und der Informationstheorie mit Hilfe von Quantenmechanik zu adressieren. In den daraus resultierenden neuen Forschungsrichtungen (beispielsweise Quantencomputing oder Quantensicherheit) dienen sogenannte Quantenbits (Qubits) als elementare Informationseinheit, welche im Gegensatz zu konventionellen Bits nicht nur einen der Basiszustände annehmen (in Dirac Notation $|0\rangle$ oder $|1\rangle$), sondern sich auch in einer beliebigen Überlagerung $\alpha_0 \cdot |0\rangle + \alpha_1 \cdot |1\rangle$ miteinander befinden können (wobei $\alpha_0, \alpha_1 \in \mathbb{C}$ und $\alpha_0 \alpha_0^* + \alpha_1 \alpha_1^* = 1$). Gemeinsam mit dem weiteren Phänomen der Quantenverschränkung (die beschreibt, wie sich die Zustände von Qubits gegenseitig beeinflussen), ergibt sich eine enorme Steigerung der Informationsdichte und potentiellen Rechenleistung für Quantensysteme [NC02].

Quantencomputing beschäftigt sich mit der Entwicklung von Algorithmen, die Quanteneffekte ausnutzen, um gewisse Problemstellungen deutlich schneller (oftmals exponentiell schneller) zu lösen als mit konventionellen Computern. Erste Anwendungen wie Grover's Suche [Gr96] oder Shor's Algorithmus, der es erlaubt Semiprimzahlen in polynomialer Zeit zu faktorisieren [Sh97], wurden bereits im letzten Jahrhundert publiziert. Seither hat sich das Anwendungsgebiet stark erweitert und beinhaltet nun auch Problemstellungen im Bereich der Chemie, der linearen Algebra, der Simulation von Quantensystemen und dem Machine Learning [Mo16]. Speziell im Bereich der Chemie (der Simulation von Molekülen) erwarten sich Forscher mit Hilfe von Quantencomputern starke Fortschritte; beispielsweise um bessere Katalysatoren für den Haber-Bosch Prozess zu finden, der 1-2 % des weltweiten Energieverbrauchs verursacht.

[1] Englischer Titel der Dissertation: "Design Automation for Quantum Computing"

[2] Institut für Integrierte Schaltungen, Johannes Kepler Universität Linz, alwin.zulehner@gmx.at

cc by sa

In letzter Zeit ist auch eine rasante Entwicklung in der Realisierung der Quantencomputer selbst beobachtbar. Während Prototypen von Quantencomputern ursprünglich an Universitäten entwickelt wurden, haben mittlerweile auch namenhafte Firmen wie *IBM*, *Google*, *Microsoft* und *Intel* als auch Startups wie *Rigetti* und *IonQ* damit begonnen, entsprechende Rechner zu realisieren [Go18]. Im Jahr 2017 startete IBM eine Initiative, um Quantencomputer über einen Cloud Service erstmals der Öffentlichkeit zugänglich zu machen. Seither wurden auf diesen Quantencomputern von mehr als 100 000 Benutzern über 6,5 Millionen Experimente durchgeführt. Im Januar 2019 präsentierte IBM außerdem *IBM Q System One*, den ersten Prototypen eines Quantencomputers, der außerhalb des Labors betrieben werden kann [IBM19].

Prognostiziert man einen gleichbleibenden Fortschritt in der Entwicklung von Quantencomputern (hinsichtlich der Anzahl der verfügbaren Qubits sowie der Genauigkeit der Operationen), werden Quantencomputer mehr und mehr für praktische Anwendungen interessant [Ar19]. Doch trotz dieser Fortschritte im Bereich der Quantenhardware werden effiziente Werkzeuge und Methoden benötigt, die den Entwurfsprozess für Anwendungen und auch für den Hardwareentwurf selbst automatisieren. Ansonsten droht eine Situation, in der Quantencomputer verfügbar sind, ihre volle Rechenkraft aber nicht genutzt werden kann. Genauer gesagt müssen insbesondere die folgenden Aufgaben effizient umgesetzt werden:

- **Simulation von Quantenschaltungen** auf konventionellen Rechnern sind wichtig, da reale Quantencomputer oft noch in ihrer Anzahl an Qubits und durch ihre Fehleranfälligkeit limitiert sind. Außerdem werden Simulatoren benötigt, um neue Quantenalgorithmen zu entwickeln, fehlerkorrigierende Codes zu evaluieren oder um die Korrektheit von Quantenschaltungen zu validieren. Darüber hinaus erlauben Simulatoren einen stärkeren Einblick in die jeweilige Anwendung, da einzelne Amplituden des aktuellen Zustands ausgelesen werden können (was bei realen Quantencomputern nicht möglich ist). Da aber sowohl Quantenzustände als auch Quantenoperationen in der Regel über (exponentiell große) Vektoren bzw. Matrizen beschrieben werden, ist die effiziente Umsetzung solcher Simulationen hochgradig komplex.
- Der **Entwurf Boolescher Komponenten** ist nötig, da sie einen wesentlichen Teil vieler Quantenalgorithmen bilden. In der Tat ist der "Quantenteil" eines zu kompilierenden Algorithmus üblicherweise bereits direkt in Form elementarer Gatter gegeben und bei vielen Quantenalgorithmen ähnlich (weshalb oft vordefinierte Bausteine zur Verfügung stehen). Bei Booleschen Komponenten (wie z.B. den häufig verwendeten Orakel-Funktionen [NC02]) ist dies jedoch nicht der Fall. Diese hängen oft von der jeweiligen Anwendung und der konkret betrachteten Instanz ab. Da Quantenschaltungen inhärent reversibel sind [NC02], benötigt man für deren Entwurf Ansätze zur Synthese von reversiblen Schaltungen. Diese Schaltungen unterscheiden sich aber fundamental von bisherigen (elektronischen) Schaltungen, weshalb hierfür komplett neue Syntheseverfahren benötigt werden.

- Schließlich sind Methoden zur **Abbildung auf reale Quantencomputer** nötig, um die jeweiligen Quantenalgorithmen (die, wie klassische Algorithmen, in der Regel in einer Hochsprache beschrieben werden) auf die elementaren Operationen der Zielarchitektur (beispielsweise den Quantencomputern von IBM oder Google) zu übersetzen. Als Zielbeschreibung dienen dabei so genannte Quantenschaltungen bestehend aus (elementaren) Quantengattern, die jeweils nur ein Qubit beeinflussen, aber von mehreren anderen Qubits kontrolliert werden können. Eine große Herausforderung besteht insbesondere darin, die logischen Qubits der Anwendung auf die physikalischen Qubits der Zielarchitektur abzubilden. Dies ist oft nicht trivial, da die möglichen Interaktionen zwischen Qubits auf den Architekturen stark limitiert sind. Aus diesem Grund muss deren Position stets angepasst werden, was zusätzliche Gatter (so genannte SWAP-Gatter zum Tauschen zweier Qubit-Werte) benötigt. Dies wiederum erhöht die Kosten und insbesondere die Fehleranfälligkeit der Berechnung. Hier effiziente Lösungen zu finden ist ein NP-hartes Problem [BKM18].

Im Bereich des Entwurfs konventioneller Schaltungen und Systeme sind ähnliche Aufgaben wohldefiniert und werden seit Jahrzehnten von Forscherinnen und Forschern bearbeitet. Dies führte zu hochentwickelten Entwurfsmethoden und Werkzeugen für Software und Hardware, die heutzutage für selbstverständlich gehalten werden und mit denen Aufgaben enormer Komplexität bearbeitet werden können (immerhin ließen sich damit heutige Schaltungen und Systeme entwerfen, die zum Teil aus über einer Milliarde Komponenten bestehen und astronomische Zustandsräume besitzen). Diese Werkzeuge und Methoden sind ein Hauptgrund dafür, dass elektronische Geräte aus praktisch allen Bereichen unseres Alltags nicht mehr wegzudenken sind.

Für Quantencomputer stellt sich diese Situation jedoch komplett anders dar: Obwohl hier in den letzten Jahren viel in die Entwicklung erster Tools investiert wurde (was z.B. zu SDKs wie IBMs *Qiskit*, Googles *Cirq*, Microsofts *Quantum Development Kit* oder Rigettis *Forest* führte), liefert keine dieser Entwicklungsumgebungen Lösungen mit einer Performanz und Qualität, wie man es aus dem konventionellen Bereich gewohnt ist. Dies liegt unter anderem daran, dass Expertise aus dem Bereich der Entwurfsautomatisierung für konventionelle Schaltungen und Systeme bisher noch nicht durchgehend in die "Quantenwelt" übertragen werden konnte.

Die Dissertation [Zu19] setzt das ambitionierte Ziel, dies zu ändern und Expertise aus jahrzehntelanger Forschung im Bereich der Entwurfsautomatisierung für den Quantencomputer nutzbar zu machen. Dies führte zur Entwicklung zahlreicher Methoden, welche von Kerntechniken der Entwurfsautomatisierung inspiriert wurden und damit die oben angeführten Aufgaben deutlich effizienter und besser bewältigen. Umfangreiche Evaluationen bestätigen, dass dadurch Ergebnisse erzielt werden können, welche die Ergebnisse der bisherigen Ansätze um mehrere Größenordnungen (in Bezug auf Laufzeit und Qualität) verbessern. In den folgenden Abschnitten werden die im Rahmen der Dissertation entwickelten Ansätze kurz vorgestellt und ihr Potential diskutiert. Sie werden zeigen, dass die Nutzung von Methoden der Entwurfsautomatisierung im Bereich des Quantencomputing letztlich wesentlich für die Etablierung dieser neuen Technologie in praxisrelevanten Anwendungen ist.

2 Simulation von Quantenschaltungen

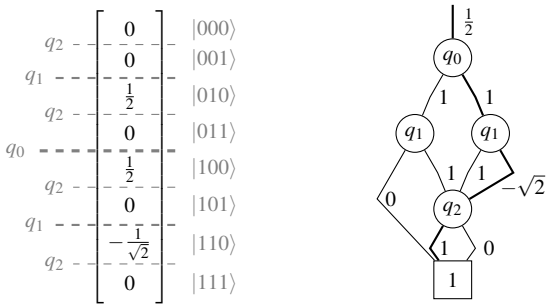
Die Simulation von Quantenschaltungen ist ein rechenintensiver Prozess, da der Zustand eines Quantensystems üblicherweise mit Hilfe eines komplexen Vektors mit exponentieller Größe beschrieben wird. Unitäre Matrizen, die ebenfalls exponentiell mit der Anzahl der involvierten Qubits wachsen, beschreiben Operationen der einzelnen Quantengatter. Die Simulation erfolgt durch schrittweise Multiplikation aller Matrizen der Gatter mit dem initialen Zustandsvektor.[2] Da der Zustandsvektor exponentiell mit der Anzahl der Qubits anwächst, sind selbst Supercomputer mit Petabytes an Speicher und tausenden Kernen auf Simulationen von Quantenschaltungen mit weniger als 50 Qubits limitiert [SSA16].

Um dieses Problem (zumindest in einigen Fällen) zu umgehen, liefert die Dissertation einen komplementären Ansatz, der auf Entscheidungsdiagrammen (eng.: decision diagrams [Br86]) basiert. Entscheidungsdiagramme haben sich bereits beim Entwurf von konventionellen Schaltungen und Systemen als überaus mächtig erwiesen und erlauben die kompakte Repräsentation zahlreicher Schaltungsbeschreibungen (z.B. in Form von Booleschen Funktionen). Die generelle Idee ist dabei, Redundanzen in der Beschreibung der Zustände und Operationen durch geteilte Strukturen zu beschreiben.

Im Kontext der Dissertation wurde eine für den Bereich Quantencomputing spezialisierte Form von Entscheidungsdiagrammen entwickelt, welche den Zustandsvektor sowie die unitären Matrizen in vielen Fällen kompakter (d.h. in nicht-exponentieller Größe) zu repräsentieren vermag. Dazu werden die jeweiligen Vektoren und Matrizen zur Beschreibung der Zustände und Operationen eines Quantensystems zerlegt und daraus resultierende Redundanzen ausgenutzt. Das soll im Folgenden anhand von Vektoren, welche Zustände eines Quantensystems beschreiben, erläutert werden.

Konkret wird o.B.d.A. ein Quantensystem bestehend aus den Qubits $q_0, q_1, \ldots q_{n-1}$ (wobei q_0 das höchstwertige Qubit darstellt) mit Hilfe eines Vektors der Größe 2^n beschrieben, welche für jede der 2^n möglichen Basiszustände die jeweiligen Amplituden $\alpha_{00\ldots0}$, $\alpha_{00\ldots1}$, $\ldots$, $\alpha_{11\ldots1}$ enthält. Die oberen 2^{n-1} Einträge beschreiben damit die Amplituden der Basiszustände mit $q_0 = |0\rangle$, während die unteren 2^{n-1} Einträge die Amplituden für die Basiszustände mit $q_0 = |1\rangle$ beschreiben. In einem Entscheidungsdiagramm wird dieser Zusammenhang nun durch einen Knoten mit der Markierung q_0 und zwei ausgehenden Kanten (einer linken Kante für den oberen Teilvektor und einer rechten für den unteren Teilvektor) dargestellt. Führt man diese Zerlegung rekursiv fort, so erhält man letztendlich Knoten, die einzelne Einträge enthalten (d.h. Vektoren der Größe 1). Eine kompaktere Darstellung des Entscheidungsdiagramms erhält man dann, indem identische Teilvektoren durch mehrfach genutze einzelne Knoten repräsentiert werden. Weitere Konzepte wie Kantengewichte ermöglichen zusätzliche Reduktionen in der Darstellungsgröße.

[2] Alternative Simulationsverfahren existieren, welche nicht alle Amplituden des finalen Zustandsvektor berechnen (sofern dies der Anwendungsfall erlaubt) [Sh17]. Dies ändert jedoch nichts an der exponentiellen Komplexität der Simulation.



(a) Zustandsvektor (b) Entscheidungsdiagramm

Abb. 1: Repräsentationen des Zustands eines Quantensystems

Beispiel 1. *Abbildung 1a zeigt einen Zustandsvektor eines Quantensystems bestehend aus drei Qubits, d.h. die Amplituden der* $2^3 = 8$ *möglichen Basiszustände (die rechts vom Vektor annotiert sind). Abbildung 1b zeigt das dazu gehörige Entscheidungsdiagramm. Um nun z.B. den Wert der Amplitude für* $|q_0 q_1 q_2\rangle = |110\rangle$ *zu erhalten, folgt man im Entscheidungsdiagramm zuerst zweimal dem rechten Nachfolger (für* $q_0 = 1$ *und* $q_1 = 1$*) und danach dem linken Nachfolger (für* $q_2 = 0$*). Dabei werden alle Gewichte entlang des Pfades multipliziert, d.h.* $|110\rangle = \frac{1}{2} \cdot 1 \cdot -\sqrt{2} \cdot 1 = -\frac{1}{\sqrt{2}}$*. Auf diese Weise lassen sich alle Amplituden kompakter darstellen als mit der bisher üblichen Vektor-Notation (konkret reichen vier Knoten im Vergleich zu acht Vektoreinträgen).*

Die oftmals viel kompaktere Darstellung von Vektoren und Matrizen erlaubt es, den für die Repräsentation von Quantenzuständen und Quantenoperationen nötigen Speicherbedarf enorm zu reduzieren. Da zudem die nötigen Matrix-Vektor-Multiplikationen direkt in den Entscheidungsdiagrammen ausgeführt werden können und deren Komplexität hier nur polynomial mit der Anzahl der Knoten wächst, kann dadurch die Simulationszeit enorm verringert werden. Dies ermöglicht es, in vielen relevanten Situationen den aktuellen Stand der Technik deutlich zu übertreffen. Beispielsweise konnte die Simulation einer Instanz des Shors Algorithmus (zur Faktorisierung von Semiprimzahlen) in weniger als zwei Minuten durchführt werden, während z.B. Microsofts Simulator LIQ*Ui*$|\rangle$ [WS14] mehr als 30 Tage für diese Aufgabe benötigte.

Implementierungen des Ansatzes (unter anderem mit verschiedenen weiteren Optimierungen und Anpassungen, die im Detail in der Dissertation [Zu19] beschrieben sind) wurden als Open Source-Projekte öffentlich gemacht. Neben der akademischen Community wurde dieser komplementäre Ansatz auch bereits von großen, im Bereich Quantencomputing tätigen Firmen aufgegriffen und anerkannt. So führten die Arbeiten am Simulator zu einem *Google Research Award* und die Integration der Implementierung in IBMs SDK *Qiskit* sowie der *Quantum Learning Machine* von Atos. Dies unterstreicht den Einfluss, den die Nutzung einer Kernmethode der Entwurfsautomatisierung (nämlich Entscheidungsdiagramme) und dessen Anpassung für den Bereich des Quantencomputings in diesem Feld haben kann.

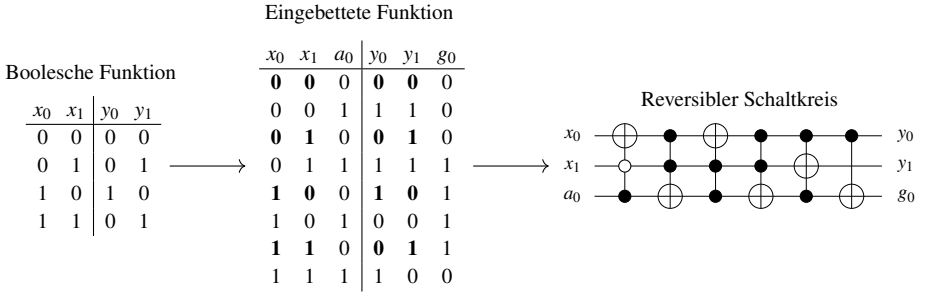


Abb. 2: Funktionale Synthese reversibler Schaltungen

3 Entwurf von Booleschen Komponenten

Die Dissertation bietet neue Ansätze zur Synthese von reversiblen Schaltungen, die für den Entwurf Boolescher Komponenten in Quantenalgorithmen benötigt werden. Obwohl diese Booleschen Komponenten (auch *Orakel* genannt) aufgrund der Reversibilität von Quantenoperationen inhärent reversibel sind, werden sie wegen ihrer Komplexität üblicherweise in nicht-reversible Komponenten zerlegt. Dies führt teils zu beträchtlichen Mehraufwänden, da diese nicht-reversiblen Teile erneut in eine reversible Funktion eingebettet werden müssen. Dies erfordert in der Regel weitere Qubits (sogenannte Hilfsqubits), welche durch die beschränkte Anzahl an physikalisch verfügbaren Qubits aber oft stark limitiert sind.

Beispiel 2. *Abbildung 2 skizziert die Synthese einer nicht-reversiblen Funktion in einen reversiblen Schaltkreis. Die in der linken Tabelle gegebene Boolesche Funktion ist nicht reversibel, da die Ausgabe* $y_0y_1 = 01$ *zweimal definiert ist (man also nicht eindeutig aus dem Ausgangsmuster das jeweilige Eingangsmuster ableiten kann). Um dies zu beheben muss eine zusätzliche Variable (ein Hilfsqubit) hinzugefügt werden, die in einem der beiden Fälle den Wert* 0 *und im anderen Fall den Wert* 1 *annimmt. Eine mögliche einbettende reversible Funktion wird in der Mitte von Abbildung 2 gezeigt. Diese Funktion wird dann mittels Syntheseansatz in einen reversiblen Schaltkreis bestehend aus Toffoli Gatter übersetzt (siehe rechte Seite von Abbildung 2).*[3]

Die große Herausforderung dieser Schritte liegt nun zum Einen in der begrenzten Skalierbarkeit der bisher verfügbaren Ansätze (die oft noch auf Wahrheitstabellen beruhen und dadurch exponentiell in der Größe der Eingangsvariablen wachsen) und zum Anderen in den zusätzlich benötigten Qubits. In der Dissertation wurden hierfür verschiedene Ideen entwickelt, die diese Probleme adressieren. Dies führte unter anderem zu einem Ansatz, der die Einbettung einer nicht-reversiblen Funktion mit der *minimalen* Anzahl an Hilfsqubits durchführt und dabei trotzdem für große Funktionen skalierbar bleibt. Ähnlich wie bei der Simulation zuvor werden dabei die einzubettenden Funktionen in Form von Matrizen beschrieben, die schließlich effizient mit Hilfe von Entscheidungsdiagrammen repräsentiert und bearbeitet werden können.

[3] Die genaue Semantik der Gatter ist in der Dissertation [Zu19] natürlich erläutert, für die Diskussion im Folgenden aber nicht nötig.

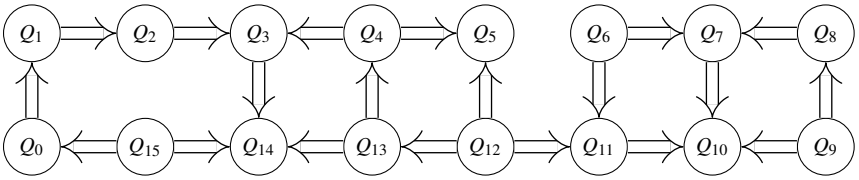


Abb. 3: IBM Q 16 Rueschlikon V1.0.0 (IBM QX3) [IB]

Weitere Verbesserungen werden erreicht, indem die beiden Schritte der Einbettung und der Synthese miteinander kombiniert werden (was zu einer sogenannten One-Pass Synthese führte). Dies verbessert die Skalierbarkeit der funktionalen Synthese weiter, da die originale (nicht-reversible) Funktion mit weniger Variablen arbeitet und dadurch das exponentielle Wachstum verringert wird. Gleichzeitig können dadurch die Kosten der resultierenden Schaltung deutlich verringert werden, da Freiheitsgrade nun gleichzeitig für Einbettung *und* Synthese ausgenutzt werden können. Und schließlich wurden noch Verfahren vorgestellt, welche mit Hilfe von Kodierungen der Ausgangsmuster erreichten, dass die Anzahl der benötigen Hilfsqubits unter die bisher bekannte minimale Anzahl reduziert werden konnte.

Insgesamt konnten damit neuartige und komplementäre Ansätze für die Synthese Boolescher Komponenten vorgestellt werden, die sowohl in Skalierbarkeit als auch Qualität dem bisherigen Stand der Technik deutlich überlegen sind.

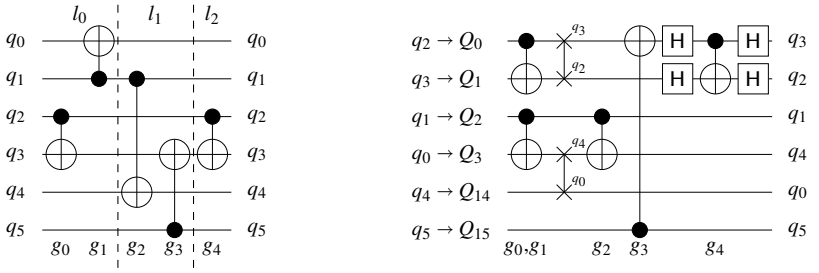
4 Abbildung von Quantenschaltkeisen auf reale Quantencomputer

Um Quantenschaltungen auf entsprechenden Quantencomputern auszuführen, müssen diese auf die Zielarchitektur abgebildet werden.[4] Genauer gesagt müssen die n logischen Qubits des Schaltkreises auf die $m \geq n$ physikalischen Qubits eines Quantencomputers abgebildet werden, während gleichzeitig sogenannte Coupling-Bedingungen erfüllt werden. Diese Bedingungen resultieren aus der physikalischen Limitation, dass nicht alle Qubits beliebig miteinander interagieren können. Tatsächlich werden die jeweils möglichen Interaktionen einer Architektur beim Entwurf eines Quantenrechners definiert und in Form einer sogenannten Coupling-Map beschrieben.

Beispiel 3. *Abbildung 3 zeigt die Coupling-Map von IBMs QX3 Quantencomputer. Die Knoten beschreiben die physikalischen Qubits, während die Kanten angeben, welche Qubits miteinander interagieren können. Die Richtung der Kante gibt zusätzlich an, welche Qubits die Operation kontrolliert und auf welches Qubit die Operation angewendet wird.*

Mit Ausnahme von trivialen Fällen existiert üblicherweise keine Abbildung, die alle Coupling-Bedingungen erfüllt. Daher werden zusätzliche Quantenoperationen (beispielsweise SWAP Operationen, die den Zustand von zwei physikalischen Qubits tauschen) in den Schaltkreis eingefügt, um die Abbildung dynamisch zu ändern.

[4] Die Dissertation geht nicht auf die Übersetzung in elementare Operationen ein, da hier bereits einige effiziente Methoden verfügbar sind [Am13].



(a) Originaler Schaltkreis (b) Abgebildeter Schaltkreis

Abb. 4: Abbildung eines Quantenschaltkreises auf IBM QX3

Beispiel 4. *Abbildung 4a zeigt einen Schaltkreis mit sechs logischen Qubits $q_0, q_1, \ldots q_5$, der auf IBM QX3 abgebildet werden soll. Abbildung 4b zeigt eine entsprechende Lösung, die alle Coupling-Bedingungen erfüllt. Die initiale Abbildung der Qubits ist dabei an der linken Seite von Abbildung 4b dargestellt. Die ersten beiden Gatter können damit problemlos ausgeführt werden, da die Interaktionen der betreffenden Qubits laut der Coupling-Map von Abbildung 3 möglich ist. Diese initiale Abbildung erlaubt aber nicht mehr die Ausführung des dritten und vierten Gatters. Entsprechend sind im Schaltkreis von Abbildung 4b SWAP Gatter hinzugefügt, welche den Zustand der Qubits tauschen, so dass die betreffenden logischen Qubits auf physikalische Qubits abgebildet werden, die miteinander interagieren können.*

Die Frage bleibt, wie diese Abbildung durchgeführt und gleichzeitig die Anzahl der zusätzlich benötigen SWAP-Gatter (die Kosten und Fehleranfälligkeit erhöhen) minimiert werden kann—ein NP-hartes Problem [BKM18]. Die Dissertation liefert mehrere Ansätze für dieses Problem.

So stellt sie ein erstes exaktes Verfahren vor, dass Lösungen mit einer minimalen Anzahl SWAP-Gatter ermittelt. Hierzu wurde eine symbolische Beschreibung des Problems entwickelt, die anschließend mit Hilfe von SAT-Beweisern bearbeitet wurde. Dadurch war erstmals die Ermittlung unterer Schranken für dieses Problem möglich. Außerdem wurde das noch freizulegende Potential aufgezeigt: So liefert IBMs eigener Ansatz zur Lösung des Problems im Schnitt um mehr als 100 % teurere Lösungen als eigentlich nötig—und das bereits bei sehr kleinen Schaltungen.

Dadurch motiviert wurden weitere Heuristiken entwickelt, die das Problem auch skalierbar lösbar machen. Erneut wurden dabei Methoden der Entwurfsautomatisierung genutzt; genauer gesagt wurden A*-Suchalgorithmen mit speziellen Optimierungen verwendet. Mit zusätzlichen Vor- und Nachbearbeitungen konnte schließlich eine Compilation-Methode entwickelt werden, die erneut derzeitigen Lösungen stark überlegen ist.

Der entwickelte Ansatz wurde zum Gewinner der *IBM Qiskit Developer Challenge* gekürt, und ebenfalls in IBMs SDK Qiskit und der Quantum Learning Machine von Atos integriert.

5 Schlussbemerkungen

Die Dissertation liefert substanzielle Beiträge für die Nutzung von Quantencomputern. Wichtige Entwurfsaufgaben wie die Simulation, der Entwurf Boolescher Komponenten und die Abbildung auf reale Quantencomputer konnten effizienter und in deutlich höherer Qualität als bisher adressiert werden. Dies wurde durch die gezielte Nutzung von Methoden und Expertise aus dem Bereich der Entwurfsautomatisierung für konventionelle Schaltungen und Systeme möglich.

Die Auswirkungen der gemachten Entwicklungen sind bereits jetzt sehr deutlich: In drei Jahren Entwicklungszeit entstanden Methoden, welche nicht nur den Gewinn der *IBM Qiskit Developer Challenge* oder die Auszeichnung des *Google Research Award* ermöglichten, sondern auch schon jetzt in Tools von IBM und Atos integriert sind. Dies zeigt deutlich das Potential der entwickelten Methoden und den Effekt, den Entwurfsautomatisierung auch im Bereich Quantencomputing hat und haben wird.

Literaturverzeichnis

[Am13] Amy, Matthew; Maslov, Dmitri; Mosca, Michele; Roetteler, Martin: A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 32(6):818–830, 2013.

[Ar19] Arute, Frank; Arya, Kunal; Babbush, Ryan; Bacon, Dave; Bardin, Joseph C; Barends, Rami; Biswas, Rupak; Boixo, Sergio; Brandao, Fernando GSL; Buell, David A et al.: Quantum supremacy using a programmable superconducting processor. Nature, 574(7779):505–510, 2019.

[BKM18] Botea, Adi; Kishimoto, Akihiro; Marinescu, Radu: On the Complexity of Quantum Circuit Compilation. In: Symposium on Combinatorial Search. 2018.

[Br86] Bryant, Randal E.: Graph-Based Algorithms for Boolean Function Manipulation. IEEE Trans. on Computers, 35(8):677–691, 1986.

[BRB90] Brace, Karl S; Rudell, Richard L; Bryant, Randal E: Efficient Implementation of a BDD Package. In: Design Automation Conf. S. 40–45, 1990.

[FDJ13] Fowler, Austin G; Devitt, Simon J; Jones, Cody: Surface code implementation of block code state distillation. Scientific reports, 3:1939, 2013.

[Go18] Gomes, Lee: Quantum Computing: Both Here and Not Here. IEEE Spectrum, 55(4):42–47, 2018.

[Gr96] Grover, Lov K.: A Fast Quantum Mechanical Algorithm for Database Search. In: Symposium on the Theory of Computing. S. 212–219, 1996.

[IB] IBM Q team: , IBM Q 16 Rueschlikon backend specification V1.0.0. `https://ibm.biz/qiskit-rueschlikon`. Accessed: 2019-06-15.

[IBM19] IBM Unveils World's First Integrated Quantum Computing System for Commercial Use. `https://newsroom.ibm.com/2019-01-08-IBM-Unveils-Worlds-First-Integrated-Quantum-Computing-System-for-Commercial-Use`, 2019. Accessed: 2019-06-15.

[Mo16] Montanaro, Ashley: Quantum algorithms: an overview. npj Quantum Information, 2:15023, 2016.

[NC02] Nielsen, Michael A; Chuang, Isaac: Quantum Computation and Quantum Information. AAPT, 2002.

[Pr18] Preskill, John: Quantum Computing in the NISQ era and beyond. Quantum, 2:79, 2018.

[Sh97] Shor, Peter W.: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM Jour. of Comp., 26(5):1484–1509, 1997.

[Sh17] Shi, Andrew: Recursive Path-Summing Simulation of Quantum Computation. arXiv preprint arXiv:1710.09364, 2017.

[SHR18] Soeken, Mathias; Haener, Thomas; Roetteler, Martin: Programming quantum computers using design automation. In: Design, Automation and Test in Europe. IEEE, S. 137–146, 2018.

[SSA16] Smelyanskiy, Mikhail; Sawaya, Nicolas P. D.; Aspuru-Guzik, Alán: qHiPSTER: The Quantum High Performance Software Testing Environment. arXiv preprint arXiv:1601.07195, 2016.

[WS14] Wecker, Dave; Svore, Krysta M: LIQUi|>: A Software Design Architecture and Domain-Specific Language for Quantum Computing. arXiv preprint arXiv:1402.4467, 2014.

[Zu19] Zulehner, Alwin: Design Automation for Quantum Computing. dissertation, Johannes Kepler University, Linz, 2019.

Alwin Zulehner erhielt seinen Master- und Doktortitel von der Johannes Kepler Universität Linz in Österreich (2015 beziehungsweise 2019). Sein Forschungsinteresse liegt im Bereich der Entwurfsautomatisierung für Quantencomputing. In diesem Bereich hat Alwin Zulehner zahlreiche Paper bei renommierten internationalen Konferenzen und in Fachzeitschriften veröffentlicht, wie beispielsweise den *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems* (TCAD), der *Asia and South Pacific Design Automation Conference* (ASP-DAC), der *Design, Automation and Test in Europe* (DATE) Konferenz, der *Design Automation Conference* (DAC), und der *International Conference on Computer-Aided Design* (ICCAD). Für seine Beiträge wurde er zudem mit mehreren *Best Student Awards* und dem *Early Research Achievement Award* der JKU ausgezeichnet. Darüber hinaus hat er die *IBM Qiskit Developer Challenge* mit seinem Compiler für Quantencomputer gewonnen und wesentlich zu einem *Google Research Award* beigetragen.

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
Vol. D-15: Ausgezeichnete Informatikdissertationen 2014
Vol. D-16: Ausgezeichnete Informatikdissertationen 2015
Vol. D-17: Ausgezeichnete Informatikdissertationen 2016
Vol. D-18: Ausgezeichnete Informatikdissertationen 2017
Vol. D-19: Ausgezeichnete Informatikdissertationen 2018
Vol. D-20: Ausgezeichnete Informatikdissertationen 2019

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 Lan-guage to Information Systems, NLDB'2001.

P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensor-gestützte 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 Hoch-schule, 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 Magen-heim (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.

P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeits-tagung.

P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Man-agement: 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 Verletz-barkeit meistern, 16. DFN Arbeitstagung.

P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informations-systeme – 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 (Ergänzungs-band).

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, Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 24. July 2003 in Darmstadt, Germany

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^2CS 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-229 Dagmar Lück-Schneider, Thomas
Gordon, Siegfried Kaiser, Jörn von
Lucke,Erich Schweighofer, Maria
A.Wimmer, Martin G. Löhe (Hrsg.)
Gemeinsam Electronic Government
ziel(gruppen)gerecht gestalten und
organisieren
Gemeinsame Fachtagung
Verwaltungsinformatik (FTVI) und
Fachtagung Rechtsinformatik (FTRI)
2014, 20.-21. März 2014 in Berlin

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

P-238 Arno Ruckelshausen, Hans-Peter
Schwarz, Brigitte Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und
Ernährungswirtschaft
Referate der 35. GIL-Jahrestagung
23. – 24. Februar 2015, Geisenheim

P-239 Uwe Aßmann, Birgit Demuth, Thorsten
Spitta, Georg Püschel, Ronny Kaiser
(Hrsg.)
Software Engineering & Management
2015
17.-20. März 2015, Dresden

P-240 Herbert Klenk, Hubert B. Keller, Erhard
Plödereder, Peter Dencker (Hrsg.)
Automotive – Safety & Security 2015
Sicherheit und Zuverlässigkeit für
automobile Informationstechnik
21.–22. April 2015, Stuttgart

P-241 Thomas Seidl, Norbert Ritter,
Harald Schöning, Kai-Uwe Sattler,
Theo Härder, Steffen Friedrich,
Wolfram Wingerath (Hrsg.)
Datenbanksysteme für Business,
Technologie und Web (BTW 2015)
04. – 06. März 2015, Hamburg

P-242 Norbert Ritter, Andreas Henrich,
Wolfgang Lehner, Andreas Thor,
Steffen Friedrich, Wolfram Wingerath
(Hrsg.)
Datenbanksysteme für Business,
Technologie und Web (BTW 2015) –
Workshopband
02. – 03. März 2015, Hamburg

P-243 Paul Müller, Bernhard Neumair, Helmut
Reiser, Gabi Dreo Rodosek (Hrsg.)
8. DFN-Forum
Kommunikationstechnologien
06.–09. Juni 2015, Lübeck

P-244 Alfred Zimmermann,
Alexander Rossmann (Eds.)
Digital Enterprise Computing
(DEC 2015)
Böblingen, Germany June 25-26, 2015

P-245 Arslan Brömme, Christoph Busch ,
Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2015
Proceedings of the 14th International
Conference of the Biometrics Special
Interest Group
09.–11. September 2015
Darmstadt, Germany

P-246 Douglas W. Cunningham, Petra Hofstedt,
Klaus Meer, Ingo Schmitt (Hrsg.)
INFORMATIK 2015
28.9.-2.10. 2015, Cottbus

P-247 Hans Pongratz, Reinhard Keil (Hrsg.)
DeLFI 2015 – Die 13. E-Learning
Fachtagung Informatik der Gesellschaft
für Informatik e.V. (GI)
1.–4. September 2015
München

P-248 Jens Kolb, Henrik Leopold, Jan Mendling
(Eds.)
Enterprise Modelling and Information
Systems Architectures
Proceedings of the 6th Int. Workshop on
Enterprise Modelling and Information
Systems Architectures, Innsbruck, Austria
September 3-4, 2015

P-249 Jens Gallenbacher (Hrsg.)
Informatik
allgemeinbildend begreifen
INFOS 2015 16. GI-Fachtagung
Informatik und Schule
20.–23. September 2015

P-250 Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Martin Mikusz, Alexander Volland (Hrsg.)
Projektmanagement und Vorgehensmodelle 2015
Hybride Projektstrukturen erfolgreich umsetzen
Gemeinsame Tagung der Fachgruppen Projektmanagement (WI-PM) und Vorgehensmodelle (WI-VM) im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V., Elmshorn 2015

P-251 Detlef Hühnlein, Heiko Roßnagel, Raik Kuhlisch, Jan Ziesing (Eds.)
Open Identity Summit 2015
10.–11. November 2015
Berlin, Germany

P-252 Jens Knoop, Uwe Zdun (Hrsg.)
Software Engineering 2016
Fachtagung des GI-Fachbereichs Softwaretechnik
23.–26. Februar 2016, Wien

P-253 A. Ruckelshausen, A. Meyer-Aurich, T. Rath, G. Recke, B. Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Fokus: Intelligente Systeme – Stand der Technik und neue Möglichkeiten
Referate der 36. GIL-Jahrestagung
22.-23. Februar 2016, Osnabrück

P-254 Andreas Oberweis, Ralf Reussner (Hrsg.)
Modellierung 2016
2.–4. März 2016, Karlsruhe

P-255 Stefanie Betz, Ulrich Reimer (Hrsg.)
Modellierung 2016 Workshopband
2.–4. März 2016, Karlsruhe

P-256 Michael Meier, Delphine Reinhardt, Steffen Wendzel (Hrsg.)
Sicherheit 2016
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 8. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
5.–7. April 2016, Bonn

P-257 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
9. DFN-Forum Kommunikationstechnologien
31. Mai – 01. Juni 2016, Rostock

P-258 Dieter Hertweck, Christian Decker (Eds.)
Digital Enterprise Computing (DEC 2016)
14.–15. Juni 2016, Böblingen

P-259 Heinrich C. Mayr, Martin Pinzger (Hrsg.)
INFORMATIK 2016
26.–30. September 2016, Klagenfurt

P-260 Arslan Brömme, Christoph Busch, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2016
Proceedings of the 15th International Conference of the Biometrics Special Interest Group
21.–23. September 2016, Darmstadt

P-261 Detlef Rätz, Michael Breidung, Dagmar Lück-Schneider, Siegfried Kaiser, Erich Schweighofer (Hrsg.)
Digitale Transformation: Methoden, Kompetenzen und Technologien für die Verwaltung
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2016
22.–23. September 2016, Dresden

P-262 Ulrike Lucke, Andreas Schwill, Raphael Zender (Hrsg.)
DeLFI 2016 – Die 14. E-Learning Fachtagung Informatik
der Gesellschaft für Informatik e.V. (GI)
11.–14. September 2016, Potsdam

P-263 Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Oliver Linssen, Martin Mikusz, Alexander Volland (Hrsg.)
Projektmanagement und Vorgehensmodelle 2016
Arbeiten in hybriden Projekten: Das Sowohl-als-auch von Stabilität und Dynamik
Gemeinsame Tagung der Fachgruppen Projektmanagement (WI-PM) und Vorgehensmodelle (WI-VM) im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V., Paderborn 2016

P-264 Detlef Hühnlein, Heiko Roßnagel, Christian H. Schunck, Maurizio Talamo (Eds.)
Open Identity Summit 2016
der Gesellschaft für Informatik e.V. (GI)
13.–14. October 2016, Rome, Italy

P-265 Bernhard Mitschang, Daniela Nicklas,Frank Leymann, Harald Schöning, Melanie Herschel, Jens Teubner, Theo Härder, Oliver Kopp, Matthias Wieland (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2017)
6.–10. März 2017, Stuttgart

P-266 Bernhard Mitschang, Norbert Ritter, Holger Schwarz, Meike Klettke, Andreas Thor, Oliver Kopp, Matthias Wieland (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2017)
Workshopband
6.–7. März 2017, Stuttgart

P-267 Jan Jürjens, Kurt Schneider (Hrsg.)
Software Engineering 2017
21.–24. Februar 2017, Hannover

P-268 A. Ruckelshausen, A. Meyer-Aurich, W. Lentz, B. Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Fokus: Digitale Transformation – Wege in eine zukunftsfähige Landwirtschaft
Referate der 37. GIL-Jahrestagung
06.–07. März 2017, Dresden

P-269 Peter Dencker, Herbert Klenk, Hubert Keller, Erhard Plödereder (Hrsg.)
Automotive – Safety & Security 2017
30.–31. Mai 2017, Stuttgart

P-270 Arslan Brömme, Christoph Busch, Antitza Dantcheva, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2017
20.–22. September 2017, Darmstadt

P-271 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
10. DFN-Forum Kommunikations-technologien
30. – 31. Mai 2017, Berlin

P-272 Alexander Rossmann, Alfred Zimmermann (eds.)
Digital Enterprise Computing (DEC 2017)
11.–12. Juli 2017, Böblingen

P-273 Christoph Igel, Carsten Ullrich, Martin Wessner (Hrsg.)
BILDUNGSRÄUME
DeLFI 2017
Die 15. e-Learning Fachtagung Informatik der Gesellschaft für Informatik e.V. (GI)
5. bis 8. September 2017, Chemnitz

P-274 Ira Diethelm (Hrsg.)
Informatische Bildung zum Verstehen und Gestalten der digitalen Welt
13.–15. September 2017, Oldenburg

P-275 Maximilian Eibl, Martin Gaedke (Hrsg.)
INFORMATIK 2017
25.–29. September 2017, Chemnitz

P276 Alexander Volland, Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Oliver Linssen, Martin Mikusz (Hrsg.)
Projektmanagement und Vorgehensmodelle 2017
Die Spannung zwischen dem Prozess und den Menschen im Projekt
Gemeinsame Tagung der Fachgruppen Projektmanagement und Vorgehensmodelle im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V. in Kooperation mit der Fachgruppe IT-Projektmanagement der GPM e.V., Darmstadt 2017

P-277 Lothar Fritsch, Heiko Roßnagel, Detlef Hühnlein (Hrsg.)
Open Identity Summit 2017
5.–6. October 2017, Karlstad, Sweden

P-278 Arno Ruckelshausen, Andreas Meyer-Aurich, Karsten Borchard, Constanze Hofacker, Jens-Peter Loy, Rolf Schwerdtfeger, Hans-Hennig Sundermeier, Helga Floto, Brigitte Theuvsen (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Referate der 38. GIL-Jahrestagung
26.–27. Februar 2018, Kiel

P-279 Matthias Tichy, Eric Bodden, Marco Kuhrmann, Stefan Wagner, Jan-Philipp Steghöfer (Hrsg.)
Software Engineering und Software Management 2018
5.–9. März 2018, Ulm

P-280 Ina Schaefer, Dimitris Karagiannis, Andreas Vogelsang, Daniel Méndez, Christoph Seidl (Hrsg.)
Modellierung 2018
21.–23. Februar 2018, Braunschweig

P-281 Hanno Langweg, Michael Meier, Bernhard C. Witt, Delphine Reinhardt (Hrsg.)
Sicherheit 2018
Sicherheit, Schutz und Zuverlässigkeit
25.–27. April 2018, Konstanz

P-282 Arslan Brömme, Christoph Busch, Antitza Dantcheva, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2018
Proceedings of the 17th International Conference of the Biometrics Special Interest Group
26.–28. September 2018
Darmstadt, Germany

P-283 Paul Müller, Bernhard Neumair, Helmut Reiser, Gabi Dreo Rodosek (Hrsg.)
11. DFN-Forum Kommunikations-technologien
27.–28. Juni 2018, Günzburg

P-284 Detlef Krömker, Ulrik Schroeder (Hrsg.)
DeLFI 2018 – Die 16. E-Learning Fachtagung Informatik
10.–12. September 2018, Frankfurt a. M.

P-285 Christian Czarnecki, Carsten Brockmann, Eldar Sultanow, Agnes Koschmider, Annika Selzer (Hrsg.)
Workshops der INFORMATIK 2018 - Architekturen, Prozesse, Sicherheit und Nachhaltigkeit
26.–27. September 2018, Berlin

P-286 Martin Mikusz, Alexander Volland, Martin Engstler, Masud Fazal-Baqaie, Eckhart Hanser, Oliver Linssen (Hrsg.)
Projektmanagement und Vorgehensmodelle 2018
Der Einfluss der Digitalisierung auf Projektmanagementmethoden und Entwicklungsprozesse
Düsseldorf 2018

P-287 A. Meyer-Aurich, M. Gandorfer, N. Barta, A. Gronauer, J. Kantelhardt, H. Floto (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Fokus: Digitalisierung für landwirtschaftliche Betriebe in kleinstrukturierten Regionen – ein Widerspruch in sich?
Referate der 39. GIL-Jahrestagung
18.–19. Februar 2019, Wien

P-288 Arno Pasternak (Hrsg.)
Informatik für alle
18. GI-Fachtagung
Informatik und Schule
16.-18. September 2019 in Dortmund

P-289 Torsten Grust, Felix Naumann, Alexander Böhm, Wolfgang Lehner, Jens Teubner, Meike Klettke, Theo Härder, Erhard Rahm, Andreas Heuer, Holger Meyer (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2019)
4.–8. März 2019 in Rostock

P-290 Holger Meyer, Norbert Ritter, Andreas Thor, Daniela Nicklas, Andreas Heuer, Meike Klettke (Hrsg.)
Datenbanksysteme für Business, Technologie und Web (BTW 2019)
Workshopband
4.–8. März 2019 in Rostock

P-291 Michael Räckers, Sebastian Halsbenning, Detlef Rätz, David Richter, Erich Schweighofer (Hrsg.)
Digitalisierung von Staat und Verwaltung
Gemeinsame Fachtagung Verwaltungsinformatik (FTVI) und Fachtagung Rechtsinformatik (FTRI) 2019
6.–7. März 2019 in Münster

P-292 Steffen Becker, Ivan Bogicevic,Georg Herzwurm, Stefan Wagner (Hrsg.)
Software Engineering and Software Management 2019
18.–22. Februar 2019 in Stuttgart

P-293 Heiko Roßnagel, Sven Wagner, Detlef Hühnlein (Hrsg.)
Open Identity Summit 2019
28.–29. März 2019
Garmisch-Partenkirchen

P-294 Klaus David, Kurt Geihs, Martin Lange, Gerd Stumme (Hrsg.)
INFORMATIK 2019
50 Jahre Gesellschaft für Informatik – Informatik für Gesellschaft
23.–26. September 2019 in Kassel

P-295 Claude Draude, Martin Lange, Bernhard Sick (Hrsg.)
INFORMATIK 2019
50 Jahre Gesellschaft für Informatik – Informatik für Gesellschaft
Workshop-Beiträge
23.–26. September 2019 in Kassel

P-296 Arslan Brömme, Christoph Busch, Antitza Dantcheva, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2019
Proceedings of the 18th International Conference of the Biometrics Special Interest Group
18.–20. September 2019
Darmstadt, Germany

P-297 Niels Pinkwart, Johannes Konert (Hrsg.)
DELFI 2019 –Die 17. Fachtagung Bildungstechnologien
16.–19. September 2019 in Berlin

P-298 Oliver Linssen, Martin Mikusz, Alexander Volland, Enes Yigitbas, Martin Engstler, Masud Fazal-Baqaie, Marco Kuhrmann (Hrsg.)
Projektmanagement und Vorgehensmodelle 2019 –Neue Vorgehensmodelle in Projekten – Führung, Kulturen und Infrastrukturen im Wandel
1Gemeinsame Tagung der Fachgruppen Projektmanagement (WI-PM), Vorgehensmodelle (WI-VM) und Software Produktmanagement (WI-ProdM) im Fachgebiet Wirtschaftsinformatik der Gesellschaft für Informatik e.V.
in Kooperation mit der Fachgruppe IT-Projektmanagement der GPM e.V.,
Lörrach 2019

P-299 M. Gandorfer, A. Meyer-Aurich, H. Bernhardt, F. X. Maidl, G. Fröhlich, H. Floto (Hrsg.)
Informatik in der Land-, Forst- und Ernährungswirtschaft
Fokus: Digitalisierung für Mensch, Umwelt und Tier
Referate der 40. GIL-Jahrestagung
17.–18. Februar 2020,
Campus Weihenstephan

P-300 Michael Felderer, Wilhelm Hasselbring, Rick Rabiser, Reiner Jung (Hrsg.)
Software Engineering 2020
24.–28. Februar 2020
Innsbruck, Austria

P-301 Delphine Reinhardt, Hanno Langweg, Bernhard C. Witt, Mathias Fischer (Hrsg.)
Sicherheit 2020
Sicherheit, Schutz und Zuverlässigkeit
17.–20. März 2020, Göttingen

P-302 Dominik Bork, Dimitris Karagiannis, Heinrich C. Mayr (Hrsg.)
Modellierung 2020
19.–21. Februar 2020, Wien

P-303 Peter Heisig, Ronald Orth, Jakob Michael Schönborn, Stefan Thalmann (Hrsg.)
Wissensmanagement in digitalen Arbeitswelten: Aktuelle Ansätze und Perspektiven
18.–20.03.2019, Potsdam

P-304 Heinrich C. Mayr, Stefanie Rinderle-Ma, Stefan Strecker (Hrsg.)
40 Years EMISA
Digital Ecosystems of the Future: Methodology, Techniques and Applications
May 15.–17. 2019
Tutzing am Starnberger See

P-305 Heiko Roßnagel, Christian H. Schunck, Sebastian Mödersheim, Detlef Hühnlein (Hrsg.)
Open Identity Summit 2020
26.–27. May 2020, Copenhagen

P-306 Arslan Brömme, Christoph Busch, Antitza Dantcheva, Kiran Raja, Christian Rathgeb, Andreas Uhl (Eds.)
BIOSIG 2020
Proceedings of the 19th International Conference of the Biometrics Special Interest Group
16.–18. September 2020
International Digital Conference

P-308 Raphael Zender, Dirk Ifenthaler, Thiemo Leonhardt, Clara Schumacher (Hrsg.)
DELFI 2020 –
Die 18. Fachtagung Bildungstechnologien der Gesellschaft für Informatik e.V.
14.–18. September 2020
Online

P-310 Anne Koziolek, Ina Schaefer,
Christoph Seidl (Hrsg.)
Software Engineering 2021
22.–26. Februar 2021,
Braunschweig/Virtuell

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