

Ein Konzept zur serviceorientierten Einbindung von Finite-Elemente-Software in Grid und Cloud Computing

Gerald Faschingbauer, Andreas Hollmann & Raimar J. Scherer
Technische Universität Dresden, Institut für Bauinformatik, Dresden, Deutschland
Andreas.Hollmann@tu-dresden.de, Gerald.Faschingbauer@tu-dresden.de, Raimar.Scherer@tu-dresden.de

Kurzfassung

Komplexe Baumaßnahmen im Grundbau sind geprägt von hohen Modellunsicherheiten. Sie erfordern daher prinzipiell eine ständige baubegleitende Überwachung sowie die Anpassung des mechanischen Modells an die vor Ort angetroffenen, durch Messungen und Beobachtungen festgestellten Zustände. Zur Realisierung dieser kontinuierlichen Systemidentifikation wird ein webbasiertes Softwarekonzept mit Integration der Finite-Elemente-Methode in einen kontinuierlich durchlaufenden Mess- und Überwachungszyklus vorgeschlagen, bei der durch gesteuerte Variation von Materialmodellen und Materialparametern Modellkandidaten erzeugt und Simulationsläufe durchgeführt werden. Die hohe Anzahl gleichzeitig zu untersuchender Modellkandidaten erfordert unter anderem die Einbindung verteilter Rechenressourcen. Die erforderliche hohe Rechenleistung soll durch Grid und Cloud Computing bereitgestellt werden. In diesem Beitrag wird die Methodik zur Systemidentifikation sowie insbesondere die Implementierung der Simulations-Kernkomponente, eines Webservices für die Berechnung von Finite-Element-Modellen vorgestellt. Desweiteren wird anhand eines Beispiels auf die zu erwartende Rechenlast eingegangen. Die theoretische Realisierbarkeit der erforderlichen Rechenressourcen durch Grid- oder Cloud-Umgebung wird skizziert.

1 Einführung

Die ständige baubegleitende Überwachung sowie die Anpassung des in der Planungsphase eines Bauprojektes aufgestellten geotechnischen Entwurfs- und Simulationsmodells an die vor Ort angetroffenen, durch Messungen festgestellten Zustände, kann bis heute nicht zufriedenstellend durchgeführt werden.

Die zutreffende Modellierung des nichtlinearen mechanischen Bodenverhaltens ist vor Baubeginn durch die große Heterogenität des Bodens schwer möglich. Während der Bauphase wäre es daher bei komplexen Projekten grundsätzlich erforderlich, das Boden-Bauwerksverhalten durch kontinuierliche sensortechnische Überwachung zu erfassen und das mechanische Modell durch Systemidentifikation an die Messdaten anzupassen, zeitnah auszuwerten und ingenieurmäßig zu interpretieren. Diese Verfahrensweise würde es ermöglichen, realitätsnäher zu planen und so während der Bauphase durch Adaption des Bauverfahrens die Sicherheit für das Bauwerk und seine Umgebung zu erhöhen, Baukosten zu reduzieren, sowie den Kenntnisstand bezüglich des Baugrundverhaltens und damit auch den wissenschaftlichen Kenntnisstand zu erweitern. Aufgrund der hohen Nichtlinearität bodenmechanischer Modelle als auch der Unbestimmtheit der Gleichungssysteme ist die zur Anpassung des mechanischen Modells erforderliche Systemidentifikation auf analytischem Wege nicht möglich. Bei der Anwendung einer simulationsbasierten Systemidentifikation, die auf der Variation der Modellparameter basiert, besteht indes das Problem, dass die Zeit bis das identifizierte geotech-

nische Ingenieurmodell als zuverlässig angesehen werden kann, die zur Verfügung stehende Zeit um Größenordnungen überschreitet. Der hohe manuelle Arbeitsanteil bei einer simulationsbasierten Systemidentifikation, angefangen von der Auswertung der Sensoren, über das Aufstellen der nichtlinearen Modellkandidaten, die nichtlineare Berechnung, iterative Neumodellierung und Neuberechnung sowie der Priorisierung und Auswertung der Simulationsergebnisse lässt eine zeitnahe Systemidentifikation, Systemdiagnose und damit die adaptive Anpassung des Bauverfahrens bisher nicht zu.

Zur Realisierung eines adaptiven Bauverfahrens, bei dem kontinuierlich erfasste Messdaten zum Erkenntnisgewinn und als Entscheidungsgrundlage für die Bauverfahrensteuerung eingesetzt werden, ist also die Integration des überwiegend materialtransformierenden Bauprozesses mit dem überwiegend informationstransformierenden Monitoringprozess erforderlich. Bau- und Monitoringprozess können hierbei nicht als unabhängige Prozesse betrachtet werden, sondern beide Prozesse müssen interagieren, um einerseits einen kontinuierlichen Wissensgewinn zu ermöglichen, der nur durch eine umfangreiche modellbasierte Analyse der während des Bauprozesses an Bauwerk und Boden erfassten Messdaten erreicht werden kann und andererseits die Rückführung der dadurch gewonnenen Erkenntnisse in den Bauprozess durch adaptive Bauverfahrensteuerung zu ermöglichen.

Da der derzeitige, auf reine Schwellwertüberwachung beschränkte, Stand der baubegleitenden Überwachung lediglich schwache Indizien für die Ursachen ungeplanten Systemverhaltens liefert, und daher zur Stellung einer

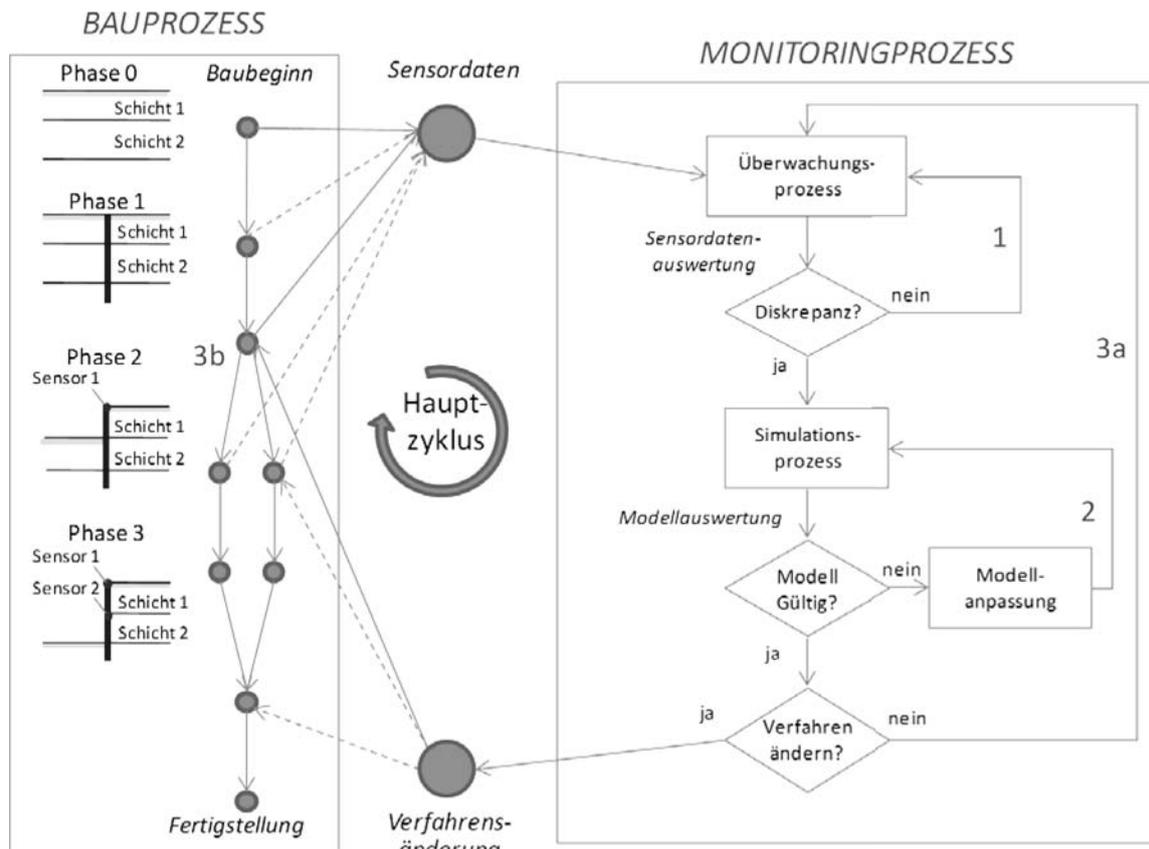


Bild 1: Kontinuierlich durchlaufender, gekoppelter Bau- und Monitoringprozess

Systemdiagnose unzureichend ist, wird in die Ergänzung einer simulationsbasierten Systemidentifikation vorgeschlagen, die durch kontinuierliche Simulation, Modellvalidierung und Modellanpassung realisiert wird. Die durch Systemidentifikation ermittelten Modelle ermöglichen eine ingenieurmäßige Interpretation und damit eine physikalisch begründbare Diagnose des Boden-Bauwerk-Verhaltens. Die hierdurch gewonnenen Erkenntnisse werden bei maschinellen Bauverfahren durch automatische Regelung und bei manuellen Bauverfahren durch verifizierte Entwurfsentscheidungen in den Bauprozess zurückgeführt. Aus den vorgenannten Überlegungen ergibt sich der in Bild 1 grob dargestellte, gekoppelte und kontinuierlich durchlaufende Bau- und Monitoringprozess.

Der Bauprozess wird begleitet durch einen nebenläufigen, kontinuierlichen Monitoringprozess, der ständig Sensordaten mit definierten Grenzwerten vergleicht (Zyklus 1) und bei Überschreitung den Simulationsprozess auslöst. Der Simulationsprozess besteht aus der Untersuchung und Auswertung mehrerer geotechnischer Modellkandidaten, aufgrund deren Ergebnisse entschieden wird, ob eine Modellanpassung notwendig ist. Repräsentiert das aktuelle geotechnische Modell die Messdaten nicht in ausreichender Übereinstimmung, wird der Simulationsprozess (Zyklus 2) so oft wiederholt, bis ein verbessertes, gültiges Modell gefunden wird. Dieser Prozess wird im Weiteren als Systemidentifikation bezeichnet. Auf der Basis eines im Rahmen der Systemidentifikation identifizierten realitätsnahen Modells kann schließlich

darüber entschieden werden, wie Baumaschinen und Bauverfahren zu steuern sind, oder ob eine gänzlich neue Situation eingetreten ist und gegebenenfalls eine Änderung des Bauverfahrens bzw. des Bauwerksentwurfs aus sicherheitstechnischen oder wirtschaftlichen Erwägungen notwendig oder möglich ist.

Ist keine Verfahrensänderung notwendig, wird der Überwachungsprozess planmäßig fortgesetzt (Zyklus 3a). Ist hingegen eine Verfahrensänderung aus wirtschaftlichen oder sicherheitstechnischen Gründen angezeigt, wird im Hauptzyklus aktiv durch Maßnahmen in den Bauprozess eingegriffen (Zyklus 3b) und das Bauverfahren entsprechend geändert.

2 Methodik zur simulations- und wissensbasierten Systemidentifikation

Eine Systemidentifikation ist nur dann kontinuierlich innerhalb sinnvoller Zeitgrenzen durchführbar, wenn sie logisch in den Gesamtprozess der Bauausführung und der Messdatenerfassung eingebettet ist. Aufgrund der Nichtlinearität realitätsnaher bodenmechanischer Modelle als auch der Unbestimmtheit des Systems ist eine Systemidentifikation auf analytischem Wege nicht möglich[4][5]. Daher wird derzeit im Projekt GeoTechControl[9] eine Methodik zur Systemidentifikation umgesetzt, die wissensbasierte Methoden mit numerischer Simulation kombiniert. Die Methodik ist in Bild 2 dargestellt und nachfolgend erläutert.

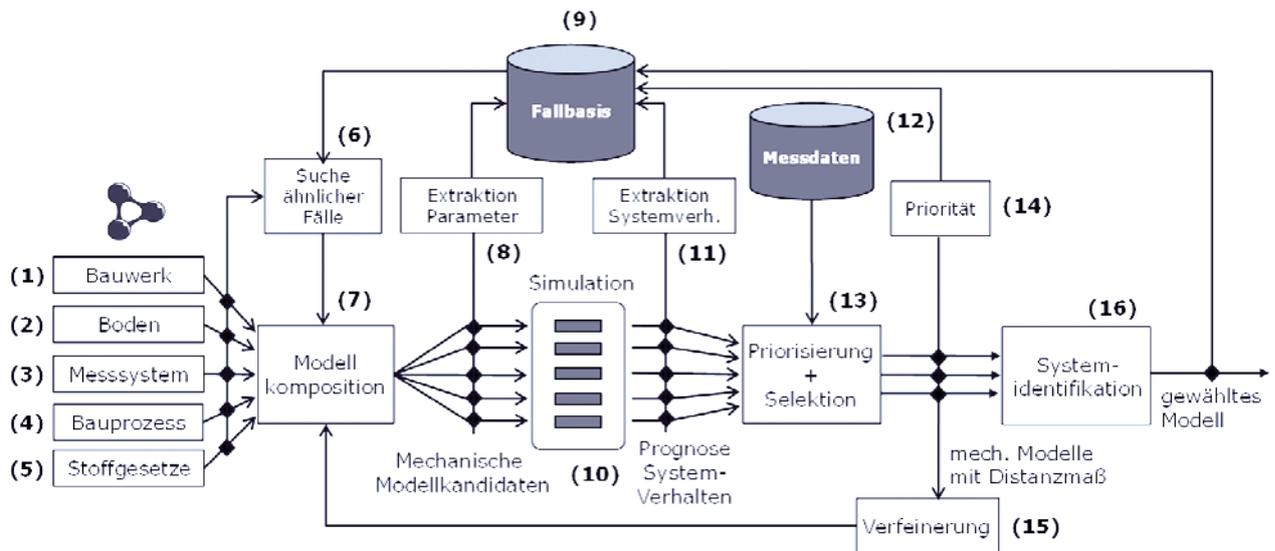


Bild 2: Methodik zur simulations- und wissensbasierten Systemidentifikation

2.1 Wissensrepräsentation der 4D-Bauwerksinformationen

Das Ingenieurbauwerk (1), der Boden (2), das Messsystem (3) sowie der Bauprozess (4) werden als objektorientiertes, parametrisches Modell in einer Wissensbasis auf der Grundlage einer OWL-DL-Beschreibungslogik unter Berücksichtigung der zeitlichen Bauzustände repräsentiert. Die Verknüpfung der Daten mit dem Prozessmodell ist im Hinblick auf die erhebliche zu bearbeitende Datenmenge ein unabdingbarer Basisbaustein für die effiziente Datenverwaltung. Dies stellt den gezielten Zugriff auf Objekt-/Prozessdaten sowie die Filterung nach bestimmten Objekt-/Prozesseigenschaften sicher und unterstützt damit durch die Abfragemöglichkeit zusammengehöriger, vergleichbarer Prognose- und Überwachungsdaten ein effizientes Monitoring des Bauzustandes. Die Repräsentation der Messprogramme, Sensoren, Labor- und Felduntersuchungen mit ihren Beziehungen zum Bauwerk, zu Baumaschinen und zum Bauablauf im Messsystemmodell (3) auf Basis der OWL-DL-Beschreibungslogik bietet durch die Anwendung von logischem Schlussfolgern die Möglichkeit, die Konfiguration des Messsystems sowohl planen als auch leicht umplanen zu können. Das eröffnet die Möglichkeit leicht Variantenuntersuchungen durchzuführen und erleichtert die automatische Datenauswertung durch die Integration von Messdaten, Bauwerksdaten und Prozessdaten. Die Messwerte selbst werden in einer relationalen Datenbank (12) abgespeichert.

Die üblicherweise als Subroutinen implementierten Stoffgesetze (5) werden dezentral abgespeichert und mit ihren Eigenschaften, Anwendungsbereichen und -grenzen durch eine Beschreibungslogik beschrieben. Die Bereitstellung als auch die Beschreibung kann durch den Stoffgesetzautor bei der Veröffentlichung des Stoffgesetzes angelegt werden. Sie enthält die logischen Fakten, die

eine Entscheidung für oder gegen ein Stoffgesetz bei einem geotechnischen Problem ermöglicht.

2.2 Wissensbasierte Modellkomposition

Der Endanwender hat im Rahmen der Modellkomposition (7) die Möglichkeit, unterstützt durch das wissensbasierte System schnell und effizient Finite-Elemente-Modelle aus den in (1) bis (5) repräsentierten Daten und Wissen zu komponieren. Hierbei wird die beschreibungslogikbasierte Registry der Stoffgesetze (5) entsprechend der Baugrundspezifika des Bodenmodells (2) gezielt nach logischen Kriterien abgefragt. Passende Stoffgesetze werden dem Ingenieur zur Auswahl angeboten und nach entsprechender Auswahl zusammen mit den Bauwerks- und Bauprozessinformationen zu FE-Simulationsmodellen kombiniert. Durch eine wissensbasierte Steuerung der Parametervariationen wird die Wahl sinnvoller, realistischer Parameterkombinationen gewährleistet. Die logikbasierte Wissensrepräsentation wird durch eine Beschreibungslogik in OWL-DL und SWRL umgesetzt und mit einer Inferenzmaschine deduktiv ausgewertet. Damit werden passend zum Bauvorhaben und zur Situation theoretisch mögliche Stoffgesetze und sinnvolle Parameter ermittelt. Der Ingenieur wählt die für die Systemidentifikation zu nutzenden Modelle aus dieser Menge aus. Die Untersuchung unrealistischer Parametersets, die durch einfache kombinatorische Verfahren ohne Berücksichtigung des geotechnischen Wissens erfolgen, ist damit reduziert. Dies soll erheblich zur Verminderung von Rechenaufwand und Fehlinterpretationen beitragen. Die Kerninformationen, die aus bereits berechneten FE-Modellen extrahiert wurden – dies betrifft sowohl Stoff- und FE-Parameter (8) als auch das ermittelte Systemverhalten (11) in Form von Spannungen und Verformungen – und die in der Fallbasis (9) hinterlegt wurden, können entsprechend der in (1) bis (4) repräsentierten projektspezifischen Informationen auf Ähnlichkeiten untersucht (6), und je nach Grad der Übereinstimmung als zusätzliche

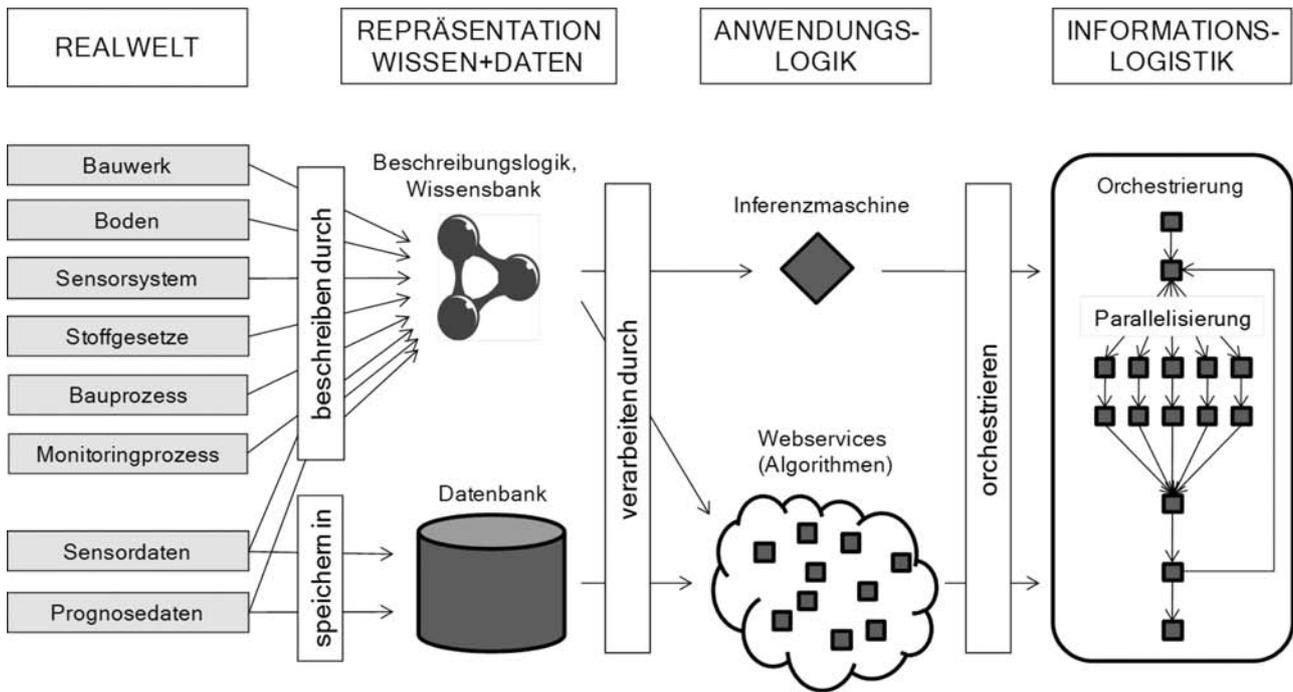


Bild 3: Prinzipieller Aufbau der Systemarchitektur

Informationsquelle zur Modellkomposition genutzt werden.

2.3 Simulation und Identifikation

Mit der durch die Modellkomposition (7) erzeugten Vielzahl mechanischer Modellkandidaten (Finite-Elemente-Modelle) wird durch einen Finite-Elemente-Webservice (FEM-Service) (10) das Systemverhalten von Boden und Bauwerk simuliert, d.h. es wird für jeden Modellkandidaten eine Prognose des Systemverhaltens erzeugt. Diese Prognosen werden anhand der in einer Messdatenbank (12) gespeicherten Sensordaten automatisch priorisiert (13). Die n-besten Modellkandidaten werden zusammen mit ihrer Priorität (14) in der Fallbasis (9) gespeichert, und bei zu geringer Übereinstimmung mit den Messdaten verfeinert (15), bis eine ausreichende Übereinstimmung erzielt wird. Der letztendlich „beste“ Modellkandidat wird im Rahmen der Systemidentifikation (16) durch einen Experten gewählt. Diese Wahl wird wiederum in der Fallbasis (9) registriert, da die Expertenwahl einer besonderen Qualitätszuordnung gleichkommt, die im Weiteren bei der Modellkomposition (7) vorrangig berücksichtigt werden soll.

Zur Vermeidung von physikalisch nicht plausiblen Modellalternativen ist geplant Trendanalysen als Entscheidungsbasis für Modelländerungen einzusetzen.

2.4 Problematik der Rechenleistung

Die erforderliche hohe Anzahl gleichzeitig zu untersuchender Modellkandidaten erfordert die Einbindung verteilter Rechenressourcen. Die vorgestellte Methodik wird daher so umgesetzt, dass durch die Anbindung an Grid- und Cloud-Umgebung die parallele Simulation der Modellkandidaten sowie eine automatisierte Auswertung der

Ergebnisse durch das Schlussfolgern aus der semantischen Beschreibung der Daten möglich ist.

Moderne Softwareentwicklungswerkzeuge ermöglichen das Aufspalten des Systems in wiederverwendbare Module, die mit relativ geringem Aufwand in bestehende Systeme mit Hilfe standardisierter Softwareschnittstellen integriert werden können[3]. Das Softwaresystem zur Systemidentifikation soll drei Kernkomponenten enthalten, die als Webservices implementiert werden. Die Komponenten werden nach ihren Funktionsaspekten unterschieden in: (1) Berechnung der FE-Modelle und Speicherung der FE-Ergebnisse, (2) Erzeugen der Stoffgesetz- und Stoffparametervariationen für die simulationsbasierte Systemidentifikation, (3) Semantische Beschreibung und Suche der FE-Modelle, Ergebnisse und verwendeten Stoffgesetze.

2.5 Architektur des GeoTechControl-Informationssystems

Die Entwicklungskriterien werden durch ein Rahmensoftwaresystem erfüllt, das auf Basis einer vierschichtigen, serviceorientierten Architektur (s. Bild 3) umgesetzt wird.

Die Aspekte der Realwelt bezüglich Bauwerk, Sensorsystem, bodenmechanisches Modell, Bauprozess, Überwachungsprozess, Sensordaten und Prognosedaten werden in einer Wissensbank als logisches System mit den wesentlichen Beziehungen und Regeln in Beschreibungslogik repräsentiert, um ihre logische Auswertbarkeit durch Schlussfolgerung zu ermöglichen. Die implizit definierten logischen Beziehungen der in der Wissensbank repräsentierten Individuen werden durch logische Schlussfolgerung mit Inferenzmaschinen (Reasoner) expliziert. Das in der Wissensbank repräsentierte Wissen wird mit der Abfragesprache SPARQL gezielt nach logischen Krite-

rien abgefragt und über das Netzwerkprotokoll SOAP im Rahmen eines Remote-Procedure-Calls an die Rechen-Webservices zur Echtzeitauswertung und Systemidentifikation übergeben. Die Informationslogistik, d. h. die Weitergabe der Daten von einem Webservice zum nächsten, wird durch eine flexibel anpassbare BPEL-Orchestrierung der Services gesteuert.

Das System soll die prozessgesteuerte Anwendungsintegration von Prognose- und Überwachungssoftware, sowie von Software für Simulation und Simulationsvorbereitung ermöglichen. Ferner soll eine Informationsintegration, die neben direkten Relationen auch transitive Zusammenhänge zwischen Informationen aus Überwachung und Simulation nutzen kann und somit modifizierbar und flexibel erweiterbar ist, realisiert werden.

3 Portierung der FE-Software als Webservice

Der FEM-Service ist eine Softwarekomponente, welche die FEM-Software von SOFiSTiK [14] als Webservice (SOAP/WSDL) für eine Verwendung in einer verteilten Anwendung bereitstellt. Dies dient letztendlich der flexiblen Einbindung in den eingangs beschriebenen komplexen Mess- und Auswertezyklus und ermöglicht zugleich die Portierbarkeit des Finite-Elemente-Kerns in eine Grid- und Cloud-Umgebung.

Zur Integration von FE-Berechnungen auf einem Webserver und zur Realisierung der SOAP-Kommunikation war die Entwicklung einiger Basismethoden notwendig: (1) Methode zur Portierung der FE-Software als Webservice, (2) Methode für das Annotieren und Initialisieren der FE-Modelle, (3) Methode der Datenselektion aus berechneten FE-Ergebnissen.

Die Portierung der FE-Software als Webservice erlaubt zum einen eine gemeinsame Nutzung eines leistungsstarken Servers für die schnellere Berechnung eines einzelnen rechenaufwendigen FE-Modells und zum anderen eine trivialparallele Berechnung der vielen FE-Modellkandidaten in einer Grid- oder Cloud-Umgebung. Der FEM-Service ist die prototypische Implementierung einer Methode zur Portierung einer FE-Software als Webservice. Die derzeitige Version FEM-Services unterstützt die Berechnung von FE-Modellen mit der SOFiSTiK-FEM-Software, ist jedoch auch für die Verwendung mit anderen kommerziellen FE-Produkten erweiterbar.

Die zu portierende FE-Software muss dabei folgende Anforderungen erfüllen: (1) die FE-Software muss die Berechnung eines FE-Modells im Batchmodus unterstützen (2) Die definierten Variablen aus dem FE-Modell müssen eindeutig erkennbar und initialisierbar sein.

Bevor ein FE-Modell berechnet werden kann muss es zum Server hochgeladen und in der Server-Datenbank gespeichert werden. Das in der Datenbank gespeicherte FE-Modell bildet das FE-Referenzmodell, also eine

Schablone des FE-Modells welche für das Generieren neuer FE-Modelle verwendet wird.

```
....
$ Block- und Variablendefinitionen
$ Variablen des WinTUBE Formelrechners
#DEFINE GTC_Boden_E = 100000.0
#DEFINE a = 41.35648
#DEFINE GTC_Boden_phi = 30.5
#DEFINE GTC_Boden_c = 5.123
....
```

Bild 4: Annotation von Variablen im FE-Modell

Die Grundlage für die Erstellung des FE-Referenzmodells bilden im Fall der Nutzung der FEM-Software von SOFiSTiK die im hochgeladenem FE-Modell mit dem „GTC_“ Präfix annotierten Variablen, die vom FEM-Service erkannt und mit neuen Werten belegt werden können. Das Konsortium des GeoTechControl-Projektes definiert eine projektspezifische Nomenklatur, die die Annotation der veränderbaren Variablen durch das Präfix "GTC_" fordert.

Die Variablen des FE-Modells sollen so durch den Ingenieur direkt aus einem FE-Modellierungswerkzeug, wie beispielsweise WinTUBE[7] von FIDES mit Präfixen annotiert und im FEM-Service wiederverwendet werden können. In Bild 4 sind die mit Präfixen annotierten Variablen fettmarkiert dargestellt. Die Variablen GTC_Boden_E, GTC_Boden_phi und GTC_Boden_c werden aufgrund dieses Präfix von dem FEM-Service automatisch erkannt und für die Generierung der neuen FE-Modelle durch die Initialisierung der annotierten Variablen mit neuen Werten verwendet.

3.1 Die Berechnung eines FE-Modells

Die Berechnung eines FE-Modells ist ein recht komplexer, parallelkonkurrierender Prozess, der wegen der Prozesssteuerung betriebssystemabhängig ist und eine betriebssystemspezifische Konfiguration auf dem Server erfordert. In Bild 5 sind die wichtigsten Schritte bei der Berechnung eines FE-Modells auf einem Server veranschaulicht.

Als erstes stellt der Nutzer des FEM-Service eine Anfrage für die Berechnung eines FE-Modells an den FEM-Service. Die Nutzeranfrage enthält die ID des in der Datenbank gespeicherten FE-Referenzmodells sowie eine Liste mit Werten für die Initialisierung der in dem FE-Referenzmodell annotierten Variablen (1).

Bevor eine FE-Berechnung auf dem Server ausgeführt werden kann, muss das FE-Referenzmodell aus der Datenbank in das Dateisystem des Servers als ZIP-Archiv geladen (2) und in einen Dateiverzeichnis dekomprimiert werden (3).

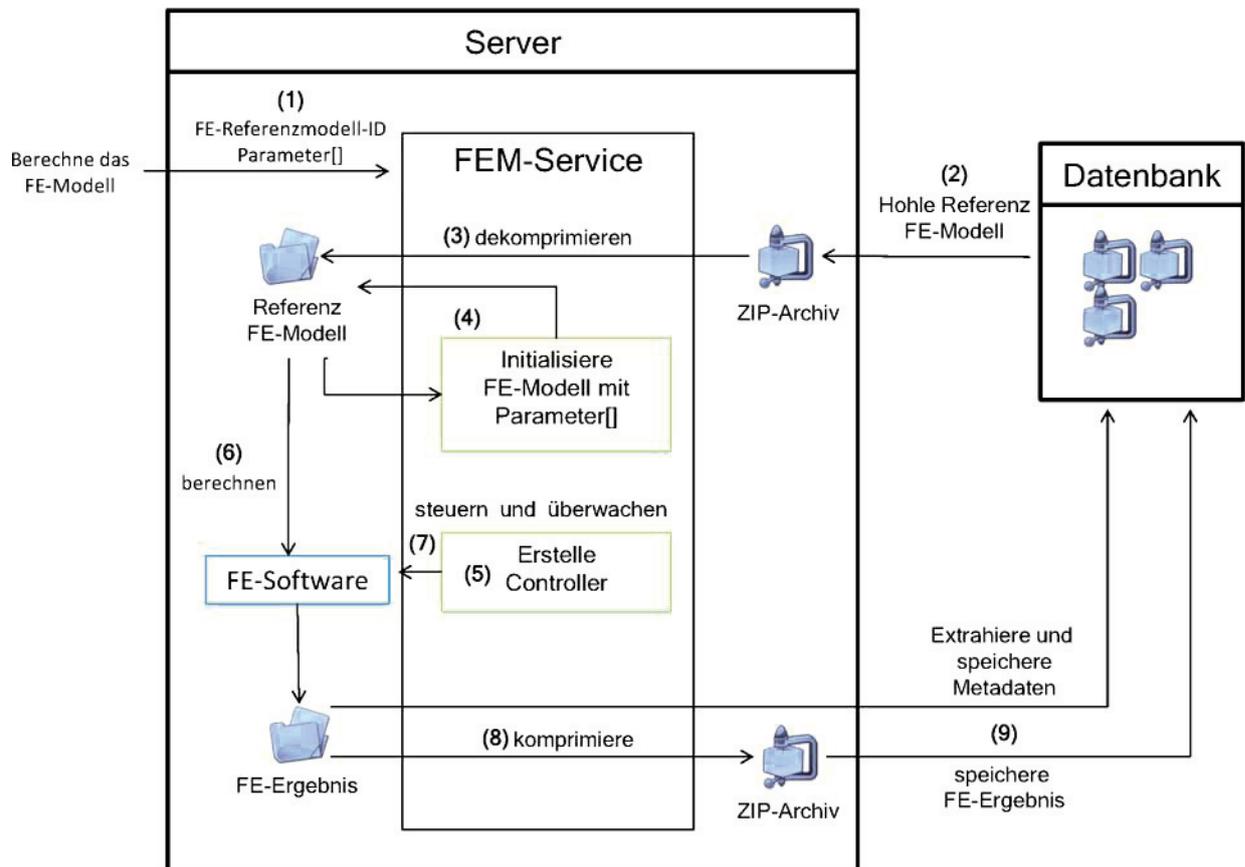


Bild 5: Berechnung eines FE-Modells auf dem FEM-Service

Nun werden die in dem FE-Referenzmodell annotierten Variablen mit neuen Werten aus der Nutzeranfrage belegt (4). Im Fall eines SOFiSTiK FE-Referenzmodells geschieht die Initialisierung der annotierten Variablen durch das Ersetzen der Textblöcke, die die Werte der Variablen in dem FE-Modell festlegen (Bild 4).

Nach dem Erstellen eines neuen FE-Modells wird ein Controllerobjekt für die Überwachung und Steuerung des Betriebssystemprozesses der SOFiSTiK-FEM-Software erstellt (5) und gestartet (6). Während der Berechnung wird die Konsolenausgabe der SOFiSTiK-FEM-Software überwacht und in einer Logdatei in dem FE-Ergebnis Verzeichnis gespeichert (7).

Nach Beendigung der Berechnung wird das FE-Ergebnis Verzeichnis zu dem ZIP-Archiv komprimiert (8) und zusammen mit aus dem FE-Ergebnis extrahierten Metadaten (s. 3.2 Datenselektionsmethode) in der Datenbank gespeichert (9).

Da die FE-Ergebnisdateien sehr groß werden können (bis zu ca. 5GB), ist es wichtig dass die Datenbank sehr große Dateien als Blob-Objekte abspeichern kann. Aus diesem Grund wird hier die HSQLDB-Datenbank verwendet[11], die das Abspeichern von bis zu 5TB großen Dateien erlaubt.

3.2 Datenselektionsmethode

Bei der Berechnung der FE-Modelle können große Datenmengen entstehen, die letztendlich über das Netz über-

tragen werden müssen und nicht mehr auf einem gewöhnlichen Server gespeichert werden können. Deswegen wurde eine Datenselektionsmethode entwickelt und implementiert, die nur die für die Systemidentifikation benötigten Daten aus berechneten FE-Ergebnissen selektiert und in der Datenbank abspeichert.

Zur Priorisierung der FE-Ergebnisse während der Systemidentifikation sind nur die Daten an vordefinierten Messpunkten von Bedeutung. Der Ersteller des FE-Modells kann die für die Selektion vorgesehenen Daten an bestimmten Messpunkten zur Abspeicherung in einer Ergebnisdatei (*.erg) unter Verwendung einer hierfür entwickelten Nomenklatur definieren.

In Bild 6 ist ein Ausschnitt aus einer *.erg Ergebnisdatei, die aus einer FE-Simulation entstanden ist dargestellt. Die fettmarkierten Zeilen veranschaulichen die für das Priorisieren der FE-Ergebnisse wichtigen Werte.

Damit die Ergebniswerte in der *.erg Datei lokalisiert werden können, muss deren Name mit dem Präfix "GTC_" anfangen. Der Rest des Namens besteht aus einzelnen Textblöcken, die durch Unterstrich voneinander getrennt sind. Die Variable GTC_MP1_LF3_UX enthält z.B. drei Textblöcke: MP1, LF3, UX. Jeder Block hat eine Bedeutung, die für die Priorisierung der Ergebnisse bzw zur semantischen Suche im Rahmen des fallbasierten Schließens verwendet werden kann. Außerdem kann jeder Textblock als eine Facette betrachtet und für die facettenbasierte Suche der FE-Ergebnisse verwendet werden. Die oben beschriebenen Textblöcke haben folgende

Bedeutung: MP1 = Messpunkt1, LF3 = Lastfall3, UX = Verschiebung in X Richtung. Die Textblöcke können durchaus einen längeren Text enthalten. Der Ergebniswert "GTC_MP1_LF3_UX" könnte beispielsweise auch mit "GTC_Messpunkt-1_Lastfall-3_Verschiebung-X-Axis" bezeichnet werden und somit folgende Textblöcke enthalten: Messpunkt-1, Lastfall-3, Verschiebung-X-Axis.

```

...
$ Messspunkt 1 ist z.B. ....
GTC_MP1_LF3_UX = 1.5611
GTC_MP1_LF3_UY = -1.2676
$ Messspunkt 2 wertet z.B. Spannungen ....
GTC_MP2_LF3_sigx = -33.6455
GTC_MP2_LF3_sigy = -82.2733
...

```

Bild 6: Selektierte Ergebniswerte

Da die Daten zusätzlich zur Ergebnisdatei auch in einer relationalen Datenbank gespeichert werden, ist der direkte Zugriff auf diese Simulationsdaten und damit der direkte Vergleich mit Messwerten möglich. Die FE-Simulation ist damit in den Bau- und Auswertezyklus eingebunden und die technischen Grundlagen für die simulationsbasierte Systemidentifikation sind gelegt. Es verbleibt noch das Problem der Einbindung von Rechenressourcen zur Behandlung des trivialparallelen Problems.

4 Erforderliche Rechenressourcen

Es entstehen durch Kombination von

- b Bauzuständen (=Bauphasen),
 - k geometrischen Modellansätzen,
 - n Stoffgesetzen,
 - m Parametervariationen (m Diskretisierungen des Parameterintervalls) über eine Anzahl von
 - p Parametern pro Stoffgesetz,
- eine Anzahl möglicher Modellkandidaten, die berechnet werden müssen. Die Anzahl ergibt sich aus der Tatsache, dass b, k, n als Skalarprodukt, und m, p als Vektorprodukt (Kreuzprodukt) miteinander kombiniert werden. Als Endergebnis ist eine begrenzte Auswahl von Modellen aus diesen vielen Modellkandidaten auszuwählen, mit dem wiederum $e * b$ Kombinationen von
- e Entwurfsvarianten für
 - b Bauzustände (Bauphasen)

berechnet werden, um auf dieser Basis letztlich zu entscheiden, ob die bisher zur Bauausführung gewählte Entwurfsvariante beibehalten, oder ob eine andere Variante gewählt werden soll. Die Berechnung von Grundwasserströmungen ist hierbei noch nicht berücksichtigt und kommt bei Bedarf noch hinzu.

Um eine Größenordnung zu nennen, sei folgendes Zahlenbeispiel gegeben: Geht man von 4 bereits ausgeführten Bauzuständen, 3 geometrischen Modellansätzen, 3 Stoffgesetzen und 9 Parameter mit jeweils nur 3 Parameterdiskretisierungen aus, so ergeben sich für die Systemidentifikation (Regelkreis 2 in Bild 1) $4 \times 3 \times 3 \times 39 = 708588$

nichtlineare FE-Berechnungen. Geht man weiterhin davon aus, dass 5 Entwurfsvarianten für 5 noch auszuführende Bauzustände mit einer Auswahl von 10 Modellen untersucht werden, sind für die Prognoseberechnung des zukünftigen Bauablaufs weitere 250 nichtlineare FE-Berechnungen erforderlich, um über Änderungen des Bauverfahrens (Regelkreis 3b in Bild 1) zu entscheiden.

Bei einer 2D-FEM-Berechnung einer Baugrube kann man von einer Rechenzeit von 5 bis 10 Minuten pro Bauzustand ausgehen. Eine Systemidentifikation mit sequenzieller Simulation der Modellkandidaten würde bei Annahme der unteren Grenze also $708.588 \times 5 = \text{ca. } 3,5$ Mio. Minuten, das sind 59.049 Rechenstunden (bzw. 6,74 Jahre), in Anspruch nehmen. Durch die Verteilung der Rechenlast über Cloud oder Grid Computing könnte der Rechenaufwand bei der Simulation der erforderlichen hohen Anzahl an Modellkandidaten durch die Parallelisierung der Berechnungen beherrschbar gemacht werden. Es ist hohes Potenzial zur Parallelisierung vorhanden, da zumindest alle Modelle, die die gleiche Bauphase beschreiben, parallel durchgerechnet werden können, d. h. die oben enthaltenen $k * n * m^p$ Modellkandidaten könnten parallel berechnet werden. Damit sind also die für jeden der 3 Bauzustände zu untersuchenden $4 \times 3 \times 39 = 236196$ Modellkandidaten parallel simulierbar. Bei paralleler Ausführung der Webserviceinstanzen mit Cloud- oder Grid-Computing und der Annahme optimaler Skalierung ergibt sich damit eine effektive Rechenzeit von insgesamt 4 Bauzuständen \times 5 Minuten = 20 Minuten. Dieses Szenario ist derzeit in der Praxis zwar noch schwer vorstellbar, jedoch keineswegs unrealistisch, wenn man als Vergleich z.B. die verteilte Grid Middleware Plattform BOINC (Berkeley Open Infrastructure for Network Computing) heranzieht. Im Jahr 2010 waren durchschnittlich 550000 Computer verfügbar, die im 24-Stunden-Durchschnitt über 5 PetaFLOPS ($=5 * 10^{15}$ Gleitkommaoperationen pro Sekunde) ermöglichen[2]. Die Tendenz der Beteiligung an solchen Rechnernetzwerken ist steigend. Ein Intel Core i7 Prozessor mit 3,2 GHz erlaubt 51,2 GigaFlops. Damit sind im BOINC-Netzwerk pro Sekunde ca. 100000 mal so viele Gleitkommaoperationen möglich, wie auf einem derzeit handelsüblichen Arbeitsplatzrechner. Da die Hard- und Softwareindustrie das Geschäftsfeld „Rechenressourcen“ für sich entdeckt hat (Amazon, Microsoft, Google), kann man also davon ausgehen, dass in absehbarer Zeit die Verfügbarkeit der Rechenleistung kein größeres Problem mehr darstellen wird. Nicht zu vernachlässigen ist allerdings die Kosten Seite. Derzeit kostet eine Recheninstanz mit 1 Rechenkern in Cloud-Computing-Umgebungen i. M. ca. 0,12 US-\$. pro Rechenstunde. Bei der oben ermittelten Gesamt-Rechenzeit (ohne Parallelisierung) von 59049 Stunden entstehen damit Kosten von insgesamt 7085 US-\$ für die Simulation der 708588 Modellkandidaten. Es handelt sich hierbei um derzeitige Kosten, die in Zukunft kontinuierlich fallen werden.

5 Zusammenfassung

Durch die Realisierung von Finite-Elemente-Software als Webservice wurde eine wesentliche Komponente zur si-

mulationsbasierten Systemidentifikation umgesetzt. Die FE-Simulation kann damit in einen durchgängigen Bau- und Monitoringzyklus eingebettet werden. Die erforderliche Rechenleistung zur Simulation einer hohen Anzahl an Modellkandidaten wird durch die Nutzung von Grid und/oder Cloud Computing erwartet. Entsprechende Konzepte zur Anbindung verteilter Rechenressourcen werden demnächst anvisiert. Da der FEM-Service selbst derzeit keine Sicherheitskonzepte implementiert, ist auch hierfür die Nutzung bestehender Grid-/Cloud-Middleware interessant. Die prinzipielle Möglichkeit zur Verwendung einer Middleware für das Anbinden eines Grid oder Cloud Rechnernetzwerks wie Amazon EC2[1] oder UNICORE[15] wurde bei der Umsetzung der Softwarearchitektur berücksichtigt.

Die flexible Architektur des FEM-Services erlaubt ferner die Portierung beliebiger FE-Software, die im Batchmodus ausgeführt werden kann und eine automatische Initialisierung der FE-Referenzmodelle erlaubt. Die Anpassung des FEM-Services an andere Fachdomäne und ein Einsatz, beispielsweise im Maschinenbau oder in der Elektrotechnik, sind aufgrund der flexiblen Softwarearchitektur des FEM-Services mit vertretbarem Aufwand möglich. Damit ist künftig auch die analoge Übertragung sowohl der Gesamtmethodik als auch der Softwarekomponenten auf eine Vielzahl möglicher Anwendungsfälle denkbar.

- [7] WinTUBE, FIDES DV-Partner GmbH Berlin München.
<http://www.fides-dvp.de/statik-hersteller/fides/tunnelbau/wintube>
- [8] Gamma, E.; Helm, R.; Johnson, R.: Entwurfsmuster . Elemente wiederverwendbarer objektorientierter Software. Addison-Wesley, 15. März 2001; ISBN-13: 2001. 978-3827318626.
- [9] GeoTechControl - Wissensbasierte Serviceplattform für Überwachung und Prognose Geotechnischer Ingenieurbauwerke.
<http://www.geotechcontrol.de>
- [10] Globus Toolkit: Build The Grid.
<http://www.globus.org>
- [11] HSQLDB - 100% Java Database.
<http://hsqldb.org>
- [12] Huang, Y.; Kintala, C.; Kolettis, N.; Fulton, D.: Software rejuvenation: analysis, module and applications. Proc 25th International Symposium on Fault-Tolerant Computing. June 1995.
- [13] Piedad, F; Hawkins, M.: High Availability - Design, Techniques and Processes. Prentice Hall, 24 September 1999; ISBN-13: 978-0130962881
- [14] Sofistik AG.
<http://www.sofistik.de>
- [15] UNICORE-Uniform Interface to Computing Resources.
<http://www.unicore.eu>
- [16] Wyld, D. C.; Wozniak, M.; Chaki, N; Meghanathan, N.; Nagamalai, D.: Trends in Networks and Communications. International Conferences NeCoM, WeST, WiMoN; Chennai, India, 15.-17. Juli 2011. ISBN 978-3-642-22542-0.

6 Literaturverzeichnis

- [1] Amazon Elastic Compute Cloud (Amazon EC2)
<http://aws.amazon.com/de/ec2>
- [2] Deventer, C.; Clarke, W. A.; Hazelhurst, S.:BOINC and CUDA:Distibuted High-Performance Computing for Bioinformatics String Matching Problems. Proc. Of Southern Africa Telecommunication Networks and Applications Conference (SATNAC). Stellenbosch, Südafrika, 2010.
- [3] Erl, T.: SOA Design Patterns; Pearson Education, Boston, 31. Dezember 2008; ISBN-13: 978-0-13-613516-6
- [4] Faschingbauer, G.; Scherer R. J.: Integrated Product- and Process Model for Online Prediction and Monitoring of Geotechnical Structures, Proceedings of the 16th EG-ICE International Workshop, Berlin, 2009.
- [5] Faschingbauer, G.; Scherer R. J.: An information management system for monitoring of geotechnical engineering structures. ECPPM 2008 – eWork and eBusiness in Architecture, Engineering and Construction – Proc. of the 7th European Conference on Product and Process Modelling (ECPPM). Netherlands, September 2008; ISBN 978-0-415-48245-5
- [6] Faschingbauer, G.; Scherer R. J.: Semi-Automation of logical decisions in monitoring procedures to realize adaptive geotechnical construction methods. Proc. 13th International Conference on Computing in Civil and Building Engineering (ICCCBE 2010) and 17th EG-ICE Workshop, Nottingham, UK., 30 June - 2 July 2010