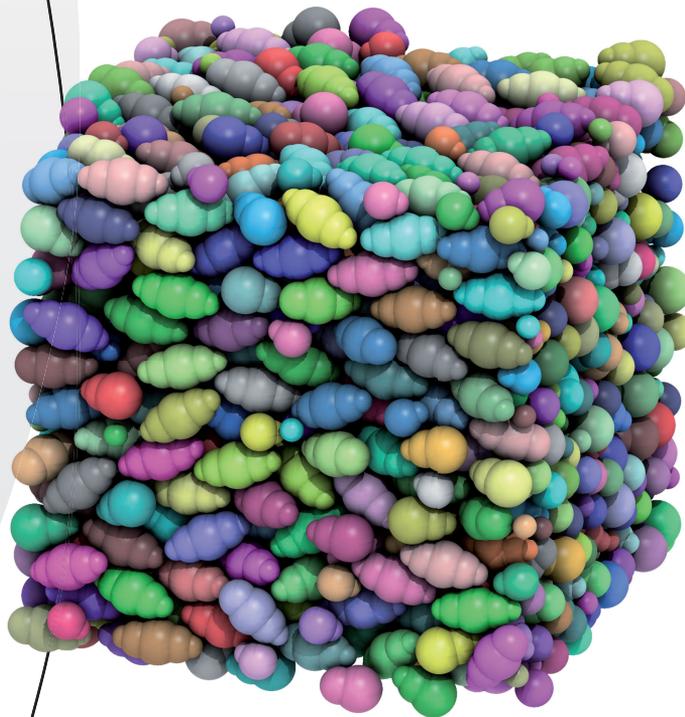




TU Clausthal

Simulationswissenschaftliches Zentrum der Technischen Universität Clausthal

Bericht 2009-2010



Postanschrift

Simulationswissenschaftliches Zentrum
Technische Universität Clausthal
Prof. Dr. Thomas Hanschke
Institut für Angewandte Stochastik und Operations Research
Erzstr. 1
38678 Clausthal-Zellerfeld
Telefon: (05323) 72-2401/-2567
Fax: (05323) 72-2304

Impressum

Herausgeber: Simulationswissenschaftliches Zentrum
der Technischen Universität Clausthal
Redaktion: Prof. Dr. Gunther Brenner, Prof. Dr. Thomas Hanschke,
Dr. Alexander Herzog
Layout und Satz: Melanie Bruchmann
Bildnachweis: S. 4, 8 (Olaf Möldner), S. 40 (Volkswagen),
S. 100 (Steffen Ottow), S. 130 (Mirco Schindler)
Hier nicht erwähnte Bilder stammen von den Autoren.
Druck: Creaktiv GmbH & Co KG, Goslar

ISBN: 978-3-942216-73-9

**Simulationswissenschaftliches Zentrum
der Technischen Universität Clausthal**

Bericht 2009–2010



Vorwort

Der vorliegende Berichtsband 2009-2010 des Simulationswissenschaftlichen Zentrums (SWZ) der TU Clausthal ist inzwischen der dritte dieser Art, der in Folge seit 2005 erscheint. Wiederum wird die Forschungstätigkeit in den beteiligten Instituten im Bereich der „Simulation“ dokumentiert. Damit wird zugleich die Aufbauarbeit zur Etablierung des SWZ als dritte Säule der hochschulübergreifenden Forschung, neben dem „Energieforschungszentrum Niedersachsen“ (EFZN) und dem „Clausthaler Zentrum für Materialtechnik“ (CZM) dokumentiert. Denn das SWZ soll in Zukunft als gemeinsame Einrichtung der Universität Göttingen und der TU Clausthal weitergeführt und ausgebaut werden. Die hierfür notwendigen Schritte wurden bereits 2008 begonnen und die wissenschaftlichen Ziele in Form eines 4.6 Millionen Euro schweren Antragspakets formuliert. Dieses wurde von der Wissenschaftlichen Kommission Niedersachsen (WKN) positiv begutachtet.

Mit der Kooperation zwischen der Universität Göttingen und der TU Clausthal wird das wissenschaftliche Spektrum der Arbeiten erheblich erweitert werden. Die bisher in Clausthal etablierten Themen werden in den drei Projektbereichen aufgehen, welche die Forschungsthemen des SWZ in den kommenden Jahren bestimmen werden:

- Simulation und Optimierung von Netzen
- Simulation von Materialien
- Verteilte Simulation

Eine kleinere Anzahl von Projektgruppen wird kurzfristig die Arbeit aufnehmen, weitere werden im Laufe der Zeit in einem dynamischen Auswahlprozess folgen. Das SWZ wird dabei als Katalysator verstanden, der es ermöglicht, interdisziplinär arbeitende Arbeitsgruppen zusammenzubringen und nachhaltige Forschungsprojekte, etwa DFG Sonderforschungsprogramme, Forschergruppen oder EU Projekte, zu initiieren.

Aber auch die bisherigen Arbeiten im SWZ haben bereits nachhaltige und nach außen sichtbare Spuren hinterlassen. Im BMBF-Foresight-Prozess werden die TU Clausthal mit ihren interdisziplinären wissenschaftlichen Einrichtungen und der Verbund mit Göttingen im Zukunftsfeld „Transdisziplinäre Modelle und Multiskalensimulation“ neben der Max-

Planck-Gesellschaft und den Universitäten Bonn, Stuttgart und Bochum zu den wesentlichen und aktivsten Institutionen gezählt (s. <http://www.bmbf.de/de/12673.php>). Weiterhin wurden erste Drittmittelprojekte im Rahmen des SWZ erworben und werden unter dessen Flagge bearbeitet.

Bereits jetzt wird absehbar, dass das SWZ in den mit der Niedersächsischen Technischen Hochschule (NTH) verbundenen Universitäten in Braunschweig und Hannover eingebunden werden wird. Als erster Erfolg kann die Teilnahme an der zweiten Phase des Auswahlprozesses zur Exzellenzinitiative des Bundes gewertet werden. Die hier geplante Graduiertenschule wird eine zentrale Rolle im Bereich der Ausbildung von Studierenden, aber auch der Entwicklung und Anwendung von Methoden zur numerischen Simulation von Mehrskalprozessen in den Ingenieurwissenschaften einnehmen.

Der vorliegende Bericht 2009/2010 orientiert sich wie in den vergangenen Jahren an der bisherigen Struktur des SWZ, in dem die Themen Materialien, Produktion, Logistik und Verkehr, Energie und Umwelt neben den Querschnittsthemen Hochleistungsrechnen und Software Engineering im Vordergrund standen. Die Anzahl der Themen wurde allerdings zu Gunsten einer ausführlicheren Darstellung einzelner Projekte etwas reduziert. Der Anfang ist gemacht. Jetzt geht es darum, die Aktivitäten der Gruppen gezielt auszubauen.

Prof. Dr. Thomas Hanschke
*Vorstand und Sprecher des
Simulationswissenschaftlichen Zentrums*

Inhaltsverzeichnis

1	Materialien.....	9
1.1	RaSim: Bestimmung der Raumausfüllung von Partikelmischungen unter Einbeziehung von Agglomeration	10
	<i>Michael Kolonko, Steffen Raschdorf, Stephan Mock</i>	
1.2	Magnetohydrodynamische Strömungen	21
	<i>Karl-Heinz Spitzer, Gunther Brenner, Yong Ma</i>	
2	Produktion, Logistik und Verkehr	41
2.1	Virtuelle Fabrikstruktur- und Layoutplanung mit erweiterten digitalen Modellen	42
	<i>Uwe Bracht, Christian Schlange</i>	
3	Energie und Umwelt.....	55
3.1	Long-term behaviour of sealed brine-filled cavities in rock salt mass – A new approach for physical modelling and numerical simulation	56
	<i>Karl-Heinz Lux</i>	
3.2	Ecological evaluation of the pulverized coal combustion in HTAC technology	80
	<i>Natalia Schaffel-Mancini</i>	
3.3	Optimierung von HT-PEM-Brennstoffzellen mittels numerischer Strömungssimulation	90
	<i>Volker Wesling, Peter Giese, Thorsten Kandelhardt</i>	
4	Hochleistungsrechnen.....	101
4.1	Analyse von dynamischen Lasten in hochbelasteten Strömungsmaschinen – Notwendigkeit der Modellvalidierung und des Hochleistungsrechnens	102
	<i>Gunther Brenner, Yang Yang, Andreas Lucius</i>	
4.2	Numerical simulation of liquid-liquid and liquid-particle flows by the Lattice Boltzmann Method	111
	<i>Ernesto Monaco, Gunther Brenner</i>	

4.3 Verfahren höherer Ordnung in Raum und Zeit
mit Anwendung auf Gleitlagerberechnungen 122
*Stefan Hartmann, Ahmad-Wahadj Hamkar,
Torben Netz*

**5 Visualisierung, Datenanalyse und
Software Engineering..... 131**

5.1 MASSim (Multi-Agent Systems Simulation)
Plattform..... 132
*Tristan Behrens, Jürgen Dix, Michael Köster,
Federico Schlesinger*

5.2 Verification of Resource-Bounded Agents 137
Nils Bulling, Jürgen Dix

5.3 Environment Interface Standard 142
Jürgen Dix, Tristan Behrens

5.4 JREP: Extending Repast Symphony for
JADE Agent Behavior Components 149
*Jana Görmer, Gianina Homoceanu,
Christopher Mumme, Jörg P. Müller, Michaela Huhn*

5.5 Simulation und Realität bei der Chipsynthese
mit VHDL am Beispiel von endlichen und
unendlichen Zählern 165
Harald Richter

**Mitglieder des
Simulationswissenschaftlichen Zentrums 186**



1 Materialien

Unter dem Begriff Materialien werden in der Technik üblicherweise alle Stoffe verstanden, die zur Produktion oder Herstellung von bestimmten Zwischen- oder Endprodukten verwendet werden. Das Thema umfasst eine unüberschaubare Anzahl an Aspekten und Teilgebieten, angefangen bei der Gewinnung und Aufbereitung der Rohstoffe, deren Veredelung und Verarbeitung zu Zwischenprodukten bis hin zur Entsorgung und Wiederaufbereitung. Wie kaum ein anderes ist das Forschungsgebiet Materialien geprägt von einer Vielzahl von Phänomenen und von einer signifikanten Disparität der Zeit- bzw. Längenskalen, die diese Phänomene charakterisieren. Deren systematische Untersuchung und Quantifizierung kann nur durch eine interdisziplinäre Zusammenarbeit unterschiedlicher Teilgebiete der Ingenieur- und Naturwissenschaften einschließlich der Mathematik erfolgen. Wie in vielen Bereichen der Natur- und Ingenieurwissenschaften nehmen auch hier mathematische und simulative Verfahren einen immer breiteren Raum ein. Sie werden eingesetzt zur Verbesserung von Materialeigenschaften, zur Optimierung von Prozessen, zur Effizienzsteigerung von Herstellungsverfahren und zur Kosteneinsparung durch Verminderung des erforderlichen experimentellen Aufwands.

In den letzten Berichten des Simulationswissenschaftlichen Zentrums wurden Ergebnisse verschiedener Arbeitsgruppen vorgestellt, beginnend bei den kleinsten atomaren Skalen bis hin zur Umformung von Blechen. Im vorliegenden Bericht erfolgt eine Fokussierung auf zwei Projekte, die Modellierung mit Multi-Skalen Methoden, bei denen das Verhalten eines Systems von Prozessen auf stark unterschiedlichen Längenskalen beeinflusst wird, sowie die Modellierung von Multi-Feld Problemen, die gekennzeichnet sind durch das Zusammenwirken verschiedenartiger Phänomene, die jeweils durch unterschiedliche Modelle, in diesem Falle partielle Differentialgleichungen, beschrieben werden. Beide werden im Rahmen von Drittmittelvorhaben durch externe Quellen (Dyckerhoff-Stiftung bzw. BMFT) finanziert und stellen erste Erfolge der systematischen Aufbauarbeit in den vergangenen Jahren dar.

1.1 RaSim: Bestimmung der Raumausfüllung von Partikelmischungen unter Einbeziehung von Agglomeration

*Michael Kolonko, Steffen Raschdorf, Stephan Mock –
Institut für Angewandte Stochastik und Operations Research*

Darstellung des Projekts

Granulare Medien sind in zahlreichen Forschungsgebieten und in der Industrie von hohem Interesse, sei es die Tablettierung in der Pharmazie, das Kollektivverhalten der Partikel in Gerölllawinen, oder die Herstellung leistungsfähiger Werkstoffe aus metallischen Rohstoffen, keramischen Pulvern und natürlichen Baustoffen in den Materialwissenschaften. So sind gerade die Bindemittelsysteme ein Beispiel für komplexe Zusammenhänge zwischen der Komposition der Ausgangsmaterialien und den Kenngrößen des Endprodukts [1]. Die vereinfachte Grundannahme lautet: Je dichter sich die Partikel in einer gemischten granularen Schüttung anordnen lassen, desto höher kann die Festigkeit des fertigen Baustoffs (Gips, Mörtel, Beton) getrieben werden.

Die Entwicklung solcher Mischungen ist bisher mit erheblichem experimentellen Aufwand verbunden, da verschiedene Mischungen hergestellt und im Labor auf ihre Raumausfüllung



Abbildung 1: Polydisperse Packung mit 1 Million Kugeln. (Alle Packungsbilder erstellt mit einer modifizierten Version von QuteMol [2].)

hin untersucht werden müssen. Daher ist man bestrebt, durch den Einsatz von Simulations- und Optimierungstechniken einen möglichst großen Teil dieser Arbeiten auf den Rechner zu verlagern. Gleichzeitig wird man dadurch in die Lage versetzt, einen wesentlichen größeren Raum möglicher Mischungen aus gegebenen Stoffen abzusuchen. Im Idealfall sollten nach Abschluss der Simulation und Optimierung lediglich wenige experimentelle Überprüfungen der besten gefundenen Mischungen erforderlich sein.

Stand der Forschung

Zur Bestimmung der Packungsdichte einer Partikelmischung mit bekannter Korngrößenverteilung existieren zwei grundsätzlich verschiedene Ansätze: Die ältere Methode der analytisch-empirischen Packmodelle betrachtet die Mischung als Kontinuum, um aus repräsentativer Korngröße sowie Eigenpackungsdichte einer jeden Komponente eine (rekursive) Formel für die Packungsdichte der Gesamtmischung herzuleiten. Durch sukzessive Hinzunahme von Komponenten können prinzipiell beliebig komplexe Mischungen untersucht werden, wobei die Interaktion zwischen Partikeln unterschiedlicher Kornfraktionen durch zuvor empirisch bestimmte Variablen abgebildet wird. Diese Faktoren sind für jedes Stoffsystem individuell zu bestimmen, sodass eine Einsparung bezüglich der experimentellen Laborarbeit nur dann erfolgen kann, wenn die Komponenten der Zielmischung feststehen und lediglich deren Anteilsverhältnisse an der Mischung variieren können.

Eine Alternative hierzu stellen die mit zunehmender Leistungsfähigkeit der Rechner populärer gewordenen Partikelsimulationen dar. Hier wird die Mischung nicht als Kontinuum betrachtet, sondern als Zusammenspiel einer Vielzahl von eigenständigen Objekten, den Partikeln. Die individuelle Bewegung jedes Objekts und seine Interaktion mit anderen Objekten lässt die Packung entstehen, ist aber auch der Grund dafür, dass die Laufzeit dieser Simulationsverfahren deutlich und überproportional von der Anzahl der verwendeten Partikel abhängt. Um diese zu beherrschen, wird daher meist auf die sonst in DEM-Simulationen übliche Berücksichtigung interpartikulärer Kräfte verzichtet und lediglich die geometrische Anordnung simuliert; eine weitere Vereinfachung ist die Approximation der Partikelform durch eine

Kugel. Die am häufigsten verwendeten Verfahren sind Random Sequential Addition (RSA) und Collective Rearrangement (CR). RSA-Algorithmen zeichnen sich durch eine sequentielle Abarbeitung der Kugeln aus. Eine einmal in der Packung platzierte Kugel kann später nicht mehr von ihrer Position weichen, was bedeutet, dass sich der Aufwand immer nur auf die jeweils aktuelle Kugel konzentriert. Eine weitere typische Eigenschaft dieser Methoden ist der sukzessive, konstruktive Aufbau der Packung, die sich zu jedem Zeitpunkt in einem validen, d.h. überlappungsfreien Zustand befindet.

Wird die Anordnung der Kugeln in jedem Iterationsschritt eines Algorithmus immer wieder (leicht) verändert, indem mehrere oder alle Kugeln verschoben werden, so handelt es sich um das CR-Verfahren. Im Gegensatz zum RSA-Ansatz kann hier jede Kugel bis zum Ende des Packungsprozesses bewegt werden, wodurch sich die Möglichkeit ergibt, auch zwischenzeitlich auftretende Lücken durch hineindrängende Kugeln wieder aufzufüllen. Diese hohe Dynamik erfordert jedoch einen deutlich größeren Rechenaufwand als bei RSA-Verfahren. Typisch für CR-Algorithmen ist außerdem die Relaxation bestimmter Anforderungen der Packung während des Packprozesses; so sind zunächst meist Überlappungen erlaubt, die dann sukzessive abgebaut werden.

Eine der größten Herausforderungen auf dem betrachteten Gebiet der Baustofftechnologie stellt die enorme Polydispersität der Partikel dar, die sich über etliche Größenordnungen vom Submikrometerbereich feinsten Zusatzstoffe bis zu zentimetergroßen Kiespartikeln erstreckt. Bereits analytisch-empirische Modelle büßen bei Betrachtung eines solch breiten Intervalls einiges an Genauigkeit ein, doch teilchenbasierte Simulationen stehen vor dem noch gewaltigeren Problem, eine für die größten Partikel ausreichende Simulationsdomäne bereitzustellen, die es dicht aufzufüllen gilt, während gleichzeitig die vorgegebenen Anteile der unterschiedlichen Kornfraktionen an der Gesamtmischung gewahrt bleiben müssen.

Forschungsbeitrag

Im Rahmen der Dissertation Raschdorf [4] wurde das Simulationssystem RaSim zur Simulation der Raumauffüllung von Par-

tikelmischungen entwickelt. Das System wird gegenwärtig in einer Anschlussarbeit um einige wichtige Komponenten erweitert. Im Folgenden werden einige der wesentlichen Aspekte des Systems RaSim geschildert.

Sampling + Hierarchisierung der Fraktionen

Eine geschickte Einteilung des gesamten Korngrößenintervalls einer Mischung in separate, unabhängig voneinander zu simulierende Intervalle, sog. Fraktionen, macht die Komplexität der Simulation beherrschbar. So bilden die Partikel der feinsten Fraktion für sich genommen eine dichte Packung, die wiederum als „Füllmasse“ für die Lücken zwischen den Partikeln der nächstgrößeren Fraktion dient. Um fraktionsübergreifende Modellierung zu unterstützen, wird im Simulationslauf jeder Fraktion eine zusätzliche Selektion größerer Partikel mitgeführt. Diese großen Partikel wirken sich als eine Art innerer Wandbereich auf die Dichte der Packung aus, indem die Raumauffüllung an diesen Rändern abgesenkt wird.

Die simulierten Packungsdichten auf der Ebene der einzelnen Fraktionen werden am Ende zu einer aggregierten Raumauffüllung der Gesamtmischung vereinigt. Der iterative Prozess eignet sich zur Umsetzung auf parallelen Rechnerarchitekturen; darüber hinaus kann durch Prüfen von Abbruchbedingungen ein schnellerer Simulationserfolg erzielt werden, ohne sämtliche Fraktionen explizit simulieren zu müssen.

Die Auswahl der in einer Fraktion zu verwendenden Partikeldurchmesser kann durch verschiedenste Samplingalgorithmen umgesetzt werden, solange die Repräsentativität der Stichprobe sichergestellt wird. Deren Kennlinie sollte möglichst stark dem Ausschnitt der Original-Korngrößenverteilung ähneln. Nach diversen Tests hat sich ein deterministisches Samplingverfahren als schnell und von ausreichender Güte erwiesen.

CR-Simulation mit dynamischer Containergröße

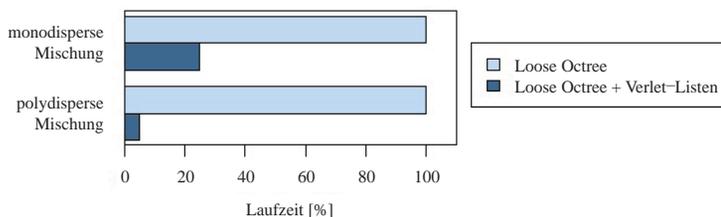
Die Simulation einer Partikelpackung auf Fraktionsebene erfolgt auf Basis eines CR-Algorithmus [3], bei dem die Kugeln initial in einem kleinen Container überlappend platziert werden. Bereits während der Platzierungsphase wird auf eine gleich-

mäßige Verteilung der Kugelüberlappungen geachtet, was die folgenden Iterationsschritte deutlich beschleunigt. Diese entzerren sukzessiv alle Überlappungen und lassen die Kugeln die noch freien Lücken im Container explorieren. Gleichzeitig sorgt eine Ablaufsteuerung unter Berücksichtigung des aktuellen Fortschritts für eine schrittweise Vergrößerung des Containers. Hier ist ein sensibles Vorgehen gefragt, sorgen große Schritte doch für eine schnellere Konvergenz, während kleine Schritte hingegen eine dichtere Packung erzeugen können. Einen guten Kompromiss unter Beibehaltung einer adäquaten Schrittweite bietet die Option der erneuten Containerverkleinerung, sobald eine überlappungsfreie Packung erreicht wurde. So wird der CR-Prozess im Prinzip neu angestoßen, nur mit einer „vorverdichteten“ Kugelanordnung, die sehr gleichmäßige Überlappungen enthält. Diese Schleife kann abhängig vom Ressourceneinsatz beliebig oft wiederholt werden.

Loose Octree + Verlet-Listen

Als Flaschenhals der Simulation stellte sich die Suche nach überlappenden Kugeln heraus, weshalb viel Anstrengung in die Entwicklung effizienter Algorithmen und Datenstrukturen zur Nachbarschaftsbestimmung geflossen ist. Der Loose Octree erwies sich letztendlich als geeignete Wahl, um den Anforderungen extrem polydisperser Mischungen gerecht zu werden: Er partitioniert die Simulationsdomäne auf hierarchische Weise in Würfel unterschiedlicher Größe. Diese sind als Knoten in einer Baumstruktur organisiert, von denen jeder nur seinen Vater- sowie seine 8 Kindknoten kennt und zudem auf die jeweils enthaltenen Kugeln referenziert. Diese Hierarchie über mehrere Größenordnungen korrespondiert sehr gut mit der Polydispersität der betrachteten Partikelmischungen, was die deutlichen Geschwindigkeitsvorteile gegenüber der Benutzung regulärer Gitter (wie sie etwa in der Molekulardynamik verwendet werden) begründet.

Abbildung 2: Performance-Gewinn durch Benutzung von Verlet-Listen, betrachtet für eine mono- und eine polydisperse Mischung aus jeweils 10.000 Kugeln



Die Effizienz der Simulation ließ sich durch den Einsatz von Verlet-Listen noch weiter steigern. Diese wirken wie ein Cache und speichern die lokale Nachbarschaft einer jeden Kugel. Deren zeitliche und räumliche Kohärenz während der CR-Simulation ermöglichte es außerdem, ein bewegungsabhängiges Aktualisierungsintervall für die Verlet-Listen zu implementieren, so dass diese immer eine Obermenge der tatsächlich in Kontakt stehenden Kugeln enthalten.

Alternative Partikelformen

Während das Kugelmodell für einige wenige Stoffe (z. B. Flugasche) eine gute Approximation darstellt, so gibt es viele weitere Materialien, deren Partikel eine von der Kugelform mehr oder weniger deutlich abweichende Gestalt besitzen. Die Annäherung dieser Form erfolgt im Simulationsprogramm nach einem konstruktiven Ansatz, bei dem mehrere Kugeln zu einem Kugelcluster zusammengefasst werden und so gemeinsam ein Partikel bilden. Ein wichtiger Aspekt ist dabei die Fähigkeit zur Bildung beliebiger Partikelformen, zugleich mussten jedoch Konzepte wie Nachbarschaft und überlappungs-basierte Verschiebung der Cluster adaptiert werden.

Die Positionen der Kugeln innerhalb eines Partikels sind im entstandenen Simulationsprogramm starr, d. h. es sind keine relativen Bewegungen der Kugeln untereinander vorgesehen. Ebenso ist dadurch keine Deformation der Partikel möglich; ein durch solch einen Kugel-Cluster dargestelltes Partikel verhält sich damit analog zu harten Kugeln. Die Nachbarschaftsverwaltung lässt sich auf einfache Weise an diese Art der Modellierung beliebiger Partikelformen anpassen, sodass sie weiterhin mit den einzelnen Kugelprimitiven arbeiten kann. Es muss lediglich sichergestellt werden, dass sich Kugeln, die Bestandteil desselben Partikels sind, nicht gegenseitig als Nachbarn betrachten können.

Neben der Translation sind die zusammengesetzten Partikel auch der Rotation ausgesetzt. Hier wird jedes Partikel um seinen eigenen Mittelpunkt gedreht, wobei die Rotation ein Resultat der Drehmomente ist, denen das Partikel durch äußere Überlappungen seiner Kugeln mit anderen Partikeln ausgesetzt ist.

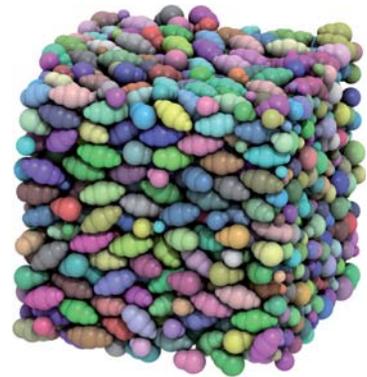


Abbildung 3: Periodische Randbedingungen: Ellipsoide gleichen ihre Achsenausrichtung einander an



Abbildung 4: Packung aus Würfeln, die sich an den Containerwänden ausrichten

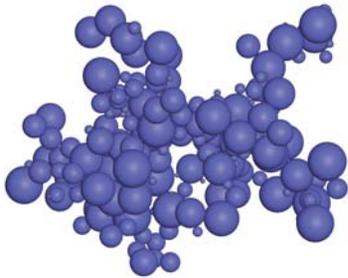


Abbildung 5: Ein mittels DLA-Algorithmus erzeugter Kugel-Cluster aus 200 Partikeln

Agglomeration

Die Agglomeration ist eine Zusammenfassung aller Prozesse, bei denen eine Verschiebung der Korngrößenverteilung von den feinen hin zu den größeren Partikelgrößen stattfindet. Ausgelöst wird dies durch interpartikuläre Kräfte, die zu einem Anhaften kleinerer Partikel und so zur Bildung größerer Strukturen führt. In der Verfahrenstechnik wird die Agglomeration unter anderem benutzt, um den Transport und die Lagerung von feinen granularen Stoffen zu erleichtern. Beispiele dafür sind das Tablettieren, Brikettieren oder Pelletieren. Bei der Entwicklung von Baustoffen ist die Bildung von Agglomeraten aber in der Regel unerwünscht, da sie der angestrebten hohen Raumaufüllung entgegenwirkt. Feine Partikel stehen nicht mehr als Füllmaterial für die Lücken zwischen den größeren Partikeln zur Verfügung, wenn sie zu Agglomeraten verbunden sind. Außerdem können Agglomerate aufgrund eines fraktalartigen Aufbaus eine sehr geringe innere Raumaufüllung haben, was auch die Raumaufüllung der gesamten Mischung herabsetzt. Um auch die Bestimmung der Raumaufüllung von Mischungen mit sehr feinen Korngrößenbereichen zu ermöglichen, muss das Simulationsprogramm daher die Auswirkungen der Agglomeration in geeigneter Weise berücksichtigen.

Eine naheliegende Methode ist die Einführung eines Faktors, der das ermittelte Simulationsergebnis, abhängig vom in der Mischung enthaltenen Feinkornanteil, korrigiert. Der verwendete Packungsalgorithmus muss dafür nicht verändert werden. Die Schwierigkeit liegt in der Ermittlung dieses Faktors, er kann durch Regression aus einer Messreihe für eine bestimmte Mischung errechnet werden. Da aber jeder Stoff anders agglomeriert, können die Ergebnisse nicht auf andere Mischungen übertragen werden.

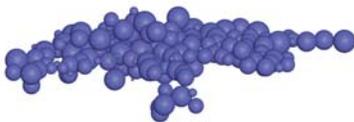


Abbildung 6: Ein mittels DLA-Algorithmus erzeugter, scheibenförmiger Kugel-Cluster

Eine andere Möglichkeit ist die Einbeziehung der Agglomeration in die Simulation. Die Verschiebung der Korngrößenverteilung kann in der Kugelpackung dadurch erreicht werden, dass kleinere, agglomerierende Kugeln gelöscht und durch größere Kugeln – die Agglomerate – ersetzt werden. Die geringere innere Raumaufüllung der Agglomerate kann entweder als zusätzliche Eigenschaft der Kugeln gespeichert

werden oder durch zusätzliche Kugeln, die als Hohlräume fungieren, dargestellt werden. Auch hier muss der Simulationsprozess selbst kaum verändert werden, es müssen lediglich unterschiedliche Arten von Kugeln verwaltet werden. Allerdings besteht auch hier das Problem, für einen gegebenen Stoff die Veränderung der Korngrößenverteilung genau zu bestimmen.

Eine weitere, realitätsnähere Möglichkeit ist die Simulation der Agglomerate durch Kugelcluster. Dies erfordert einen höheren Aufwand, ermöglicht aber den direkten Vergleich mit der Mikrostruktur einer realen Packung. Da das Simulationsprogramm bereits in der Lage ist, von der Kugelform abweichende Partikel als Kugel-Cluster zu simulieren, können auch Agglomerate auf diese Weise behandelt werden. Um Kugel-Cluster mit den erforderlichen Eigenschaften zu generieren, wurde eine auf einem DLA-Algorithmus (diffusion limited aggregation) [7] basierende Software entwickelt. Dieser Algorithmus baut sequentiell einen Cluster auf, indem eine erste sogenannte Saatkugel im Koordinatenursprung platziert wird. Weitere Kugeln werden auf einer, den Cluster in ausreichendem Abstand umgebenden Startsphäre zufällig positioniert. Von dort aus vollführen sie einen Zufallslauf, der einen Diffusionsprozess simuliert, bis sie den Cluster berühren. Auf diese Weise entstehen fraktale Kugel-Cluster mit einer sehr geringen inneren Raumauffüllung. Neben der Einstellung der Anzahl an Kugeln, bietet die Software auch vielfältige Möglichkeiten weitere Parameter zu beeinflussen. Durch eine Änderung der Zufallsverteilung bei der Positionierung der Kugeln auf der Startsphäre können z.B. Kugel-Cluster mit unterschiedlicher äußerer Form erzeugt werden (vgl. Abb. 7 und Abb. 8).

Ein Problem bei der Einfügung derartiger Agglomerate in das bestehende RaSim-System bildet gegenwärtig noch die Bewegung der Cluster während des Überlappungsabbaus. Die mit dem DLA-Algorithmus erzeugten fraktalartigen Cluster haben die Neigung, sich mit ihren 'Armen' zu verhaken. Hier wird an einem hierarchischen Konzept gearbeitet, das zunächst grobe Bewegungen ohne Berücksichtigung der Feinstruktur des Agglomerats erlaubt und dann eine genauere lokale Bewegung vornimmt.

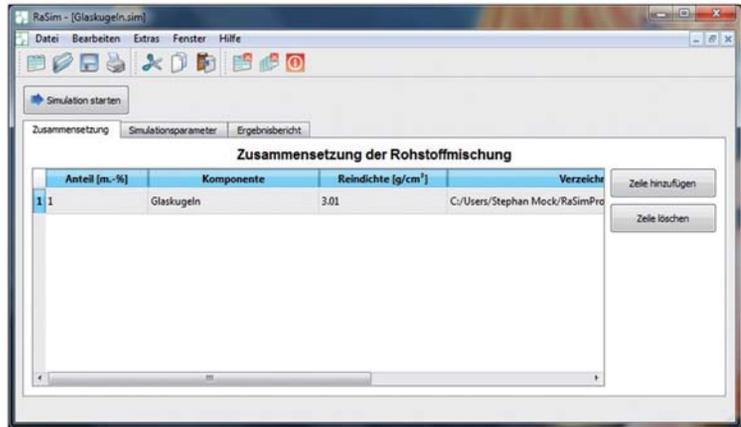


Abbildung 7: Grafische Benutzeroberfläche „RaSim“

Grafische Benutzeroberfläche

Im Rahmen der Weiterführung dieses Projektes wurde eine grafische Oberfläche für RaSim entwickelt. Dies stellt einen ersten Schritt auf dem Weg vom Simulationswerkzeug zum Anwendungsprogramm dar. Wie schon bei der Implementierung des Simulationsprogramms wurde auch hier auf Plattformunabhängigkeit geachtet. Um zukünftige Entwicklungen nicht zu erschweren, ist die grafische Oberfläche als separates Frontend zum eigentlichen Simulationsprogramm entwickelt worden, das weiterhin unabhängig gestartet werden kann. Über die grafische Oberfläche ist es möglich, alle für die Simulation relevanten Daten zu verwalten wie z.B. die Rohdaten der Ausgangsstoffe, die Parametereinstellungen für die Simulation und die Simulationsergebnisse. Ferner können Profile für verschiedene Personen oder Arbeitsbereiche angelegt werden und die Anzahl der benötigten Fraktionen können automatisch berechnet werden.

Über diese Programmversion ist es erstmals möglich, Simulationsläufe durchzuführen ohne genauere Kenntnisse des Simulationsprogramms. Das Programm befindet sich bereits im Institut für Nichtmetallische Werkstoffe im Einsatz.

Fazit und Ausblick

Von der Modellierung bis zur Implementierung wurde in der aus dem Projekt entstandenen Dissertation [4] auf eine univer-

selle Umsetzung geachtet, sodass über die Bindemittelkomponenten hinaus nicht nur das gesamte Korngrößenspektrum bis hin zum groben Zuschlag abgedeckt werden kann – was die Simulation von Packungen aus Zement, Gips, Mörtel und Beton einschließt –, sondern die Raumauffüllungen beliebiger granularer Medien bestimmt werden können.

Die Einbeziehung nicht-kugelförmiger Partikel und von Partikel-Agglomeraten ist im Prinzip bereits jetzt möglich, hier ist jedoch noch einiges an Modellierungs- und Anpassungsarbeit zu leisten, um realitätsnahe Simulationen beliebiger Korngrößenverteilungen zu ermöglichen.

Die schnelle und genaue Simulation einer gegebenen Korngrößenverteilung ist wiederum die Voraussetzung für die rechnergestützte Suche nach Partikelmischungen mit vorgegebener oder möglichst hoher Raumauffüllung. Diese Such- und Optimierungskomponente soll mit naturanalogen Verfahren wie genetischen Algorithmen arbeiten.

Beteiligte Partner

Dieses Projekt wird in enger Zusammenarbeit mit dem Institut für Nichtmetallische Werkstoffe (Prof. Dr. Wolter) und dem Institut für Informatik (Prof. Dr. G. Zachmann) an der TU Clausthal betrieben. Finanziert wird es durch mehrere Promotionsstipendien und weitere Fördermittel der Dyckerhoff-Stiftung (Projektnummer T218/15631/2006).

Partikelpackungen, die mit dem hier vorgestellten Simulationsprogramm erzeugt wurden, befinden sich auch am Institut für Technische Mechanik sowie am Institut für Polymerwerkstoffe und Kunststofftechnik als Ausgangssituation für weitergehende Simulationen (Strömungsmechanik, FEM) im Einsatz.

Literatur

- [1] S. Palm: Optimierung der Raumauffüllung und der Komponentenverteilung von Multikompositzementen, Dissertation TU Clausthal, 2009.
- [2] M. Tarini, P. Cignoni, C. Montani: Ambient Occlusion and Edge Cueing for Enhancing Real Time Molecular Visuali-



Abbildung 8: Tetris-artige Partikelformen demonstrieren die Möglichkeiten heteromorpher Packungen

- zation, IEEE Transactions on Visualization and Computer Graphics, 12(5):1237-1244, 2006.
- [3] D. He, N. N. Ekere, L. Cai: Computer Simulation of random packing of unequal particles, Physical Review E, 60(6):7098-7104, 1999.
 - [4] S. Raschdorf: Bestimmung der Raumauffüllung von Partikelmischungen – Modelle und Datenstrukturen für die Simulation durch Kugelpackungen, Dissertation TU Clausthal, 2010.
 - [5] M. Kolonko, S. Raschdorf, D. Wäsch: A hierarchical approach to simulate the packing density of particle mixtures on a computer, Granular Matter, 12(6):629-643, 2010. Web: <http://dx.doi.org/10.1007/s10035-010-0216-5>.
 - [6] S. Raschdorf, M. Kolonko: A comparison of data structures for the simulation of polydisperse particle packings, International Journal for Numerical Methods in Engineering, 85(5):625-639, 2011. Web: <http://dx.doi.org/10.1002/nme.2988>.
 - [7] P. Meakin: Progress in DLA research, Physica D, 86:104-112, 1995.

1.2 Magneto hydrodynamische Strömungen

Karl-Heinz Spitzer – Institut für Metallurgie

Gunther Brenner, Yong Ma – Institut für Technische Mechanik

Zusammenfassung

Das hier beschriebene Projekt ist ein Bestandteil der im Rahmen der SWZ angestrebten Schwerpunktbildung im Bereich der „Computational Materials Science“. Das Projekt wird derzeit aus anderen Drittmittelquellen (BMBF Vorhaben 033R007B) finanziert. Eine Förderung im Rahmen des SWZ ist zu einem späteren Zeitpunkt angestrebt.

Gegenstand des Projektes ist die numerische Berechnung von magneto hydrodynamischen (MHD) Strömungen mit freien Oberflächen. Hintergrund ist die Analyse und Optimierung von neuartigen Herstellungsverfahren für Stahlbleche, dem Bandgießverfahren. Dieses benötigt ein System zur gleichmäßigen Aufgabe der Schmelze auf ein umlaufendes Band. Hier kommt der Wechselwirkung zwischen der flüssigen Schmelze und einem externen elektromagnetischen Wechselfeld besondere Bedeutung zu, mit dessen Hilfe eine gleichmäßige Dicke über der gesamten Breite des Bandes gewährleistet werden soll. Aus Sicht der Modellierung und Simulation handelt es sich hierbei um ein klassisches Mehrfeldproblem, da neben den hydrodynamischen Modellgleichungen, d.h. den Navier-Stokes Gleichungen, die Maxwell-Gleichungen gekoppelt gelöst werden müssen. Hinzu kommt die Berücksichtigung von beweglichen Phasengrenzen und des komplexen Materialverhaltens der Schmelze. Ziel des Projektes ist die Weiterentwicklung von numerischen Berechnungsverfahren für derartige Mehrfeldprobleme. Diese Modelle sollen anhand einer Modellkonfiguration untersucht werden. Die Konfiguration besteht aus einer Gerinneströmung mit freiem Rand; Form der freien Oberfläche und somit die Filmdicke der Gerinneströmung hängen vom aufgetragten Magnetfeld ab.

Technologischer Hintergrund, Stand der Forschung

Aufgrund der großen technologischen und wirtschaftlichen Bedeutung von hochwertigem Feinblech, d.h. hochfesten Stahlblechen mit einer Dicke unter 3mm, wird kontinuierlich an der Weiterentwicklung von effizienten Herstellungsverfahren gearbeitet. Dies gilt insbesondere vor dem Hintergrund der Notwendigkeit der Steigerung der Energieeffizienz dieser Verfahren. Diesbezüglich ist das seit den 60er Jahren etablierte Verfahren, bei dem Brammen durch Warmwalzen und Kaltwalzen auf die gewünschte Enddicke gebracht werden, nicht optimal. Bei diesem CSC Verfahren wird der flüssige Stahl mit dem Verteiler aus Pfannen in Kokillen geführt und zu einem gekühlten Strangführungssystem weiter transportiert. Sobald der flüssige Stahl auf die gekühlte Kokillenwand und den Kaltstrang trifft, beginnt er zu erstarren. Je nach betrieblichen Möglichkeiten können die erzeugten Brammen noch warm in die Walzwerksöfen eingesetzt werden, um so Energie zu sparen. Meistens lässt man sie allerdings auf Kühlstrecken soweit abkühlen, dass sie mit Magneten gehandhabt werden können und stapelt sie dann ab. Dadurch geht die gesamte thermische Energie verloren und die Brammen müssen vor dem Walzen wieder aufgeheizt werden. Als Weiterentwicklung wurde in den späten 1980er bis 1990er Jahren das Dünnbrammengießverfahren (engl. Compact Strip Production, kurz CSP) eingeführt. Dabei werden Brammen von 250–300mm (beim CSC) Dicke auf 50–100 mm verringert und ohne Zwischenabkühlung direkt zu Warmband ausgewalzt. Die Vorteile dieses Verfahren sind nicht nur die Abnahme der Banddicke, sondern auch die Verringerung des Energieverbrauchs durch Vermeiden des Wiederaufheizens.

Die nächste Entwicklung in der Feinblechproduktion ist das Dünnbandgießen, auch Castrip-Prozess genannt, bei dem die hergestellten Metallbänder von nur noch 1–15 mm Dicke erstarren und nicht oder nur in wenigen Stichen warm umgeformt werden müssen. Bei diesem Verfahren wird der flüssige Stahl zwischen zwei gegeneinander liegenden wassergekühlten Walzen vollständig als Band erstarrt und schließlich durch den hohen Druck zwischen den Walzen auf eine Dicke bis 1.5 mm gebracht. Außer der dünneren Banddicke und des geringen Energieverbrauchs benötigt der Castrip-Prozess noch einen geringen Walzenaufwand und ist in relativ kleinen Men-

gen kostengünstig herstellbar. Der große Nachteil des Castrip-Prozesses liegt darin, dass die mechanischen Eigenschaften des erzeugten Feinblechs nicht durch das Kaltwalzen eingestellt werden können und dass die gegossene Struktur praktisch vollständig erhalten bleibt.

Das an der TU Clausthal entwickelte Verfahren, der Direct Strip Casting (DSC)-Prozess, wie er in Abbildung 1 dargestellt ist, vermeidet die Nachteile des Castrip-Verfahrens bei gleichzeitiger Reduktion des Investitions- und Energieaufwands gegenüber den Dünnbrammengießverfahren. Beim DSC-Verfahren wird die Schmelze direkt auf ein durch Wasser gekühltes Stahlblechband transportiert. Die seitliche Begrenzung besteht aus ebenfalls wassergekühlten Kupfersegmenten und läuft mit derselben Geschwindigkeit wie das Band mit. Um eine schnelle Erstarrung und gleichzeitige Reduktion des Kohlenstoffgehaltes der Stahloberfläche zu erzielen, werden eine wassergekühlte Abdeckung und ein aus Argon bestehendes Schutzgas eingesetzt. Durch das DSC-Verfahren wird eine übermäßige Porosität der Oberfläche vermieden und eventuelle Lunker bilden sich als Vakuumlunker nur im Kern des Bandes, wo sie durch das anschließende Warmwalzen wieder verschweißt werden können. Mit deutlich erhöhter Qualität bietet der DSC-Prozess mit ca. 15 mm Brammendicke nach einigen Warmwalzschritten eine Banddicke von etwa 8 mm an.

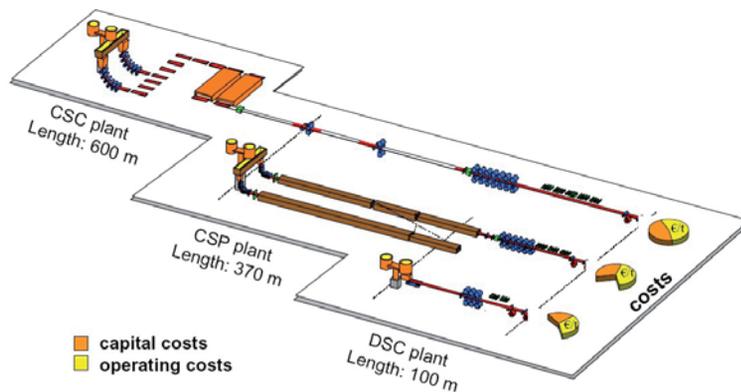
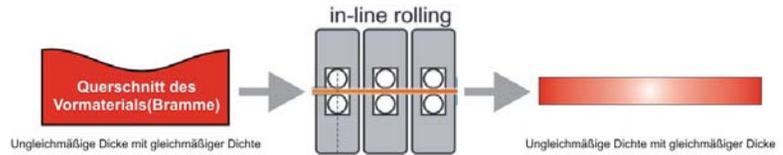


Abbildung 1: Schematische Darstellung des Platz- und Energiebedarfs von drei verschiedenen Verfahren zur Herstellung von Feinblech

Abbildung 2: Zur Veranschaulichung des Zusammenhangs zwischen Warmbanddicke und Dichte nach dem Walzvorgang



Der Energiebedarf kann demnach auf 24 % gegenüber dem CSC-Verfahren reduziert werden. Gleiches gilt für die Investitionskosten durch den Wegfall von Walzgerüsten und Hallenfläche.

Da der Umformgrad der erzeugten Feinblechprodukte bei gegebener Enddicke entscheidend von der Dicke des gegossenen Vormaterial abhängt, ist die gleichverteilte Dichte, bzw. gleichmäßige Dicke des Vormaterials, eine wichtige Einflussgröße für die mechanischen Eigenschaften des fertigen Warmbandes. Abbildung 2 veranschaulicht das Verhältnis zwischen Dichte- und Dicken-Verteilung nach dem Walzen. Aus diesem Grunde ist die Kontrolle der Schichtdicke des noch flüssigen Stahls von entscheidender Bedeutung, um Inhomogenitäten z.B. durch ungleichmäßig schnelles Abkühlen des Bandes zu vermeiden.

Damit die Schichtdicke des flüssigen Stahls in der Bandgießanlage gesteuert werden kann, werden Magnetfelder als eine implementierbare Maßnahme eingesetzt und erprobt. Wie in der Abbildung 3 dargestellt wird durch das Magnetfeld und die von diesem induzierte Stromdichte eine Lorentzkraft eingeführt, welche das Geschwindigkeitsprofil und die Schichtdicke des noch flüssigen Stahls beeinflusst (magnetisches Rühren). Am Institut für Metallurgie der TU Clausthal werden im Technikumsmaßstab experimentelle Untersuchungen zur Optimierung des Prozesses realisiert, beispielsweise durch Anordnung oder Stärke der Magnetfelder.

Eine exakte Quantifizierung der Wirkung der Magnetfelder ist de facto nicht möglich, da dies die Messung von Geschwindigkeiten im Inneren der Bandgießanlage erfordern würde. An dieser Stelle sollen daher numerische Berechnungsverfahren eingesetzt werden, in denen diese Phänomene abgebildet werden und auf deren Basis Parametervariationen zur Optimierung durchgeführt werden können.

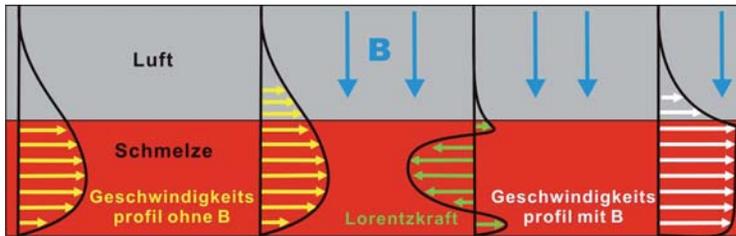


Abbildung 3: Veranschaulichung des Zusammenwirkens von Geschwindigkeitsfeld und Magnetfeld

Der Gegenstand des Projektes ist demnach die numerische Berechnung von magnetohydrodynamischen (MHD) Strömungen mit freien Oberflächen. Aus Sicht der Modellierung und Simulation handelt es sich hierbei um ein klassisches Mehrfeldproblem, da neben den hydrodynamischen Modellgleichungen, d.h. den Navier-Stokes-Gleichungen, die Maxwell-Gleichungen gekoppelt gelöst werden müssen. Hinzu kommen die Berücksichtigung von beweglichen Phasengrenzen und das komplexe Materialverhalten der erstarrenden Schmelze. Ziel des Projektes ist die Weiterentwicklung von numerischen Berechnungsverfahren für derartige Mehrfeldprobleme. Diese Modelle sollen anhand einer Modellkonfiguration untersucht werden. Die Konfiguration besteht aus einer Gerinneströmung mit freiem Rand; Form der freien Oberfläche und somit die Filmdicke der Gerinneströmung hängen vom aufgeprägten Magnetfeld ab.

Die grundlegenden Modellgleichungen zur Beschreibung der Bewegung eines Fluides (Gas oder Flüssigkeit) in Wechselwirkung mit elektrischen und magnetischen Feldern basieren auf den Navier-Stokes-Gleichungen, d.h. den Erhaltungsprinzipien für Masse, Impuls und Energie eines viskosen Fluides, sowie den Maxwell'schen Gleichungen. Hierzu gehören das Gauss'sche, Faraday-Konstante und Ampere-Maxwell'sche Gesetz sowie Erhaltungsgleichungen für den magnetischen Fluss und die elektrische Ladung. Eine zusammenfassende Darstellung der EMHD Modellbildung findet sich z.B. in [1, 2, 3]. Für die Metallurgie sind zudem komplexe Materialgesetze für die (erstarrenden) Metallschmelzen von Bedeutung.

Die EMHD kombiniert die hydro- und elektrodynamischen Eigenschaften von leitfähigen Flüssigkeiten. Somit sind Materi-

albewegungen und das Verhalten des elektrischen Feldes und des Magnetfeldes nicht unabhängig voneinander. Plasmabewegungen führen zu weiteren magnetischen und elektrischen Feldern; umgekehrt wird das elektromagnetische Feld die Fluidbewegung beeinflussen. Eine numerische Simulation dieser gekoppelten Felder ist mit den derzeit vorhandenen Werkzeugen nicht möglich. Es lassen sich Einzelphänomene isoliert betrachten, die aus einer Vereinfachung der allgemeinen EMHD Gleichungen hervorgehen. In der klassischen MHD werden quasi-statische magnetische Felder betrachtet, womit sich die Maxwell-Gleichungen erheblich vereinfachen lassen. Hinzu kommen vielfach weitere Annahmen, z.B. die des idealen elektrischen Leiters oder die Vernachlässigung magnetostriktiver Anteile in der hydrodynamischen Impulsgleichung. Diese Modelle stellen die Basis für weitergehende Betrachtungen, z.B. zur Modellierung der Turbulenz in MHD Strömungen [4, 5], dar. Für die vorliegende Fragestellung kommt hinzu, dass das Integrationsgebiet freie, bewegliche Ränder aufweist. In [6] werden offene Gerinneströmungen modelliert, allerdings unter der vereinfachenden Annahme einer konstanten Dicke der Filmströmung. In [8] wird die Wirkung von magnetischen Wechselfeldern auf die Ausbildung einer freien Oberfläche im sogenannten Kaltwandofen mittels kommerziellen numerischen Verfahren (ANSYS, Fluent) berechnet. Die Berechnung der hydrodynamischen und magnetischen Gleichungen erfolgt hier entkoppelt, d.h. die Maxwell-Gleichungen werden gelöst und damit konstante magnetische Kräfte als Quellterme in den Navier-Stokes-Gleichungen vorgegeben.

Die mit der Durchführung des Projektes betrauten Arbeitsgruppen am Institut für Technische Mechanik und am Institut für Metallurgie haben einschlägige Erfahrungen im Bereich der Modellierung von Schmelzen [9–11] sowie der Modellierung von MHD Prozessen [12, 13]. Technologische Aspekte des Projektes werden in [3] erläutert.

MHD Modellierung

Im Rahmen des Projektes sollen Berechnungsverfahren für die Navier-Stokes'schen Gleichungen und für die Maxwell-Gleichungen weiterentwickelt werden, wobei der Fokus auf der effizienten Kopplung der Verfahren liegt. Die Programmierung erfolgt

auf der Basis der open-source Programmbibliothek OpenFOAM. Das Ziel ist, ein modulares Paket aufzubauen, das es ermöglicht die oben beschriebenen Fragen zu beantworten. Neben der Analyse und Optimierung von Bandgießverfahren, dem eigentlichen Ziel des BMBF Projektes, besteht das Interesse, das Thema der Simulation von „Mehrfeldproblemen in den Materialwissenschaften“ längerfristig zu etablieren. So bestehen bereits jetzt Verknüpfungen zu einer Arbeitsgruppe, die sich mit elektrohydrodynamischen Problemen beschäftigt. Diese Aktivitäten sollen im weiteren Verlauf des Projektes ausgebaut werden.

Im laufenden Projekt ist vorgesehen, schrittweise die Modellierung magnetohydrodynamischer Probleme am Beispiel des Bandgießverfahrens aufzubauen und zu validieren. Im ersten Schritt werden offene Gerinneströmungen untersucht. Es werden numerische Verfahren zur Modellierung der freien Phasengrenzfläche und der Materialgesetze in OpenFOAM erprobt. Zu einem späteren Zeitpunkt wird die Modellierung der magnetischen Kraftdichte in den hydrodynamischen Gleichungen vorgenommen. Deren Berechnung erfolgt ebenfalls mit OpenFOAM; die Kopplung erfolgt zunächst nur unidirektional. Das eigentliche Wirkprinzip der elektromagnetischen Homogenisierung des Stahlfilms basiert darauf, dass die veränderliche Schichtdicke des Stahlfilms auf die Verteilung des EM-Feldes rückwirkt. Diese Kopplung wird im folgenden Arbeitspaket berücksichtigt, wobei geklärt werden muss, in welchen Intervallen diese Kopplung zu erfolgen hat. Zudem ist die Modellierungstiefe zu bewerten. Letztlich sollen in diesem Projekt Parameterstudien durchgeführt werden, um den Einfluss des Magnetfeldes auf die Ausbildung der freien Oberfläche bewerten zu können. Die Erstarrung der Schicht ist nicht Gegenstand dieses Projektes.

Bei den normalerweise verwendeten Verfahren der numerischen Strömungsmechanik (engl. CFD) lassen sich Strömungen in einem Gebiet Ω mit den Navier-Stokes-Gleichungen beschreiben. Die Navier-Stokes-Gleichungen sind definiert als

$$\frac{\partial(\rho\vec{u})}{\partial t} + \nabla \cdot (\rho\vec{u}\vec{u}) = -\nabla p + \nabla \cdot (\mu \nabla\vec{u}) + S \quad (2.1.1)$$

Darin ist ρ die Dichte, \vec{u} der Geschwindigkeitsvektor, p der Druck, μ die dynamische Viskosität und S der Quellterm, wie etwa Gravitation, Oberflächenspannung. Gleichung (2.1.1) ohne Quellterm wird in OpenFoam wie folgt abgebildet:

$$\text{fvm::ddt}(U) + \text{fvm::div}(\text{phi}, U) - \text{fvm::laplacian}(\nu, U) == -\text{fvc::grad}(p)$$

In OpenFoam wird das Finite-Volumen-Verfahren eingesetzt, um die Impulsgleichung zu diskretisieren. Zusammen mit einer Linearisierung resultiert daraus ein lineares Gleichungssystem, welches in OpenFOAM mit einem iterativen Gleichungslöser relaxiert wird. Die Lösung von Gleichung 2.1.1 erfolgt für eine inkompressible Strömung, d.h. unter Berücksichtigung der Divergenzfreiheit, mit dem SIMPLE und PISO Druckkorrekturverfahren.

Für die Modellierung der freien Oberfläche wird das „Volume of Fluid“ (VOF) Verfahren eingesetzt, bei dem die Konvektion einer skalaren Größe α betrachtet wird. Diese ist ein Maß für den Füllstand eines finiten Kontrollvolumens. Es gilt:

$$\alpha = \begin{cases} 0 & , \text{leere Zelle} \\ (0, 1) & , \text{Grenzflaechenzelle} \\ 1 & , \text{gefüllte Zelle} \end{cases}$$

Desweiteren sind die Dichte ρ und dynamische Viskosität μ bei dem VOF Verfahren nicht mehr global konstant, sondern werden mit dem passenden Füllstand der aktuellen Zelle verbunden. Deshalb ist eine neue Funktion ζ für die globale Verteilung der Dichte und Viskosität als

$$\zeta(\rho) = \alpha\rho_1 + (1 - \alpha)\rho_2$$

$$\zeta(\mu) = \alpha\mu_1 + (1 - \alpha)\mu_2$$

definiert. Die Bestimmung der Volumenfraktion α wird durch eine Kontinuitätsgleichung

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \vec{u}) = 0 \tag{2.3.2}$$

modelliert. Da Gleichung 2.3.2 die Erhaltungsbedingung nicht erfüllt (siehe Rusche [15]), wird zusätzlich noch ein Korrekturterm eingesetzt.

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \vec{u}) + \nabla \cdot (\vec{u}_c \alpha (1 - \alpha)) = 0. \quad (2.3.3)$$

Die in Gleichung 2.3.3 verwendete Größe \vec{u}_c bezeichnet man hier als „relative Geschwindigkeit“, welche direkt als Fluss berechnet wird:

$$\vec{u}_c = \vec{n} \min \left(cAlpha \frac{|u_f|}{|S_f|}, \max \left(\frac{|u_f|}{|S_f|} \right) \right),$$

wobei *cAlpha* eine einstellbare Konstante (siehe OpenFOAM User Guide) ist und $\vec{n} = \frac{\nabla \alpha}{|\nabla \alpha|}$ der normierte Normalenvektor, der die Ausbreitungsrichtung des Fluids angibt, ist. Darüberhinaus muss bei freien Oberflächen speziell der Einfluss der Gravitation und der Oberflächenspannungskräfte modelliert werden. Dies erfolgt in den Navier-Stokes-Gleichungen durch zusätzliche Quellterme, d.h.

$$\frac{\partial(\rho \vec{u})}{\partial t} + \nabla_\phi \cdot (\rho \vec{u} \vec{u}_\phi) = -\nabla p + \nabla_\phi \cdot (\mu \nabla \vec{u}_\phi) + \rho g + f_\sigma. \quad (2.3.4)$$

In Gleichung 2.3.4 bezeichnen ρg die Schwerkraft und $f_\sigma = \sigma \cdot k \cdot \vec{n}$ die Oberflächenkräfte, wobei σ die Oberflächenspannung, $\vec{n} = \frac{\nabla \alpha}{|\nabla \alpha|}$ der Normalenvektor und $k = \nabla \cdot \vec{n}$ die lokale Krümmung der freien Oberfläche ist.

Zur Beschreibung der magnetohydrodynamischen Strömung wird eine Kopplung der hydrodynamischen Gleichungen zusammen mit den Maxwell-Gleichungen und dem Ohmschen Gesetz notwendig. Das Ziel ist es, einen neuen Term, die sogenannte Lorentz-Kraft, in der Navier-Stokes-Gleichung zu bestimmen, damit die durch den Einsatz des Magnetfelds \vec{B} erzeugte Wirkung im Strömungsfeld modelliert wird. Sie werden im Folgenden beschrieben:

Maxwell-Gleichungen:

$$\nabla \cdot \vec{J} = 0 , \quad (2.4.1)$$

$$\nabla \cdot \vec{B} = 0 , \quad (2.4.2)$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} , \quad (2.4.3)$$

$$\nabla \times \vec{B} = \mu_m \vec{J} . \quad (2.4.4)$$

Ohmsches Gesetz:

$$\vec{J} = \sigma_e \left(\vec{E} + \vec{u} \times \vec{B} \right) . \quad (2.4.5)$$

Lorentz Kraft:

$$\vec{F}_L = \vec{J} \times \vec{B} . \quad (2.4.6)$$

Mit der Maxwell-Gleichung 2.4.4 wird die Lorentzkraft erweitert, so dass folgt

$$\vec{F}_L = \frac{1}{\mu_m} \left(\nabla \times \vec{B} \right) \times \vec{B} . \quad (2.4.7)$$

Mit der Identität $\nabla (\vec{A} \cdot \vec{C}) = \nabla_A (\vec{A} \cdot \vec{C}) + \nabla_C (\vec{A} \cdot \vec{C})$ folgt für den speziellen Fall, dass $\vec{A} = \vec{C}$ weiterhin:

$$\nabla (\vec{A} \cdot \vec{A}) = 2 \left(\vec{A} \times (\nabla \times \vec{A}) + (\vec{A} \cdot \nabla) \vec{A} \right) \rightarrow \vec{A} \times (\nabla \times \vec{A}) = \frac{1}{2} \nabla (\vec{A} \cdot \vec{A}) - (\vec{A} \cdot \nabla) \vec{A}$$

Weil $\vec{A} \times (\nabla \times \vec{A}) = -(\nabla \times \vec{A}) \times \vec{A}$ gilt, lautet die Gleichung 2.4.7 wie folgt:

$$\vec{F}_L = -\frac{1}{2\mu_m} \nabla \vec{B}^2 + \frac{1}{\mu_m} \left(\vec{B} \cdot \nabla \right) \vec{B} . \quad (2.4.8)$$

Für die Impulserhaltungsgleichung 2.3.4 folgt damit

$$\frac{\partial(\rho\vec{u})}{\partial t} + \nabla_{\phi} \cdot (\rho\vec{u}\vec{u}_{\phi}) = -\nabla p + \nabla_{\phi} \cdot (\mu \nabla \vec{u}_{\phi}) + \rho g + f_{\sigma} - \frac{1}{2\mu_m} \nabla \vec{B}^2 + \frac{1}{\mu_m} (\vec{B} \cdot \nabla) \vec{B} \quad (2.4.9)$$

Aus Gleichung 2.4.8 wird deutlich, dass zur Bestimmung der Lorentzkraft nur das Magnetfeld \vec{B} benötigt wird. Um das Magnetfeld \vec{B} zu bestimmen, werden die Maxwell-Gleichung 2.4.3 und das Ohmsche Gesetz benötigt:

$$\nabla \times \vec{E} = -\frac{\partial B}{\partial t} \quad \text{wobei} \quad \vec{E} = \frac{\vec{J}}{\sigma_e} - \vec{u} \times \vec{B}. \quad (2.4.10)$$

Mit der Maxwell-Gleichung 2.4.4 wird die Gleichung 2.4.10 erweitert, so dass folgt:

$$\nabla \times \left(\frac{\nabla \times \vec{B}}{\sigma_e \mu_m} \right) - \nabla \times (\vec{u} \times \vec{B}) = -\frac{\partial B}{\partial t}. \quad (2.4.11)$$

Falls die elektrische Leitfähigkeit σ_e und magnetische Leitfähigkeit μ_m konstant sind, können wir die Gleichung 2.4.11 mit dem Vektoroperator $\nabla \times (\nabla \times \vec{B}) = \nabla (\nabla \cdot \vec{B}) - \nabla^2 \vec{B}$ weiter vereinfachen

$$\frac{1}{\sigma_e \mu_m} (\nabla (\nabla \cdot \vec{B}) - \nabla^2 \vec{B}) - \nabla \times (\vec{u} \times \vec{B}) = -\frac{\partial B}{\partial t}. \quad (2.4.12)$$

Die oben erstellte Gleichung 2.4.12 wird nach 2.4.2 $\nabla \cdot \vec{B} = 0$ weiter vereinfacht:

$$-\frac{1}{\sigma_e \mu_m} \nabla^2 \vec{B} - \nabla \times (\vec{u} \times \vec{B}) = -\frac{\partial B}{\partial t}. \quad (2.4.13)$$

Letztendlich wird Gleichung 2.4.13 mit der Maxwell-Gleichung 2.4.2 und dem Vektoroperator

$$\nabla \times (\vec{A} \times \vec{B}) = \vec{A} \nabla \cdot \vec{B} - \vec{B} \nabla \cdot \vec{A} + (\vec{B} \cdot \nabla) \vec{A} - (\vec{A} \cdot \nabla) \vec{B}$$

zusammengefasst:

$$\frac{\partial B}{\partial t} - \frac{1}{\sigma_e \mu_m} \nabla^2 \vec{B} - (\vec{B} \cdot \nabla) \vec{u} + (\vec{u} \cdot \nabla) \vec{B} = 0. \quad (2.4.14)$$

Bis hier wurden zunächst die wesentlichen Bestandteile der Modellierungen vorgestellt. Der zugehörige Lösungsalgorithmus ist zusammenfassend in Abbildung 4 dargestellt.

Ergebnisse

Im Folgenden werden exemplarisch einige Ergebnisse von Berechnungen vorgestellt, in denen die Komplexität des Problems sukzessive gesteigert wurde. Dadurch kann einerseits im Sinne eines Suppressionstest eine Plausibilitätskontrolle zur Validierung erreicht werden. Zum anderen lassen sich auf diese

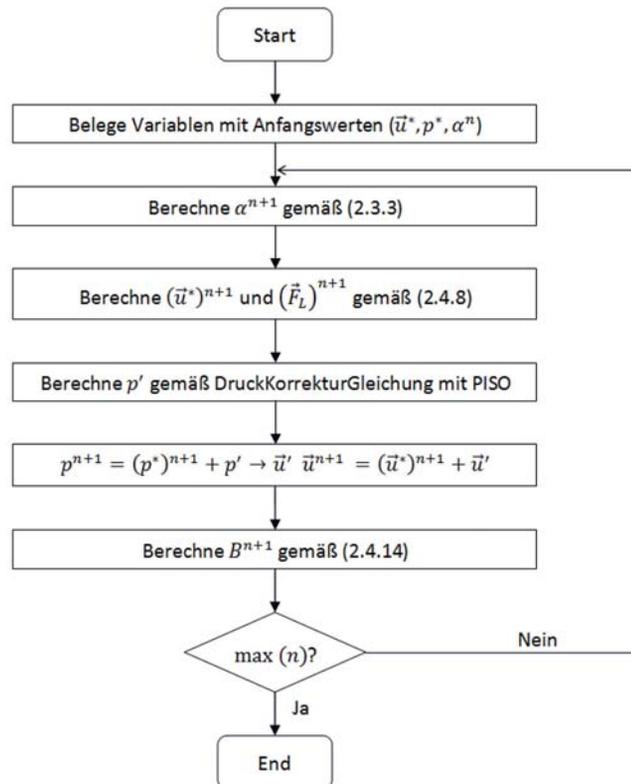


Abbildung 4: Lösungsalgorithmus des MHD Verfahrens

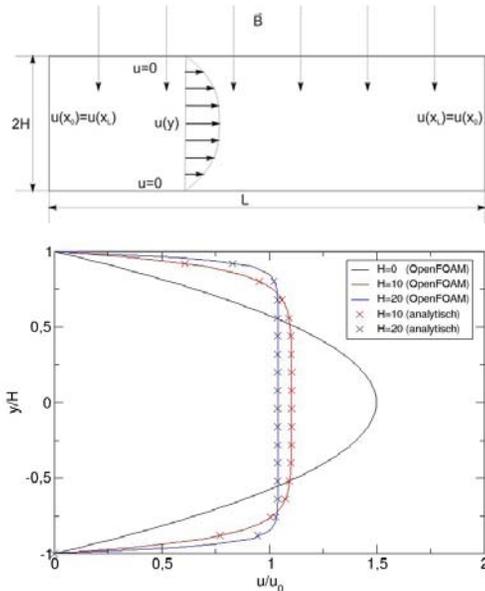


Abbildung 5: Prinzipskizze und berechnete Geschwindigkeitsverteilung zur Verifikation des Berechnungsverfahrens

Weise einige grundlegende Phänomene der Strömung unter Magnetfeldeinfluss herausarbeiten.

Zur Verifikation des Modells wird in einem ersten Schritt die „Hartmann“ Strömung betrachtet. Eine Hagen-Couette-Strömung zwischen zwei parallelen Wänden wird durch ein Magnetfeld senkrecht zur Strömungsrichtung verändert. Für diese Konfiguration ist eine analytische Lösung vorhanden, anhand derer die numerischen Ergebnisse überprüft werden können. Eine Prinzipskizze ist in Abbildung 5 zusammen mit analytischen Lösungen für verschiedene magnetische Feldstärken dargestellt. Betrachtet wird ein kartesisches Rechengebiet mit periodischen Randbedingungen in Strömungsrichtung. Die Strömung wird durch einen konstanten Druckgradienten getrieben und ist laminar, so dass sich ohne Magnetfeld eine parabolische Geschwindigkeitsverteilung einstellt. Das Magnetfeld führt zu einer Lorenzkraft, die umso größer ist, je höher die Geschwindigkeit ist. Daher wird mit Magnetfeld die Geschwindigkeit in der Kanalmitte stärker verringert als nahe den Wänden. Für die vorliegenden Berechnungen wurden Materialeigenschaften von flüssigem Eisen verwendet.

Tabelle 1: Materialeigenschaften von Luft und flüssigem Eisen

Geschwindigkeit Schmelze $\bar{u} \left(\frac{m}{s} \right)$	Dichte Luft $\rho \left(\frac{kg}{m^3} \right)$	Viskosität Luft $\nu \left(Pa \cdot s \right)$	Elektrische Leitfähigkeit Luft $\sigma_{Luft} \left(\Omega m \right)^{-1}$	Magnetische Leitfähigkeit Luft $\mu_{Luft} \left(\frac{H}{m} \right)$
0.1	1	1.48e-05	1e-06	1e-06
Oberflächen- spannung Schmelze $\sigma \left(\frac{N}{m} \right)$	Dichte Schmelze $\rho \left(\frac{kg}{m^3} \right)$	Viskosität Schmelze $\nu \left(Pa \cdot s \right)$	Elektrische Leitfähigkeit Schmelze $\sigma_{Steel} \left(\Omega m \right)^{-1}$	Magnetische Leitfähigkeit Schmelze $\mu_{Steel} \left(\frac{H}{m} \right)$
1.5	6490	8.8339e-07	7.14e05	1e-03

In einem weiteren Schritt der Verifikation wurde die Modellierung der freien Oberfläche mit einbezogen. Hier existiert nicht mehr die Möglichkeit, eine analytische Lösung zu Vergleichszwecken zu verwenden. Die Konfiguration ist ähnlich der zuvor betrachteten, mit dem Unterschied, dass eine Phasengrenzfläche vorhanden ist. Das Magnetfeld wirkt nur in einem definierten Abschnitt der Strömung. Das Fluid ist auch hier durch konstante Materialdaten definiert, d.h. eine Phasenumwandlung wird hier noch nicht betrachtet.

Auch hier führt die Lorenzkraft zu einer Abbremsung des Fluids und einer Verstetigung des Geschwindigkeitsprofils, wie in Abbildung 6 dargestellt. Deutlich zu erkennen ist die Verzögerung der Strömung im Kernbereich, während nahe der Wand eine Beschleunigung stattfindet.

Schließlich wird der Einfluss von lokalen Magnetfeldern auf dreidimensionale Strömungen untersucht. In Abbildung 7a ist die Konfiguration dargestellt, ein Kanal mit einer Breite von 0,05 m, einer Höhe von 0,1 m und einer Länge von 1 m. Die verwendeten Parameter sind in Tabelle 1 aufgelistet.

Es werden zwei Szenarien untersucht. In der ersten Variante ist das Magnetfeld in einem engen Bereich lokalisiert, während sich das Magnetfeld in der zweiten Variante über die gesamte Kanalbreite erstreckt, wie in Abbildungen 7b und 7c dargestellt ist.

Abbildung 8 zeigt die Lage der freien Oberfläche und die Ge-

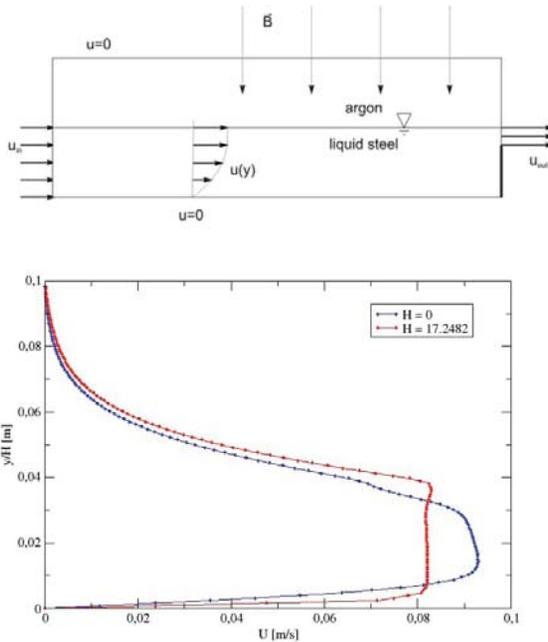
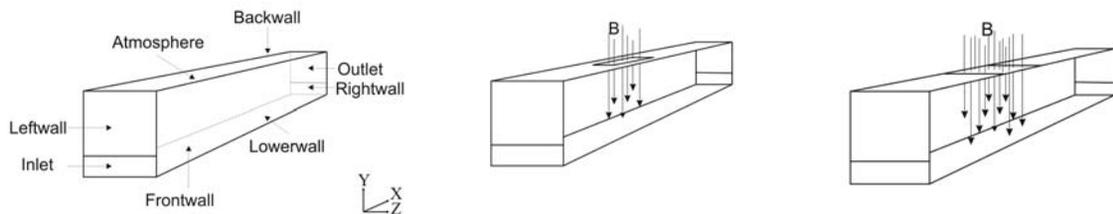


Abbildung 6: Prinzipskizze zur Verifikation des Berechnungsverfahrens mit freier Oberfläche

schwindigkeitsverteilung in einigen ausgewählten Ebenen, wobei zunächst kein Magnetfeld berücksichtigt wird. In Abbildung 9 wird der Einfluss des lokal aufgeprägten Magnetfeldes deutlich. Durch die Verzögerung des Fluids im Bereich der hohen magnetischen Feldstärken kommt es in den Randbereichen zu einer Beschleunigung des Fluids, da ein vorgegebener Massenstrom durch den Kanal befördert werden muss. Zugleich wird eine deutliche Verzerrung der freien Oberfläche sichtbar. Im Falle eines Magnetfeldes, das sich über die gesamte Kanalbreite erstreckt, ist die Geschwindigkeitsabnahme und damit die Auffaltung der freien Oberfläche noch deutlicher. Zugleich wird eine Sekundärströmung in der flüssigen Phase induziert.

Abbildung 7: Prinzipskizze des Berechnungsgebietes und Anordnung des magnetischen Feldes in drei Dimensionen



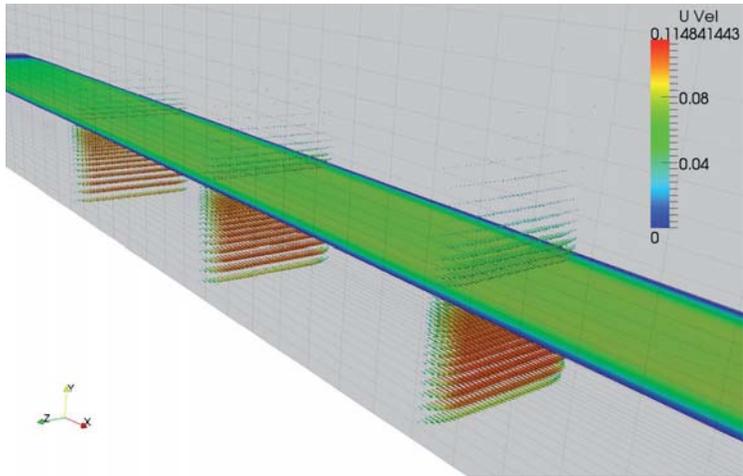


Abbildung 8: Dreidimensionale Strömung im Kanal ohne Magnetfeld

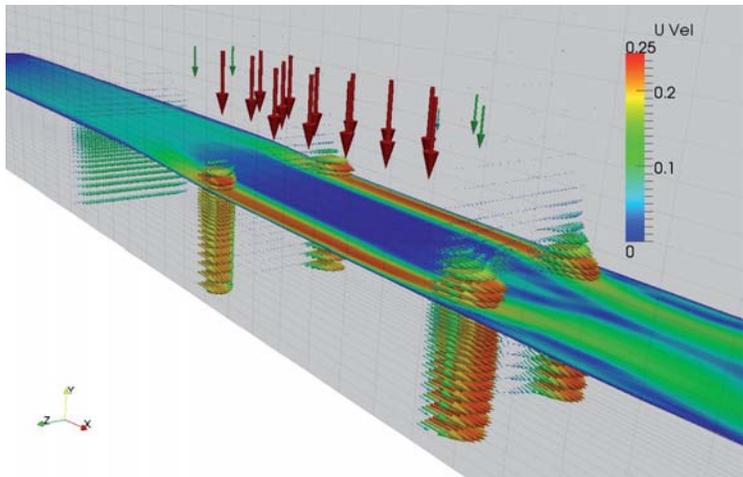


Abbildung 9: Dreidimensionale Strömung im Kanal. Das Magnetfeld wirkt lokal in der Mitte des Kanals

Die hier gezeigten Phänomene verdeutlichen den Einfluss des magnetischen Feldes auf die Fluidbewegung. Ob und in welchem Maße diese Effekte unter den beim Bandgießen vorliegenden Parametern auftreten, wird in den kommenden Untersuchungen gezeigt werden.

Zusammenfassung und Schlussfolgerung

Ziel der bisher durchgeführten Berechnungen ist die Verifi-

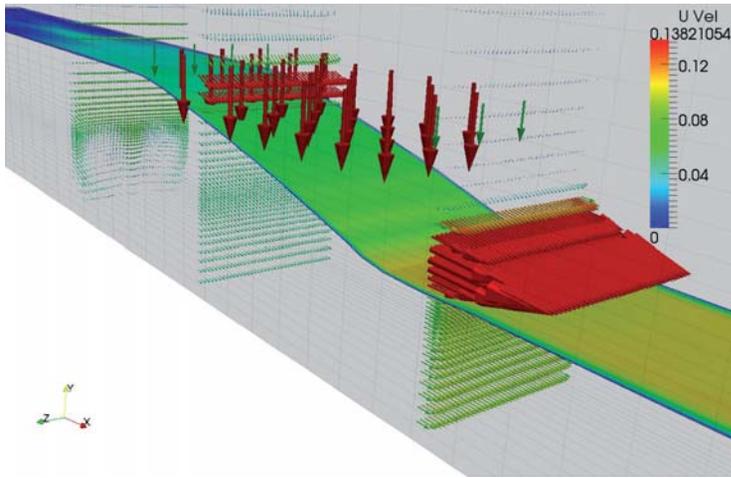


Abbildung 10: Dreidimensionale Strömung im Kanal. Das Magnetfeld ist konstant über dem Querschnitt

kation des physikalischen Modells und dessen Implementierung in OpenFOAM zur Berechnung von inkompressiblen Strömungen unter dem Einfluss von externen Magnetfeldern. In den folgenden Untersuchungen steht die Modellierung des Wärmetransports und der Erstarrung des Stahls im Vordergrund. Aus numerischer Sicht besteht hier die Herausforderung darin, dass sich die Modellgleichungen zur Beschreibung des Problems lösungsabhängig verändern. In der flüssigen Phase kann von einer Newtonschen Flüssigkeit mit temperaturabhängiger Viskosität ausgegangen werden. Sobald die Erstarrung einsetzt, wird latente Wärme freigesetzt und das spezifische Volumen sowie die Rheologie des Materials ändern sich rapide. Schließlich liegt nach der Erstarrung ein Festkörper vor. Um diesen Übergang in einem konsistenten Ansatz modellieren zu können, sind weitere Arbeiten notwendig. An dieser Stelle können ggf. Phasenfeldmethoden eingesetzt werden. Die numerischen Verfahren, z.B. zur Druck-Geschwindigkeitskopplung in der flüssigen Phase, müssen ebenfalls überdacht werden.

Literatur

- [1] G. S. Dulikravich and S. R. Lynn, Unified electro-magneto-fluid dynamics (EMFD): a survey of mathematical models, Inf. J Non-Linear Mechanics, Vol. 32, No. 5, pp. 923-932, 1997.

- [2] A. C. Eringen and G. A. Maugin, *Electrodynamics of Continua I; Foundations and Solid Media*. Springer, New York (1990).
- [3] A. C. Eringen and G. A. Maugin, *Electrodynamics of Continua II; Fluids and Complex Media*. Springer, New York (1990).
- [4] S. Kenjeres, K. Hanjalic, A direct-numerical-simulation-based second-moment closure for turbulent magnetohydrodynamic flows, *Phys. Fluids* 16(5), 2004.
- [5] H. Kobayashia, Large eddy simulation of magnetohydrodynamic turbulent channel flows with local subgrid-scale model based on coherent structures, *Phys. Fluids* 18, 2006.
- [6] N. B. Morley, P. H. Roberts, Solutions of uniform, open-channel, liquid metal flow in a strong, oblique magnetic field, *Phys. Fluids* 8 (4), 1996.
- [7] Y. Fautrelle, A. Sneyd, J. Etay, Effect of AC Magnetic Fields on Free Surfaces, in: *Fluid Mechanics and Its Applications* Vol. 80, *Magnetohydrodynamics*, Springer, p 345-355, 2007.
- [8] A. Umbrashko, E. Baake, A. Jakovics, Melt Flow and Skull Formation Modelling Possibilities for TiAl Melting Process in Induction Furnace With Cold Crucible, *Modelling for Electromagnetic Processing*, Hannover, October 27-29, 2008.
- [9] Kumar, V.; G. Biswas; G. Brenner; F. Durst, Effect of thermocapillary convection in an Industrial Czochralski crucible: numerical simulation, *Int. J. Heat and Mass Transfer*, 46, 2003, 1641-1652.
- [10] Kumar, V.; B. Basu; S. Enger; G. Brenner; F. Durst, Role of Marangoni convection in Si-Czochralski melts. Part I: 3D Predictions without crystal. *J. Crystal Growth* 253, 2003, 142-154.
- [11] Kumar, V.; B. Basu; S. Enger; G. Brenner; F. Durst, Role of Marangoni convection in Si-Czochralski melts. Part II: 3D Predictions with crystal rotation, *J. Crystal Growth* 255, 2003, 27-39.
- [12] K.-H. Spitzer, Application of Rotating Magnetic Fields in Czochralski Crystal Growth, *Progress in Crystal Growth and Characterization of Materials*, 38, 1999, 39.
- [13] K.-H. Spitzer, O. Pesteanu, Application of Travelling Magnetic Fields in Metallurgy“, *Proc. EPM-2000*, Nagoya, Japan, April 3-6., 2000, 409.

- [14] K.-H. Spitzer, R. Scholz, J. Kroos, K.H. Hower, R. Nyström, E. Burström, W. Reichelt, M. Dubke, "Entwicklungsstand beim DSC-Bandgießverfahren", Stahl u. Eisen, 121 2001, Nr. 5, 73.
- [15] Henrik Rusche, „Computational fluid dynamics of dispersed two-phase flows at high phase fractions“, PhD Thesis, Imperial College, London, 2003.



2 Produktion, Logistik und Verkehr

In der Produktionsplanung spielt die frühzeitige Abbildung und Simulation von realen Prozessen in Form einer sogenannten Digitalen Fabrik eine stetig größer werdende Rolle. Dabei steht der Begriff der Digitalen Fabrik nicht nur für die Abbildung von Maschinen und Strukturen in 3D-CAD und Virtual Reality Werkzeugen, sondern insbesondere für eine Modellierung und Untersuchung der Abläufe im Produktionsprozess. Dies geschieht durch eine möglichst vollständige Integration von Informationen über die Funktionsweise von Maschinen sowie die gesamten Abläufe. Auf diese Weise entsteht als Simulationsmodell ein umfassendes Netzwerk aus digitalen Modellen, Methoden und Werkzeugen, die über gemeinsame Datenstrukturen ansprechbar sind und durch ein durchgängiges Datenmanagement vernetzt sind.

Mit Hilfe der Digitalen Fabrik können die Modelle bereits vor dem Bau der realen Fabrik auf ihr Potential in Bezug auf Funktion, Kollisionen von Materialströmen und Werkzeugnutzungen sowie ihre Ergonomie hin untersucht werden. Damit können Engpässe bereits im Planungsstadium erkannt und behoben werden. Ein wichtiger Aspekt hierbei ist eine möglichst hohe Integrationsdichte: Bei der Verwendung wenig integrierter Einzelwerkzeuge entfällt ein beträchtlicher Anteil der gesamten Entwicklungskosten auf die Einbindung und Verknüpfung der einzelnen Applikationen. Nur durch den aufwandsarmen Import und Export von Modellen, Parametern und Ergebnissen ist die Nutzung der vollen Potentiale der Planungsapplikationen auf allen Ebenen der Fabrikstruktur- und Layoutplanung möglich.

2.1 Virtuelle Fabrikstruktur- und Layoutplanung mit erweiterten digitalen Modellen

*Uwe Bracht und Christian Schlange –
Institut für Maschinelle Anlagentechnik und Betriebsfestigkeit*

Einleitung

Die effektive Nutzung der Virtuellen Realität (VR) in der Fabrikplanung setzt neben leistungsfähigen Algorithmen zur Echtzeitvisualisierung ebenfalls eine durchgängige Integration in die bestehende Prozess- und Systemlandschaft voraus. Notwendig dafür sind unter anderem der aufwandsarme Import von 3D-Fabrikmodellen sowie die Übernahme von Modellparametern und Ergebnissen, beispielsweise aus Planungsapplikationen der Digitalen Fabrik. Entsprechend dieser Anforderungen wurde die im vorliegenden Beitrag beschriebene Gesamtplanungslösung TeamPlan entwickelt und in der VR-gestützten Struktur- und Layoutplanung zur Anwendung gebracht.

In einer zweistufigen Vorgehensweise werden die Geometriedaten der Fabrikanlage zunächst aus dem Autorensystem (3D-CAD-System oder 3D-Modeler) übernommen. Durch Nutzung aktueller CAD-Austauschformate, wie beispielsweise JT, können auch komplexe Fabrikszenarien geometrisch korrekt und unter Erhalt der Modellstruktur integriert werden.

Über eine objektorientierte Erweiterungssystematik kann das Modell in der zweiten Stufe mit frei konfigurierbaren Informations- bzw. Funktionskomponenten verknüpft werden. Auf Grundlage dieser Erweiterten Virtuellen Modelle (EVM) bietet die beschriebene Gesamtplanungslösung ein Framework zur aufwandsarmen Implementierung und interaktiven Anwendung von Planungsverfahren in virtuellen Umgebungen.

Die praktische Anwendung der Lösung wird anhand eines typischen Restrukturierungsprojektes aufgezeigt. Dazu wird eine Methodik zur Aufnahme der Ist-Situation vorgestellt, bei der erfasste Informationen direkt im virtuellen Modell hinterlegt werden können. Des Weiteren wird die Umsetzung einer sta-

tischen Materialfluss- und Prozessvisualisierung beschrieben, die auf effizienten Wegefindungsalgorithmen im 3D-Layout basiert.

Integration in der Digitalen Fabrik

Ein wichtiges Erfolgskriterium zur nachhaltigen Anwendung der Digitalen Fabrik ist ein hoher Integrationsgrad der eingesetzten Systeme und Modelle. Die VDI-Richtlinie 4499 [VDI-4499, 2008, S. 2] definiert die Digitale Fabrik in diesem Zusammenhang als „umfassendes Netzwerk aus digitalen Modellen, Methoden und Werkzeugen, die durch ein durchgängiges Datenmanagement vernetzt werden“ und fordert damit gleichzeitig effiziente Integrationsprinzipien. Unterstrichen wird diese Notwendigkeit durch Studien, nach denen bis zu 35 % der Entwicklungs- und Wartungskosten eines typischen Softwareprojektes durch die Integration bestehender Applikationen verursacht werden (Heutschi, 2007) sowie ein hoher Anteil der Integrationsprojekte ihre Ziele innerhalb des vorgesehenen Zeitplans oder Budgets nicht erreichen (Siegenthaler u. Schwinn, 2006).

Beginnend mit dem Austausch von Planungsergebnissen zwischen beteiligten Planungsapplikationen bis hin zur Kopplung einzelner Planungssysteme kann die Integration dabei auf verschiedenen Ebenen erfolgen. Aus technischer Sicht werden die Ansätze unter anderem nach Daten-, Modell-, Prozess- sowie Systemintegration unterschieden (Jung, 2006). Um Softwaresysteme in einem integrierten Gesamtplanungsprozess zur Anwendung zu bringen, müssen zunächst die zugrunde liegenden Daten bzw. digitalen Modelle integriert werden. Datenintegration kann dabei treffend als Problem der Kombination von Daten aus unterschiedlichen Quellen und der Bereitstellung einer für die Benutzer einheitlichen Sicht auf diese Daten definiert werden (Lenzerini, 2002). Zur technischen Umsetzung können Datenbestände dazu physikalisch zusammengeführt oder aber als weiterhin eigenständige Einheiten über eine logische Sicht zugreifbar gemacht werden. Um darüber hinaus die Anwendung der Daten in unterschiedlichen Applikationen zu ermöglichen, sind ebenfalls die zu Grunde liegenden digitalen Modelle zu integrieren. Dabei rücken neben dem Modellsyntax ebenfalls die Modellbedeu-

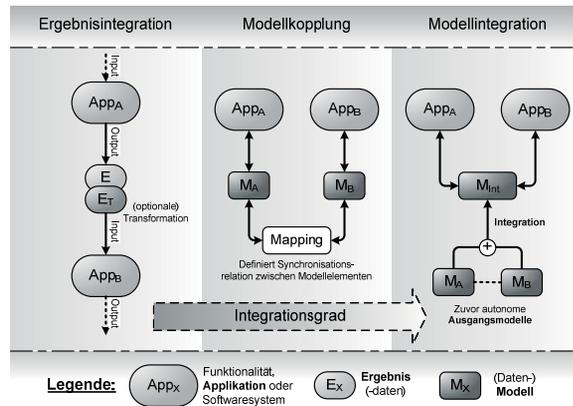


Abbildung 1: Ausprägungen der Modellintegration

tung (Semantik) sowie die praktische Nutzung (Pragmatik) in den Blickpunkt. Unterschiedliche Modellwelten müssen in Begrifflichkeit und Bedeutung untersucht und in Beziehung gesetzt werden (Wenzel, 2004).

Die Integration von Planungsapplikationen bzw. -funktionalitäten und der genutzten Modelle in Anwendungen der Digitalen Fabrik kann in mehreren Ausprägungen erfolgen (siehe Abbildung 1). Bei der reinen Ergebnisintegration werden lediglich Ergebnisse der einzelnen Applikationen ausgetauscht, bei Bedarf unter Nutzung einer Transformation der Datenformate. Die technische Umsetzung kann dabei durch die Nutzung standardisierter Austauschformate, beispielsweise durch JT im CAD-Umfeld oder durch XML-basierte Formate für alphanumerische Daten, vereinfacht werden. Ebenfalls erfolgversprechend ist der Ergebnisaustausch über Serviceorientierte Architekturen (SOA), der eine Trennung zwischen Anwendungslogik und Ergebnis ermöglicht (Heutschi, 2007).

Weiter als die Ergebnisintegration geht die Kopplung von Modellen, die beispielsweise bei der Verbindung von Ablaufsimulatoren mit Visualisierungs- bzw. VR-Werkzeugen zu beobachten ist. In diesem Fall existieren unterschiedliche Modelle zur Abbildung ein und desselben Sachverhalts. Zur Synchronisation der Modellwelten sind eintretende Ereignisse bzw. Aktionen zu übertragen, was beispielsweise durch entsprechende Mapping-Relationen umgesetzt werden kann (Wenzel, 2001).

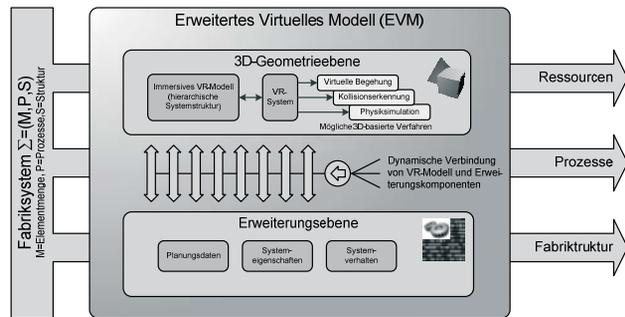
Bei der reinen Modellintegration werden vormals autonome Modelle in einem integrierten Modell zusammengeführt. Als Arbeitsmedium für verschiedene Applikationen bietet dieses Modell eine zentrale und redundanzfreie Datenhaltung mit einheitlicher Semantik, die unter anderem die direkte Nutzung erzielter Ergebnisse in nachgelagerten Planungsschritten ermöglicht. Diesem Vorteil steht allerdings ein Mehraufwand beim Integrationsprozess gegenüber, da die Anforderungen der beteiligten Applikationen sowie die Semantik der Ausgangsmodelle detailliert zu berücksichtigen sind.

Erweiterungskonzept für virtuelle Modelle

Aufbauend auf der bereits erfolgreichen Anwendung der Virtuellen Realität in der Fabrikplanung, beispielsweise zur virtuellen Begehung geplanter Produktionsanlagen (Bracht et al., 2005), wird im Folgenden ein Ansatz zur Nutzung integrierter Modelle in der VR-gestützten Struktur- und Layoutplanung beschrieben. Dieser basiert auf der dynamischen Verknüpfung herkömmlicher geometriebasierter VR-Modelle mit Zusatzinformationen und aktiven Funktionskomponenten. Durch diese als Modellerweiterung bezeichnete Technik wird der zeitsparende Aufbau problemspezifischer Modelle ermöglicht und eine geeignete Grundlage für die Anwendung rechnergestützter Planungsverfahren geschaffen. Das zu modellierende System kann in systemtheoretischer Sichtweise dabei als Paar $S = (T, R)$ aufgefasst werden, wobei es sich bei T um die Menge der Systemelemente und bei R um eine Menge von Relationen über T handelt, siehe beispielsweise (Klir, 2001). Ausgehend von dieser Definition können alle grundlegenden Systemeigenschaften wie Funktion, Struktur und Subsystemhierarchie abgeleitet werden, die bei der Bildung fachspezifischer Systemmodelle zu berücksichtigen sind. Bei (Schmigalla, 1995) wird die Fabrik beispielsweise als Tripel $\hat{a} = (M, P, S)$ aufgefasst, wobei M die Menge der Fabrikelemente (Ressourcen technischer und personeller Art) wiedergibt. Die Menge P gibt mögliche Produktionsprozesse wieder, die als n-stellige Relationen über M modelliert werden. Die Struktur des Fabriksystems wird in ähnlicher Weise über die zweistellige Relation S abgebildet.

In Anbetracht der erwarteten hohen Anzahl möglicher Fragestellungen und Modellformen wurde zur Umsetzung der

Abbildung 2: Konzept der Erweiterten Virtuellen Modelle (EVM)



Modellerweiterung eine dynamische Verknüpfung von Informations- und Funktionskomponenten mit dem geometriebasierten virtuellen Modell gewählt (siehe Abbildung 2). Damit können unter anderem Ressourcen, Prozesse sowie Struktureigenschaften des Fabriksystems erweitert bzw. modelliert und in VR-gestützten Planungsverfahren genutzt werden.

Als Träger der erweiterten Information bzw. Funktion werden Objekte im Sinne der objektorientierten Programmierung eingesetzt, im Folgenden als digitale Fabrikobjekte (DFO) bezeichnet. Dadurch können bestehende Systemelemente, beispielsweise die 3D-Objekte eines virtuellen Szenarios, nicht nur mit Informationen (Eigenschaften) sondern auch mit Funktion bzw. Verhalten (Methoden) hinterlegt werden. Zudem handelt es sich bei Klassen bzw. Objekten als deren Instanziierung um ein anerkanntes Werkzeug der objektorientierten Softwareentwicklung, das höchstmögliche Flexibilität bei der digitalen Systemmodellierung bietet.

Im Gegensatz zu einer vollständigen programmtechnischen Neuentwicklung eines Modells und der darauf aufbauenden Planungsverfahren ist die Technik der Verknüpfung mit externen Objekten aufwandsarm und auch ohne Neuerstellung des Programms möglich. Die Erweiterung kann dabei ohne Eingriff in das Ausgangsmodell erfolgen, was als Nebeneffekt einen hohen Grad an Unabhängigkeit vom eingesetzten VR-System mit sich bringt. In der Umsetzung erhält der Planer die Möglichkeit, geeignete Fabrikobjekte aus einer Objektbibliothek zu wählen, zu instanzieren und dynamisch, d.h. zur Laufzeit, mit einem bestehenden virtuellen Modell zu verknüpfen. Dabei kommt das aus der Softwareentwicklung bekannte

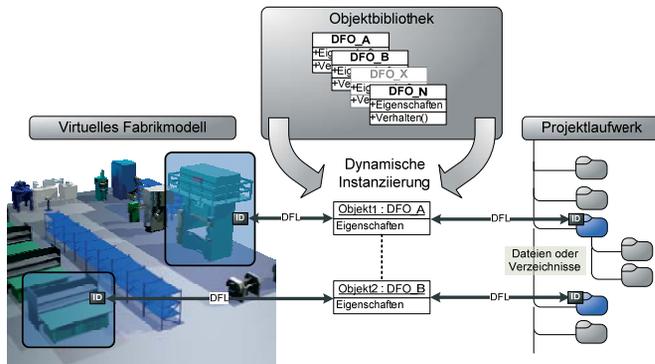


Abbildung 3: Beispielhafte Anwendung der Modellerweiterung

Entwicklungspattern „Prototyp-Instanz“ zum Einsatz (siehe beispielsweise Gamma et al. , 1995). Grundsätzlich erlaubt das eingesetzte Prinzip die Erweiterung weiterer digital vorliegender Elemente, unter anderem von Daten oder Verzeichnissen, sofern diese durch eine eindeutige ID als Ziel der Verknüpfung identifizierbar sind. Abbildung 3 fasst das beschriebene Erweiterungskonzept anhand eines beispielhaften Modellszenarios zusammen, bei dem sowohl die Elemente eines virtuellen Modells als auch eine Verzeichnisstruktur durch digitale Fabrikobjekte erweitert wurden.

Zur Persistierung, d. h. dauerhaften Speicherung der digitalen Fabrikobjekte, wurde ein objektorientiertes Datenbankmanagementsystem (OODBMS) eingesetzt. Gleiches gilt für vorhandene Verknüpfungen, die ebenfalls in Objektform vorliegen und somit gespeichert werden müssen. Im Gegensatz zu objektrelationalen Ansätzen, bei denen Objektstrukturen in Schemata eines herkömmlichen relationalen Datenbankmanagementsystems (RDBMS) abgebildet werden, ist die direkte Speicherung der Objekte insgesamt fehlerunanfälliger und auf Grund der erreichten Performanz besser für die VR-Echtzeitanwendung geeignet. Für Details zur technischen Umsetzung der Erweiterungssystematik sei auf (Schlange, 2010) verwiesen.

Anwendung der Gesamtplanungslösung TeamPlan

Um die gezeigte Methodik der Modellerweiterung in der Praxis einsetzen zu können, wurde eine VR-gestützte Gesamtplanungslösung zur teambasierten Struktur- und Layoutplanung,

kurz TeamPlan, entwickelt. Grundlage dieser Software ist eine modulare und offene Systemarchitektur auf Basis der .NET-Plattform, in die vorhandene bzw. neu zu entwickelnde Softwarekomponenten in einfacher Form integriert werden können. Ausgangspunkt der Modellerweiterung sind 3D-Fabrikmodelle, die mittels CAD-Schnittstelle in das VR-System importiert werden. Dabei hat sich insbesondere das im Umfeld der Digitalen Fabrik verbreitete JT-Format als geeignet erwiesen, mit dem Struktur, Geometrie und Materialeigenschaften des Originalmodells weitgehend unverfälscht in ein VR-Modell übertragen werden können. Im Kern der Gesamtlösung steht ein VR-System, das neben der eigentlichen Darstellung (Rendering) der virtuellen Fabrikmodelle unter anderem für die Benutzerinteraktion verantwortlich ist. Auf Grund der Komplexität typischer 3D-Fabrikzenarien werden dabei hohe Anforderungen an Renderingleistung und Darstellungsqualität gestellt. Um eine Bearbeitung des Layouts zu ermöglichen, muss das VR-System darüber hinaus über effiziente Verfahren zur Objektselektion sowie –manipulation verfügen. Wichtig ist ebenfalls Funktionalität, um mehrere Layouts in unterschiedlichen Fenstern bearbeiten bzw. virtuell begutachten zu können. In Verbindung mit einer freien Konfigurierbarkeit der graphischen Oberfläche wird damit ein wichtiger Grundstein für den werkzeugübergreifenden Einsatz gelegt. TeamPlan wird unter anderem für den Betrieb einer VR-Grossprojektionsanlage sowie eines typischen 2D/3D-Fabrikplanungstisches genutzt. Im letzteren Fall lässt sich die Oberfläche der Software in zwei Bereiche aufteilen, eine orthogonale Ansicht (Draufsicht) zur Bearbeitung des Layouts auf dem Planungstisch sowie einer 3D-Ansicht, die beispielsweise mittels Projektor großflächig dargestellt werden kann (siehe Abbildung 4 links bzw. rechts).

Abbildung 4: Beispielhafte Oberflächenkonfiguration von TeamPlan



Im Folgenden wird die Anwendung der Gesamtplanungslösung am Beispiel einer Restrukturierungsplanung für ein mittelständisches Unternehmen aufgezeigt und dabei insbesondere auf die Möglichkeiten der Modellerweiterung eingegangen. Die Grundvoraussetzung für eine erfolgreiche Projektdurchführung ist die anforderungsgerechte und korrekte Durchführung einer Ist-Aufnahme, die einen erheblichen Aufwandsanteil im Verlauf eines Struktur- und Layoutplanungsprojektes annehmen kann. Der bereits von (Aggteleky, 1982) angegebene Wert von 30–40 % erscheint auch in aktuellen Struktur und Layoutplanungsprojekten realistisch zu sein. Ein ähnlich hoher Gesamtaufwand für die Ist-Aufnahme bzw. Datenerfassung ist in typischen Simulationsprojekten zu beobachten (siehe beispielsweise Fritz, 2007). Der Schwerpunkt liegt hier allerdings stärker auf der genauen Erfassung der Prozessabläufe und weniger auf der Dokumentation der räumlichen Struktur des Produktionsbereiches.

Zur Unterstützung der Ist-Aufnahme wurde daher eine Vorgehensweise konzipiert und umgesetzt, um vor Ort erfasste Daten in ein virtuelles Modell der Produktionseinrichtung integrieren zu können. Da in frühen Projektphasen häufig keine detaillierten 3D-Daten vorliegen, wird dazu zunächst ein auf Stellvertretermodellen basierendes Grobmodell erstellt. Mittels aktueller CAD-Systeme ist dieser vorbereitende Schritt aufwandsarm möglich, indem beispielsweise vorhandene 2D-CAD-Pläne mittels Extrusionsoperationen in ein 3D-Layout überführt werden. Dieses wird anschließend über das JT-Format in eine mobile Version von TeamPlan importiert und steht dort zur Datenerfassung zur Verfügung. Unter Nutzung der Modellerweiterung wurde dazu ein digitales Fabrikobjekt zur Speicherung der Ist-



Abbildung 5: Ist-Aufnahme im virtuellen Modell

Daten entwickelt. Neben Angaben zu Funktion und Dimension der einzelnen Ressourcen (Maschinen, Fördertechnik etc.) können hier unter anderem auch verarbeitete Produktgruppen sowie Vorgänger-/Nachfolgerbeziehungen aufgenommen werden. Dazu wird eine Instanz des Fabrikobjektes einem virtuellen Objekt, beispielsweise einer Maschine, zugeordnet und kann über einen entsprechenden Benutzerdialog befüllt werden (siehe Abbildung 5).

Nach erfolgter Eingabe wird das Fabrikobjekt automatisch in der Objektdatenbank abgelegt und damit dauerhaft gespeichert. Insgesamt steht bei Nutzung der beschriebenen Vorgehensweise bereits in einer frühen Projektphase ein virtuelles Modell mit hinterlegten planungsrelevanten Daten für teambasierte Planungssitzungen zur Verfügung. Auf Grund der freien Konfigurierbarkeit der gewünschten Informationen ist eine Anpassung an weitere Anwendungsfälle möglich, beispielsweise zur Unterstützung bei der Parametrierung von Simulationsmodellen.

Um die Restrukturierung der Produktionsanlage materialflussoptimiert durchführen zu können, verfügt TeamPlan über die Möglichkeit einer wegetreuen Materialfluss- und Prozessvisualisierung im 3D-Layout. In Abbildung 6 ist die Vorgehensweise zur Nutzung dieser Funktionalität dargestellt. Zunächst sind vom Benutzer Marker (sogenannte Wegepunkte) im 3D-Grobmodell zu platzieren und durch Wege zu verbinden (1). Unter Anwendung eines effizienten Wegefindungsalgorithmus kann nun ein kürzester Pfad zwischen zwei gewählten Layoutele-

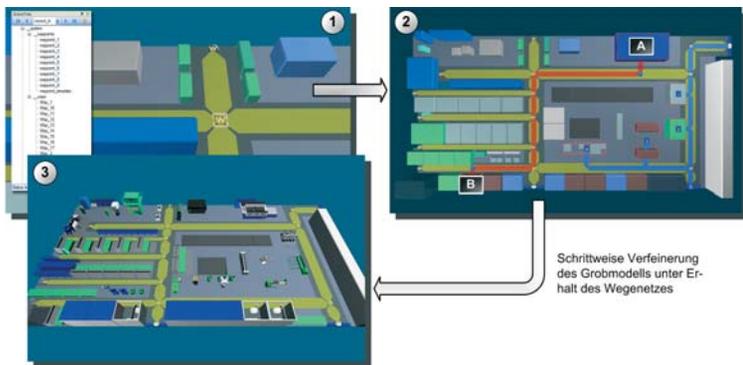


Abbildung 6: Wegetreue Materialfluss- und Prozessvisualisierung

menten bestimmt werden (2, roter Pfad). Diese Funktionalität kann ebenfalls zur Visualisierung von Prozessabläufen einzelner Produktgruppen genutzt werden (2, blauer Pfad). Dazu wird ausgehend von einem gewählten Startelement der jeweils kürzeste Pfad zum nächsten im Prozessverlauf anzusteuern Layoutelement bestimmt und somit sichergestellt, dass die definierte Produktionsfolge bei gleichzeitig optimalen Transportrouten durchlaufen wird. Die umgesetzte Prozessvisualisierung ist statischer Natur, d.h. berücksichtigt weder Fehlersituationen noch zustandsabhängige Verzweigungen. Ist eine dynamische Absicherung der Produktionsprozesse gefordert, sollte daher in jedem Fall ein entsprechendes Modell in einem Ablaufsimulator genutzt werden.

Im vorliegenden Fall werden sowohl das Wegenetz als auch die zur Wegefindung notwendige Funktionalität in Form eines digitalen Fabrikobjekts abgebildet. Aufwändige Eingriffe in das VR-System sowie die Definition von Speicherformaten, beispielsweise für das Wegenetz, entfallen damit. Ausgehend vom Grobmodell kann jetzt eine schrittweise Detaillierung erfolgen, indem einzelne Stellvertretermodelle durch realistische 3D-Modelle ersetzt und die Anordnung der Layoutelemente im Bezug auf Materialfluss und Prozessablauf optimiert werden (3).

Zusammenfassung

Mit dem beschriebenen Ansatz wurde eine Möglichkeit aufgezeigt, um geometriebasierte VR-Modelle durch Informationen und aktive Funktionskomponenten zu erweitern und damit die Implementierung und Anwendung layoutbezogener Planungs- und Visualisierungsverfahren im virtuellen Modell zu vereinfachen. In der technischen Umsetzung werden dazu bestehende Modellelemente mit .NET-Objekten verknüpft, die in einer objektorientierten Datenbank gehalten werden. Durch die beispielhafte Umsetzung einer Vorgehensweise zur Ist-Aufnahme sowie einer wegetreuen Materialfluss- und Prozessvisualisierung im virtuellen Modell konnte der Mehrwert des Ansatzes für die VR-gestützte Struktur- und Layoutplanung belegt werden.

Literatur

- [1] Aggteleky, Béla: Fabrikplanung Band 2 – Werkentwicklung und Betriebsrationalisierung. München Wien: Carl Hanser Verlag, 1982.
- [2] Bracht, Uwe, Schlange, Christian, Eckert, Clemens, Masurat, Thomas: Datenmanagement für die Digitale Fabrik, In: wt Werkstatttechnik online 95 (2005), H. 4, S. 197–204.
- [3] Fritz, Jürgen: Zielführende Modellierung und Analyse schlanker Fertigungssysteme mit der Digitalen Fabrik, Dissertation Universität des Saarlandes, 2007.
- [4] Gamma, Erich, Helm, Richard, Johnson, Ralph, Vlissides, John: Design Patterns (Gang of Four). Longman Amsterdam: Addison Wesley Verlag, 1995.
- [5] Heutschi, Roger: Serviceorientierte Architektur – Architekturprinzipien und Umsetzung in die Praxis. Berlin: Springer-Verlag, 2007.
- [6] Jung, Reinhard: Architekturen zur Datenintegration. Wiesbaden: Deutscher Universitätsverlag (Springer), 2006.
- [7] Klir, George. J.: Facets of Systems Science. Second Edition. Springer US, 2001.
- [8] Lenzerini, Maurizio: Data Integration: A Theoretical Perspective. In: Popa, Lucian (Hrsg.): Proceedings of the Twenty-first ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. New York. ACM Press, 2002, S. 233–246.
- [9] Schlange, Christian (2009): Integrierte Struktur- und Layoutplanung unter Nutzung erweiterter virtueller Fabrikmodelle. Dissertation TU Clausthal, Aachen: Shaker-Verlag, voraussichtlicher Erscheinungstermin: 10/2010.
- [10] Schmigalla, Hans: Fabrikplanung: Begriffe und Zusammenhänge. REFA-Fachbuchreihe Betriebsorganisation. München: Carl Hanser-Verlag (Kommissionsverlag), 1995.
- [11] Siegenthaler, Andreas, Schwinn, Alexander: Modellierung von Integrationsaspekten in Applikationslandschaften. In: Schelp, Joachim, Winter, Robert (Hrsg.): Integrationsmanagement - Planung, Bewertung und Steuerung von Applikationslandschaften. Berlin: Springer-Verlag, 2006, S. 203–230.
- [12] VDI-4499: VDI-Richtlinie 4499 Blatt 1 – Digitale Fabrik – Grundlagen. Berlin: Beuth-Verlag, 2008.

- [13] Wenzel, Sigrid, Jessen, Ulrich: The Integration of 3-D Visualization into the Simulation-based Planning Process of Logistics Systems, In: Simulation 77 (2001), H. 3/4, S. 114–127.
- [14] Wenzel, Sigrid: Die Digitale Fabrik – Ein Konzept für interoperable Modellnutzung. In: Industrie-Management 20 (2004), H. 3, S. 54–58.



3 Energie und Umwelt

Auf die Energieversorgung der Bundesrepublik Deutschland kamen in jüngster Zeit die wohl größten Anforderungen ihrer Geschichte zu. Als Folge der 1998 eingeführten Liberalisierung des Strom- und Gasmarktes entstand ein internationaler Energiehandel mit Energieflüssen, für die die europäischen Netze nicht ausgelegt waren. Zusätzliche Belastungen entstehen in den Stromnetzen durch die politische Neubewertung der Kernenergie-Risiken und den beschlossenen Ausstieg aus dieser Technologie. Die Erkenntnis, dass die Nutzung der fossilen Energiequellen wegen ihrer Begrenztheit und der schädlichen Auswirkung auf die Umwelt nicht im bisherigen Umfang fortgesetzt werden kann, führte zum Energiekonzept der Bundesregierung von 2010, das einen Umstieg von fossilen auf regenerative Energiequellen vorsieht. In der Stromversorgung soll der regenerative Anteil bis 2050 auf 80 % steigen, wobei hier laut der „Leitstudie 2010“ des BMU Fotovoltaik (PV) und Windenergie dominierend sein werden.

PV und Wind gehören zu den nicht beeinflussbaren und stark fluktuierenden Energieformen. Daraus resultieren große Herausforderungen an das zukünftige Energieversorgungssystem: Im elektrischen Netz kann (im Gegensatz zum Gasnetz) keine nennenswerte Energie gespeichert werden, weswegen stets ein Gleichgewicht zwischen Erzeugung und Verbrauch erforderlich ist. Eine wichtige Voraussetzung hierfür ist die zuverlässige Prognose der Wind- und PV-Leistung. Zusätzlich werden Speicherkapazitäten für die Stromversorgung benötigt, die weit über den Möglichkeiten der heute eingesetzten Speichersysteme liegen. Eine Option ist die elektrolytische Erzeugung von Wasserstoff und dessen Einkopplung in das Gasversorgungsnetz. Damit werden die dort vorhandenen großen Speicher, z.B. Kavernenspeicher, für das Stromnetz nutzbar. Das gespeicherte Gas kann in thermischen Kraftwerken oder Brennstoffzellen zur Stromerzeugung genutzt werden. Die im Folgenden vorgestellten Projekte zur Optimierung von Kavernenspeichern, Brennern und Brennstoffzellen erbringen einen wertvollen Beitrag für das zukünftige Energieszenario.

3.1 Long-term behaviour of sealed brine-filled cavities in rock salt mass – A new approach for physical modelling and numerical simulation

Karl-Heinz Lux – Institut für Aufbereitung, Deponietechnik und Geomechanik

Abstract

The load-bearing behaviour of caverns constructed in salt formations to extract minerals or to store energy have been the focus of scientific research for many decades. This research previously concentrated on confirming the stability and safety of the caverns during construction and operation. During the past 10 years, the focus has also partially fallen on the questions involving cavern abandonment and their long-term behaviour. Various options are available for abandoning caverns: the preferred option in most cases is filling the caverns with water / brine and sealing the access borehole. The load-bearing behaviour of fluid-filled closed caverns has therefore been the focus of technical and scientific investigations. This paper presents a new approach to the analysis of the load-bearing behaviour of sealed fluid-filled salt caverns. This primarily concerns the process of brine infiltration into the originally impermeable salt rock, and the physical modelling and numerical simulation of this process [7].

Introduction

Salt caverns have been constructed in many countries around the world for several decades. These salt caverns may be a by-product of the solution mining of halite or specifically planned for the storage of hydrocarbons such as oil and natural gas. The load-bearing behaviour of salt caverns during the construction and operation phases has also been the focus of scientific and technical investigations at a national and international level for many decades, e.g. Lux (2006) [1] and the references contained in this paper. In the case of waste disposal caverns in Germany, recommendations have already been elaborated concerning the stipulated confirmation of stability, tightness and environmental protection, especially long-term safety. This excludes the disposal of liquid waste in salt caverns, because

investigations carried out at the time were unable to rule out fluid-pressure related failure of the rock mass formation.

The scientific and technical investigations concentrate on further developing the principles for physically modelling the geomechanical and geohydraulic processes (improving the understanding of the processes) and the computer simulation (improving reliability of the load-bearing structural analysis). The ultimate aim of this work is to improve the precision of the confirmation required to guarantee static stability and tightness, as well as economic optimisation (e.g. increasing cavern size or extending the operating parameters and operating procedures) by reducing uncertainties or lowering conservative assumptions and safety margins.

Another question which was raised at around the start of the 80s was how to deal with the salt caverns after shutting down leaching operations once the authorised final configuration had been reached (brine extraction) or after shutting down storage operations (oil / natural gas storage). At a national and international level, experts came to the conclusion over the course of time that salt caverns scheduled for abandonment should be filled with water / brine, plugged after a certain undetermined time (dedicated to temperature equalization between fluid and rock mass) with a long-term fluid-tight seal in the borehole, and then left to their own devices. The aspect that requires clarification for the official abandonment operating planning procedure is how a sealed brine-filled cavern behaves in the long-term, e.g. with respect to pressure build-up as a result of rock mass convergence, and the capacity of the salt rock to withstand fluid pressure, especially at levels higher than the rock mass stresses of the surrounding salt rock mass.

State of the art

In Germany the long-term behaviour of sealed brine-filled caverns has been the subject of scientific investigations for many years (a list of references is given in Lux (2006) [1]). The basic question looked at by the theoretical studies during the 80s was the size of the hydraulic pressure built up in a sealed cavern as a result of formation convergence, and whether the fluid pressure causes fracturing in the cavern roof zone. Another ques-

tion concerned the actual size of the frac pressure involved. Some investigations indicated that the rock would frac when the minimum formation pressure was reached. On the other hand, other investigations also indicated that because of the hydraulic tensile strength of the salt frac-failure only occurred when much higher hydraulic pressures were reached.

To provide a final and convincing answer to these controversial thesis which have been the subject of discussions for many years, a consortium of cavern operators decided to supplement the theoretical studies by conducting a large-scale field test from 1991 to 1994 in a representatively selected and properly technically equipped cavern in Germany. This was carried out at the Etzel K102 cavern (Consortium 'Druckaufbautest' 1994) [2]. The aim of this test was to determine the fluid pressure level at which the formation would start to fail and whether this failure involved macrofracturing analogous to the fracturing of the rock during a hydraulic fracturing operation, or whether the loss of formation integrity was due to a different failure mechanism. Other field tests were conducted in France and Germany (see Lux (2006) [1] for a reference list). A summary of the state of developments at the end of the last decade, and the discussions on the abandonment of salt caverns is reported in Crotagino & Schmidt (1999) [3].

The latest field experiments dealing with this topic involved carrying out a pressure build-up test in a cavern-like cavity specially excavated in a salt mine in Germany for this purpose. The cavity had a volume of approx. At the end of the experiment a gallery was driven into the cavern to enable direct visual inspection of the penetration of the tracers brine into the rock immediately surrounding the cavity.

Pressure-driven infiltration as the process mechanism

According to the current level of understanding, liquid and gas tight halite formations under primary formation pressure become permeable under low to negligible pressure build-up rates with brine as a pressure fluid when a fluid pressure level is reached that approximately corresponds to the minimum formation pressure of homogenous and texturally uniform halite. This is a result of the progressive in-filtration of brine through

the micro-migration paths formed as a result of this fluid pressure build-up. The following hypothesis can be formulated for the physical modelling and numerical simulation of this infiltration process when supplementing the SMRI-model that is based on an approach which assumes already slightly permeable halite a priori:

- The undisturbed halite formation is liquid tight and gas tight under primary rock mass pressure conditions.
- Secondary migration paths are formed at a micro and meso level along the grain boundary surfaces in the grain fabric of the salt rock under a specific constellation of conditions (still to be determined in more detail) involving brine pressure and the minimum formation stress.
- The volume-related intensity (= secondary porosity) and propagation rate (= microfrac formation rate) of the secondary migration paths are de-pendant on the rock type and the stress state, and can be estimated, in an integral way at least, from laboratory tests.
- The mean (integral) fracture propagation in the rock mass follows the rules of hydrofrac theory with respect to their general direction when considered at a macroscopic level. The local stress field and the rock fabric have the most significant influence on the main direction of fracture propagation.
- Dispersive fracture propagation is ignored in the first modelling step but should be integrated within the theory at a later date.
- The main parameters determining the progression of the brine front in the salt rock at isothermal conditions are salt rock mass convergence (de-pendent on the creep behaviour of the salt rock, the depth of the cavern and the cavern geometry as well as the formation stress), brine compressibility, secondary porosity, and the difference between the minimum formation stress and fluid pressure at the fracture propagation front.

This physical model approach has two main advantages compared to the concept described in SMRI (2002):

- The basic assumption that the halite rock in an undisturbed state is fluid tight and gas tight under the influence of formation pressure does not have to be abandoned for modelling reasons by specifying an interlinked fluid-filled fabric of pore spaces with primary permeability.
- The secondary more permeable formation zone is not derived from a stress-state-related condition, and its 3-dimensional development is also not only restricted to the immediate periphery of the cavern. Instead, its development in time and space is individually determined as a result of the physical modelling / numerical simulation by the secondary migration paths induced by the brine infiltration in the rock. This involves estimating the secondary permeability from the secondary porosity, which is then used to characterise the hydraulic properties of the infiltrated part of the rock.

Basics of physical modelling and numerical simulation

The computer code INFIL

A computer code for the numerical simulation of the infiltration process was developed based on the observations from laboratory tests (fluid infiltration in halite specimens) and the still only verbally formulated approaches presented here on the start of infiltration and the propagation of micro-fractures as the basis for the physical modelling. This software called INFIL does not just include the program to implement the physical model of a time and stress / fluid pressure controlled infiltration process in a primarily impermeable continuum. It is also the central control program of a software package for the numerical simulation of the geomechanical processes taking place prior to cavern sealing, and particularly, the geomechanical / (geo)hydraulic coupled processes taking place after cavern sealing, via pressure build-up, and infiltration all the way through to the ultimate flow through the secondary migration paths induced by the infiltration process, which can be subdivided into a pressure build-up phase, an infiltration phase and a flow phase.

Pressure build-up phase

The pressure build-up phase means a simulation of the past

history, and then after sealing, a simulation with the MISES3 simulator (FEM-software) of the cavern convergence and brine temperature change related pressure build-up in the brine-filled sealed cavern. If the brine heats up in relation to a temperature gradient between the formation and the brine, the mechanical-hydraulic simulation is superimposed by an independent thermodynamic simulation which quantifies the time-dependent changes in brine temperature and the associated brine expansion and resulting change in brine pressure.

Infiltration phase

Physical modelling and numerical simulation of the infiltration process give rise to specific approaches concerning the start of infiltration, the direction of infiltration, the infiltration rate and the infiltration paths, as well as the infiltration intensity and infiltration propagation.

Start of infiltration

Brine infiltration into the salt rock mass begins when the fluid pressure exceeds the minimum principal stress in a directly fluid-pressurised rock mass element:

$$p_F > \min \sigma. \quad (1)$$

If the fluid pressure is lower than the minimum principal stress, there is no infiltration.

Direction of infiltration

The infiltration direction is derived from the direction in which the secondary migration paths are induced. Applying the hydro fracturing theory to a material with an isotropic fabric implies that fracture propagation takes place normal to the smallest principal stress and therefore vice versa in the direction of the largest principal stress. More complex relationships exist in anisotropic rock fabrics where empirically the migration paths develop preferentially along the bedding or cleavage planes. Additional assumptions must be incorporated in the model in such cases.

Infiltration rate

The infiltration rate is lithology-dependent and must be deter-

mined in the laboratory. Laboratory investigations reveal the dependence of infiltration rates on the hydromechanically effective fluid pressure (stress difference between fluid pressure and minimum formation stress), which is termed $\Delta\sigma_p$. The empirically selected approach is

$$\bar{v}_{\text{inf}} = a \cdot \exp(b \cdot \Delta\sigma_p), \quad (2)$$

where a and b are dependent on the infiltrated rock mass material and the infiltration medium (e.g. tracer fluid, brine or gas).

Infiltration path

The physical model assumes constant conditions / states for time increment Δt . The infiltration path derived from equation (2) is therefore:

$$\Delta u_{\text{inf}} = \bar{v}_{\text{inf}} \cdot \Delta t. \quad (3)$$

Infiltration intensity

The infiltration intensity describes the fluid volume accommodated i.e. stored in the secondary induced pore space within a rock element. The secondary induced pore space is determined in the laboratory and characterised by the secondary porosity ϕ^s . The infiltration volume (storage volume) for a rock element and a time increment Δt is defined as

$$\Delta V_{\text{inf}} = \phi^s \cdot \Delta V_0 = (m \cdot \Delta\sigma_p + c) \cdot \Delta V_0. \quad (4)$$

Infiltration propagation

When infiltration has begun, infiltration continues into the neighbouring rock when the complete infiltration (saturation) of the observed rock element has taken place after a period of time and when, taking into account the changed fluid pressure conditions (infiltration-related decompression, creep and temperature-related compression) as well as the changed formation stress field (creep-related stress relocations), the conditions have been fulfilled for further infiltration into the next rock element adjacent to the originally observed rock element (infiltration conditions).

Flow phase

If the infiltration front reaches a permeable salt solution or

groundwater bearing horizon, the infiltration process changes to a flow process. The flow here follows the secondary migration paths formed during the infiltration phase. The salt rock mass continues to be impermeable away from the infiltration zone.

When modelling the flow process, it is assumed that lamina flow takes place in the brine-saturated interlinked secondary migration paths in the infiltration zone in accordance with the Darcy flow model

$$v_f = k_f \cdot i. \quad (5)$$

The hydraulic conductivity is derived here from the secondary permeability which itself is related to the secondary porosity and the secondary stress field via PoroPerm-Model. Various PoroPerm-Models are known from the literature (a reference list is given in Lux (2006) [1]).

If looked at in more detail, the infiltration zone can be considered as a porous water-saturated medium that mechanically – via the (disturbed) mineral grain fabric – as well as hydraulically – via the pore water pressure – has a supporting effect on the surrounding impermeable rock, and vice versa is accordingly placed under stress as a result of the damage to the grain fabric caused by the creation of secondary migration paths and the pore water pressure. However, it does not have the same mechanical properties as the neighbouring non-infiltrated rock. This change to the stress conditions and material properties of the rock in the infiltration zone (brine-infiltrated halite) needs to be taken into consideration when analysing the load-bearing behaviour, or determined by carrying out suitable laboratory tests.

Some results of laboratory investigations

Laboratory investigations on relatively large specimens measuring $l/d = 30/15$ cm were carried out in test apparatus specially designed and constructed for these tests to determine the infiltration behaviour of halite. The testing principle involves placing the specimens under triaxial stress and simultaneously applying an axial fluid pressure. The fluid pressure is the technical test variable and can be adjusted to lie above or below the lateral (minimal) mechanical stress. The axial stress is usually higher than the fluid pressure. This test constellation leads in theory (in isotropic fabrics) to an axially-aligned propaga-

tion of the infiltration front with respect to the specimens. The horizontal stresses occurring at the end of the specimen at the boundary to the pressure piston of the test apparatus, arising from the blocked lateral extension in addition to the externally applied lateral stresses, are estimated to be minor. Fluid infiltration takes place via an artificially created slot or small drill-hole a few centimetres long to simulate an imperfection in the fabric of the fluid-pressure stressed rock surface. The mechanical stresses, i.e. the axial pressure σ_1 and the lateral pressure $\sigma_{2,3}$ ($\equiv \min \sigma$) as well as the fluid pressure p_{Fi} , are measured during the test. The fluid pressure p_{Fi} is kept constant by adding pressure liquid if infiltration takes place. The associated fluid volume V_{inf} , which corresponds to the infiltration volume, is also measured.

The test finishes at the end of a defined test or infiltration time, or if a fluid breakthrough and an approximately stationary flow has been reached. The specimen is dissected at the end of the test. Because the pressure fluid contains a tracer, the infiltrated zones can be visualised under UV-light. Figure 1 shows a typical infiltration test result.

Representative measurement data of an infiltration test are shown in Figure 2. The diagram shows the mechanical and hydraulic pressures adjusted to remain constant, as well as the amount of fluid injected into the specimen to maintain a con-

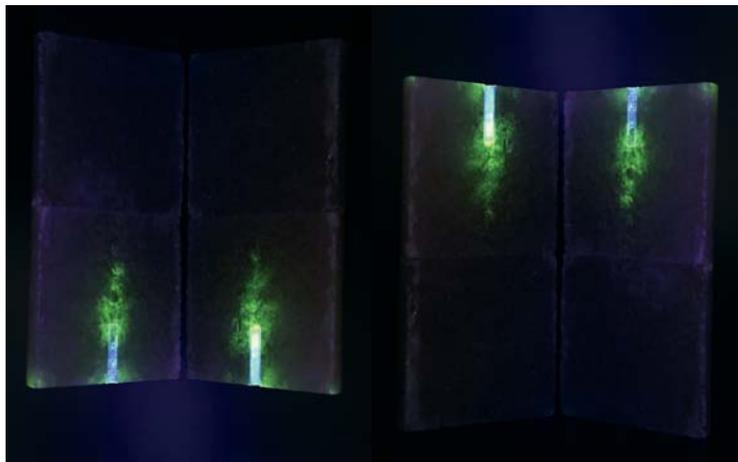


Figure 1: Formation of the infiltration zone in a sample (laboratory test, left figure) and according to the circumstances in a cavern (right figure)

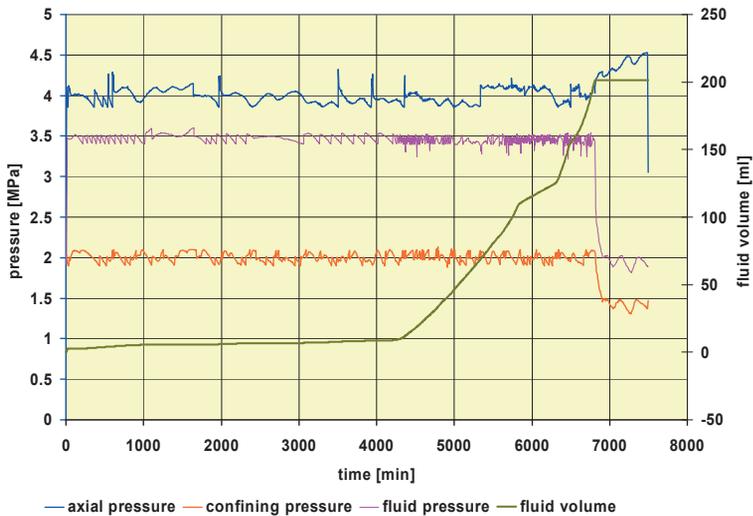


Figure 2: Measurement data of an infiltration test

stant fluid pressure. This fluid volume therefore corresponds to the volume of fluid infiltrated into the specimen and therefore also corresponds to the infiltration volume.

The tests reveal that the injection or infiltration volume plotted against time can be divided into two periods characterised by approximately constant but very different infiltration (= injection) rates. In the interpretation, the first part of the curve with the much lower rate is seen as corresponding to the actual infiltration process: driven by the pressure which is adjusted to remain constant, the fluid creates inter-granular migration paths in the crystal grain fabric when it reaches an adequately high pressure level compared to the lateral mechanical (minimum) stress. The fluid penetrates the inter-granular migration paths which are then propagated as the fluid moves forward. This gives rise to an inter-linked fabric of microfractures whose main propagation direction corresponds to the direction of the largest main stress. The microfracture fabric also propagates laterally at the same time although to a subordinate degree. This phase of the test is called the infiltration phase.

The infiltration phase ends relatively abruptly – clearly shown in the measurement data by the almost instant significant rise in injection rate at a specific point in time. This time is inter-

preted to correspond to the hydraulic breakthrough of the fluid and the creation of a hydraulic connection between two reservoirs with different fluid pressures. The second part of the curve with the higher injection rate is therefore interpreted as corresponding to the flow process. The flow paths are the secondary migration paths created during the infiltration process. Because this flow process continues in principle at a constant flow rate and under otherwise constant conditions, it can be assumed that no new migration paths are created during this period despite the change in pore pressure conditions corresponding to the hydraulic potential field.

First basic numerical simulations and results

Based on the proposed physical model a computer program has been developed for numerical simulation of cavern behaviour. Different phases have to be considered: pressure build-up phase, infiltration phase and flow phase. This computer code called INFIL is described in detail in Wolters (2005) [4].

In one of the first numerical simulations, a typical salt cavern from a larger cavern field is considered which has a cavern roof at a depth $z_D = -1115\text{m}$, a cavern sump at a depth $z_S = -1315\text{m}$ and therefore a cavern height of $h = 200\text{m}$, as well as a cavern diameter of $d = 80\text{m}$. Top salt lies at a depth of $z = -715\text{m}$. Top salt is overlain by a porous water saturated formation.

To model and simulate the load-bearing behaviour of the shut-down cavern, it is initially assumed that sufficient time has elapsed for temperature equalisation to take place between the rock and the brine between the flooding of the cavern with water / brine and the installation of the cavern seal – making it possible initially to ignore the thermally-related increases in pressure. For the numerical simulation, the pressure build-up in this case starts after installation of the fluid-tight cavern seal under isothermal conditions and a hydrostatic internal pressure level ($\gamma_{\text{Fl}} = 0.012 \text{ MN/m}^3$). The influence of thermally-induced pressure increases in the context of the modelling approach presented here is discussed in Lux (2006) [1].

The analysis at first looks at the geohydraulic / geomechanical processes taking place, and the load-bearing behaviour of the

rock, taking into consideration a variable rock stress state during the infiltration process reflecting internal cavern pressure and time. The point in time at which infiltration begins arises automatically from the first fulfilment of the infiltration conditions in the calculation zone in accordance with equation (1).

For the defined rock model with a homogenous structure and isotropic material properties, it can be assumed empirically that the infiltration process in principle follows the line of decreasing lithostatic depth pressure, and spreads upwards from the cavern roof through the overlying salt in the direction of top salt.

Pursuant to the modelling approach, the infiltration process changes to a flow process after reaching a permeable horizon. As a consequence, analysis of the infiltration process is followed by analysis of the subsequent flow process – modelled in a simplified form as lamina and stationary Darcy flow over the defined time increment.

The following figures show some selected typical results of the numerical simulation:

- Figure 3 firstly shows that around 30 years pass in this example until the internal pressure in the cavern has built up to the necessary level required to initiate the infiltration process. This internal pressure is then maintained at a fairly constant high level for a relatively long period of around 100

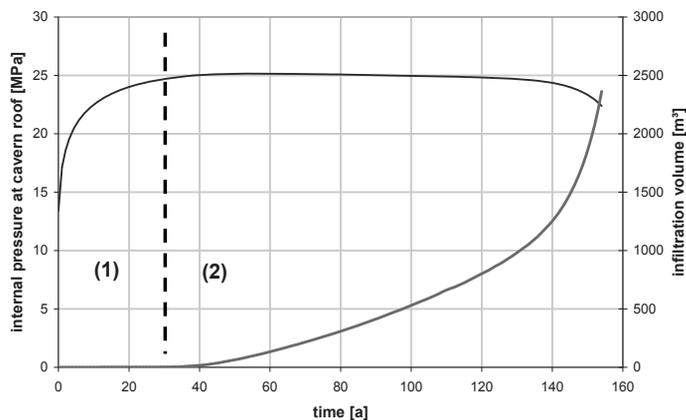


Figure 3: Time-dependent development of internal pressure at cavern roof and of infiltrated brine volume with (1) pressure increasing phase and (2) infiltration phase until hydraulic break-through

years with respect to the duration of the infiltration process of around 125 years in total. Figure 3 also shows the volume of brine infiltrated into the rock salt during this process. The strong rise in the last two decades prior to hydraulic breakthrough is attributable to the increasing infiltration rate. This is itself a reflection of the de-creasing depth pressure.

- Figure 4 shows the corresponding calculated propagation of the infiltration zone at different times during the infiltration process. This reveals that the infiltration front only develops very slowly at the beginning in the zone close to the cavern but then spreads relatively fast later on in the more distal locations from the cavern. The continuously high internal pressure during the in-filtration phase gives rise to a period of only 125 years from the start of infiltration to hydraulic breakthrough. Assuming adequately reliable modelling / simulation, this reveals that the time period during which the load-bearing system of the sealed cavern and the surrounding halite formation could be sustainably altered by the infiltration process, and its transformation to a flow process along the secondary migration paths created by infiltration, are of an order of magnitude suitable for engineering treatment. Indeed several factors could enlarge this time spans significantly: lower creep rates than modelled at low deviatoric stresses, a creep limit, capillary forces between fluid and salt rock crystal surfaces in the micro-pathways.

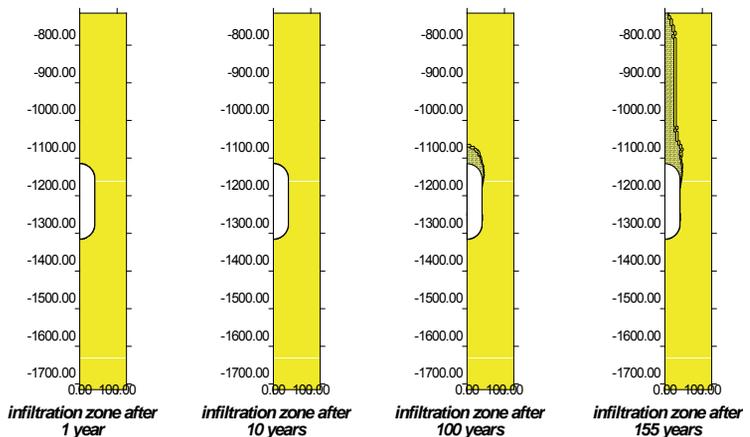


Figure 4: Propagation of the infiltration front at different points in time

- Following hydraulic breakthrough into a porous water-saturated formation a stationary Darcy flow takes place after the infiltration process in this simulation example, the volume flow of the brine extruded as a result of rock convergence is $Q \approx 48 \text{ m}^3/\text{a}$ corresponding to a convergence rate of $K \approx 0.005 \text{ \%}/\text{a}$. The convergence of a sealed cavern after breakthrough and hydraulic connection is significantly dependent on the transmissivity of the infiltration zone and the creep behaviour of the surrounding salt under the relevant stress conditions. It is important to point out again here the key influence of the location-specific rock mass convergence – i.e. convergence under the conditions affecting the sealed cavern and therefore the very low deviator stresses – on the long-term load-bearing behaviour of the cavern, and the effects on the surrounding geology and the ground surface.

In the long term context, the final flow phase is of crucial importance to the load-bearing behaviour of the load-bearing system formed by the salt rock mass and the sealed fluid-filled cavern, as well as its effects at the ground surface.

To demonstrate the general capability of the IN-FIL-code, the so-called Etzel Field Test has been re-analysed.

Etzel field test and back-analysis

Test procedure and basic results

The Etzel field test refers to a pressure build-up test carried out between 1991 and 1994 at cavern K102 in the Etzel cavern field. This test is used here as the first validation of the INFIL software. Figure 5 shows the cavern position and the geometry of the cavern. The test cavern is a relatively slim but extremely high cavern with a height of $h = 662\text{m}$. The pressure build-up test was carried out to determine whether, and if yes, how, and at which pressure level the rock salt surrounding the cavern loses its integrity, i.e. its fluid tightness, during a slow increase in internal pressure. The conditions during the pressure build-up test, and in particular the time involved, were therefore to correspond as closely as possible to the behaviour of a sealed cavern (low pressure build-up rate, long test duration) also taking into consideration economic factors.

Figure 6 shows the test data and specifically the change in internal pressure against time (brine-side head pressure). The pressure build-up phase and the pressure reduction phase are marked by sections in which the fluid pressure changes as a result of brine injection and brine extraction, and other sections where the fluid pressure remains roughly constant when no brine is injected or extracted, so the test can be subdivided into injection and extraction phases as well as observation phases.

Based on the understanding at the time (planning for the test began around the middle of the 80s), two different types of potential failure were looked at, namely

- the loss of formation integrity (tightness) as a result of the formation of macrofracs at fluid pressure gradients considerably higher than the primary formation pressure gradient (an additional liner was installed to ensure that the system was technically capable of testing this condition) or alternatively

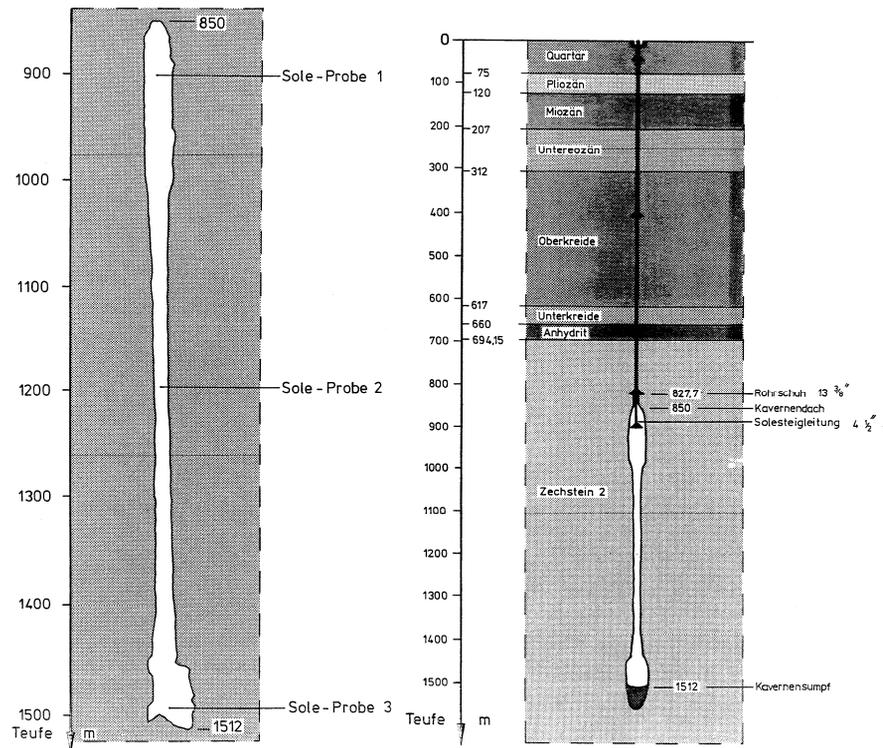


Figure 5: Cavern Etzel K102

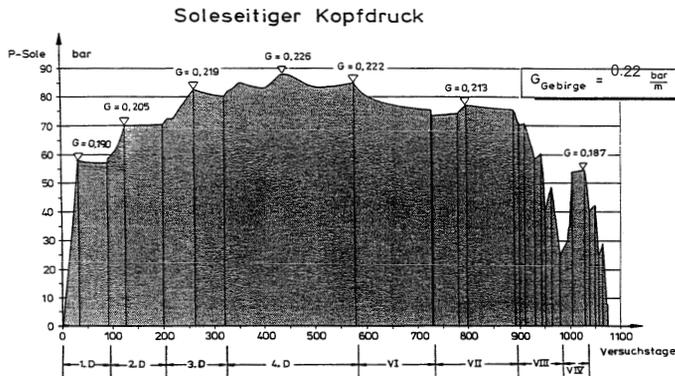


Figure 6: Time-dependent development of fluid pressure

- the loss of formation integrity as a result of gradual microfracture creation at fluid pressure gradients at around the same level as the primary formation pressure gradients.

The test curve in Figure 6 (time-dependent fluid pressure) shows that during the first three test phases (pressure build-up and observation phases) up to the achievement of fluid pressure gradients of $p'_F = 0.219 \text{ bar/m}$ at the pre-defined brine injection rate, although the pressure build-up intensity decreases there are still overall clear increases in brine pressure. The first significant drop in pressure occurred during the observation phase after stopping brine injection during the third test phase. This indicates brine infiltration of the rock. During the subsequent fourth test phase, it was still possible to achieve another increase in pressure to a gradient of $p'_F = 0.226 \text{ bar/m}$ only by doubling the injection rate for a much longer injection time and therefore at a significantly lower pressure build-up rate. After this point, the fluid pressure drops to a gradient of $p'_F = 0.222 \text{ bar/m}$ despite continuing injection. During the subsequent observation phase, the pressure continued to drop continuously. This was taken as clear evidence that the fluid pressure strength level of the rock had been reached or clearly exceeded during this test. However, with respect to the type of failure, instead of the formation of singular macrofracs (also not excluded a priori) the failure took the form of fluid-pressure-induced zonal secondary migration paths consisting of microfissures.

The key findings of the field experiment were therefore:

- The failure of the rock (loss of integrity) during a slow increase in pressure does not take place as a result of spontaneously formed local macrofractures but rather as a result of gradual zonal infiltration in fluid-pressure-induced microfissures (= formation of a linked secondary micro migration path system).
- The fluid pressure level during failure corresponds roughly to the primary pressure depth level.
- 1257.2 m³ of injected brine remained in the formation after returning the pressure to the initial level.

Comments on modelling and numerical simulation

The calculation model with the discretisation of the cavern K102 is shown in Figure 7. This reveals that the calculation model corresponds relatively closely to the cavern roof zone as well as the other cavern configuration. The salt formation is modelled as being homogenous and isotropic. The Lubby2 elastic-viscous

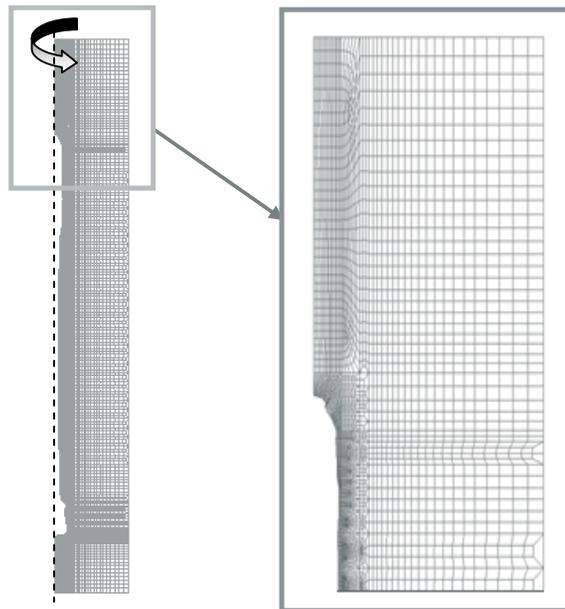


Figure 7: FEM-model of cavern K102

constitutive model is used to model the creep behaviour. The material parameters correspond to the parameters for the salt deposit derived from laboratory tests. When modelling / simulating the infiltration process, the material parameters are set as in the preceding calculation example because of the absence of location-specific data.

The numerical simulation of the field test covers a period of 723 days. As shown in Figure 6, the fourth observation phase ends at the end of this period. The pressure is then reduced in several stages, but the process of pressure reduction is not part of the simulations made here.

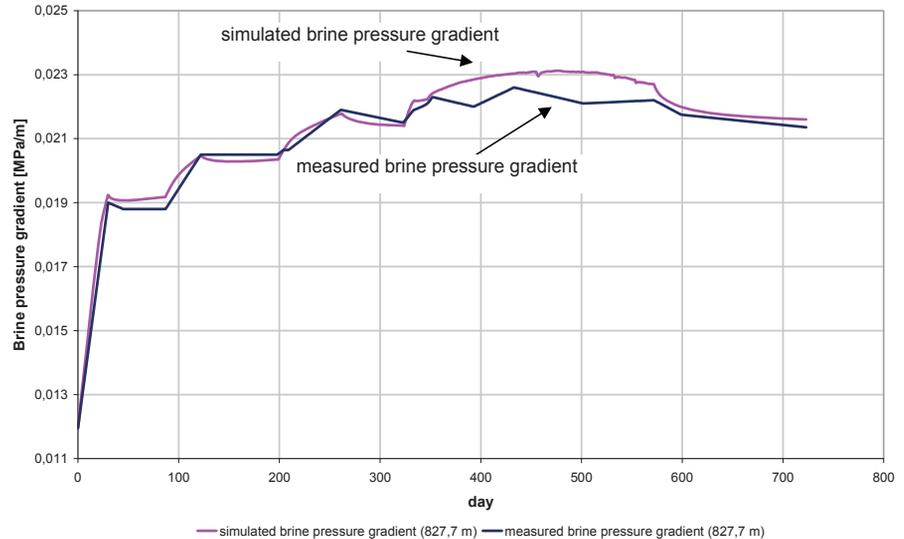
Calculation results

The start and / or continuation of infiltration is decided as part of the infiltration model used here with the help of the criterion equation (1).

A detailed look at the way the internal pressure builds up during the first two pressure build-up phases actually shows that the first pressure build-up phase takes place at a constant pressure build-up rate under constant injection rate conditions (linear pressure build-up). During the first observation phase, the pressure level already reached may decline marginally, but basically remains constant (microfissures at cavern wall?). The second pressure build-up phase with a lower injection rate initially shows a correspondingly smaller pressure build-up rate. By the end of the second pressure build-up phase however the pressure build-up rate grows much more strongly even though the injection rate remains constant. During the subsequent observation phase, the fluid pressure remains constant over a period of approx. 75 days until the start of the third pressure build-up phase.

This development of fluid pressure build-up is not proportional to the externally applied injection rate but appears to be irregular, and can only be explained when incorporating the secondary stress field combined with the development of the infiltration front and possibly also to an inhomogeneous and anisotropic rock fabric.

Figure 8: Time-dependent development of brine pressure gradient (measured and simulated)



14 parameter variations were calculated to perform a sensitivity analysis. The sensitivity analysis procedure involved varying only one parameter at a time in each case to determine its precise effect on the pressure build-up, infiltration volume and infiltration zone parameters. Note that assumed dependencies were sometimes confirmed by the numerical simulation but sometimes also refuted. Renewed analysis of the reaction of the load-bearing system to the variation in parameters then provided new information and a better understanding of the process.

Regarding the calculation result which produced the best match between the calculation data and the measurement data, it can therefore be concluded that given the extent of abstraction, the approximate nature of the model approach, and the numerous parameters (and in particular when also considering the parameters not varied but treated instead as determined) the surprisingly good match between the calculation data and the measurement data according to Figure 8 was achieved at an unexpectedly high level of quality and sensitivity in its response to the changes in the test conditions (injection rate).

Figure 9 shows that at calculation time $t = 566d$ the infiltration front has expanded considerably further into the rock mass

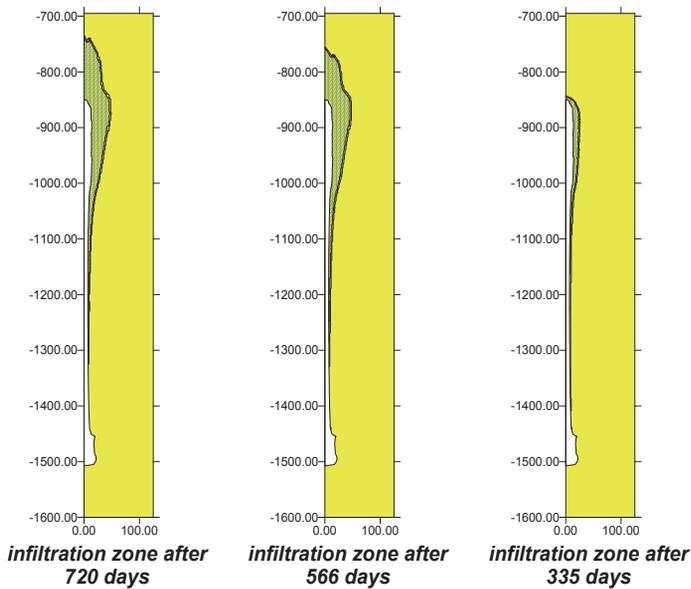


Figure 9: Propagation of infiltration front in time and space

both in the direction of top salt as well as laterally at the depth of the lower cavern roof zone:

however, at the end of the pressure build-up test it only reaches a depth of $z = -760\text{m}$ and therefore extends around 90m from the cavern into the overlying rock, and is therefore still approx. 60m below top salt.

The subsequent observation phase only leads to a further small extension of the infiltration front in the direction of top salt because in the absence of brine injection, the progression of the infiltration front leads to a reduction in fluid pressure and thus a lowering of the effective fluid pressure.

Consequences from the analysis of the load-bearing behaviour of sealed fluid-filled salt caverns with respect to abandonment feasibility

The numerical simulations of the long-term load-bearing behaviour of fluid-filled sealed salt caverns on the basis of the modelling approach in line with the infiltration hypothesis presented here graphically show

- the time-dependent course of fluid infiltration within the rock,
whereby
- the lateral propagation of the infiltration zone and the development of secondary permeability are spatially restricted to a narrowing zone of salt rock mass extending from the cavern roof in the direction of top salt and
- the salt rock mass lying outside of this infiltration zone still retains its primary permeability and can therefore be considered to be impermeable.

The numerical simulations also enable further quantification of the mechanical and hydraulic variables used to analyse and evaluate the load-bearing behaviour of the rock taking into con-

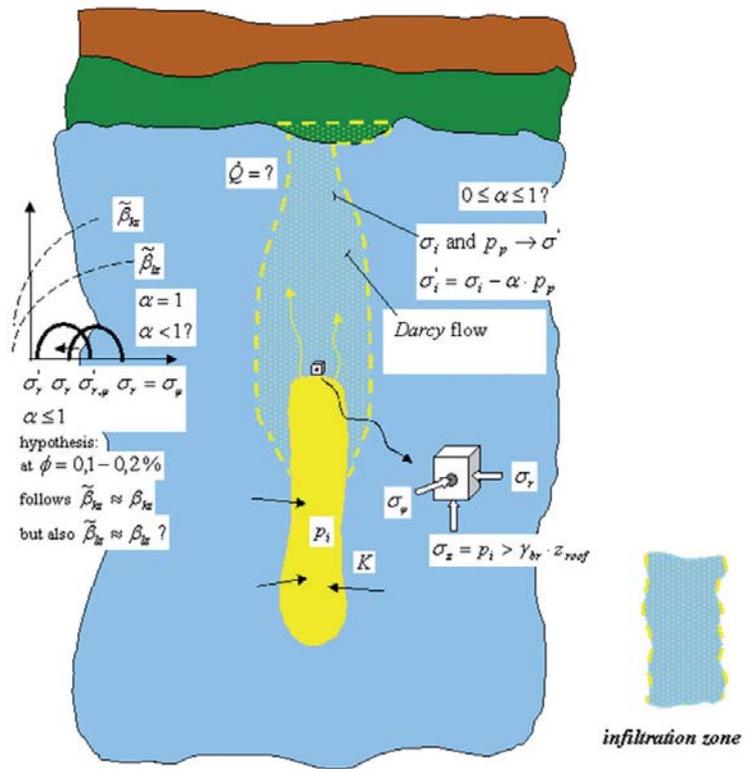


Figure 10: Questions concerning the long-term behaviour of sealed liquid-filled salt cavities, especially after breakthrough of fluid into the non-saline cap rock

sideration stability, integrity and third party protection (surface protection, environmental protection).

Figure 10 provides an initial summary of the questions which arise from the identified load-bearing behaviour of the rock mass under the influence of hydromechanically coupled processes.

In this context, the focus is less on the specific numerical values of each of the variables, but on the resulting proof or observation aspects.

The following questions were identified concerning proof aspects within the context of the post-operational abandonment of a fluid-filled and sealed cavern in creepable salt rock taking into consideration stability, integrity and third-party protection. The questions include specification of the location-specific conditions in each case but do not necessarily include all of the potential questions:

- Does an infiltration zone develop which propagates progressively until it reaches a permeable formation?
- Do all rock salt facies behave more or less the same or are there significant principle differences regarding the process of fluid infiltration?
- How much time does the infiltration process take until hydraulic breakthrough?
- How do the mechanical properties of the rock change in the infiltration zone?
- Which long-term hydraulic situation establishes itself after hydraulic breakthrough (pore pressure, internal cavern pressure)?
- Does more pressure-induced fracturing of the rock take place after hydraulic breakthrough?
- Is the load-bearing capacity of the infiltrated salt rock mass above the roof of the cavern / infiltration zone guaranteed in the long-term (pore pressure level and internal pres-

sure level, mechanically-effective pore pressure, mechanical stress) or is there a risk of progressive collapse of the roof of the cavern with a tendency to develop a collapse cone?

- Can the released cavern fluid (brine, storage medium residues) be discharged in an environmentally compatible way into the (deep) groundwater?
- Do the results have an influence on the specifications for the construction of the borehole plug?
- How long does it take before final convergence is reached?
- Is the surface subsidence rate and the final amount of surface subsidence acceptable?
- Is there a long-term risk of collapse up to the ground surface?

With respect to cavern abandonment, this clearly shows that in addition to the construction of the borehole plug, there are still some scientific and technical aspects which require clarification from a rock-mechanical point of view. Work on these aspects is to be carried out in the near future. It also clearly shows that after shutting down operations, an observation phase of some years should first be planned in prior to sealing the cavern.

References

- [1] Lux, K.-H. 2006. Zum langfristigen Tragverhalten von verschlossenen solegefüllten Salzkavernen – ein neuer Ansatz zu physikalischer Modellierung und numerischer Simulation. EEK, 121 Jg. Heft 11 / 122 Jg. Heft 4 / 122 Jg. Heft 11.
- [2] Consortium 'Druckaufbautest' 1994. Druckaufbautest in der Kaverne K102. Research Project Report (unpublished).
- [3] Crotogino, F. & Schmidt, U. 1999. Endverwahrung von Salzkavernen – Stand der Entwicklung. Forschung, Technik und Innovation, Preussag – Interne Veröffentlichungen, H. 23.
- [4] Wolters, R. 2005: INFIL – Ein EDV-Programm zur numerischen Simulation des Infiltrationsprozesses von fluiden Phasen in impermeable Salzgesteinsformationen. Handbook, TU Clausthal, Professur für Deponietechnik und Geomechanik (unpublished).

- [5] Lux, K.-H., Rokahr, R.B. & Kiersten, P. 1986. Gebirgsmechanische Anforderungen an die untertägige Deponierung von Sonderabfällen im Salzgebirge. Köln: Forschung und Praxis U-Verkehr und unterirdisches Bauen, STUVA.
- [6] SMRI – Rokahr, R.B. et al, Crotogino, F. & Rolfs, O. 2002. High Pressure Cavern Analysis. Solution Mining Research Institute, Research Project Report (unpublished).
- [7] Lux, K.-H. 2007: Long-term behaviour of sealed brine-filled cavities in rock salt mass – A new approach for physical modelling and numerical simulation, in: The mechanical Behaviour of salt – Understanding of THMC Processes in Salt, Taylor & Francis Group, London, UK.

3.2 Ecological evaluation of the pulverized coal combustion in HTAC technology

Natalia Schaffel-Mancini – Institut für Energieverfahrenstechnik und Brennstofftechnik

Summary

The Ph.D. thesis: **Ecological evaluation of the pulverized coal combustion in HTAC technology** was realized in the frame of the agreement between Silesian University of Technology and Clausthal University of Technology for Ph.D. projects.

State of the art

Combustion technology provides more than 90 % of our world-wide energy demand. Severe environmental requirements and international agreements on reduction of pollutants emissions (CO_2 , CO, NO_x , soot, particles etc.) raise a continuous demand for improved combustion technologies [1, 2, 3, 4]. Application of HTAC technology to boilers fired with pulverized coal could be one of the future coal combustion technologies for the clean power generation. High Temperature Air Combustion (HTAC or HiTAC) [5] is named also as: Excess Enthalpy Combustion (EEC) [6], FLameless OXidation (acronym FLOX) [7], MILD (Moderate and Intensive Low-oxygen Dilution) combustion [10]. The most important feature of HTAC technology is an existence of an intense recirculation of combustion products inside the chamber. This recirculation causes that both the combustion air stream and the fuel stream being diluted before the ignition occurs. Therefore, the temperature peaks are suppressed. The second result of these specific mixing conditions is homogeneity of both the temperature and the species concentrations fields. Consequently, HTAC technology features low NO_x and CO emissions and high and uniform heat fluxes. So far, HTAC technology was implemented mainly in industrial furnaces fired either with gaseous fuels or light oils [11]. In most of industrial applications, the technology is combined with heat recovery systems and such a combination typically results in

substantial fuel savings. In this work, the application of HTAC technology in power boilers fired with pulverized coal has been investigated. The following advantages are expected:

- In HTAC furnaces, the radiative heat fluxes are significantly higher than in traditional ones. If the same could be achieved in boilers, then the size of the heat transfer area, and thus boiler investment costs, could be significantly reduced. The boiler could be build using high temperature alloys allowing an increase of superheated steam parameters to an ultra-supercritical level and, subsequently, an increase of the thermal efficiency of the entire cycle.
- The next advantage is in significant reduction of pollutants emissions, mainly nitric oxide which allows eliminating expensive flue gas cleaning installations. Summarizing, application of HTAC technology to boilers fired with pulverized coal could be one of the future coal combustion technologies for the clean power generation. Technical and ecological aspects of such an application are analyzed and discussed in this report.

Research task

The main objective of this work is to investigate applicability of HTAC technology to power station boilers fired with pulverized coal for environmental friendly electricity production. In order to achieve this goal, several technical objectives have been formulated.

The first objective is to examine how accurately HTAC combustion of coal can be predicted using numerical modeling methods. To this end the mathematical model has been validated against the IFRF measurements. The CFD-based simulations have been performed using FLUENT code. The mathematical model selected in the validation and verification process is then used in all subsequent investigations.

The second objective is to develop a conceptual design of a pulverized coal fired boiler utilizing HTAC technology. This involves determination of the combustion chamber shape, its dimensions, distances between individual burners and positions of the burner block.

The third objective involves the examination of the environmental aspects of the HTAC technology implementation. Here one focuses on NO_x , CO, unburned hydrocarbons and char burnout.

The fourth objective is to examine the HTAC boiler operation under different operating conditions like: low combustion air temperatures, low combustion air jet momentum and a low excess air ratio. The fifth objective is to investigate whole steam cycle in order to estimate the efficiency of electricity production using such a HTAC boiler.

Model validation

In this part of the work, the mathematical model describing coal combustion in HTAC technology has been validated against the data generated during an IFRF experiment called HTAC 99 [12]. Several sub-models have been tested during the validation and verification procedure and as a result the following sub-models have been selected:

- $k - \epsilon$ model for turbulence
- Eddy Dissipation Model for interaction between chemistry and turbulence
- discrete phase model for description of particles behavior
- CPD model for devolatilization process
- Intrinsic model for char burnout
- Discrete Ordinates model for radiation
- nitric oxides formation and destruction considering the fuel, thermal, prompt and N_2O paths, as well as NO reburning in the gaseous phase and on the char surface

It was concluded that the predicted values are, generally, in very good accordance with the measurements. Therefore, such a validated model has then been used in the boiler design studies. For more details about the validation procedure see in [13].

Design of the HTAC boiler

HTAC combustion process is very different from conventional combustion. Therefore there is a need for an innovative design of boilers utilizing this technology. In this part of the work several particular boiler concepts have been analyzed in the con-

text of the following three key points: existence of an intensive in-furnace recirculation, homogeneity of both the temperature and the chemical species fields, and uniformity of heat fluxes. Several simulations have been performed in order to find the shape of the boiler and its dimensions, to optimize both the distance between burners and location of the burner block. For more details about the boiler design procedure see in [14].

Final HTAC boiler design

Based on the previous investigations, the final configuration of the HTAC boiler was selected and it is presented in Fig. 1. The boiler is 13 m high and has a 7 m times 6 m cross section. It is equipped with a burner block that consists of 5 identical burners located at the top wall thus the boiler is a down-fired one. The flue gas outlets are also located at the top wall and they are symmetrically positioned on both sides of the burner block. The outlets have a quadrate form with lateral length of 1 m. Each of the five burners is equipped with a central injector of hot air and two coal guns positioned on both sides from the air injector. Pulverized coal is introduced into the furnace by nozzles of 15 mm diameter and the combustion air by 48 mm nozzles. The boiler is equipped with two ash hoppers.

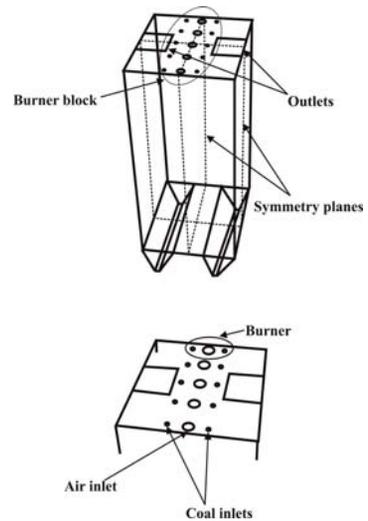


Figure 1: Final geometry of the HTAC boiler

The combustion air is preheated to 1200 K and the coal together with its transport air is supplied at ambient temperature (300 K). The feeding rate of coal is of 3.2 kg/s, and of its transport air almost twice a high. The mass flow of combustion air is equal to 33.1 kg/s. The air jet is supplied at a high velocity (120 m/s) and coal jet has the velocity of 30 m/s. The boiler is operated at 130 MW total thermal input. The fuel thermal input is equal to 100 MW so each burner operates at 20 MW fuel power. Both the combustion and the transport air streams contain 23% mass oxygen and 77% mass nitrogen. The wall temperature is constant in the final boiler design calculations and it is equal to 800 K. The final boiler design leads to the intensive recirculation and the dead zones are small, as can be observed in Fig. 2. The whole volume of the chamber participates in the combustion process. The internal recirculation of the combustion products creates homogenous both the temperature and the chemical species concentration fields. Further, due to dilution of the combustion air and fuel jets the coal ignition takes

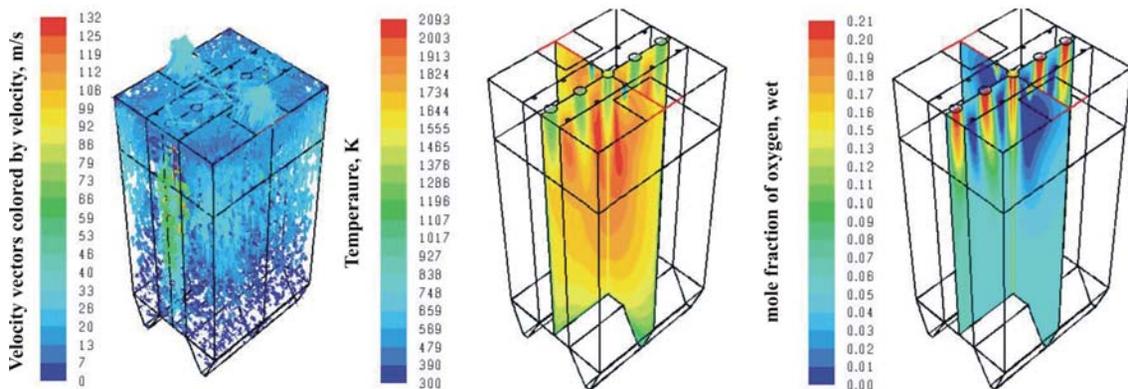


Figure 2: Velocity vectors (left), temperature field (centre) and oxygen concentration field (right) inside the HTAC boiler

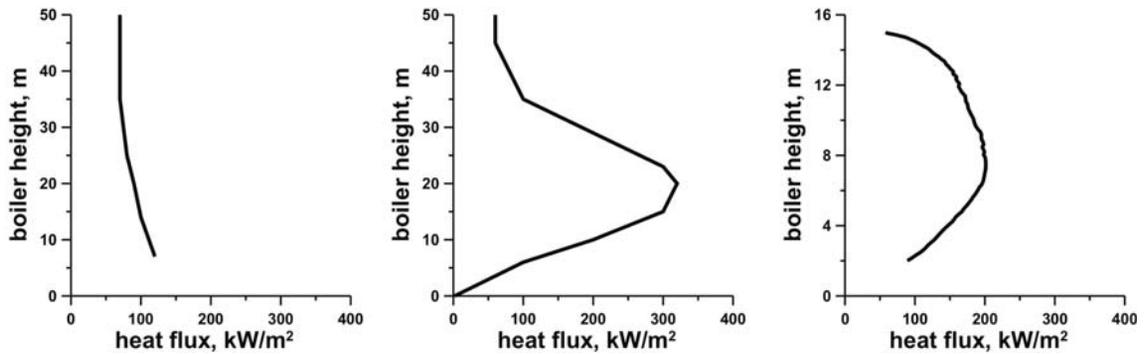
place in low oxygen concentration environment, and therefore the temperature peak is suppressed. Strong recirculation of the hot combustion gas eliminates the problems related to the coal ignition.

The whole boiler is filled up with combustion products of 1600-2000K temperatures.

The temperature peak is substantially suppressed in comparison with conventional combustion using such high air preheat, and it is about 2100 K. The furnace exit temperature is around 1400 K, and the furnace exhaust gas has a high enthalpy. This enthalpy must be recovered in a heat exchanger and utilized to preheat the combustion air.

The oxygen concentration in almost the entire boiler is in a range of 3–5 % while in the flue gas is equal to 3.4 %. Uniform and low oxygen concentrations result in the temperature peak reduction.

As a result of the strong recirculation inside the combustion chamber and as a consequence of the uniform temperature field, the heat fluxes are high and almost constant along the height of the HTAC boiler (see Fig. 3, right). For the sake of comparison typical heat flux profiles for fluidized bed boilers (see Fig. 3, left), and conventional wall fired boilers (see Fig. 3, centre). The HTAC boiler has two advantages: uniform heat fluxes along the boiler height (as in fluidized bed boilers) and high heat fluxes values (as in wall fired pulverized coal boilers).



Heat transfer due to radiation plays a dominant role. Its share is 83% of the total heat transfer. The rest is due to the convection.

For more details about the final boiler design see in [14].

Environmental issues

Most of NO is generated in the region between the burners, as can be seen in Fig. 4, left. The NO concentration peak is equal to 1195 ppm. Downstream of this region the nitric oxide concentrations are low and they are in a range from 300 up to 400 ppm. In the HTAC boiler, 98 % of nitric oxide is formed via fuel mechanism. The NO reburning mechanism plays an important role. As a result, the nitric oxide concentrations at the boiler outlet are low and equal to 298 ppm.

The long particles residence time and recursive recirculation of the combustion products improve the burnout of the CO and

Figure 3: Heat flux along the height of a fluidized bed (left) boiler [15], conventional pulverized coal (centre) boiler [15], and the simulated boiler (right)

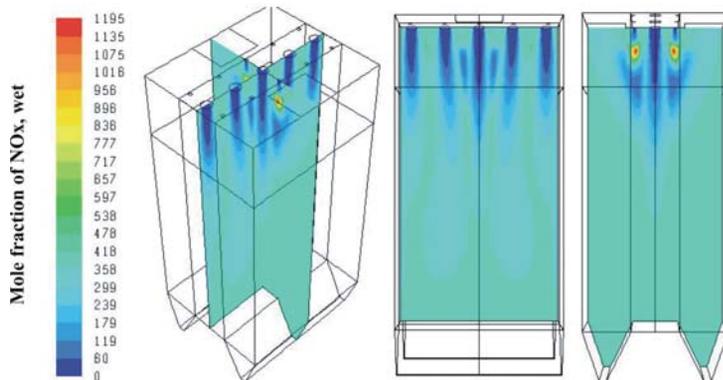


Figure 4: Concentrations of nitric oxide (left); devolatilization (centre) and char burnout (right) regions inside the HTAC boiler

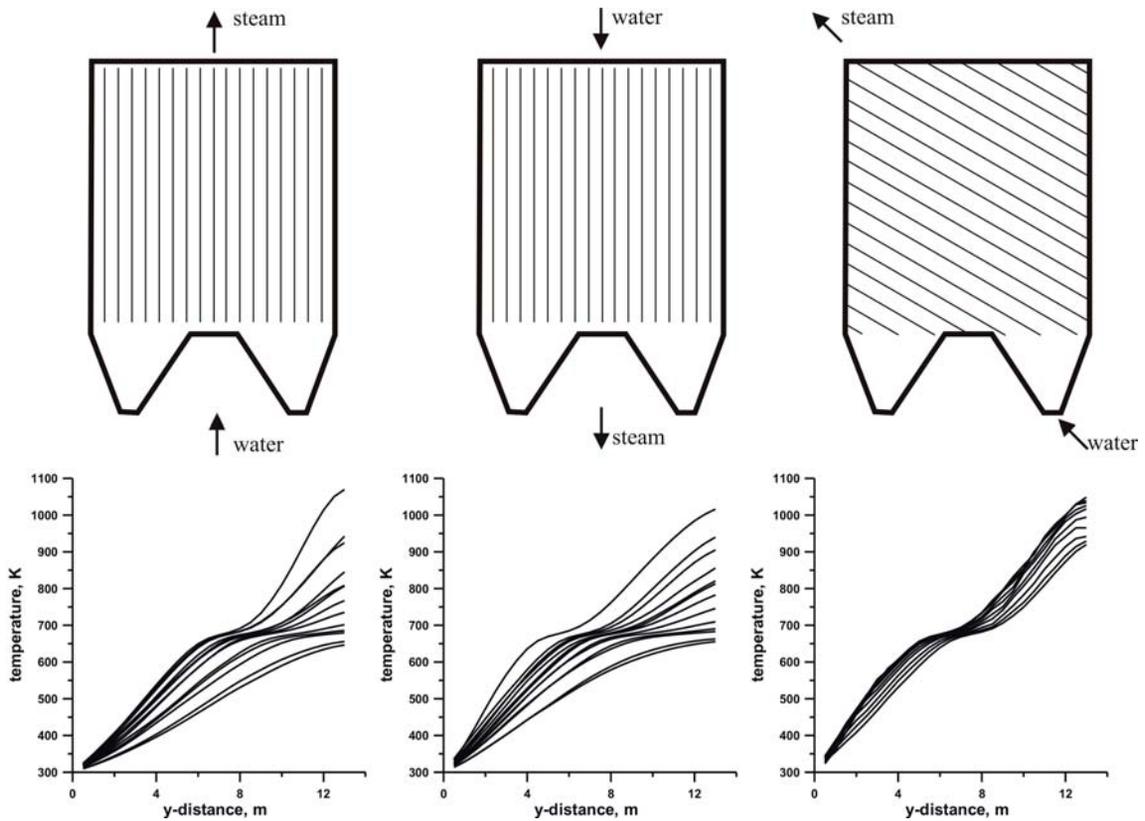
volatiles (see Fig. 4, centre) and of the char (see Fig. 4, right). Very stable combustion process and simple burner construction offer the possibility of using low rank coals.

Effects of selected operating parameters

In the next part of the work, an impact of three important parameters: the combustion air preheat, the combustion air jet velocity and the air excess ratio on the HTAC boiler performance have been tested. Thus, for all calculations the final HTAC boiler geometry is retained and the same boundary conditions are applied. However, boiler operating conditions are different and these are specified for each computational run. The final boiler design is the reference case. It was concluded that the HTAC technology in boilers can work correctly at different levels of the air preheating, as well as without air preheating. Further, it was confirmed that the high velocity (and at the same time high momentum) of the strong combustion jet is needed in order to realize HTAC technology. Additionally, it was noticed that could be possible to operate the HTAC boiler at the air excess ration near the stoichiometric values. For more details about the effects of selected operating parameters see in [14].

Coupling between the HTAC boiler and the steam cycle

Boiler design procedures involve an examination of the combustion process as well as the steam cycle. Both issues are strongly coupled due to the heat transfer proceeding from the combustion products inside the chamber to the water/steam mixture inside the boiler tubes. In this simulation series the coupling between HTAC boiler and the entire steam cycle is taken into consideration. The final boiler geometry as well as the operating and boundary conditions are the starting point for these calculations. User define subroutine has been written to perform the calculations. As a result of these calculations, the temperature profiles at the boiler walls are obtained. The HTAC boiler is proposed in this thesis as an ultra-supercritical boiler with the once-through type of the water circulation. Three commonly used configurations of the once-through boiler tubes were tested in this series of calculations: vertical down-up, up-down and spiral (see Fig. 5, top). The most uniform temperature profile can be found in the boiler with the spiral



tubing as can be found in Fig. 5, bottom. Comparing the two vertical tubes arrangements, it can be noticed that the up-down configuration works even worse than the down-up one. It was concluded that the spiral tubes configuration is the most suitable for the HTAC boiler designed in this work. However, this configuration is technically most complicated. The Rankine cycle efficiency of the steam cycle coupled with the HTAC boiler is calculated to be above 50%. For more details about the coupling between the HTAC boiler and the steam cycle see in [14].

Figure 5: Heat flux along the height of the boiler for the simulated tubes arrangements: vertical down-up (left), vertical up-down (centre) and spiral (right)

Summary and Outlook

Overall, the present study confirmed that HTAC technology could be a practicable, efficient and clean technology for pulverized coal fired boilers.

Summarizing, the most important advantages of the pulverized coal fired boiler operating under HTAC conditions are as following. Firstly, heat fluxes emitted during combustion process are high and uniform which results in the high firing density and consequently the small size of the boiler. Secondly, low NO_x emissions in comparison with the standard PC boilers. Then, burners have a very simple construction: without air staging, flame stabilizer or swirl, which are commonly used in the commercial pulverized coal burners.

Although HTAC technology seems to be attractive for power boilers fired with pulverized coal, there are still many open questions. First of all, the particulate concentration in the flue gas for the coal flame is probably too high to use the technique in conjunction with regenerative heat exchangers. It is a need to find in the future a way to utilize the enthalpy of the boiler exhaust gas. However, it was proved that the boiler can be operated without air preheating even if the ignition mechanism has to be examined in more details. Secondly, it could be also a problem with erosion because of the strong recirculation and high velocity of the coal particles.

Partners

Silesian University of Technology
Faculty of Energy and Environmental Engineering
Institute of Thermal Technology

Literature

- [1] Key World Energy Statistics, International Energy Agency, IEA, 2008, (www.iea.org).
- [2] International Energy Outlook, Energy Information Administration, EIA, 2008, (www.eia.doe.gov).
- [3] Survey of Energy Resources, World Energy Council, WEC, 2007, (www.worldenergy.org).
- [4] Deciding the Future: Energy Policy Scenarios to 2050, World Energy Council, WEC, 2007, (www.worldenergy.org).
- [5] M. Katsuki, T. Hasegawa: The Science and Technology of in Highly Preheated Air, Proc. Combust. Inst., 27:3135-3146, 1998.

- [6] F. Weinberg, D. Hardesty: Burners producing large excess enthalpies, *Combust. Sci. Tech.*, 8:201-214, 1974.
- [7] J. Wüning: Flammenlose Oxidation von Brennstoff mit hochvorgewärmter Luft, *Chem.-Ing.-Tech.*, 63: 1243-1245, 1991. (in German)
- [8] W. Kordylewski: *Spalanie i paliwa*, Oficyna Wydawnicza Politechniki Wrocławskiej, 2008. (in Polish)
- [9] R. Wilk, A. Szlęk, T. Misztal, K. Malczyk, A. Zajdel: Badania procesu spalania oleju lekkiego przy wysokiej temperaturze powietrza, *Archiwum Spalania*, 2:77-95, 2002. (in Polish)
- [10] A. Cavaliere, M.D. de Joannon: *Mild Combustion*, 30:329-366, 2004.
- [11] D. Szewczyk, J. Sudoh, A. Świdzki, B. Forsberg: Over decade of the industrial experience in high temperature air combustion applied with HRS regenerative burners, 6th HiTACG Symposium, Essen, Germany, 2005.
- [12] S. Orsino, M. Tamura, P. Stabat, S. Costantini, O. Prado, R. Weber: Excess enthalpy combustion of coal Results of High Temperature Air Combustion Trials, HTAC99, IFRF Research Station b.v., IFRF Doc.No.F46/y/3, IJmuiden, January 2000.
- [13] N. Schaffel, M. Mancini, A. Szlęk and R. Weber, *Mathematical Modeling of MILD Combustion of Pulverized Coal, Combustion and Flame*, 156:1771-1784, 2009.
- [14] N. Schaffel-Mancini, M. Mancini, A. Szlek, R. Weber. Novel conceptual design of a supercritical pulverized coal boiler utilizing high temperature air combustion (HTAC) technology, *Energy*, 35:2752-2760, 2010.
- [15] R. Lundquist, A. Shrief, P. Kinnunen, K. Myohanen, M. Seshamani: *A Major Forward-The Supercritical CFB Boiler*, Power-Gen International 2003, Las Vegas, USA, 2003.

3.3 Optimierung von HT-PEM-Brennstoffzellen mittels numerischer Strömungssimulation

Volker Wesling, Peter Giese, Thorsten Kandelhardt – Institut für Schweißtechnik und Trennende Fertigungsverfahren

Die Effizienzsteigerung von Brennstoffzellen auf der Grundlage einfacher und kostengünstiger Bauelemente, die das Potential zu einer Massenproduktion haben, ist das Ziel unseres Forschungsvorhabens, welches durch die N-Bank finanziert wird. Neben neuartigen Membranen aus neutralen Monomeren sollten spezielle Kanalgeometrien zur Verteilung der Reaktionsgase entwickelt werden.

Brennstoffzellen, die als Energieerzeuger aus Wasserstoff und Sauerstoff zu den erneuerbaren Energien gehören, bekommen derzeit eine wichtige Rolle zugeschrieben. Dennoch sind Probleme wie zum Beispiel die geringere Leistungsdichte gegenüber Verbrennungen immer wieder ein Thema. Ein Bereich, um die Energiedichte zu verbessern, liegt in der strömungstechnischen Optimierung der Kanalstrukturen (Flowfields). Durch den Einsatz von systematischen Lösungsfindungen, kombiniert mit der Entwicklung von Brennstoffzellensimulationen, sollte die Leistungsfähigkeit von Hochtemperaturbrennstoffzellen verbessert werden.

Die Fragen hinsichtlich der Verbesserung des Flowfields sind wegen der zum Teil widersprüchlichen Anforderungen nicht pauschal zu beantworten. Zu den Aufgaben der Flowfields gehören neben der Sammlung der Edukte und Produkte über der gesamten Elektrodenoberfläche, eine optimale Versorgung mit Luft und Wasserstoff sowie die gleichmäßige Verteilung der Gase. Dem entgegen steht eine möglichst großflächige Kontaktierung zur Reduzierung des Übergangswiderstandes und zur Verteilung der Montagekräfte. Ein möglichst geringer Druckverlust, der vor allem in Bezug auf die Verlustleistung und Kompressorleistung gewünscht ist, ist ebenfalls in den Anforderungskatalog mit aufzunehmen.

Stand der Technik

Bekannte Niedertemperaturbrennstoffzellen, die als PEM oder Methanol-Brennstoffzelle ausgeführt sind, haben typische Arbeitstemperaturen zwischen 25°C und 80°C. Es können Leistungen von 50KW und mehr erreicht werden. Die geringen Arbeitstemperaturen erleichtern die Handhabbarkeit und reduzieren die Vorwärmzeiten enorm. Der Schadstoffausstoß bei PEM-Brennstoffzellen ist äußerst gering. Die saure Umgebung, in der sich die Katalysatoren befinden, erfordert jedoch teures Platin. Zudem kann, wenn Luft als Reaktionsgas verwendet wird, der Katalysator kontaminiert werden, was die Funktionsfähigkeit der Brennstoffzelle stark einschränkt.

Die Bipolarplatten, in die sowohl die Wasserstoff- als auch die Sauerstoffkanäle hineingefräst werden, werden aufeinander gestapelt und als Stack bezeichnet. Durch die an den Rändern der Platten angebrachten Verspannungen werden die Platten verformt und lokal stark belastet. Die Flächenpressung ist aus diesem Grund nicht über die gesamte Zelle konstant, was zu großen Dichtungsproblemen führt. Versuche mit verschiedenen Flowfields sowohl bei PEM-Brennstoffzellen als auch bei anderen Brennstoffzellen wurden schon in [1] und [2] durchgeführt. Die Einflüsse der Kanalstruktur auf die unterschiedlichen Eigenschaften haben die Vielfalt neuer Strukturen stark reduziert. Dennoch beschränken sich die Verteilerstrukturen in Brennstoffzellen auf einzel- und mehrfachmäander oder parallele Strukturen.

Strömungssimulationen für Brennstoffzellen bieten wenige Unternehmen in kommerzieller Software wie zum Beispiel Ansys und Comsol an. Vor allem Comsol hat in den letzten Jahren ein eigenes Brennstoffzellen-Modul entwickelt, mit dem die Simulation von NT-PEM und HT-PEM-FCs möglich ist. Ein Modul für Ansys CFX, welches für die Software Fluent entwickelt wurde, ist mittlerweile vollständig integriert. Auch wenn diese Modelle sehr genau arbeiten, besitzen sie einige Nachteile: Um die Genauigkeit realisieren zu können, müssen die Simulationsmodelle entsprechend groß werden, so dass die Rechenleistung und

die damit verbundene Rechenzeit entsprechend groß sind. Dadurch ist es im Allgemeinen nur möglich, einzelne kleine Zellen in angemessener Zeit zu berechnen [6].

Grundsimulation der Kanalströmung

Als Werkzeug zur Aufgabenbewältigung wurden Simulationen eingesetzt. Ziel war es, die Interaktionen zwischen Anode und Kathode, sowie Gaskonzentrationen in der Ebene der Membran auch bei komplexen und großen Brennstoffzellen realitätsnah zu simulieren. Alle nachfolgenden erwähnten und dargestellten Berechnungen wurden mit dem kommerziellen CFD-Programm Ansys CFX mit einem turbulenten Ansatz durchgeführt.

Aufgrund der Tatsache, dass in die an dem Strömungskanal (Flowchannel) anliegende Diffusionsschicht kontinuierlich Masse übergeht, um mit dem gegenüberliegenden Medium zu reagieren, verändert sich die Umgebung des Strömungsmediums in Abhängigkeit des Massentransports durch die

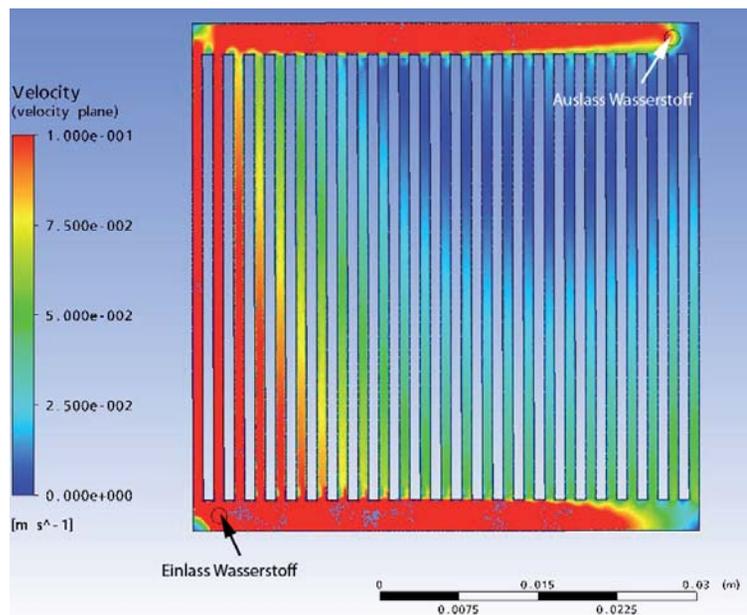


Abbildung 1: Simulation einer parallelen Struktur

Diffusionsschicht. Je nachdem, welcher Strömungskanal (Wasserstoffseite, Sauerstoffseite) betrachtet wird, wird der Volumenstrom entweder größer (Kathodenseite) oder kleiner (Anodenseite).

Die simulative Aufgabe der Geschwindigkeitsverteilung unter Berücksichtigung des Einströmens von Gas in die Diffusionsschicht steht im Vordergrund der grundlegenden Berechnung. Um das Strömungsverhalten realitätsnah abzubilden, ist es erforderlich, dem Strömungskanal mit Hilfe von Oberflächenquellen Masse zu entziehen. Durch das Variieren von z.B. Druck und Temperatur der Medien kann der Betriebszustand der Brennstoffzelle verändert werden. Um die Simulationen untereinander vergleichbar zu machen, wurde der Betriebszustand der Brennstoffzelle in der Simulation konstant (Stromdichte 200 mA/cm^2) gehalten.

In den Ergebnissen ist ein signifikanter Unterschied gegenüber einer herkömmlichen Simulation ohne Oberflächenquellen vor allem in der Geschwindigkeitsverteilung zu erkennen.

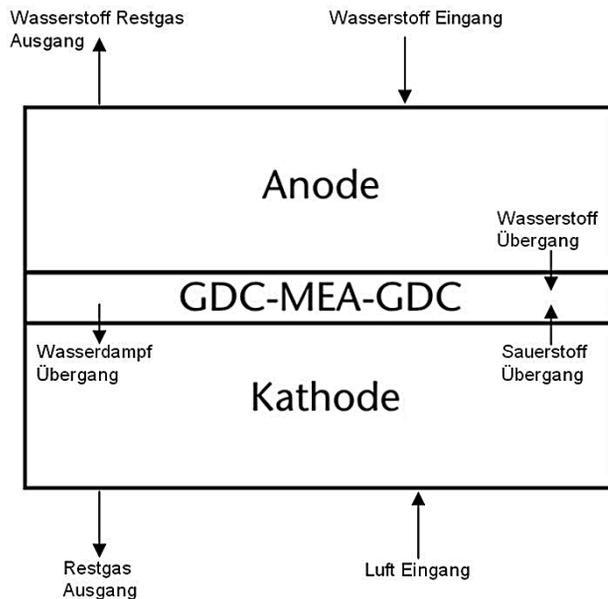


Abbildung 2: Simulation einer parallelen Struktur

Im Beispiel einer parallelen Verteilerstruktur (siehe Abbildung 1) weist, entgegen der Erwartung einer ähnlichen, eventuell nur langsameren Strömungsgeschwindigkeit zum Ausgang hin, die Simulation mit Oberflächenquellen eine starke Rückströmung auf. Eine derartige Strömungsverteilung ist für die Funktion und Leistung einer Brennstoffzelle negativ zu bewerten. Dementsprechend ist eine parallele Struktur trotz des immensen Differenzdruckvorteils gegenüber einer Mäanderstruktur benachteiligt. Dieses ließ sich in einem Versuch bestätigen. Eine genauere Beschreibung der Funktionsweise ist in [3] zu finden.

HT-PEM-FC-Simulationsmodul

Neben der Grundsimulation des Gastransportes ist die Simulation der Diffusionseigenschaften in die Gasdiffusionsschicht und die daraus resultierenden Interaktionen zwischen der Anode und Kathode ein weiterer entscheidender Schritt für die Entwicklung einer Brennstoffzelle. Aus diesem Grund wurde ein eigenes geeignetes PEM-FC-Modell entwickelt. Ein in Ansys CFX implementiertes FORTRAN-Programm ermöglicht es, auch größere, komplexere Modelle zu berechnen. Damit dies möglich ist und dennoch eine realistische Strömungsverteilung resultiert, wurden die Gasdiffusionsschicht, die Katalysatoren und die Membran zu einem gemeinsamen Volumen zusammengefasst (vgl. Abbildung 2).

Dieses Programm bildet die Diffusion der Gase und die Gasverteilung in den Gasdiffusionsschichten ab und ermöglicht somit Reaktionsmaxima und -minima zu erkennen. Zusätzlich werden hier die Reaktionsgase und Produkte (inkl. Stickstoff) berechnet. Selbst eine Berechnung eines kompletten 3-er Stacks wurde mit dieser Modifikation durchgeführt. Somit wurden auch die Unterschiede in den einzelnen Zellen und Gasdiffusionsschichten sichtbar (vgl. Abbildung 3a, 3b).

Temperaturentwicklungssimulation von HT-PEM-Brennstoffzellen

Bei einer höheren Betriebstemperatur der Brennstoffzelle ist die genaue Temperaturentwicklung von Interesse, da eventuell die Temperaturbelastung innerhalb bestimmter Flächen

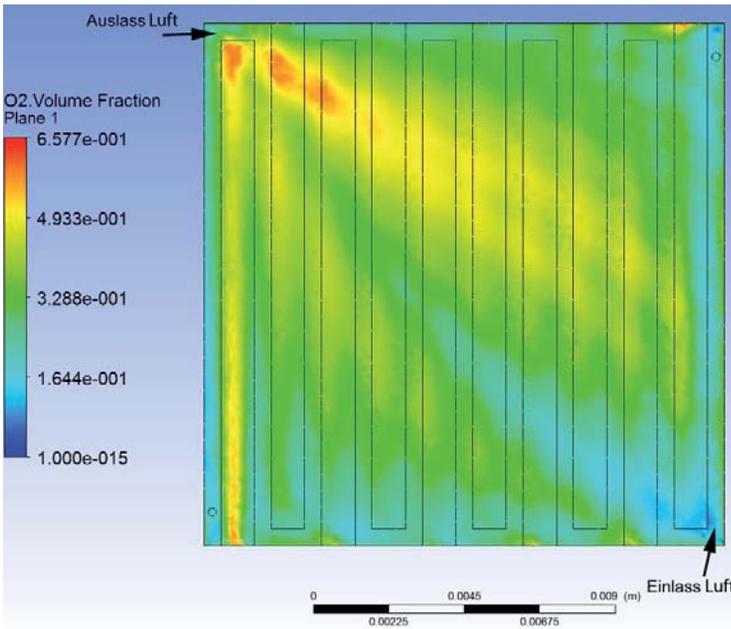
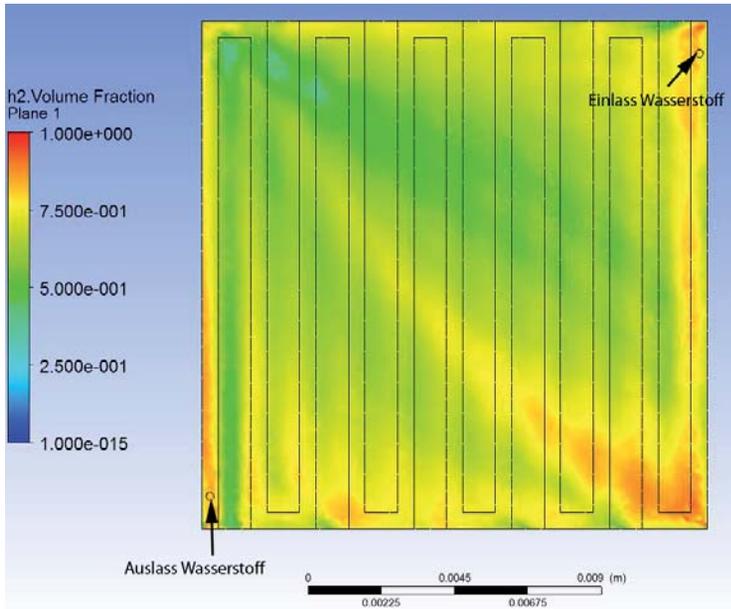


Abbildung 3a und 3b: Gasverteilung auf Höhe der Membran von zwei Mäanderstrukturen

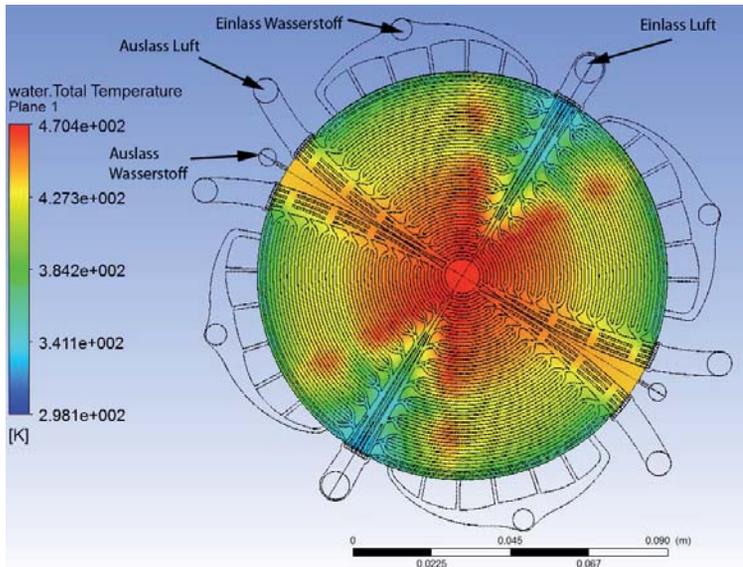
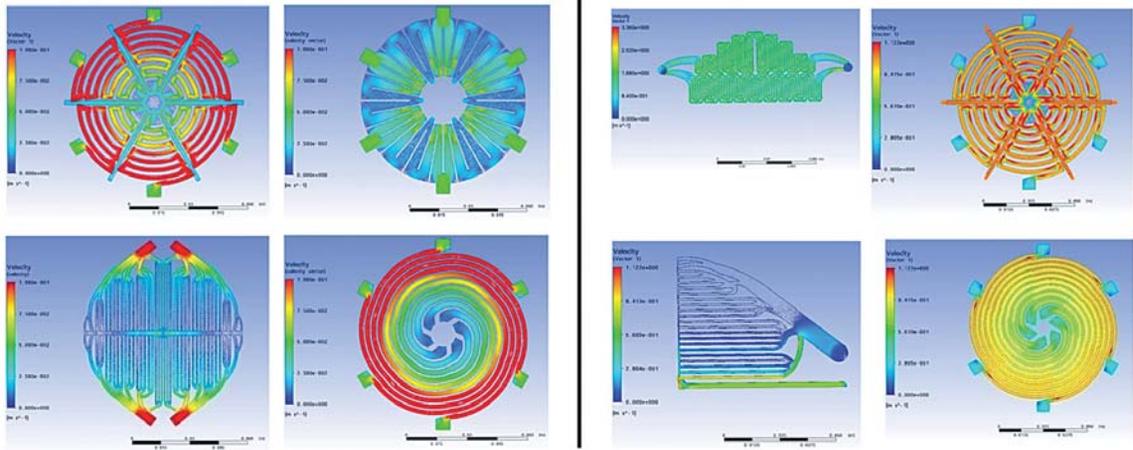


Abbildung 4: Simulierte Temperaturverteilung

der Brennstoffzelle kritische Werte annehmen und damit die Lebensdauer reduzieren könnte. Besonders kritisch sind dabei nicht unbedingt nur die hohen absoluten Temperaturen, sondern besonders große lokale Temperaturdifferenzen. Ziel ist es daher, die Brennstoffzelle so zu konstruieren, dass im Idealfall eine konstante Temperaturverteilung über die gesamte Zelle vorliegt, da die Materialbelastung der Brennstoffzelle so am geringsten ist. Eine entscheidende Möglichkeit, die Temperaturentwicklung lokal zu verändern, ist über die Form und den Verlauf der Gaskanäle für Wasserstoff und Sauerstoff bzw. Luft gegeben. Man kann annehmen, dass die Reaktionsdichte vom Abstand der beiden Kanäle abhängt, und davon die Temperaturerhöhung.

Die Simulation selbst, wobei die Modellierung grundlegend nach dem vorhergehenden Abschnitt dargestellten Schema vorgenommen wurde, ist durch die Einbeziehung der Temperaturverteilung etwas vergrößert worden. Mögliche Temperaturabstrahlungen an die Umgebung werden nicht berücksichtigt. Neben der Betrachtung der Temperaturverteilung in der Zelle selbst ist es nun auch möglich, die lokalen Temperaturveränderungen in den Zellplatten anschaulich zu machen. Da-



für sind am besten transiente Simulationen geeignet. Für die Berechnung und Darstellung von Temperaturgradienten bzw. Temperaturverteilungen ist die stationäre Simulationsberechnung vorzuziehen.

Abbildung 5: Ergebnisse aus der systematischen Lösungsfindung von Anode (vier links) und Kathode (vier rechts)

Zusammenführung von Simulation und Konstruktion

Auf Basis eines runden radialsymmetrischen Flowfields mit sechseckiger Grundplatte mussten Kanalgeometrien mit einem Durchmesser von 120 mm konstruiert werden.

Eine durchgeführte systematische Lösungsfindung (vgl. Abbildung 5), bestehend aus den Wirkprinzipien Gaszuführung,

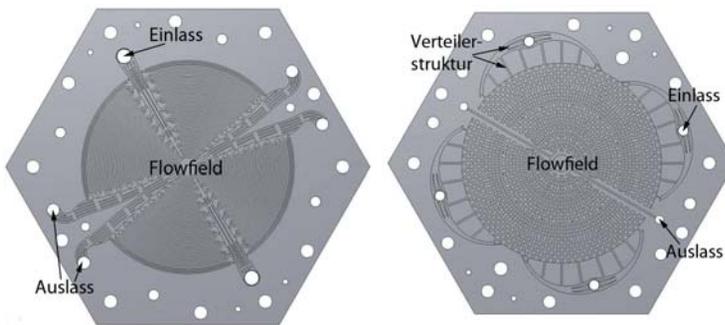


Abbildung 6: links: Kathode der optimierten Brennstoffzelle; rechts: Anode der optimierten Brennstoffzelle [4]

Gasverteilung und Gasausleitung, liefert die grundlegenden Flowfieldstrukturen. Diese stellen auf der Anode eine Pin-Struktur und auf der Kathode eine Mäanderstruktur bei unterschiedlichen Gaszu- und -abführungen dar. Der Grund verschiedener Strukturen für Anode und Kathode liegt an den unterschiedlichen Bedingungen auf der jeweiligen Elektrode. Die Kathode hat bei PEM-Brennstoffzellen neben der Verteilung auch das gesamte Produktwasser aufzufangen und abzuführen.

In mehreren Schritten wurden beide Flowfields hinsichtlich geringen Druckverlusts, gleichmäßiger Strömungsverteilung, kleiner Temperaturgradienten und guter Reaktionsdichte optimiert. Die zum Patent angemeldeten und optimierten Gasverteilungsstrukturen sind in Abbildung 6 zu sehen.

Die gegenüber konventionellen Lösungen vorgenommenen Veränderungen in den Zuführungen und Verteilerstrukturen erzeugen eine homogene Strömungsverteilung der Gase während des Betriebs. Auf der Anode wurden vier Eingänge und zwei Ausgänge verwendet. Die Kathode hingegen besitzt zwei Eingänge und vier Ausgänge. Die parallelen Kanäle der Kathode sind nunmehr radial angeordnet. Diese Anordnung verringert die Verlustleistung, die ein Produkt aus Differenzdruck und Volumenstrom darstellt.

Literatur

- [1] Can Kreuz, PEM-Brennstoffzellen mit spritzgegossenen Bipolarplatten aus hochgefülltem Graphit-Compound, Universität Duisburg-Essen, April 2008, Dissertation.
- [2] Oliver Christopher Niemzig, Entwicklung eines portablen PEM-Brennstoffzellensystems mit Bipolarplatten aus einem elektronisch leitfähigen thermoplastischen Compound-Material, Universität Duisburg-Essen, Juli 2005, Dissertation.
- [3] V. Wesling, P. Giese, T. Kandelhardt, Single-Fuel-Cell Simulation With CFX Under Consideration of Multi-Phase-Domains and Hydrogen-Oxygen-Interactions, ISSN: 1869-8018, Januar 2010.
- [4] V. Wesling, P. Giese, T. Kandelhardt, Entwicklung eines neuartigen strömungstechnisch-optimierten Brennstoff-

zellenflowfields, Aktenzeichen 10 2010 035 254.3 und Aktenzeichen 10 2010 035 259.4.

- [5] V. Gurau, F. Barbir, H. Liu: An Analytical Solution of a Half-Cell Model for PEM Fuel Cells, *Journal of The Electrochemical Society* 147, 2000.
- [6] Christian Siegel, George C. Bandlamudi, Nadine van der Schoot, Angelika Heinzl, Large Scale 3D Flow Distribution Analysis in HTPEM FuelCells Proceedings of the European COMSOL Conference 2009, 14-16. Oct. 2009, Milan, Italy.



Seth
RESE
ER

Ma/A
Lissi

C...

S/S

097 GND2-C12

OK1

097 GND2-C12

OK1

097 GND2-C12

OK1

4 Hochleistungsrechnen

Simulation spielt neben Theorie und Experiment eine zunehmend wichtige Rolle, um komplexe Phänomene und Prozesse nicht nur vorherzusagen sondern auch besser verstehen zu können. Die Umsetzung von Simulationen erfordert geeignete Rechnerressourcen und Algorithmen. In der Struktur des SWZ wird diesem Umstand Rechnung getragen, indem neben den anwendungsorientierten Bereichen das Thema Hochleistungsrechnen als Querschnittsdisziplin implementiert wurde.

Die Grundlage jeder Computersimulation ist eine Problemformulierung durch ein Modell als Abbild der Realität, das mit Hilfe geeigneter mathematischer Algorithmen auf dem Rechner gelöst werden soll. Im Bereich der Kontinuumstheorie sind dies beispielsweise partielle Differentialgleichungen, die auf Basis finiter Approximationen gelöst werden. Daraus resultieren Gleichungssysteme mit mehreren Millionen bis Milliarden Unbekannten zu deren Lösung neben effizienten Algorithmen leistungsfähige Software für Parallelrechner benötigt werden.

Die Anforderungen, die an die Rechnerressourcen gestellt werden, sind je nach zu beantwortender Fragestellung enorm. So erfordert die Berechnung der Strömung im Schaufelrad einer Turbomaschine während einer einzigen Umdrehung, die in der Realität wenige hundertstel Sekunden dauert, einen Rechenbedarf von mehreren hundert Stunden. Derartige Berechnungen sind nur praktikabel, wenn entsprechend viele Prozessoren parallel verwendet werden. Die Zunahme der Leistungsfähigkeit der Rechner, die in der Vergangenheit maßgeblich durch die Erhöhung der Taktfrequenz erzielt wurde, wird in Zukunft durch massive Parallelität erreicht werden. Dies erfordert den Einsatz und die Anpassung geeigneter Berechnungsprogramme.

Der Schwerpunkt der Arbeiten im SWZ im Bereich Hochleistungsrechnen liegt auf der Anpassung und Weiterentwicklung von Algorithmen für moderne Rechnerarchitekturen. Hierzu gehören beispielsweise kinetische Berechnungsverfahren und effiziente Zeitintegrationsverfahren. Zukünftige Projekte werden sich insbesondere auf die Nutzung von spezieller Hardware für numerisch intensive Berechnungen konzentrieren.

4.1 Analyse von dynamischen Lasten in hochbelasteten Strömungsmaschinen – Notwendigkeit der Modellvalidierung und des Hochleistungsrechnens

Gunther Brenner, Yang Yang, Andreas Lucius – Institut für Technische Mechanik

In vielen Industriebereichen werden Strömungsmaschinen als wesentliche Bestandteile von Produktionsanlagen eingesetzt. Sie dienen der Förderung von Flüssigkeiten oder Gasen im Produktionsprozess z.B. in der Verfahrenstechnik, in der Lüftungs- und Klimatechnik oder für Trocknungsprozesse. In den genannten Anwendungen werden Pumpen und Ventilatoren bzw. Verdichter als Arbeitsmaschinen eingesetzt, um das Fluid durch die Anlage zu transportieren. Die notwendige Förderleistung wird dabei durch die Prozessführung vorgegeben. Hierbei kann ein Trend zu einer immer höheren Belastung der Strömungsmaschinen beobachtet werden. Höher belastete einzelne Laufräder erlauben eine Reduktion der insgesamt notwendigen Anzahl von Maschinen, um die notwendige Fördermenge bzw. Druckerhöhung zu erreichen. Insbesondere für Radialventilatoren wird eine Erhöhung der Leistungsfähigkeit in den Bereich von Verdichtern angestrebt, dies geschieht im Wesentlichen über die Steigerung der Umfangsgeschwindigkeit am Laufradaustritt. Durch die einfache Blechkonstruktion ergibt sich ein Kostenvorteil gegenüber der komplexen Laufradgeometrie eines Verdichters.

Die Belastungen der Laufräder nehmen jedoch mit der Umfangsgeschwindigkeit stark zu. Die Fliehkraft als eine wesentliche Belastung nimmt quadratisch mit der Drehzahl der Maschine zu. Eine andere Quelle, deren Abschätzung für die Auslegung mit deutlich größerer Unsicherheit behaftet ist, ist das Auftreten von Strömungskräften. Insbesondere bei Abweichung vom Auslegungspunkt ergeben sich schwankende Strömungskräfte, die sich in Form von fluidinduzierten Schwingungen bemerkbar machen. Prominente Beispiele für diese sogenannten Teillastinstabilitäten sind das rotierende Abreißen (rotating stall) bzw. das Pumpen, welches vor allem in Verdichtern auftritt. Die Bestimmung dieser Lasten wird mit Hilfe von

modernen Simulationsverfahren ermöglicht. Die instationäre Simulation ist ein mächtiges Hilfsmittel, um die dynamischen Lasten, welche im Betrieb der Maschine auftreten, korrekt zu erfassen. Die ermittelten Lastkollektive können dann für die Dimensionierung der Laufräder herangezogen werden.

Derzeit wird in 2 Projekten am Institut für Technische Mechanik die Belastung von Laufrädern durch Strömungskräfte untersucht. Der Schwerpunkt liegt dabei auf der Ermittlung der instationären Fluidkräfte mit Hilfe von CFD (Computational Fluid Dynamics). Das ZIM Projekt „Untersuchungen zur Realisierbarkeit von Hochleistungsradialventilatoren zur Anwendung in der Brüdenkompression“ befasst sich mit der Untersuchung von Hochleistungsventilatoren mit Umfangsgeschwindigkeiten von über 325 m/s. Im Rahmen des AiF Projekts „Dynamische Lasten“ werden in Zusammenarbeit mit der Universität Magdeburg Simulationen und zeitaufgelöste Messungen durchgeführt, um den Erfolg von Rechenverfahren bei der Vorhersage instationärer Phänomene zu beurteilen. Beide Forschungsprojekte werden in enger Zusammenarbeit mit der Industrie durchgeführt. Im Folgenden werden Ergebnisse aus den Projekten präsentiert.

Numerische Simulation von Strömungen in Turbomaschinen – High Performance Computation

Numerische Verfahren zur Lösung der strömungsmechanischen Grundgleichungen haben sich in der Industrie als Standardwerkzeug zur Optimierung und Analyse von Strömungsmaschinen etabliert. Simulationsläufe im Rahmen des Designprozesses lassen sich innerhalb kurzer Zeit auf Arbeitsplatzrechnern durchführen. Die heute verfügbaren Rechenprogramme und Rechentechniken erlauben es, Simulationsmodelle mit 10^6 Gitterpunkten auf einem PC innerhalb von Stunden stationär zu berechnen. Allerdings steigen mit der Verfügbarkeit von Ressourcen und schnellen Methoden auch die Ansprüche an die Modellierung. Für die Mehrzahl der Fälle wird eine stationäre Simulation ausreichend sein, oftmals unter Verwendung von Vereinfachungen bzgl. der Geometrie des Rechengebietes. Ein Beispiel ist die Ausnutzung der Rotationssymmetrie des Rotors und Beschränkung auf einen Schaufelkanal mit periodischen Randbedingungen. Tatsächlich ist die Strömung in der

Maschine allerdings immer instationär. Bedingt durch die Rotation des Laufrades in einem Gehäuse ergibt sich eine instationäre Rotor-Stator-Interaktion, welche mit Drehfrequenz bzw. Schaufelfrequenz fluktuierende Fluidkräfte verursacht. Weiterhin treten turbulente Fluktuationen der Strömungsgrößen auf, die unter Umständen mit dem Turbulenzmodell aufgelöst werden sollen. Im Teillastbereich können zusätzlich instabile Strömungszustände wie das „rotating stall“ Phänomen auftreten, in diesem Fall sind niederfrequente Anregungen unterhalb der Drehfrequenz zu erwarten. Das rotierende Abreißen und andere instabile Strömungszustände stehen im Fokus des AiF Projektes „Dynamische Lasten“.

Um mit einem Simulationsmodell das rotierende Abreißen zu berechnen, sind Arbeitsplatzrechner nicht mehr ausreichend. Die Rotationssymmetrie der Strömung ist hier nicht mehr gegeben, d. h. es muss immer der gesamte Rotor mit allen Schaufeln gerechnet werden. Zudem werden Einlaufgebiete und Austrittsgebiete (z.B. Spiralgehäuse) benötigt, um das Simulationsmodell realitätsnah abzubilden. In diesem Fall sind Modelle mit 10^7 Gitterpunkten durchaus üblich. Die geforderte Auflösung hängt aber sehr stark von der geforderten Genauigkeit und dem gewählten Turbulenzmodell ab. Bedingt durch den hohen Speicherbedarf eines solchen Modells sind Hochleistungsrechner zur Durchführung der Simulationen notwendig. Durch massive Parallelisierung lassen sich Probleme dieser Größenordnung innerhalb weniger Stunden stationär lösen. Stationäre Lösungen sind in der Nähe des Auslegungspunktes möglich, allerdings hat sich gezeigt, dass transiente Simulationen bessere Übereinstimmung mit Messungen liefern. Im Fall massiver Ablösungen oder Strömungsinstabilitäten muss dagegen immer instationär gerechnet werden.

Für eine effiziente Parallelisierung sollten nicht weniger als 100.000 Knoten pro Prozessor eingesetzt werden, damit der Kommunikationsaufwand klein gegenüber dem eigentlichen Rechenaufwand bleibt. Für Modelle mit 10 Mio. Knoten werden daher sinnvollerweise 64 bis 128 Prozessoren eingesetzt. Im Rahmen des Projektes wurden in der Regel 64 oder 96 CPUs parallel zur Berechnung benutzt. Für instationäre Simulationen erhöht sich die Rechenzeit nochmals um ca. eine bis zwei Größenordnungen. Um die transienten Effekte zu erfassen

werden kleine Zeitschritte gefordert. Pro Umdrehung werden einige hundert bis zu über 1.000 Zeitschritte benötigt, wobei die Zeitschrittweite vom Turbulenzmodell und auch der Gitterauflösung abhängig ist. Feine Gitter erfordern in der Regel kleinere Zeitschritte damit Fluktuationen während eines Zeitschrittes nicht über viele Gitterzellen transportiert werden. Dazu kommt, dass für eine sinnvolle Bildung von Mittelwerten einige Umdrehungen benötigt werden. Ist man an niederfrequenten Phänomenen wie Rotating Stall interessiert, werden 10 oder besser 20 Umdrehungen benötigt, unter Umständen auch mehr. Insgesamt wurden z.B. für die Simulation des rotierenden Abreißens in einer Kreiselpumpe im AiF Projekt „Dynamische Lasten“ in der Größenordnung 10^4 Zeitschritte pro Fall berechnet. Die gesamte Rechenzeit für einen solchen Fall liegt trotz des Einsatzes von massiv parallelen Rechnern immer noch im Bereich von Wochen.

Neben der eigentlichen Simulation ist auch die Auswertung großer Datensätze zeitkritisch. Die Dateigröße für die Ergebnisdateien mit allen Variablen beträgt 5 bis 10 Gigabyte (Binärformat), für Zwischenergebnisse beträgt die Dateigröße noch bis zu 1 GB. Dazu wurde die Anzahl gespeicherter Variablen auf das Nötigste reduziert. Der gesamte Bedarf an Festplatten Speicher liegt bei einigen Terrabyte. Das Laden und Auswerten dieser Dateien für eine große Anzahl von Zeitschritten kann ebenfalls durch den Einsatz moderner Programme parallelisiert und damit wesentlich verkürzt werden. Die Arbeiten zur Auswertung der Daten stehen aktuell im Fokus der Arbeiten am Institut für Technische Mechanik.

Modellierung der turbulenten Strömung

In einer Turbomaschine sind immer hohe Reynoldszahlen zu erwarten, d. h. die Strömung ist in der Regel stark turbulent. Die turbulenten Fluktuationen erhöhen den Impulsaustausch und sind damit für erhöhten Reibungswiderstand verantwortlich. In einigen Anwendungen ist die Turbulenz aber auch durchaus erwünscht z.B. erhöhte Stabilität turbulenter Grenzschichten gegenüber Ablösungen.

Die dem Problem zu Grunde liegenden Navier-Stokes'schen Gleichungen sind auch für turbulente Strömungen gültig. Für

eine numerische Simulation der Turbulenz müssen aber alle Fluktuationen in Raum und Zeit aufgelöst werden. Man nennt diese Technik direkte numerische Simulation – kurz DNS. Es kann gezeigt werden, dass mit steigender Reynoldszahl die Größe der turbulenten Strukturen und damit die nötige Gitterweite stark abnimmt. DNS ist heute für Reynoldszahlen von einigen 10.000 in einfachen Geometrien wie Kanälen unter Einsatz von tausenden Prozessoren möglich. Im Turbomaschinenbau werden Reynoldszahlen von 10^6 bis 10^7 erreicht, damit ist eine DNS für diesen Fall selbst bei einer wesentlichen Steigerung der verfügbaren Rechnerkapazitäten ausgeschlossen.

Der Ingenieur ist in der Regel nicht an der turbulenten Fluktuation von Strömungsgrößen auf kleinsten Zeit- und Längenskalen interessiert. Von Interesse sind integrale Beiwerte wie etwa der Wirkungsgrad einer Maschine oder die Druckerhöhung. Stand der Technik ist die Modellierung der Effekte turbulenter Fluktuationen nach dem RANS (Reynolds Averaged Navier Stokes) Ansatz. Die meisten dieser Modelle basieren auf dem Konzept der Wirbelviskosität, d.h. die Wirkung der Turbulenz wird durch erhöhte Viskosität modelliert. Die Gitterweite ist für diese Modelle nicht mehr physikalisch motiviert wie im Falle der DNS, sondern rein durch numerische Genauigkeit.

Da sich gezeigt hat, dass die Modellannahmen von RANS Modellen in einigen Fällen nicht zu guten Übereinstimmungen mit Messergebnissen führen, hat sich eine Technik etabliert, bei der die großen turbulenten Strukturen mit dem Gitter aufgelöst werden und die Turbulenz auf kleineren Skalen modelliert wird (Large Eddy Simulation – LES). Dies ist sinnvoll, da die großen Strukturen den wesentlichen Anteil der kinetischen Energie beinhalten. Gegenüber der DNS ergibt sich eine deutliche Einsparung, allerdings sind die notwendigen Gitterweiten i.A. wesentlich kleiner als für RANS Rechnungen. Gut aufgelöste LES Rechnungen für komplexe Geometrien bei hohen Reynoldszahlen sind für heutige Höchstleistungsrechner immer noch zu aufwändig. Schätzungen zufolge sind innerhalb der nächsten Dekaden LES Simulationen für komplexe Geometrien wie Tragflügel bei realistischen Reynoldszahlen auf Superrechnern möglich.

Um die Lücke zwischen Turbulenz auflösenden Modellen (LES / DNS) und RANS Modellen zu schließen, wurden hybride LES RANS Modelle entwickelt, die Simulationen technischer Strömungen bereits heute ermöglichen. Dabei wird ein Teil des Rechengebietes im LES Modus (z.B. Ablösegebiete hinter stumpfen Körpern) simuliert, wogegen z.B. anliegende Grenzschichten mit RANS gerechnet werden. Solche Modelle sind aktueller Forschungsgegenstand und sollen im Rahmen der Projekte verifiziert und validiert werden. Für einige Testfälle konnte für diese Modellklasse eine Überlegenheit gegenüber RANS bei reduziertem Aufwand gegenüber LES erreicht werden. Aus der Vielzahl von Ansätzen wurde das SAS (Scale-Adaptive Simulation) Konzept für die Simulationen ausgewählt.

Forschungsergebnisse

In den genannten Forschungsprojekten steht die Simulation instationärer Strömungen in Pumpen und Ventilatoren und die Bestimmung der fluktuierenden Strömungskräfte im Vordergrund.

Das AiF Projekt ist fokussiert auf die Validierung von Strömungsberechnungen im Teillastbereich der Kennlinie. Dazu wurden

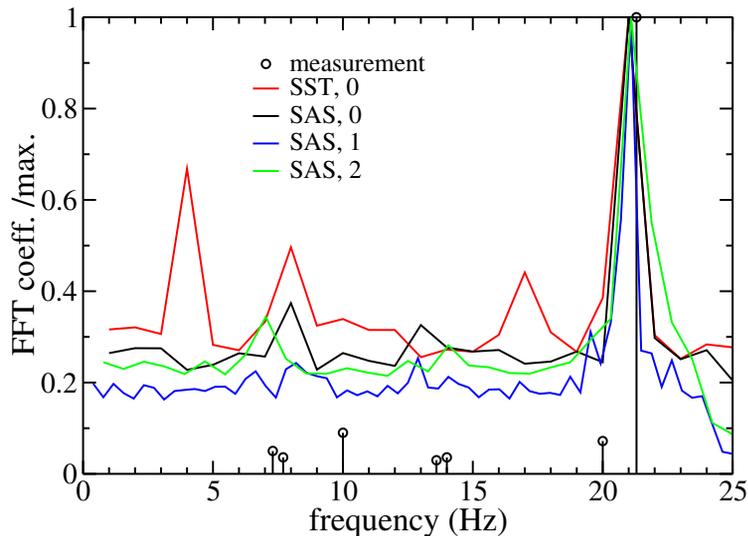


Abbildung 1: Vergleich der Umlauffrequenz des rotating stall, Messungen vs. CFD

erstmals zeitlich hoch aufgelöste 3D PIV (Particle Image Velocimetry) Messungen in einem Modellventilator am Lehrstuhl für Strömungstechnik und Strömungsmechanik der Uni Magdeburg durchgeführt. Die Strömungsgeschwindigkeiten sind hier gering, so dass die Strömung inkompressibel behandelt werden kann. Die Validierung erfolgte ebenfalls mit „time resolved PIV“ Messungen des rotating stall in einer Kreiselpumpe. Diese Messungen wurden am selben Lehrstuhl im Rahmen früherer Projekte durchgeführt und dokumentiert. Es zeigt sich, dass mit einer ausreichenden Auflösung (15 Mio. Gitterpunkte, Fall 2 in Abbildung 1) und höherwertigem Turbulenzmodell (SAS) die Frequenz des rotating stall mit guter Genauigkeit vorhergesagt werden kann. Das SST Modell, als Referenz für ein in

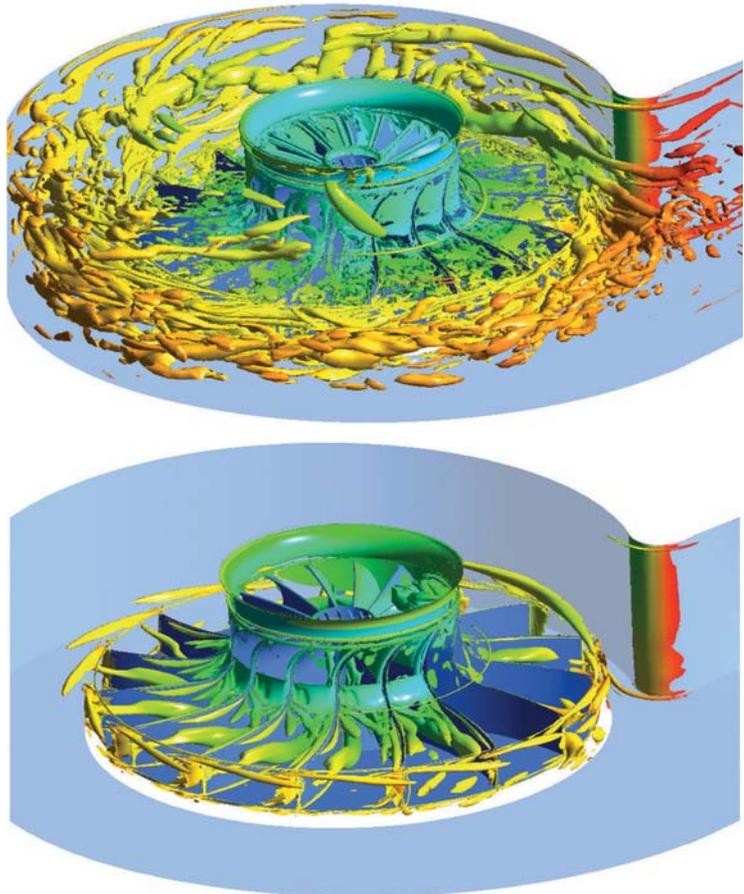


Abbildung 2: Vergleich der berechneten Wirbelstrukturen in einem Hochleistungsradialverdichter mit dem SAS Modell (Oben) und dem SST Modell (Unten), $\lambda_2 = -2,4 \times 10^6$

Turbomaschinen weit verbreitetes RANS Modell, ist nicht in der Lage, die kleineren Peaks im Spektrum korrekt vorherzusagen. Im ZIM - Projekt werden Hochleistungsradialventilatoren untersucht. Die Simulationen sind durch vergleichsweise hohe Kompressibilität gekennzeichnet (Machzahl 0,6). Es wurden transiente Simulationen im Betriebspunkt und bei Teillast mit RANS (SST Modell) und SAS durchgeführt. Beide Modelle geben in guter Übereinstimmung mit Messungen die integralen Kennwerte des Ventilators (Druck- und Temperaturerhöhung) wieder. Deutliche Unterschiede werden erkennbar, wenn man die Strömung im Detail betrachtet. Hierzu ist in Abbildung 2 das λ_2 -Kriterium für die Detektion von Wirbelkernen ausgewertet. Man erkennt, dass das SST Modell die grobe Struktur der Wirbelentwicklung wiedergeben kann. Es ist aber nicht in der Lage, die kleinskaligen Strukturen aufzulösen. Eine Frequenzanalyse ergibt im Wesentlichen Druckschwankungen mit der Schaufelfrequenz. Im Rahmen des Projektes werden numerisch Designstudien durchgeführt, um z.B. den Einfluss des Spiralgehäuses bei hohen Machzahlen zu untersuchen.

Fazit und Ausblick

Die Ergebnisse zeigen, dass mit Hilfe von numerischer Strömungssimulation nicht nur integrale Kenngrößen wie Wirkungsgrad oder Druckerhöhung, sondern auch das dynamische Verhalten in instabilen Strömungszuständen vorhergesagt werden können. Dabei sind Vorteile für Turbulenz auflösende Modelle ermittelt worden, welche allerdings die Verwendung von feineren Gittern erfordern. Für die Abarbeitung der transienten Simulationen sind Hochleistungsrechner aufgrund des großen Bedarfs an Hauptspeicher und Rechenzeit unbedingt erforderlich.

Die Bewertung der ermittelten Lasten durch Simulationen der Struktur ist Ziel des AiF Projektes. Zur Validierung der am Rechner ermittelten Strukturschwingungen werden Schwingungsmessungen am Ventilator durchgeführt. Damit soll das Potential einer einseitig gekoppelten FSI Simulation (Fluid Struktur Interaktion) für die Berechnung von fluid induzierten Schwingungen an Ventilatoren gezeigt werden.

Beteiligte Partner

Die Forschungsprojekte laufen in enger Zusammenarbeit mit den folgenden Partnern aus der Industrie:

- Piller Industrieventilatoren GmbH Moringen
- Forschungsvereinigung Luft- und Trocknungstechnik

Für die Arbeiten am AiF Projekt sind zwei weitere Forschungsstellen an der Otto-von-Guericke Universität Magdeburg beteiligt:

- Lehrstuhl für Strömungstechnik und Strömungsmechanik (LSS)
- Institut für Mechanik (IFME)

Die Projekte werden durch das Bundesministerium für Wirtschaft und Technologie im Rahmen von ZIM (Zentrales Innovationsprogramm Mittelstand), sowie über die AiF (Arbeitsgemeinschaft industrieller Forschungsvereinigungen „Otto von Guericke“) gefördert.

Literatur

- [1] J. Fröhlich D. von Terzi: Hybrid LES/RANS methods for the simulation of turbulent flows, Progress in Aerospace Sciences 44, 2008, pp 349-377.
- [2] F. R. Menter, M. Kuntz, R. Vender: A Scale-Adaptive Simulation Model for Turbulent Flow Predictions, AIAA Paper, 2003-0767.
- [3] P.R. Spalart, Strategies for turbulence modelling and simulations, International Journal of Heat and Fluid Flow 21, 2000, pp 252-263.
- [4] A. Lucius, G. Brenner, Unsteady CFD simulations of a pump in part load conditions using scale-adaptive simulation, International Journal of Heat and Fluid Flow 31, 2010, pp 1113 – 1118.

4.2 Numerical simulation of liquid-liquid and liquid-particle flows by the Lattice Boltzmann Method

Ernesto Monaco and Gunther Brenner – Institut für Technische Mechanik

Introduction

Multiphase flows are encountered in many natural phenomena and industrial processes like fuel engineering, cooling systems and nuclear reactors technology. The range of characteristic length and time scales is extremely various: in fact, while simple fluids are characterized by their atomic size, there are matters like suspensions, gels, foams or emulsions characterized by one or more additional length scales which are intermediate between atomic ($O(10^{-10} \text{ m})$) and macroscopic levels and are referred to as “mesoscopic” scales ($O(10^{-9} - 10^{-6} \text{ m})$). Besides, simple fluids are usually Newtonian, while these so-called *soft matter system* exhibit complex nonlinear rheological properties.

The complexity of multiphase flows makes theoretical solutions available only for relatively simple cases, while experiments are often extremely difficult or even impossible to perform. Therefore numerical simulations can greatly help in understanding the behaviour of multiphase flows, because they allow to investigate the influence of interphase interactions, microstructure, composition and geometry on the flow. To be truly effective a numerical scheme has to fulfil substantially two requirements: first of all, it has to properly model the *hydrodynamic interactions* occurring at interfaces. The expression hydrodynamic interaction designates for instance the dynamic particle correlation induced by diffusive momentum transport through the fluid, at a rate governed by fluid kinematic viscosity. Finally, the algorithm is required to be efficient and computationally cheap in order to track for instance the position and velocity of hundreds or thousands of droplets or particles. Conventional schemes based on the discretization of Navier-Stokes equations like finite volume method (FVM) or finite elements method (FEM) have problems especially with a large number of particles in 3D simulations, because there is the need of generating at any time step new geo-

metrically adapted grids. The lattice-Boltzmann method (LBM) [1] fulfils instead both these requirements.

The rest of this report is arranged as follows: section II introduces the LBM, underlying its particular nature when compared to traditional numerical techniques and presenting the approaches used in this study for the different classes of multiphase flows. These models are applied in section III and IV to study two different cases, namely the liquid-liquid mixing in a microscopic T-sensor and the wall-particle interactions in creeping shear flows. Finally, section V concludes the paper.

Computational Method

The LBM derives from a description of the fluid state based on the kinetic theory. In essence, the LBM is a discretization of the Boltzmann equation in terms of discrete velocity distribution functions f : of particle populations. This discretization leads to an explicit second order accurate finite difference scheme. The algorithm is characterized by collision and streaming steps. Given a certain node, streaming involves only next-neighbour nodes while the collision step is completely local, making the algorithm suit for massively parallel computations. The LBM has an underlying fixed (Eulerian) Cartesian lattice grid as consequence of the symmetry of discrete velocity set and the fact that lattice space δx is related to time step δt by $\delta x = c \delta t$, being c the basic unit of discrete velocity set. The advantage of LBM as compared to other numerical schemes is its ability to simulate dynamic processes at fairly low computational costs. Boundary conditions (BC) are expressed in terms of particle distributions. The most common way of modelling the no-slip boundary condition is the so-called “Bounce-Back” (BBK) scheme [1], in which distribution functions are reflected back into the computational domain after the streaming step. This boundary conditions is easily to implement, even if only first –order accurate.

For liquid-liquid or liquid-gas flows the Shan-Chen (SC) model [2] consists in introducing an interparticle potential, the gradient of which is the force that modifies the local velocity of the particles. The original formulation of this model was thermodynamically inconsistent, and allowed to reach small density ratios between the liquid and the gas phases ($O(10)$). In [3] it

was found that realistic density ratios could be reached by simply changing the equation of state of the model (EOS). This improved SC model has been applied in [4, 5] to study the binary droplet collisions at high inertia and density ratio (from 150 to 1000). In [6, 7] the LBM was successfully applied to particle-fluid suspensions. Because of its *Eulerian* grid at any time step the surface of the generic particle cuts some lattice links connecting grid nodes inside and outside the particle: for each of this inside-outside couple of nodes it is possible to define a *boundary node*, lying exactly in the middle of the cut link. The boundary nodes allow extending the BBK scheme to moving objects. This approach is simple to code and allow to overcome the limitations of FEM and FV, but has some drawbacks: first of all, since the particle frontier is approximated by a stepwise boundary, a large number of nodes is necessary to represent accurately particle surfaces; secondly it causes fluctuations in computed forces and torques which can led to instabilities at high Reynolds numbers Re . To update particle positions the molecular dynamics approach is used. It consists of Lagrangian tracking of individual spherical particles and considers the conservation of linear and angular momentum as well as the momentum exchange due to particle collisions. This formulation allows a simple calculation of forces and torques acting on every particle, except when two or more particles are in contact: in this case, the hydrodynamic (lubrification) forces have to be modelled and applied to the scheme, following the approach proposed in [8].

Two phase flow in a Mikromixer

The mixing of two fluids is an essential process in Lab-on-a-chip devices, which require rapid (order of milliseconds) and effective mixing of samples flowing in microchannels. Since typically $Re \ll 1$ in microflows turbulence cannot be used to promote mixing, which instead relies on diffusion. Unfortunately diffusion is quite a slow process, therefore it proves to be inefficient unless the mixing length amounts to a few microns: if that is not the case the channel length required for a complete mixing would get unacceptable because longer channels mean higher pressure gradient, with consequent problems in the design of the micropump driving the flow. In *passive* micromixers this issue is tackled by using the energy due to pumping or hydrodynamic potential to restructure the flow in a way which results

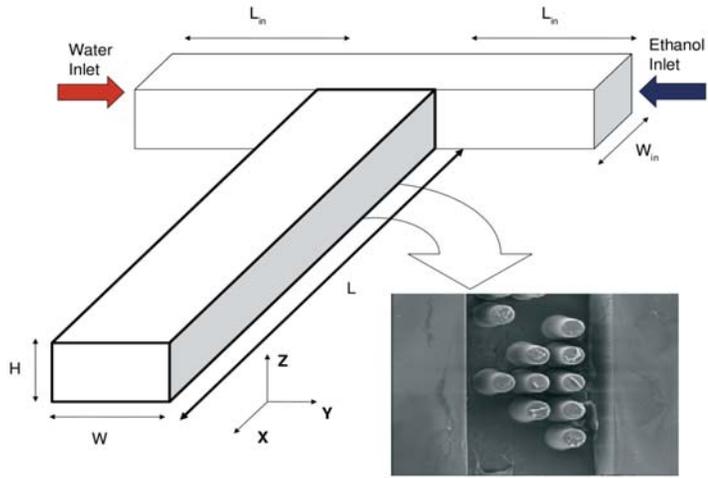


Figure 1: Sketch of T-sensor. The photographic image from [9] refers to part of one of the layouts tested in the experiments

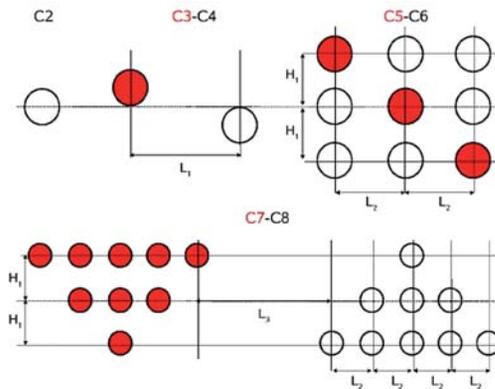


Figure 2: Geometry of the different configurations tested and comparison between the most relevant lengths in [9] and in the present study. The values in parentheses are referred to configuration 8

	L	H	W	W_{in}	L_{in}	D_{Obst}	L_1	L_2	L_3	H_1
Wang (μm) [9]	1200 (2000)	100	300	200	-	60	-	100	300	90
LU	800 (1000)	-	150	100	300	40 (30)	100	50	150	45

in faster mixing. The simplest passive mixer is the T-shaped micromixer, constituted by two inlets leading two fluid streams into a main microchannel where they flow parallel. In order to increase the mixing efficiency of such a device, in [9] it was proposed to place obstacles in the main channel. Obstacles do not generate turbulence because of the low Re flow regime characterizing the flow, but their effect is to stir the fluid creating

transversal mass transport. In [9] the effect of eight different circular obstacle layouts on the mixing efficiency of an Y-sensor was evaluated by two-dimensional numerical simulations employing the MemCFD code by CoventorWave™. A sketch of this device is reported in Figure 1.

The LBM was adopted to evaluate the efficiency of a T-mixer under different flow conditions [8]. The approach is firstly validated with the cases reported in [9]: the different configurations tested are described in Figure 2, while Figures 3 and 4 illustrate the performance of two of these layouts. The efficiencies achieved by the different layouts are reported in Table I and compared to the corresponding ones in [9]. Very good agreement can be observed.

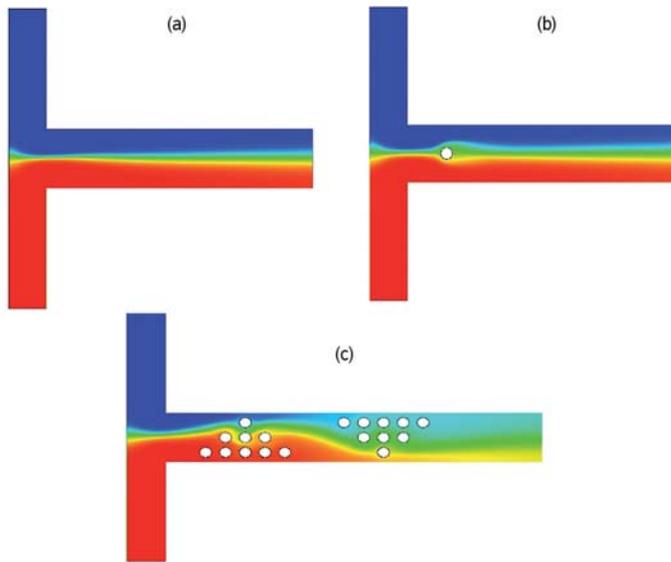


Figure 3: LBM simulations of T-shaped micromixer for configurations 1 (a), 2 (b) and 8 (c) from [9]. The pictures represent the mass fraction of water. The wider the green zone, the better the mixing

Configuration	Efficiency (%) [9]	Efficiency (%) (LBM)
1	18	17.2
2	21	22
3	42	31.5
4	44	43.8
5	46	46.2
6	32	31.4
7	53	51.3
8	62	60.8

Table 1: Mixing efficiencies evaluated on the outflow section of the device either in [9] or in the present study for the different obstacle layouts

Wall-particles interactions in creeping flows

Particles distribution in non-Brownian suspensions is greatly influenced by hydrodynamic cross-streamline migration induced by particle encounters. To achieve a complete understanding of such phenomena it is worth to start with the simplest possible configuration, consisting of two spherical particles interacting in a shear flow. At creeping-flow regime the Stokes equation holds: according to that in an unbounded flow the faster (trailing) particle overtakes the slower (leading) one and both particles return to their original transverse position once the encounter terminates. That means at least three particles are required for cross-streamline diffusion in absence of non-hydrodynamic forces breaking the symmetry of the flow. The presence of walls can significantly change this picture: as shown in [10], if the flow is bounded by walls and the channel height is compatible with the particle diameter a new class of trajectories can be observed in which the trailing particle does not overtake the leading one: the transverse component of the relative particle velocity changes sign, leading to an exchange of the vertical positions of the particles. The occurrence of these swapping trajectories depends on the initial particle displacement: in [10] the domain of the swapping trajectories was identified for the case of spheres having the same diameter. A Cartesian-representation algorithm was used in their study. The same case was studied with the LBM in [11] and it is here summarized.

The configuration of the system considered is sketched in Figure 4: the dispersed phase consists of two non-Brownian spherical particles, identified respectively as the faster, or “trailing” (particle 1) and the slower or “leading” (particle 2) one, characterized by diameters d_1 and d_2 . The characteristic length of the problem d is always assumed to be the diameter of the biggest particle. They move in a Couette flow characterized by a shear rate α in a channel bounded by walls on x - z planes separated by the distance H . The continuous phase has the same density of the particles and kinematic viscosity ν . We study the relative motion of particle 2 with respect to particle 1, thus we introduce the relative coordinates $\Delta r = (\Delta x, \Delta y, \Delta z)$. The unperturbed speed at a given height y is $U = \alpha y e_y$, while the maximum speed on top wall is U_0 . The other boundaries are periodic. The flow

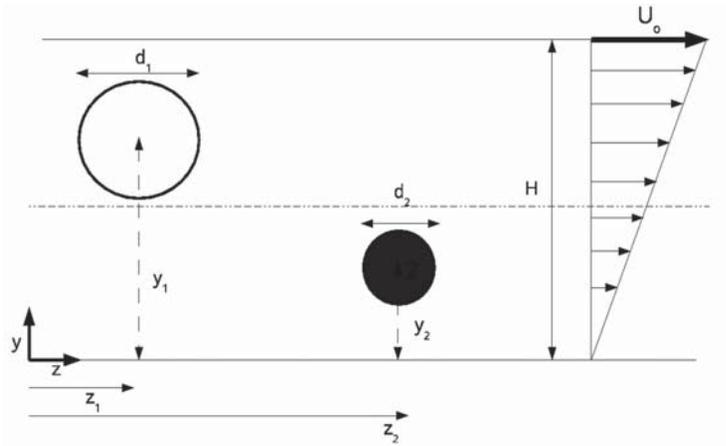


Figure 4: System description

is characterized by the shear Reynolds number $Re_s = \alpha(d/2)^2/v$, which is held equal to 0.001 in this study. The particle diameter is 20 in lattice units, unless differently specified. Three configurations have been considered in this study: firstly the case in which both particles have the same diameter, $d_1 = d_2$, then the case in which the leading particle has twice the diameter of the trailing one $d_1 = 0.5d_2$ and finally the case $d_1 = 2d_2$.

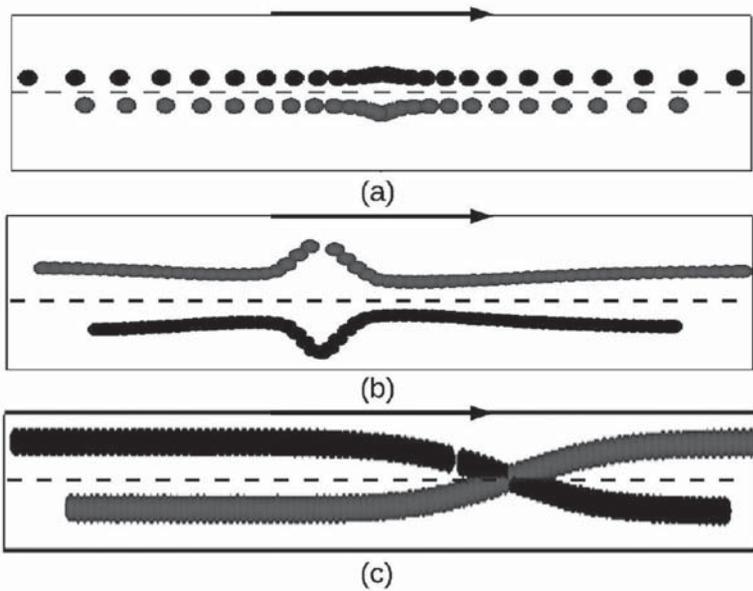


Figure 5: Different kind of observable particle trajectories in a Couette flow, with moving wall indicated by an arrow: a) unbounded case; b) bounded, non-swapping; c) bounded, swapping

All the different observable trajectories are sketched in Figure 5. In an unbounded shear flow the relative Δy increases while the leading particle is approached by the trailing one till a maximum is reached at $\Delta z=0$; the particles return then to their initial transverse positions. For bounded flow the outcome depends on the initial displacement Δy_0 . For sufficiently large Δy_0 the evolution is similar to the unbounded case, except for a minimum reached before $\Delta z=0$; decreasing Δy_0 can lead to completely different trajectories, characterized by a change in sign of Δy : the leading particle will never be reached by the trailing one, while there will be a swap of the transverse coordinates. In [10] it was shown that the swapping trajectories depend on the walls which scatter the perturbation flow induced by a moving particle, inducing motion towards the channel centre of the second particle.

Figure 6 reports the trend with time of the relative interparticle gap, defined as $\varepsilon = |\Delta r| / d - 1$ for the different initial transverse offsets Δy_0 reported in [10]. Here $H/d=5$. It is possible to observe how the occurrence of swapping prevents particle contact. In fact for swapping trajectories $\varepsilon = O(1)$, while the gap for non-swapping case can decrease down to $\varepsilon = 10^{-3}$. For the cases $d_1 = 0.5d_2$ (a) and $d_1 = 2d_2$ (b). The behavior is similar to the previous case, with the only difference that the minimum distance for the non-swapping cases is higher.

In order to evaluate the influence swapping trajectories exert on non-Brownian dilute particle suspension we have considered the Couette flow in a parallel-wall channel with $H/d=5$ of a mixture of two species of particles with same diameter. Particles of species one (indicated by diamonds in Figure 7a) are initially placed in the region $y < H/2$, while particles of species two (indicated by squares in Figure 7a) are initially placed in the $y > H/2$ region. The grid in this case is given by $100 \times 100 \times 600$ lattice nodes. A total of 1100 particles whose diameter is 10 lattice units, resulting in a volume fraction of $\phi = 0.1$, are placed in this domain. Ten simulations were performed averaging the results in order to reduce statistical fluctuations. The final condition related to one of these runs can be observed in Figure 7b, from which a substantial amount of mixing, which is entirely due to the swapping mechanism because of the low ϕ , can be appreciated.

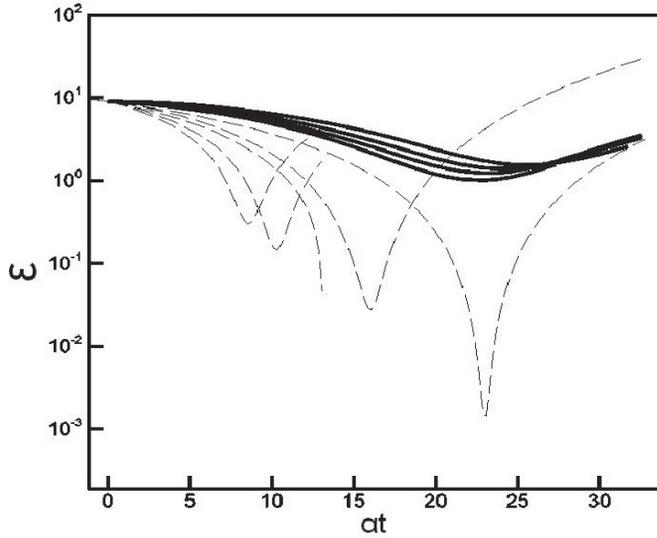
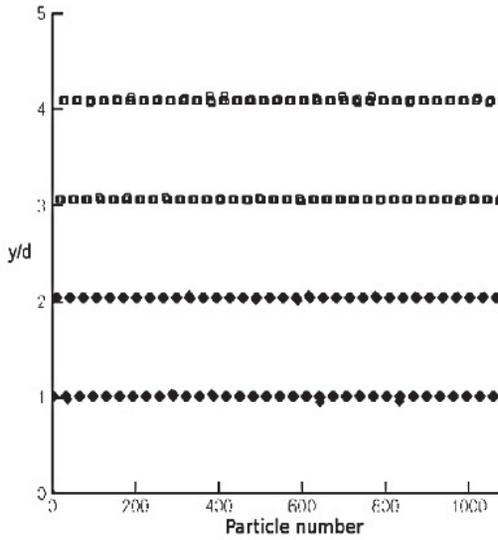
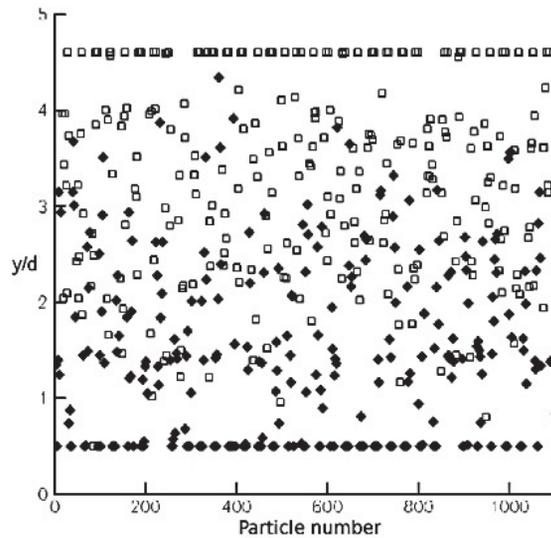


Figure 6: Temporal evolution of the interparticle relative distance (ϵ) for particles with same diameter. Swapping trajectories are indicated by dark curves



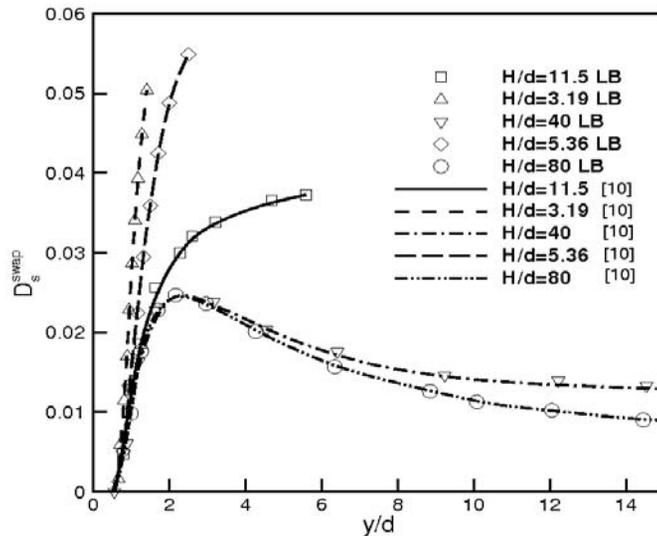
(a)



(b)

Figure 7: Initial (a) and final (b) distribution of particles of species 1 (squares) and 2 (diamonds)

Figure 8: Variation of D_s^{swap} with respect to y/d at different H/d values. The results are compared to those reported in [10]



To make a quantitative analysis we computed the dimensionless contribution D_s^{swap} given by the swapping trajectories to the self-diffusivity coefficient [10]: as it can be inferred from Figure 8 a satisfactory comparison with the analogous results reported in [10] for different values of H/d is achieved. In conclusion all the different kinds of trajectories predicted in [10] for particles with the same diameter were observed, reconstructing the range of initial particle displacement leading to particle swapping. The results match those reported in [10]. Besides some preliminary extension to the case of particles with different diameters was reported.

Conclusions

The goal of this report is to present the LBM as a valid and versatile tool for the numerical simulation of multiphase flows. The kinetic nature of the method allows a simple modeling and incorporation of hydrodynamical interactions, while its streaming-collide algorithm makes the method computationally very efficient and extremely suitable for modern processor systems. Both of the two cases presented will be extended in the near future: concerning the micromixer the next step is to pass to 3D simulations and to test different obstacle shapes, while the in-

fluence of swapping has to be assessed for bidisperse suspensions; Further extensions may consist respectively in studying the T-shaped sensor as a microreactor and in evaluating the influence of rough walls on the occurrence of swapping.

1. Literatur

- [1] S. Chen and G. Doolen, Lattice Boltzmann Method for fluid flows. *Ann. Rev. Fluid Mech.*, 30 (1999).
- [2] X. Shan and H. Chen, Lattice Boltzmann model for simulating flows with multiple phases and components, *Phys. Rev. E*, 47 (1993).
- [3] P. Yuan and L. Schaefer, Equations of state in a Lattice Boltzmann model, *Phys. Fluids* 18, (2006).
- [4] K.H. Luo, J.Xia and E. Monaco, Multiscale modeling of multiphase flows with complex interactions, *Journal of Multiscale Modeling*, 1 (2009).
- [5] E. Monaco, K. H. Luo and G. Brenner, Lattice Boltzmann simulations of binary droplet collisions by a multiphase multiple relaxation time model. "Lecture Notes in Computational Science and Engineering Series: Proceedings of Parallel CFD conference 2008". Springer Verlag (2009).
- [6] A. J. C. Ladd, Numerical Simulations of Particulate Suspensions via a discretized Boltzmann Equation. Part I. Theoretical Foundation, *J. Fluid Mech.*, 271 (1994).
- [7] N. Q. Nguyen and A. J. C. Ladd, Lubrification corrections for LB simulations of particle suspensions, *Phys. Rev. E*, 66 (2002)
- [8] E. Monaco, K. H. Luo and G. Brenner, Numerical investigation on the efficiency of a passive micromixer with the lattice Boltzmann method. In proceedings of ECCOMAS 2010, Fifth European Conference on Computational Fluid Dynamics". 14-17 June 2010. Lisbon, Portugal.
- [9] H. Wang, P. Iovenitti, E. Harvey and S. Masood, Optimizing the layout of obstacles for enhanced mixing in microchannels, *Smart Mat. Struct.*, 11 (2002).
- [10] M. Zurita-Gotor, J. Blawdziewicz and E. Wajnryb, Swapping trajectories: a new wall-induced mechanism in a dilute suspension of spheres, *J. Fluid. Mech.*, 592 (2007).
- [11] E. Monaco and G. Brenner, Influence of walls on the migration of non-Brownian spherical particles in creeping shear flow: a lattice Boltzmann study, accepted for publication in *Proc. R. Soc. Lond. A*.

4.3 Verfahren höherer Ordnung in Raum und Zeit mit Anwendung auf Gleitlagerberechnungen

Stefan Hartmann, Ahmad-Wahadj Hamkar, Torben Netz – Institut für Technische Mechanik

Darstellung des Projekts

In dem durch das Simulationswissenschaftliche Zentrum der TU Clausthal geförderten Projekt *Verfahren höherer Ordnung in Raum und Zeit mit Anwendung auf Gleitlagerberechnungen* war das Fernziel der Aufbau eines Simulationswerkzeuges zur Behandlung tribologischer Systeme. Hintergrund hierfür ist, dass tribologische Systeme derzeit vorwiegend mit einfachen Vorgehensweisen behandelt werden, die zum Teil aus der Dimensionsreduktion sowie aus der Vermischung von Raum- und Zeitdiskretisierung heraus resultieren. Der Nachteil, der hieraus entsteht, ist die nur grobe Vorhersage der gesamten physikalischen Problemstellung. Die Gesamterfassung des gekoppelten thermischen und dynamischen Systems aus unterschiedlichen Materialien und verschiedenen Raum- und Zeitskalen mit modernsten Diskretisierungsverfahren ist derzeit gerade in den Anfängen. Für kontaktfreie Problemstellungen ist insbesondere die Raum- sowie effiziente Zeitdiskretisierung von hohem Interesse.

Stand der Forschung

Die Erfassung des gesamten Prozesses eines belasteten Gleitlagers (es wird hier eine Beschränkung auf Radial-Gleitlager vorgenommen) behandelt

- das Fluid in Form des Schmierfilms,
- die Dynamik (Mehrkörperdynamik und Schwingungsphänomene) des Gesamtproblems bestehend aus dem Antrieb, dem Zapfen, der Lagerschale, etc.,
- die thermischen Effekte zum Beispiel durch Aufweiten des Zapfens und der Lagerschale,
- die Verschleißbestimmung durch lokale Kontaktsituationen,
- die Veränderungen der Materialeigenschaften des Fluides und der beschichteten Lagerbauteile sowie
- die vorliegende Fluid-Struktur Interaktion des gesamten tribologischen Systems.

Die beschichteten Lagerbauteile sind zudem thermisch und zyklisch stark belastet und stellen selbst wieder funktional gradierte Bauteile dar, siehe [2]. Gleitlager werden üblicherweise mit elastodynamischen Betrachtungen behandelt, d.h. wenn die Finite-Element Methode herangezogen wird¹, so dient hierzu die Annahmen des elastischen Festkörpers, siehe [9-10]. Aufgrund von hohen thermischen und mechanischen Belastungen ist die Eigenschaft der Elastizität jedoch in den oberflächennahen Schichten nicht gegeben. Zudem werden lediglich sehr vereinfachte Diskretisierungsmethoden herangezogen. Bei der Verwendung der Methode der finiten Elemente sind dies lineare oder maximal quadratische Ansatzfunktionen für die Raumapproximation sowie im Zeitbereich maximal Verfahren zweiter Ordnung. Diese Verfahren verletzen zum großen Teil die Energieerhaltung bei elastischen Systemen und liefern somit ungewollte numerische Dämpfungseigenschaften. Insbesondere die linearen Ansätze bei der Methode der finiten Elemente führen zu lediglich stückweise stetigen Verläufen der geometrischen Approximation und stellen daher bei Radial-Gleitlagern mit den dort vorliegenden gekrümmten Rändern den Nachteil einer groben Approximation der Geometrie dar. Dies kann nur umgangen werden, in dem eine sehr feine Raumdiskretisierung verwendet wird, was wiederum zu sehr rechenintensiven Problemen führt. Um diesen Themenkomplex insgesamt erfassen zu können, muss zunächst eine Reduktion der Fragestellung herbeigeführt werden, wobei in diesem Projekt nur der Festkörper und quasi-statische Berechnungen betrachtet werden sollen. Hierzu sollen modernste und hoch-effiziente Verfahren im Rahmen der Methode der finiten Elemente herangezogen werden.

In der Arbeitsgruppe Festkörpermechanik der TU Clausthal ist ein Finite-Elemente Programm für thermo-mechanisch gekoppelte, quasistatische Strukturberechnungen unter Berücksichtigung von Materialnichtlinearitäten in den letzten Jahren entstanden. Dieses basiert auf der vertikalen Linienmethode, bei der die Raumdiskretisierung mit Hilfe von finiten Elementen und der Zeitdiskretisierungsschritt mit Integrationsverfahren höherer Ordnung erfolgt. Das Programm TASA-FEM verwendet insbesondere schrittweitengesteuerte Zeitintegrationsverfahren höherer Ordnung in Form von steif-genauen diagonal-impliziten Runge-Kutta Verfahren sowie Verfahren

¹ Eine andere auf halb-analytischen Näherungen basierende Vorgehensweise ist zum Beispiel in [13,14] und der darin zitierten Literatur zu finden.

vom Rosenbrock-Typ. Die Anwendung von Rosenbrock-Verfahren haben bei glatten Fragestellungen den Vorteil, dass sie auf Rechenzeiten von lediglich 20 % - 30 % klassischer Finite-Elemente Berechnungen führen, da sie komplett iterationsfrei sind und lediglich auf lineare Gleichungssysteme führen, siehe [3, 6, 8]. Jedoch sind derzeit nur Materialmodelle der Viskoelastizität für kleine und finite Verzerrungen sowie ein Materialmodell der endochronen Viskoplastizität (fließflächenfreie Formulierung für Elastomere und Kunststoffe) angewendet worden. Diese Modelle stellen aus mathematischer Sicht "glatte" Problemstellungen dar, siehe [4, 5]. Im Bereich der Raumdiskretisierung liegen insbesondere klassische Finite-Elemente und eine gemischte Elementformulierung vor. Eine derzeit laufende Entwicklung stellt die Einbindung von Finiten Elementen höherer Polynomordnung dar (p-Version der Methode der finiten Elemente). Deren Vorteil liegt in der (natürlichen²) Berücksichtigung der exakten Geometrie in Form von "blending-functions" bei der innerhalb der FEM auftretenden Koordinatentransformation, siehe [1, 11], womit die Fehler in der Geometrieapproximation erheblich reduziert werden. Betrachtet man die jeweilige Fragestellung der Raum- und Zeitdiskretisierung, so stellen die Verfahren höherer Ordnung im Vergleich zu Verfahren erster Ordnung (sowohl im Orts- als auch im Zeitbereich) bei gleichen Genauigkeiten erheblich effizientere Vorgehensweisen dar, so dass damit dreidimensionale Berechnungen für die hier vorliegende Problematik sinnvoll möglich werden. Der Inhouse-Code TASA-FEM ist derzeit eines der sehr wenigen FE-Programme, welches Verfahren höherer Ordnung unter Berücksichtigung von Materialnichtlinearitäten ermöglicht. In der derzeitigen Version des Programmes liegt lediglich eine Kombination der p-Version der Methode der finiten Elemente mit steif-genauen, diagonal-impliziten Runge-Kutta Verfahren vor. Alle diese Betrachtungen sind derzeit für quasi-statische Berechnungen sowie instationäre Wärmeleitungsprobleme (und der Kopplung) erfolgt. Die Erweiterung auf dynamische Fragestellungen ist hierbei noch offen, jedoch ist aufgrund der Struktur des Programms die Erweiterung effizient möglich. Erste eindimensionale Untersuchungen hierzu sind in [12] für Runge-Kutta-Nyström Verfahren zu finden, die in ersten Voruntersuchungen als interessante Möglichkeit zur Behandlung in einem 3D-Code sinnvoll erscheinen.

2 Ohne die Verwendung von blending-functions erscheint die Einführung von p-Elementen unter Verwendung von Legendre-Polynomen nicht als geeignet.

Forschungsaufgaben

Um die genannten Lagerberechnungen behandeln zu können, sollen erste Voruntersuchungen und Weiterentwicklungen vorangetrieben werden. Zu den Zielen liegen nachfolgende Arbeitspunkte vor. Als erster Schritt sollen Rosenbrock-Verfahren höherer Zeitintegrationsordnung mit der p-Version der Methode der finiten Elemente gekoppelt werden. Zunächst ist hierbei ein Materialmodell der Viskoelastizität bei kleinen und großen Verzerrungen vorgesehen. Hierzu muss die Zeitdiskretisierung mit den von dem Datenmanagement komplexer aufgebauten p-Elementen implementiert und ausgetestet werden. Die Kopplung zu steif-genauen, diagonal-impliziten Runge-Kutta Verfahren, welche auf die Lösung großer nichtlinearer Gleichungssysteme in einem Zeitpunkt und damit auf eine iterative Vorgehensweise führen, ist der Ausgangspunkt der Untersuchungen und dient als Referenzlösung. Die Erweiterung auf Rosenbrock-Verfahren, deren Vorteil in einer komplett iterationsfreien Vorgehensweise liegt, ist in dem Projekt vorgenommen worden. Hierbei steht zunächst der Code-Verifikationsaspekt im Vordergrund. Als Standard-Beispiel eines Materialmodells wird ein Modell der finiten Viskoelastizität zur Berücksichtigung nichtlinearer Geschwindigkeitsabhängigkeit für Elastomere sowie ein Modell der finiten Viskoplastizität für Metalle gewählt. „Finit“ bedeutet in diesem Kontext die Behandlung beliebig

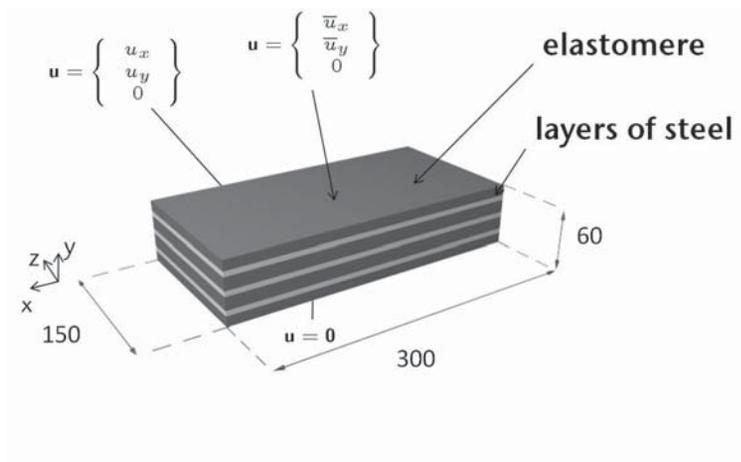
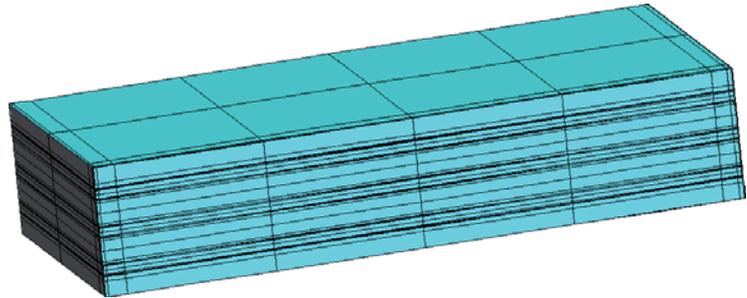


Abbildung 1: Geschichtetes Lagerbauteil aus Elastomer- und Metallschichten

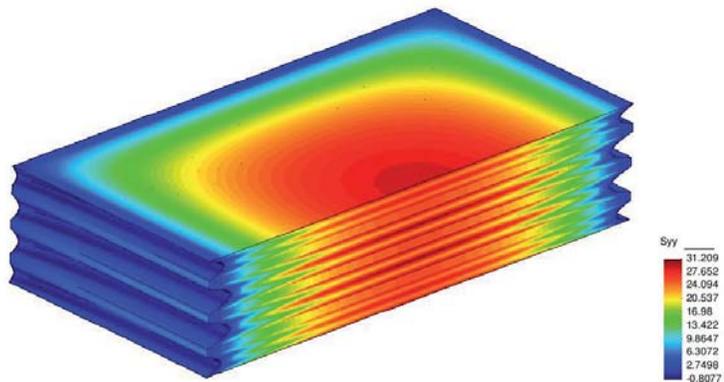
Abbildung 2: Diskretisierung mit p-Elementen



großer Verzerrungen und Verschiebungen. In Voruntersuchungen ergab sich, dass ein Polynomgrad von sechs erforderlich ist, um das nahezu inkompressible Materialverhalten des Elastomerwerkstoffes vernünftig zu approximieren, siehe [15]. Im Zeitbereich sind die numerischen Zeitintegratoren durch Ordnungsuntersuchungen auf deren erreichbare Ordnung zu untersuchen. Hierbei konnten die erwarteten Ordnungen der Verfahren erreicht und zeitadaptive Berechnungen durchgeführt werden. Abbildung 1 zeigt ein geschichtetes Lager und Abbildung 2 die zugehörige p-Elemente Diskretisierung. Die Deformation sowie die Spannungsverteilung einer Spannungskomponente im Lager zeigt Abbildung 3, siehe hierzu [18].

Ein weiterer Schritt ist die Erweiterung auf thermische und thermo-mechanisch gekoppelte Problemstellungen. Für den Fall der Thermo-Hyperelastizität zeigt sich auch, dass Rosenbrock-Verfahren zu sehr effizienten Berechnungen führen, siehe [16].

Abbildung 3: Normalspannungsverteilung und Deformation des geschichteten Lagers



Insbesondere der Vergleich verschiedenster Zeitintegratoren spielt zur Untersuchung der Effizienz und der Anwendbarkeit eine große Rolle [17].

Fazit und Ausblick

Der Effizienzgewinn bei der p-Version der Methode der finiten Elemente und der komplett iterationsfreien Zeitintegrationsverfahren vom Rosenbrock-Typ liegt nicht so hoch wie bei klassischen h-Elementen. Dies liegt daran, dass der wesentliche Aufwand bei p-Elementen in dem Aufbau der Steifigkeitsmatrizen und nicht in der Lösung der großen linearen Gleichungssysteme liegt. Trotzdem ist die Halbierung der Rechenzeiten beachtlich, da sie bei gleichen Genauigkeiten im Vergleich mit derzeitigen FE-Programmen (auch kommerzieller Art) noch einen sehr großen Vorteil haben. Eine Parallelisierung der Elementberechnung ist hierbei noch ausgeschlossen, sollte jedoch den Effizienzgewinn gegenüber klassischen Elementen noch erheblich steigern. Insbesondere die Verwendung der Rosenbrock-Verfahren, die zu einer komplett iterationsfreien Vorgehensweise führt, ist ein zukunftssträchtiger Ansatz.

Beteiligte Partner

Für Diskussionen zur Implementierung der p-Version der Methode der finiten Elemente sei Herrn Professor Alexander Düster (TU Hamburg-Harburg) recht herzlich gedankt. Im Rahmen der Zeitintegration lagen enge Verbindungen mit Herrn Dr. Joachim Rang (Institut für Wissenschaftlichen Rechnen, TU Braunschweig; ehemals Doktorand der TU Clausthal, Institut für Mathematik) vor. Hierbei entstand ebenfalls eine eingereichte Publikation, siehe [7].

Literatur

- [1] Düster, A.: High order finite elements for three-dimensional, thin-walled nonlinear continua. Shaker Verlag, Aachen, 2002.
- [2] Hagemann, T: Entwicklung eines thermodynamischen Berechnungsprogramms für kreiszylindrische Radialgleitlager mit dreidimensionaler Temperaturverteilung. Master's thesis, Institut für Tribologie und Energiewandlungsma-

- schinen, Clausthal University of Technology, Clausthal-Zellerfeld, Germany, 2006.
- [3] Hamkar, A.-W. und Hartmann, S.: Application of Rosenbrock-type methods to a constitutive model of polymeric material. Proceedings in Applied Mathematics and Mechanics, 7:4060045–060046, 2007.
 - [4] Hartmann, S.: Computation in finite strain viscoelasticity: finite elements based on the interpretation as differential-algebraic equations, Computer Methods in Applied Mechanics and Engineering, 191(13-14):1439–1470, 2002.
 - [5] Hartmann, S.: A thermomechanically consistent constitutive model for polyoxymethylene: experiments, material modeling and computation, Archive of Applied Mechanics, 76:349–366, 2006.
 - [6] Hartmann, S. und Hamkar, A.-W.: Rosenbrock-type methods applied to finite element computations within finite strain viscoelasticity, Computer Methods in Applied Mechanics and Engineering, 199(23-24):1455–1470, 2010.
 - [7] Hamkar, A.-W., Hartmann, S., Rang, J.: A stiffly accurate Rosenbrock-type method of order 2, submitted for publication, Applied Numerical Mathematics, 2011.
 - [8] Hartmann, S. und Wensch, J.: Finite element analysis of viscoelastic structures using Rosenbrock-type methods, Computational Mechanics, 40:383–398, 2007.
 - [9] Knoll, G., Backhaus, K., Lang, J., und Wilhelm, K.: Berechnung von Motorengleitlagern unter Berücksichtigung der Deformation, MTZ Motortechnische Zeitschrift, 59:652–656, 1998.
 - [10] Knoll, G., Reipert, P., Voigt, M., und Lang, J.: Tribologische und strukturdynamische Auslegung des Kolben-Zylinder-Systems durch rechnergestützte Simulation, <http://www.istaachen.com/pimo1.pdf>, Institut für Maschinenelemente und Konstruktionstechnik, Universität Kassel, 1998.
 - [11] Szabo, B. und Babuska, I.: Finite Element Analysis. Wiley, New York, 1991.
 - [12] Xin, P.: Untersuchung zeitadaptiver p-FEM bei dynamischen Systemen unter Berücksichtigung von Materialnichtlinearitäten. Diploma thesis, Institute of Applied Mechanics, Clausthal University of Technology, Clausthal-Zellerfeld, Germany, 2009.
 - [13] Wegmann, R. und Brökel, K.: Aktuelle Probleme der numerischen Simulation des hydrodynamischen und hydrosta-

- tischen Verhaltens von Spaltströmungen. In Brökel, Feldhusen, Grote, Riege, und Stelzer, editors, 6. Gemeinsames Kolloquium Konstruktionstechnik 2008, Nachhaltige und effiziente Produktentwicklung, Schriftenreihe Produktentwicklung und Konstruktionsmethodik, pages 291–303, Aachen. Shaker Verlag, 2008.
- [14] Wegmann, R., Kreja, S.-U., und Brökel, K.: Modernes Verfahren zur Simulation stationärer und instationärer Belastungen beim hydrostatischen und hydrodynamischen Betrieb von Radialgleitlagern unter Berücksichtigung von Wellenbiegung. In Gleit- und Wälzlagerungen. Gestaltung, Berechnung, Einsatz, Nr. 2069 in VDI-Bericht, pages 161–169. VDI, Düsseldorf, 2009.
- [15] Heisserer, U., Hartmann, S., Düster, A. und Yosibash, Z.: On volumetric locking-free behavior of p-version finite elements under finite deformations. *Communications in Numerical Methods in Engineering* 24, 1019-1032, 2008.
- [16] Hamkar, A.-W., Hartmann, S.: Theoretical and numerical aspects in weak-compressible finite strain thermo-elasticity, *Journal of Theoretical and Applied Mechanics*, accepted for publication, 2010.
- [17] Rothe, S., Hamkar, A.-W., Quint, K. J., Hartmann, S.: Comparison of diagonal-implicit, linear-implicit and half-explicit Runge-Kutta methods in non-linear finite element analyses, *Archive of Applied Mechanics*, submitted for publication, 2011.
- [18] Netz, T., Hartmann, S.: High-order time integration in the p-version of finite elements, *PAMM Proceedings in Applied Mathematics and Mechanics* 10, 201-202, 2010.

C-1

15:52:03

Start Simulation

A	B	C	D	E
T-0986-037	20110304-1558	Amsterdam	15:50	-1 min.
T-0986-038	20110304-1558	Amsterdam	15:54	2 min.
T-0141-006	20110304-1633	Bern	15:58	6 min.
T-0284-020	20110304-1709	Neu-Delhi	16:02	10 min.
T-0226-033	20110304-1640	Kopenhagen	16:06	14 min.
T-0226-016	20110304-1640	Kopenhagen	16:10	18 min.
T-0663-008	20110304-1655	Bukarest	16:14	22 min.
T-0141-027	20110304-1633	Bern	16:18	26 min.
T-0359-040	20110304-1735	Sydney	16:22	30 min.
T-0284-035	20110304-1709	Sydney	16:26	34 min.
T-0951-017	20110304-1748	Neu-Delhi	16:26	38 min.
T-0327-024	20110304-1719	London-Heathrow	16:30	42 min.
T-0663-045	20110304-1700	London-Heathrow	16:34	46 min.
T-0193-007	20110304-1709	Hongkong	16:38	50 min.
T-0284-036	20110304-1735	Bukarest	16:42	54 min.
T-0359-027	20110304-1655	Malmö	16:46	58 min.
T-0663-027	20110304-1719	Neu-Delhi	16:50	62 min.
T-0078-008	20110304-1709	Sydney	16:54	66 min.
T-0327-027	20110304-1735	Bukarest	16:58	70 min.
T-0284-021	20110304-1655	FEIERABEND	17:02	74 min.
	20110304-1800	Hongkong	17:06	
	20110304-1748	Neu-Delhi		
	20110304-1709			

C1
C2
C3
C4
IT-Ecosystem

5 Visualisierung, Datenanalyse und Software Engineering

In diesem Kapitel werden aktuelle Arbeiten aus dem Bereich der Modellierung, Simulation und Visualisierung sowie dem Software Engineering vorgestellt.

Im MASSim (Multi-Agent Systems Simulation) Projekt werden Verfahren der agentenbasierten Simulation auf Basis formaler Modelle untersucht. Eine Besonderheit von MASSim ist, dass seine Infrastruktur modular ist und es autark bzgl. der gewählten Agentenimplementations-technik ist.

Das Projekt „Verification of resource bounded agents“ beschäftigt sich mit Agenten, die zum Ausführen von Aktionen Ressourcen verbrauchen und daher nicht immer die theoretisch optimale Aktion ausführen können. Hier geht es also um die Modellierung von ressourcenbeschränkten Systemen, etwa in der verteilten Simulation, und die Möglichkeit, solche Systeme zu verifizieren.

Das Projekt „Environment Interface Standard“ stellt einen Ansatz dar, den wir in den letzten Jahren durch die Organisation unseres jährlichen Agent Contests entwickelt haben. Ausgangspunkt war die Feststellung, dass es keinen allgemeingültigen Standard gab, der die Trennungslinie zwischen Umgebung und Multiagentensystem festlegt. Ohne ein genau definiertes Interface muss für jedes Agentensystem die Simulation neu programmiert werden.

„Extending Repast Symphony for JADE Agent Behavior Components“ beschreibt ebenfalls eine Simulationsplattform, die auf dem Paradigma der Multiagentensysteme basiert. Für komplexe Softwaresysteme ist die Skalierbarkeit ein großes Problem, da die einzelnen Teile massiv miteinander interagieren. Methoden der Agentensysteme bieten sich hier an, da sie erlauben, verschiedene Sichtweisen (Mikro und Makro level) gleichzeitig zu modellieren.

Die Verifikation einer formal per VHDL beschriebenen Schaltung erfolgt heute üblicherweise teilweise per Software-Simulation und teilweise mit Hilfe von programmierbaren Logikbausteinen. In dem Projekt „Simulation und Realität bei der Chipsynthese mit VHDL“ wird untersucht, in wie weit sich dieser Prozess durch die gänzliche Umstellung auf softwarebasierende Simulation vereinfachen lässt.

5.1 MASSim (Multi-Agent Systems Simulation) Plattform

*Tristan Behrens, Jürgen Dix, Michael Köster, Federico Schlesinger
– Institut für Informatik*

Einordnung in die Forschungsgebiete

Multiagentensysteme sind mittlerweile als mächtige Werkzeuge zur Simulation von komplexen, dynamischen Systemen anerkannt. In den letzten Jahren wurden immer mehr Multiagentensysteme zur Simulation und zum besseren Verständnis von sozialen Institutionen, physikalischen, ökonomischen und biologischen Systemen eingesetzt.

Die MASSim Plattform [1] lässt sich zur Visualisierung, der Datenanalyse und zur Evaluierung des Multiagentensystems benutzen und somit in den Projektbereich „Verteilte Simulation“ einordnen.

Sie ist eine stabile Testumgebung für dezentrale Architekturen und für Konzepte bzgl. der Simulation. Das bedeutet, dass neben der Evaluierung der Koordinations- und Kooperationsansätze von Multi-Agenten-Systemen gerade auch die Funktion als Simulator für komplexe Szenarien bereitgestellt wird.

Eine Besonderheit ist, dass sie autark bzgl. der gewählten Agenten-Implementationstechnik ist. Zusätzlich ist durch die Modularität der Infrastruktur wenig Aufwand nötig, um das Simulationsszenario auszutauschen. Für die spätere Analyse werden sowohl eine Echtzeitüberwachung und Echtzeitbetrachtung des Simulationsverlaufs als auch zusätzlich animierte Videosequenzen bereitgestellt.

Lag der Fokus verteilter Simulationen bislang in der performance-verbessernden Parallelisierung und der Verteilung von Berechnungen, so wird es für die Zukunft vermehrt notwendig sein, das Verhalten von komplexen Systems of Systems zu analysieren und zu simulieren. Unter Systems of Systems versteht man dabei Systeme, die wiederum aus komplexen und heterogenen Systemen bestehen, in loser Kopplung miteinander interagieren und ein emergentes Verhalten haben.

Die traditionellen Methoden und Vorgehensmodelle der Simulation stoßen bei den Anforderungen dieser komplexen Systeme an ihre Grenzen, da sie die Dezentralität weder in der Ausführung, noch in der Modellerstellung nicht oder nur in Ansätzen unterstützen.

Darstellung des Projekts

Die MASSim Plattform basiert auf der Idee einer diskreten runden- und zeitbasierten Spielesimulation. Durch die Plug-in-Architektur ist es jedoch möglich, davon stark abweichende Szenarien zu erstellen. Solch ein Szenario, als Simulationsplug-in eingebunden, erlaubt die Realisierung einer Umgebung für eine MAS-Simulation.

Abbildung 1 zeigt die technische Infrastruktur. Die MASSim Plattform besteht aus verschiedenen Komponenten, die über mehrere Computer verteilt sein können.

Der eigentliche Kern (der Simulationskreislauf) ist in Java geschrieben und dient als zentrale Komponente für die Interaktions- und Kooperationsmöglichkeiten der einzelnen Elemente untereinander. Außerdem implementiert er den Ablaufplan des Turniers.

Das Szenario Simulationsplug-in, implementiert in einer Java-Klasse, beschreibt ein diskretes, schrittbasierendes Spiel. Außerdem beinhaltet es die Logik sowie die Funktionalität der MAS-Umgebung.

Bei der Kommunikation werden zwei Arten unterschieden: die Agenten-Server- und die Agenten-Agenten-Kommunikation. Während erstere über den Verbindungsmanager mittels TCP/IP und XML erfolgt, ist die zweite nicht über die Plattform realisiert und dem Agentenentwickler freigestellt. Eine Kommunikation der Agenten über den Server ist jedoch einfach zu implementieren.

Die MASSim Plattform besteht zusätzlich aus den Agententeams. Diese laufen nicht auf dem Server, sondern auf der Infrastruktur der Teilnehmer. Damit wird weder die Implementierungstechnik noch die Agentenkommunikationsmethode oder der Kooperationsmechanismus eingeschränkt.

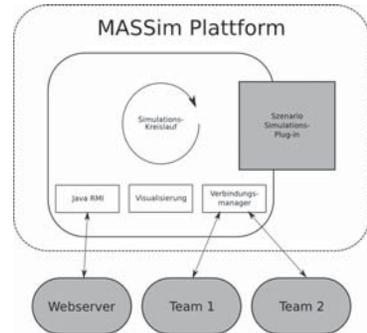


Abbildung 1: Übersicht über die MASSim Plattform

Eine weitere Komponente, die Visualisierung, ermöglicht die Darstellung des Spielgeschehens nach Simulationsablauf. Dazu existiert eine Java-Bibliothek, die beliebig skalierbare Animationen mithilfe von SVGs, also skalierbare Vektorgraphiken, und JavaScript erstellt. Sie sind plattformunabhängig in jedem Browser mit SVG-Unterstützung betrachtbar.

Um den Simulationsverlauf, also die Zwischenergebnisse schon während der Ausführung betrachten zu können, existiert zusätzlich ein Web-Interface. Es ist über Apache/Tomcat und Java Server Pages realisiert und läuft unabhängig von der Kernkomponente. Mit dieser kommuniziert es über Java RMI.

Stand der Forschung

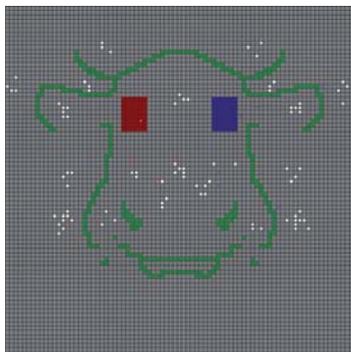


Abbildung 2: Spielfeld Cowboy-Szenario 2008

Die MASSim Plattform ist in der Praxis erfolgreich benutzt worden, so wurde sie dieses Jahr zum fünften Mal beim Multi-Agent Programming Contest [1] eingesetzt. In den Jahren 2006 bis 2007 ist das sogenannte Goldgräber-Szenario [2] verwendet worden. In 2008 [4] jedoch haben wir das Szenario (Abbildung 2) wesentlich erweitert und verändert. In einer Gitter-förmigen Umgebung versuchen Cowboys, die Agenten, in Kooperation mit ihren Teammitgliedern, Kühe in einen Stall zu treiben. Dabei können neben statischen Hindernissen, wie Bäumen, die Agenten des gegnerischen Teams die Arbeit der eigenen stark behindern. Wie auch bei allen vorherigen Szenarien haben die Agenten nur lokale Informationen, was bedeutet, der Agent sieht nur einen kleinen Ausschnitt der Karte und muss sich mit Hilfe seiner Team-Kameraden ein Bild von der Umwelt machen. Dieser Ansatz fördert also die Kooperationsfähigkeit der Agenten. Neben dieser entscheidenden Sichteinschränkung ist eine weitere wichtige Eigenschaft hervorzuheben: Aktionen des Agenten können fehlschlagen. Dadurch wird die Entwicklung von robusten und stabilen Agenten gefördert und ein fehler-tolerantes Verhalten belohnt. Die Kühe haben dabei ein unabhängiges Verhalten, sie folgen einem Herdentrieb, zerstreuen sich und meiden die Cowboys. Diese scheinbar kleine Änderung hat das Verhalten der Agenten grundlegend verändert. War es vorher noch möglich, dass Agenten ohne Kooperation wenigstens teilweise Erfolg haben konnten, so erzwingt die neue Beschreibung eine sorgfältig geplante Koordination und Kooperation. Auch bei diesem Szenario hat der einzelne Agent

nur lokale Informationen zur Verfügung und muss mit fehlgeschlagenen Aktionen umgehen können.

Für 2009 [1] (Abbildung 3) ist das Cowboy-Szenario leicht geändert worden. Die Kühe werden nicht mehr vom Spielfeld entfernt, sobald sie im Stall sind, sondern bleiben bis zum Spielende im Spiel. Außerdem gibt es ein neues Element: Zäune. Diese können die Agenten nur mithilfe mindestens eines anderen Agenten überqueren. Schließlich ist das Verhalten der Kühe verbessert worden. All diese Änderungen haben zum Ziel gehabt, sowohl den Wettbewerb um die Kühe zu verschärfen, als auch weitergehende Kooperation zu erzwingen.

Die wichtigste Neuerung für das Szenario in 2010 [1] war neben kleinen Änderungen an dem Verhalten der Kühe die Einführung eines neuen Scoring-Algorithmus. Anstatt wie in früheren Versionen die Gesamtzahl an Kühen für das Ergebnis heranzuziehen, wird nun ein Durchschnitt über alle Kühe pro Schnitt gebildet. Damit ist es nicht mehr möglich, destruktive Strategien kurz vor Ende der Simulation anzuwenden.

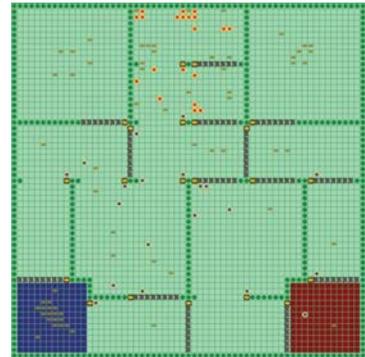


Abbildung 2: Kuh-Szenario 2009-2010

Forschungsaufgaben

Nicht nur die stetig steigende Anzahl der Teilnehmer und der damit verbundenen deutlich längeren Simulationsdauer, sondern auch die steigende Komplexität der einzelnen Simulation, verlangen nach Methoden und Mechanismen, ganze Simulationen oder aber auch Teile davon parallelisiert auszuführen. Neben der Möglichkeit damit auch wirtschaftliche Prozesse wie das Supply Chain Management auf elegante Weise simulieren zu können, verspricht dieser Parallelisierungsansatz, komplexe Simulationen zu ermöglichen.

Zusätzlich verstärkt dieser Ansatz die Robustheit der Plattform, da man mehrere Komponenten derselben Art im so verteilten MASSim-Server vorhalten kann.

Bei der Verteilung der Simulation ist jedoch zu beachten, dass bestimmte Safety-, Liveness- oder Fairnessbedingungen eingehalten werden. Dafür bietet es sich an, die Simulationen um logikbasierte Methoden zur formalen Verifikation zu erweitern. Zum Beispiel ließen sich Teile der Simulation mit Logik-Formeln

annotieren und dann mithilfe des Modell-Checkings [3] überprüfen.

Fazit und Ausblick

Der Agent Contest ist mittlerweile in seiner sechsten Ausführung und die Teilnehmerzahl steigt stetig an. Das zeigt, dass die MASSim Plattform eine wichtige Rolle für die Entwicklung von Multiagentensystemen spielt. Für die Zukunft soll deshalb zum Einen der Kooperationsaspekt weiter gefördert werden, zum Anderen soll die Plattform auch stärker für anders geartete Simulationen benutzt werden.

Beteiligte Partner

Während die Entwicklung der Plattform von der TU Clausthal vorangetrieben wird, wird der Agent Contest zusammen mit der Utrecht University (Mehdi Dastani), der Czech Technical University Prag (Peter Novák) und der Federal University of Santa Catarina, Brasilien (Jomi Hübner) organisiert.

Literatur

- [1] T.M. Behrens, M. Dastani, J. Dix, M. Köster, P. Novák, The multi-agent programming contest from 2005–2010, *Annals of Mathematics and Artificial Intelligence*, Springer Netherlands; Volume 59, Seite 277-311, 2010.
- [2] Dix, Jürgen, Behrens, Tristan, Köster, Michael, Novák, Peter. Agenten als Goldgräber und Cowboys, *KI - Zeitschrift Künstliche Intelligenz*; Volume 23(2):50–53, 2009.
- [3] Clarke, E., Grumberg, O., Peled, D.: *Model checking*. Springer, 1999.
- [4] T.M. Behrens, Mehdi Dastani, Jürgen Dix, and Peter Novák. Agent contest competition: 4th edition. In Koen V. Hindriks, Alexander Pokahr, and Sebastian Sardiña, editors, *Proceedings of the sixth international workshop on Programming Multi-Agent Systems (ProMAS'08)*, volume 5442 of *Lecture Notes in Computer Science*, Estoril, Portugal, 2009. Springer.

5.2 Verification of Resource-Bounded Agents

Nils Bulling and Jürgen Dix – Department of Informatics

Project Description

In this project we develop formal tools to model and to reason about resource-bounded systems. We are particularly interested in the verification of multi-agent systems in which agents do only have limited resources. We investigate the boundaries of what can and cannot be verified about such agents. We believe that for distributed simulation fairness and liveness properties might play a role. In addition, parts of such simulations might be model-checkable under certain abstractions. The theory developed in this project is a first step in this direction.

Introduction

The verification of multi-agent systems, in particular the model-checking problem (i.e. whether a given property holds in a given model), has attracted much attention in recent years [8,9,1,11,12,10]. Most of these results focus on well-established logics like the computation tree logics or alternating time temporal logics [9,1]. Just recently these logics have been extended to verify various aspects of rational agents [5,6]. However, the basic idea of rational agents being autonomous entities perceiving changes in their environment and acting according to a set of rules or plans in the pursuit of goals does not take into account resources. But many actions that an agent would execute in order to achieve a goal can in real life only be carried out in the presence of certain resources. Without sufficient resources some actions are not available, leading to plan failure.

The analysis and verification of agent systems with resources of this kind is still in its infancy; the only work, apart from our own, we are aware of in this direction is [2,3]. The authors extend **ATL** and coalition logic with resources. However, they only allow the production of resources which yields a decidable model checking property. The authors do also consider axiomatizations of their logics.

In our work we are rather interested in the boundaries of what can and cannot be verified. For this purpose, we also allow actions to produce resources which makes the problem undecidable in the general cases.

Framework and Results

The start of our project is formed by the analysis of single-agent resource-bounded systems. The proposed logics, resource-bounded tree logics [7], generalize the well-known computation tree logics by adding resources. Instead of asking for the mere existence of an infinite path satisfying some temporal property, this path must also be feasible given a set of available resources. The main operator $\langle \rho \rangle \gamma$ expresses that there is a computation feasible with resources ρ along which the (temporal) property γ holds. We show that the model checking problem for **RTL**, the less expressive variant of the resource-bounded tree logics, is decidable. In order to show this, we reduce the problem to non-trivial reachability problems over Petri nets. Other restricted settings which have a decidable model checking problem are identified.

After having obtained these results, we extend the setting to multiple agents. In our analysis it turns out that the handling of resources-in particular the production of resources-is harder than it may seem at first glance: We prove that in many settings the model checking problem is undecidable [4]. The underlying logics, resource agent logics [4], come in several versions of different expressibility. All of them generalise the strategic alternating-time temporal logics. The latter provide means to reason about strategic abilities of groups of agents. The operator $\langle\langle A \rangle\rangle \gamma$ expresses that the coalition A of agents can cooperate to guarantee γ regardless of what the other agents do. More precisely, agents in A have a winning strategy to enforce γ . In mathematical terms a strategy is a function that fixes the behaviour of an agent; that is, it determines an action for each situation. We consider two types of situations which affect the complexity of model checking. Agents can base their decisions on the current state only; or, they can base their decision on the whole history of events which have occurred. For the plain strategic case (i.e. the case of the alternating-time temporal logics) it is well-known that model checking is decidable

for both types of strategies (the complexity ranges from P to $2EXPTIME$, depending on the concrete setting [1]).

In order to take resources into account we assume that actions, executed by agents, may consume and produce resources. The proposed class of logics is called resource agent logics. It turns out that this simple idea has strong implications on the complexity of model checking. We show that all straight extensions of the alternating-time temporal logics have an undecidable model checking problem even for the simpler setting of memoryless (i.e. state-based) strategies. These results render the general problem of verifying agents with bounded resources as “unsolvable”. We consider restrictions with respect to several dimensions:

- Memory: Do agents base their decision on the current state only or on the complete history of events that have happened?
- Languages/temporal dimension: We introduce three languages which restrict the use of temporal operators.
- Resource-flatness: We restrict the use of nested operators.
- Proponent-restrictiveness: Do all agents have limited resources or only some of them (in particular, the opponents)?
- Semantics: What happens if we focus on infinite computations?

Combinations of these properties give rise to 32 logics. In our work we were able to prove that the great majority of them has an undecidable model checking problem. Some cases remain open and are left for future research. Surprisingly, all our undecidability results require only one or two agents. These results show that stronger restrictions seem necessary to make verification possible:

- Bounded settings: We put a bound on the number of resources an agent can have.

With a bound on the number of resources model checking becomes (trivially) decidable; however, we suspect that the computational complexity is huge. This is an open issue which we are going to analyse in this project.

Conclusions and Future Work

In the project we consider a setting important for rational decisions. So far, in most logics for strategic abilities actions do not

have any costs assigned to them. However, in practical applications this assumption is often unrealistic. The costs of actions significantly influence the selection of them. Rational agents should take this into account. Moreover, agents may not even have enough resources for some actions in some point in time. Our proposed logics, resource-bounded tree logics and resource agent logics, address these aspects. The focus has been on the question whether it is possible to verify agents which act rationally in this sense. We have shown that these modification essentially change the picture when it comes to the complexity of model checking. We have identified several variants and have shown that the interplay between specific properties (e.g. perfect recall vs. memoryless strategies) may yield an undecidable model checking problem in general.

In our future research we plan to close the open cases and plan to identify tractable settings by imposing appropriate restrictions to the logics. More precisely, we would like to address the following points:

- The decidability/undecidability of open fragments of the resource-bounded agent logics should be closed.
- The computational complexity of the decidable fragments of the resource-bounded agent logics are still open. It would be interesting to identify tractable fragments by suitable restrictions on top of the already implemented restrictions.
- A detailed analysis of the model checking complexity and the decidability question of the full resource-bounded tree logic is also still open. It would be particularly interesting to identify constraints which make the logics' model checking problems efficiently decidable for a relevant class of multi-agent systems.

Partners

Dr. Berndt Farwer, University of Glamorgan, U.K.

References

- [1] R. Alur, T. A. Henzinger, and O. Kupferman, 'Alternating-time Temporal Logic', *Journal of the ACM*, 49, 672–713, (2002).

- [2] N. Alechina, B. Logan, N. Hoang Nga, and A. Rakib, ‘A logic for coalitions with bounded resources’, in Proceedings of the Twenty First International Joint Conference on Artificial Intelligence, ed., Craig Boutilier, volume 2, pp. 659–664, Pasadena CA, USA, (July 2009). IJCAI/AAAI, AAAI Press.
- [3] N. Alechina, B. Logan, N. Hoang Nga, and A. Rakib, ‘Verifying properties of coalitional ability under resource bounds’, in Proceedings of the Second International Workshop on Logics for Agents and Mobility (LAM’09), ed., Berndt Farwer, Los Angeles CA, USA, August 2009.
- [4] N. Bulling and B. Farwer, ‘On the (Un-)Decidability of Model-Checking Resource-Bounded Agents’, in Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010), ed., Helder Coelho and Michael Wooldridge, Lisbon, Portugal, 2010.
- [5] N. Bulling, W. Jamroga, and J. Dix, ‘Reasoning about temporal properties of rational play’, *Annals of Mathematics and Artificial Intelligence*, 53(1-4), 51–114, 2009.
- [6] N. Bulling and W. Jamroga, ‘What agents can probably enforce’, *Fundamenta Informaticae*, 93, 81–96, 2009.
- [7] N. Bulling and B. Farwer, ‘Expressing properties of resource-bounded systems: The logics RBTL and RBTL’, in Post-Proceedings of CLIMA ’09, eds., J. Dix, M. Fisher, and P. Novak, LNCS, Springer, 2010.
- [8] E.M. Clarke, O. Grumberg, and D. Peled, *Model Checking*, MIT Press, 1999.
- [9] E.M. Clarke and E.A. Emerson, ‘Design and synthesis of synchronization skeletons using branching time temporal logic’, in Proceedings of Logics of Programs Workshop, volume 131 of Lecture Notes in Computer Science, pp. 52–71, (1981).
- [10] W. Jamroga and J. Dix, ‘Model checking abilities of agents: A closer look’, *Theory of Computing Systems*, 42(3), 366–410, 2008.
- [11] O. Kupferman, M.Y. Vardi, and P. Wolper, ‘An automata-theoretic approach to branching-time model checking’, *Journal of the ACM*, 47(2), 312–360, 2000.
- [12] A. Pnueli and R. Rosner, ‘On the synthesis of a reactive module’, in POPL ’89: Proceedings of the 16th ACM SIGPLAN-SIGACT symposium on Principles of programming languages, pp. 179–190, New York, NY, USA, 1 ACM, 1989.

5.3 Environment Interface Standard

Jürgen Dix und Tristan Behrens – Institut für Informatik

Motivation und Übersicht

Der vorliegende, technische Bericht hat den environment interface standard, kurz EIS, zum Inhalt. Eingordnet wird dieser in der künstlichen Intelligenz, genauer im sogenannten agent-oriented software engineering, das heißt in dem Teil der künstlichen Intelligenz, der sich mit dem Entwurf, der Entwicklung und der Analyse von Software beschäftigt, welche auf dem Agenten-Prinzip beruht. Dieses Paradigma unterscheidet zwischen Agenten, das heißt zielgerichteten, proaktiven, reaktiven und sozialen Softwarekomponenten, und Umgebungen, also den passiven Softwarekomponenten, in denen die Agenten situiert sind. Die Interaktion zwischen Agenten und der Umgebung basiert auf Wahrnehmungsobjekten und Aktionsobjekten. Ein Wahrnehmungsobjekt ist hier ein Objekt, welches einen Teilzustand der Umgebung repräsentiert. Eine Aktion ist hier ein Objekt, welches, falls ausgeführt, den Zustand der Umgebung verändert. Agenten setzen in der Regel eine sense-think-act-Schleife um, das heißt, dass der Agent wiederholt die Umgebung wahrnimmt und sich aufgrund seines Wissens über die Welt und den gesetzten Zielen entscheidet, welche Aktion er ausführen soll.

Der kritische Punkt bei unseren Betrachtungen ist in dieser Arbeit die Verbindung zwischen Agenten und der Umgebung, also der Schnittstelle, welche Aktionsobjekte transportiert und Wahrnehmungsobjekte liefert. Es hat sich ergeben, dass verschiedene, agenten-orientierte Plattformen, etwa 2APL, GOAL und Jason, jeweils eine eigene, plattformspezifische Umgebungsschnittstelle definieren. Diese Umgebungsschnittstellen stellten sich als nicht übertragbar heraus. Dies bedeutet, dass Umgebungen, welche für eine der genannten Plattformen entwickelt wurden, nicht ohne einen nicht zu vernachlässigen Aufwand unter den Plattformen ausgetauscht werden können.

EIS ist ein designierter Standard für Umgebungsschnittstellen, welcher mit der Absicht entwickelt wurde, den Austausch von

Umgebungen zwischen verschiedenen Agentenplattformen zu erleichtern.

Prinzipien, Metamodell und Implementation

Der Entwurf der Umgebungsschnittstelle folgt den folgenden Prinzipien, die sich aus der Untersuchung der oben genannten, agenten-orientierten Programmierumgebungen ergeben:

1. **Portabilität:** In der Regel soll der Austausch von Umgebungsschnittstellen wie folgt geschehen. Als Erstes wird die spezifische Umgebungsschnittstelle in Form einer sogenannten Jar-Datei geladen. Als Zweites wird das Multiagentensystem auf die Verwendung der Schnittstelle angepasst. Und schließlich wird das System ausgeführt. Wir glauben, dass die Verwendung von Jar-Dateien ideal für unsere Zwecke ist.
2. **Generalität:** Die Umgebungsschnittstelle soll nur minimale Anforderungen an Agentenplattformen und Umgebungen stellen. Es soll keine Ablaufplanung in Bezug auf die Ausführung von Aktionen geben. Dies bedeutet, dass Aktionen nur von der Plattform oder der Umgebung selbst, aber nicht von der Umgebungsschnittstelle geplant und ausgeführt werden dürfen. Weiterhin werden keine Anforderungen an eine etwaige Agenten-Kommunikations- oder Organisations-Struktur gestellt. Kommunikation kann sowohl auf der Plattformseite oder auf der Environment-Seite stattfinden. Die Art der Verbindung zwischen Umgebung und Umgebungsschnittstelle ist beliebig, d.h. dass etwa Techniken wie TCP/IP, Java Remote Method Invocation oder Java Native Interface ausdrücklich erlaubt sind.
3. **Trennung der Belange (separation of concerns):** Wir nehmen an, dass Agenten von der Umgebung getrennt sind und auf der Plattformseite leben. Agenten generieren in der von uns angenommenen Perspektive Aktionsobjekte und verarbeiten Wahrnehmungsobjekte. Auf der Umgebungsseite sind sogenannte kontrollierbare Entitäten (controllable entities), welche wiederum Aktionsobjekte verarbeiten und Wahrnehmungsobjekte generieren. Auf der Implementationsseite ist dies gleichzusetzen mit der Prämisse, dass es

ausdrücklich verboten ist, Agenten-Objekte auf der Umgebungsseite zu speichern. Zum Zwecke der Kopplung von Agenten-Objekten auf der Plattformseite und der kontrollierbaren Entitäten auf der Umgebungsseite verwaltet die Umgebungsschnittstelle Referenzen auf Instanzen beider Objektklassen in Form von Identifikatoren und eine Relation, der sogenannten Agenten-Entitäten-Relation (agents-entities-relation), die dafür sorgt, dass Agenten auf kontrollierbare Entitäten zugreifen können, um so mit der Umgebung zu interagieren.

4. Vereinheitliche Verbindungen: Die Umgebungsschnittstelle soll vereinheitlichte Mittel für die Verbindungen zwischen Agenten, der Plattform und der Umgebung anbieten, welche bestehende Ansätze nicht einschränkt. Agenten-Methoden sollen erlauben zu agieren und die Umgebung passiv bzw. aktiv wahrzunehmen. Plattform-Methoden sollen erlauben, dass Agenten kontrollierbaren Entitäten zugeordnet werden können. Und schließlich soll über Umgebungs-Verwaltungs-Methoden die Ausführung der Umgebung zu steuern sein.
5. Standards für Aktionsobjekte/Wahrnehmungsobjekte et cetera: Wahrnehmungsobjekte und Aktionsobjekte sollen konform zu einer Sprachdefinition, der interface intermediate language, sein.

Von diesen Prinzipien leitet sich das Metamodell (siehe Abbildung 1) ab. Das Metamodell besteht aus diesen Komponenten:

- Agent: Ein Agent ist ein Software-Objekt, welches agieren und wahrnehmen kann und darüber hinaus keinen weiteren Einschränkungen unterliegt.
- Umgebungsmodell (environment model): In der Umgebung leben kontrollierbare, welche Sensoren bzw. Aktuatoren mit begrenzter, lokaler Wahrnehmungs- bzw. Wirkungsreichweite zusammenfassen.
- Plattform: Diese ist dafür verantwortlich, Agenten zu instanzieren und auszuführen. Weiterhin nehmen wir an, dass die Plattform dafür sorgt, dass Agenten mit kontrollierbaren Entitäten verbunden werden.

- Umgebungs-Verwaltungs-System (environment management system): Jenes bietet ein Repertoire an Aktionen zur Verwaltung von Umgebungen an. Damit meinen wir die Initialisierung der Umgebung über Startparameter, Starten der Umgebung, Pausieren der Umgebung und Auflösen der Umgebung im Sinne der Freigabe all seiner Ressourcen (siehe Abbildung 2).
- Agenten-Entitäten-System (agents-entities-system): Dieses System verdrahtet einzelne Agenten fest mit einzelnen, kontrollierbaren Entitäten und leitet Wahrnehmungs- bzw. Aktionsobjekte weiter (siehe Abbildung 3).
- Umgebungs-Abfrage-Schnittstelle (environment querying interface): Diese erlaubt den Zugriff auf interne Daten von außen, etwa zum Erstellen von Statistiken. Dies umfasst sowohl Daten der Umgebungsschnittstelle als auch der Umgebung selbst, falls diese es zulässt.

Die Implementation von EIS basiert auf Java 1.6. Die generische Umgebungsschnittstelle liegt in zwei Formen vor. Ein Java Interface wird angeboten, welches zur Schaffung einer speziellen Umgebungsschnittstelle implementiert werden muss. Da dieses Vorgehen mit einem großen Programmieraufwand verbunden ist, wenn man die oben umrissenen Prinzipien wahren und das Metamodell beherrsigen will, wird auch eine abstrakte Java Class angeboten. Diese default implementation stellt bereits das Gros der benötigten Funktionalität bereit und muss nur noch durch wenige Methoden, welche umgebungsspezifisch sind, angepasst werden. Die interface intermediate language liegt in der Form eines abstrakten Syntaxbaums (abstract syntax tree, AST, siehe Abb. 4) vor.

EISifizierte Umgebungen und verwandte Projekte

Bis dato wurden mehrere Umgebungen EISifiziert. Wir gehen nur kurz auf ein paar ein. An der Universität Delft in den Niederlanden wurde eine Umgebungsschnittstelle für das Computerspiel Unreal Tournament entwickelt. Somit ist es möglich, Agenten zu programmieren, die die Akteure in dieser virtuellen Welt steuern. Weiterhin wurde eine Umgebungsschnittstelle für einen Aufzug-Simulator geschaffen. In einem simulierten

Hochhaus sollen Aufzüge intelligent programmiert werden um die Wartezeiten für einzelne Menschen so klein wie möglich zu halten. Schließlich wurde eine Umgebungsschnittstelle für das aktuelle Szenario des Multi-Agent Programming Contest implementiert.

Das erste verwandte Projekt ist MASSim. MASSim ist die Software-Architektur des bereits erwähnten Multi-Agent Programming Contests. Ein weiteres, verwandtes Projekt ist APLTK, das agent programming language toolkit, welches zum Ziel hat, diverse Agenten-Interpreter, Tools und Umgebungen miteinander zu verknüpfen.

Beteiligte Partner

Dr. Mehdi Dastani, Universität Utrecht, Niederlande.
Dr. Koen Hindriks, Universität Delft, Niederlande.
Dr. Rafael Bordini, Universität Rio Grande do Sul, Brasilien.
Dr. Jomi Hübner, Universität Santa Catarina, Brasilien.
Dr. Lars Braubach, Universität Hamburg.
Dr. Alexander Pokahr, Universität Hamburg.

Literatur

- [1] EIS Homepage, <http://sf.net/projects/apleis>.
- [2] Multi-Agent Programming Contest Homepage, <http://multiagentcontest.org>.
- [3] Behrens, Tristan, Hindriks, Koen, Dix, Jürgen: Towards an environment interface standard for agent platforms, *Annals of Mathematics and Artificial Intelligence*, Springer Netherlands, 2010.
- [4] Behrens, Tristan M. and Hindriks, Koen and Hübner, Jomi and Dastani, Mehdi: Putting APL Platforms to the Test: Agent Similarity and Execution Performance, 2010.
- [5] Behrens, Tristan M. and Dix, Jürgen and Hindriks, Koen V.: The Environment Interface Standard for Agent-Oriented Programming Platform Integration Guide and Interface Implementation Guide, 2009.
- [6] Behrens, Tristan M. and Dix, Jürgen and Hindriks, Koen V.: Towards an Environment Interface Standard for Agent-Oriented Programming, 2009.

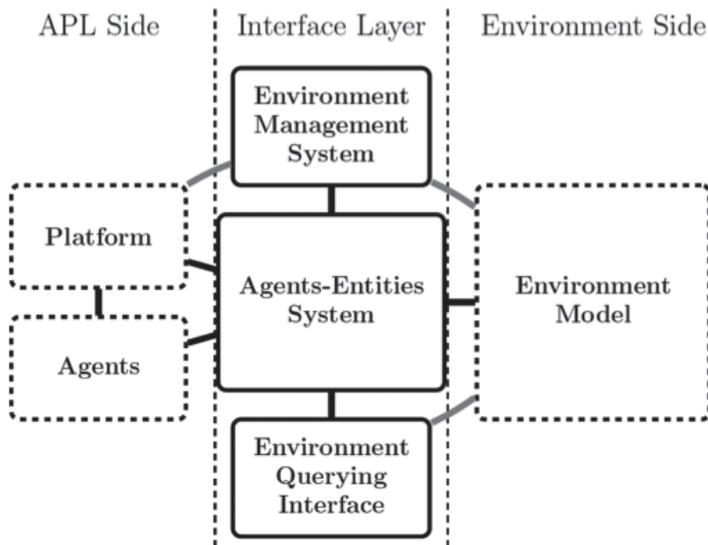


Abbildung 1: Das zugrundeliegende Metamodell

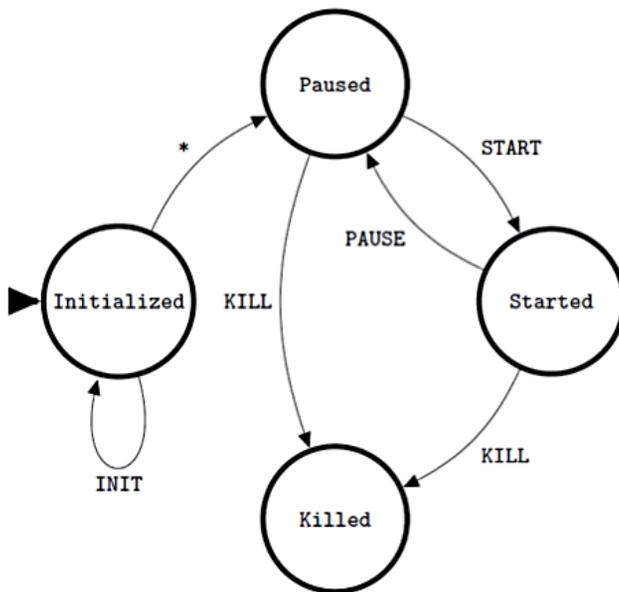


Abbildung 2: Der Kern des Umgebungs-Verwaltungssystems dargestellt als endlicher Automat

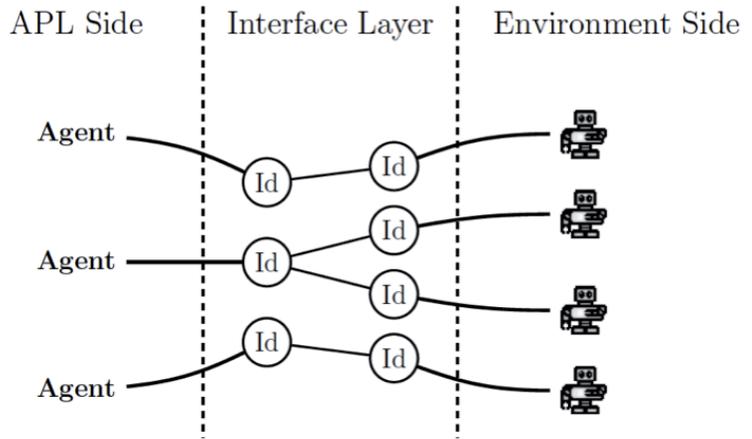


Abbildung 3: Das Agenten-Entitäten-System

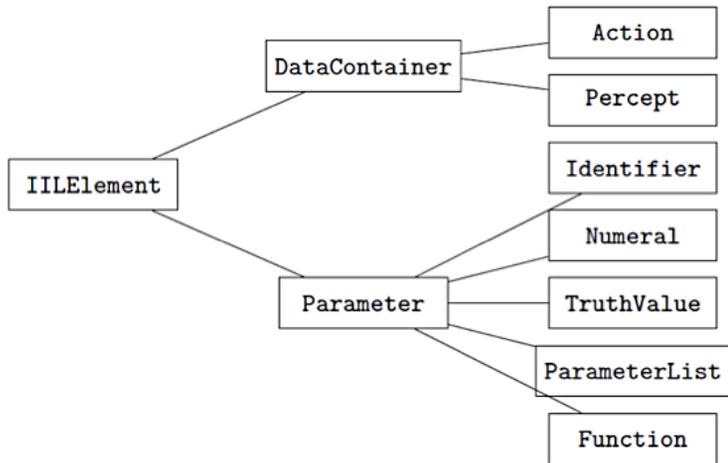


Abbildung 4: Die interface intermediate language als abstrakter Syntaxbaum

5.4 JREP: Extending Repast Symphony for JADE Agent Behavior Components

Jana Görmer, Gianina Homoceanu, Christopher Mumme, Jörg P. Müller, Michaela Huhn – Department of Informatics

Introduction

Classical approaches of computer science do not scale well for complex software systems. They are connected among each other and interact massively, therefore they cannot be considered in isolation but as parts of a complex system [8]. Nowadays, intensive modeling and simulation is part of various design and analysis phases of a system and its parts. As the requirements of various stakeholders as well as the different spheres of influence have to be taken into account: the macro level (i.e. the overall system level as a "God view" or "aggregated bottom-up view" evolved from a complex system which typically exhibits hierarchical self-organization under selective pressures), the micro level (i.e. agent behaviors), as well as the linking component the interaction level are relevant.

Multi-agent systems (MAS) [13] are a promising paradigm for constructing such systems with multiple methods for modeling and simulating. However, when developing a system, designers need to specify organizational structure, the individual agents' behavior, and their communication. Existing agent frameworks focus on either the macro for simulating Swarms or on the micro perspective to show agent's behavior.

Thus, our goal is to provide an efficient and usable simulation platform for complex systems supporting the macro and micro perspective linked by interaction. Technically, we combine Repast Symphony (Repast S) [7] and JADE [1].

The paper is structured as follows: in Section 2 we outline requirements for simulating a smart airport as one example of a complex MAS; in Section 3 we give an argumentation for using the platforms Repast S for macro level and JADE the micro level. Section 4 describes our JRep and in Section 5 we present our JRep tool on a coin flip for evaluating the performance and

on the smart airport showing details on the macro/micro/interaction level integration. Finally, Section 6 summarizes and concludes the paper with future work.

Requirements for Simulating a Smart Airport

In our research we focus on complex MAS which considers numerous heterogeneous agents and their interaction, an environment with different subsystems and control structures to handle the complexity of the system. As the requirements of various stakeholders as well as the different spheres of influence have to be taken into account, the macro perspective, i.e. the overall system level, the micro (objects' behaviors) view and the interaction (the intermediate layer between the micro-macro view) are relevant.

In the following we describe our example scenario and the resulting requirements for a simulation infrastructure: We evaluate our research topics by considering a smart airport where autonomous agents (TA) transport passengers between stations (entrances, check-in counters, gates, and plane parking positions). These stations are connected by two-lane roads and we define to be a subsystem (a region with its own norms) each area between two stations. Thus, in these regions agents have different rights and influences.

The TAs are equipped with batteries, which have to be recharged after some time at a charging station. The charging stations are designed as agents with the assignment to handle traffic jams that can occur, i.e. when a high amount of TAs want to recharge simultaneously. The TAs are signed in a service that collects passenger orders (pickup/drop positions, times). Each successful transportation of the passenger means a reward for the TA. This leads to competition on tickets, roads and charging stations in order to transport as much passengers as possible. For this, they can communicate with other agents that are within their horizon and perform joint tasks/goals.

To model the airport we propose three levels and further we will describe the requirements on each level:

Macro level technologies and techniques related to the system as a whole (i.e. structure). Therefore, the airport should be

viewed as a whole to see the agents move within the different regions. Analysis tools, good performance on execution speed and other general purposes and support for decision making processes should be available. In order to combine all the different subsections a step-by-step simulation with predefined function is helpful. Monitoring functions to see the simulation progression in real-time, displaying functions to show different data on predefined performance criteria are nice add-ons for the entire system view. The macro level represents the complex system with many interdependent individual actors working like clockwork. Agents do not have a full global view of the system and no central controlling agent [9], the macro level (complex behavior of the entire system) is built based on the individual strategies and the interaction (i.e. cooperation) with other individual agents. So we consider a bottom-up or decentralized approach to form the complex airport system. We also assume the existence of trusted institutions associated to different subsystems, which provide norms (e.g. traffic rules) and mechanisms to regulate the agent behavior in order to balance between different interests and to establish and sustain certain notion of stability at the global level.

Micro level (agent level) technologies and techniques concerned only with individual agents (i.e. procedures for agent reasoning and learning). Since an airport has many heterogeneous actors, it should be possible to implement different agents from a very simple execution robot to an intelligent agent which is able to make own decision based on its picture of the environment and goals. The agent needs to perceive the environment and act accordingly and autonomous.

Interaction level technologies and techniques that concern the communication between agents (i.e. communication languages, interaction protocols and resource allocation mechanism). Macroscopic behavior of the total system is constructed through local interactions or microscopic behavior of the agents. Therefore, a major role in such a system is the interaction between agents. Interactions play a crucial role also in terms of regulating the system, because global system state emerges as a side effect of the interactions of subsystems. Therefore, to achieve a stable equilibrium state we have to influence these interactions using regulation mechanisms. Such mechanisms are achieved

using the notion of norms and institutions to restrict the agents' autonomy and influence system behavior. This is primarily accomplished through communication, the essential functionality underlying an agent's social ability [13]. Through different competences and abilities, agents need to coordinate themselves for achieving their goals, also in an airport. Communication (i.e. exchange information) is necessary, but due to the agents' heterogeneity ontologies are also needed in order to be able to speak and understand the language. The agents have to have a common understanding of the language.

Rationale for choice of platform

There are several toolkits available for modeling and simulating complex systems as JADE, Swarm, NetLogo, Mason, Repast etc. and different survey over these toolkits have emerged in the past few years as [6]. Considering the requirements of our approach stated in the previous section, we have chosen Repast Symphony and JADE:

Repast Symphony is pure Java, which extends the Repast (open-source set of tools, originally based on Swarm) portfolio for developing agent-based models. One main feature of Repast S is the run-time GUI that allows the user to control the simulation during run-time by pausing, resetting and restarting different models. For example, the user can easily compare start configuration of different models with their behavior during run-time. Repast S provides also statistical functions such as graphs which are helpful for simulation monitoring and for real-time monitoring. Through defined variables for monitoring, graphs can be designed during development time. During run-time these graphs are displayed with actual variable's values and its history can be saved. Repast S supports the usage of external statistical programs, like R¹, for monitoring the simulation at run-time. Thus both features apply the model in the loop idea [10], so that the construction of a complex system, such as our airport, can be supported.

One primary purpose of Repast S platform is to control which specific actions are executed when, in simulated time. For this, Repast S provides explicit methods for scheduling actions, both in fixed time steps and dynamically.

1 <http://www.r-project.org/>

Furthermore in [11] is provided a comparison of Repast S with other software platforms (Swarm, NetLego, MASON) based on several criteria as model structure, scheduling or execution speed. The authors concluded that Repast S is a good Java simulation platform which can guarantee good performance on execution speed, but it has received poor feedback for modeling agents and their structure.

For our purpose Repast S incorporates the **macro level**. It is used for simulating the environment of our airport and for visualizing and monitoring the emergence of the macro behavior from collective behavior of different agents composing it.

JADE has a distributed system topology with peer-to-peer networking, and software component architecture with agent paradigm. JADE provides an agent programming framework, while in Repast S any Java class without an agent-based structure, can be defined as agent [11]. All tasks of agents are modeled as behavior objects. It implements either the entire task or only the intentions for implementing more complex ones. Thus JADE has no specific agent architecture. One can argue that Jadex could be also a solution, but it limits us to use its BDI agent architecture. Another advantage of JADE, it includes FIPA(-ACL) and interaction protocols where FIPA agent communication specifications deal with Agent Communication Language (ACL) messages.

Considering the previous stated advantages and the conclusion drawn from several evaluation reports that JADE is highly efficient in terms of performance on the agent message transport layer [12], the internal database access and message exchange capabilities [2], let us choose JADE as a modeling platform. Mainly, JADE considers the micro level, used for defining and implementing our own agent architectures (TA, charging station, etc.).

Interaction level plays a major role in our approach because it represents the intermediate layer that makes the connection between the macro and the micro level. As stated in Section 2 this is realized through interaction between agents. Here we make use of one of the JADE advantages, namely the support for FIPA ACL protocol. In ACL messages interaction protocols

are exchanged, which are based on the speech act theory. This includes communicative acts and content language representations [13]. JADE agents are processes of their own, which communicate remote and work decentralized with each other.

The combination of the both platforms makes use of the complement features provided by each platform. One example is the limitation of Repast S with the absence of an agent structure which is seen as a positive property that enables the combination with JADE, because the agent model in the simulation must be based on the JADE framework [14]. Another example is the set of predefined functions offered in Repast S for allowing step-by-step simulation, planning schedules, visualization and monitoring of the global systems (the macro view) which are the main issue in JADE.

Combining JADE and Repast S: JREP

In this section we cover the JREP architecture by the following key issues:

- **Macro level:** For the overall system we present the technical implementation of our JREP architecture and the scheduling process of the agents during runtime.
- **Micro level:** For a single agent we created our architecture in Jade and we explain the registration of an agent to the Repast S environment.
- **Interaction level:** The interaction between agents is provided by FIPA ACL.

JREP Architecture

As stated in Section 3 we use JADE and Repast S for a common JREP platform. One goal is that JADE agents can act in Repast S. Thus, we assume three minimum requirements needed by agents for acting in a simulation environment (cf. black box agent model by [5]):

- 1) Perception (Input): Agents must be able to sense the environment. Thus, they must perform “Perception” and thereby receive information regarding the environment.
- 2) Agents must be able to make plans and execute them, in

order to achieve goals (e.g. path finding algorithms in the airport).

- 3) Action (Output): Agents must perform actions and they also have to be informed about the success or failure by performing that action.

To enable these requirements, agents must have access to the simulation environment. In addition, the simulation environment must inform the agents about the current simulation time.

Considering the previous statements, an architecture of the common platform is developed (see Figure 1) with the following components.

The main idea to integrate JADEAgents into Repast S is to use an agent wrapper class that is implemented in Repast S:

RepastAgentRepresentation: Each agent has exactly one representation within Repast S, which is responsible for the execution of *Perception* and *Action*. Therefore, the class *RepastAgentRepresentation* possesses the methods *perception()* and *scan()* (cf. Fig. 1). Furthermore the agents are informed of each new Tick by the *RepastAgentRepresentation*.

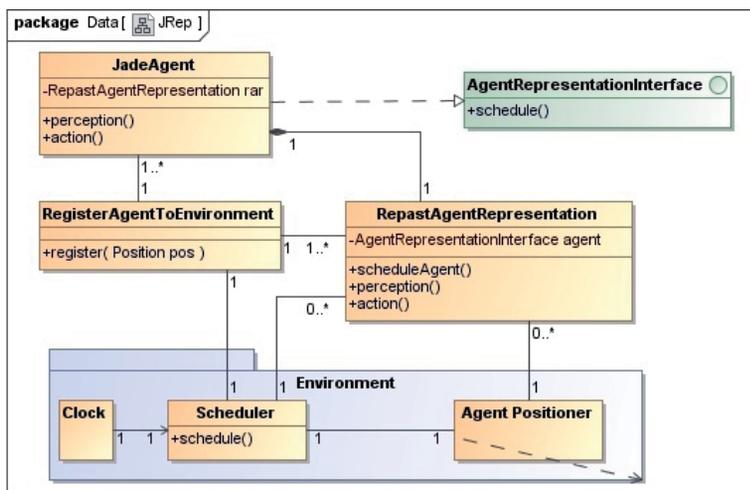


Figure 1: Class diagram of the JRep approach

JADEAgent: The JADE agents can be connected to Repast S (and/or the appropriate *RepastAgentRepresentation*) directly by references. In order to avoid the problem of *Polling* occurring in [14], the communication between agents and the Repast S simulation environment must be made possible in both directions. First agents are informed about the new system time of Repast S through the `scheduleAgent()` method of the *RepastAgentRepresentation* class and then they can act directly on the Repast S platform (cf. Fig. 1). For this, it is necessary that the agents implement the Interface *AgentRepresentationInterface* with the `schedule()` method. Thus, a *RepastAgentRepresentation* object receives the reference from the agent (that is of the type *AgentRepresentationInterface* at the same time). The agents instantiate a *RepastAgentRepresentation* object independently when they log into the environment and also have the reference on the *RepastAgentRepresentation* object (cf. Section 4.3). In a similar way, one can connect also Repast S agents to the simulation environment by using references.

Repast S already possesses an internal clock, where each *Tick* represents a simulation step. However, in JADE there is no given internal clock. If a JADE agent acts in a Repast S environment, it must be synchronized with Repast S's internal clock. Thus, our JREP architecture provides the concepts of *Clock* and *Scheduler*:

Clock: The Repast S Clock gives the cycle of the simulation time and informs the Scheduler component.

Scheduler: The Scheduler conducts the execution of actions by Repast S connected agents. In each Tick the *RepastAgentRepresentation* is informed about executed actions.

Further we will describe the scheduling process, providing first the necessary steps for this (see Figure 2). The Tick is triggered by the system clock (1.) and the scheduler successively contacts all connected agents in the loop (e.g., randomly). Thus the `scheduleAgent()` method (2.) of the *RepastAgentRepresentation* class is called. At the same time a flag is set in this class (3.), to denote that the agent was informed about the new Tick. The flag is released after the agent takes an action (a movement to the north, south, east, or west, or a delay) (16.). This is necessary to guarantee that the scheduler allows only one agent to

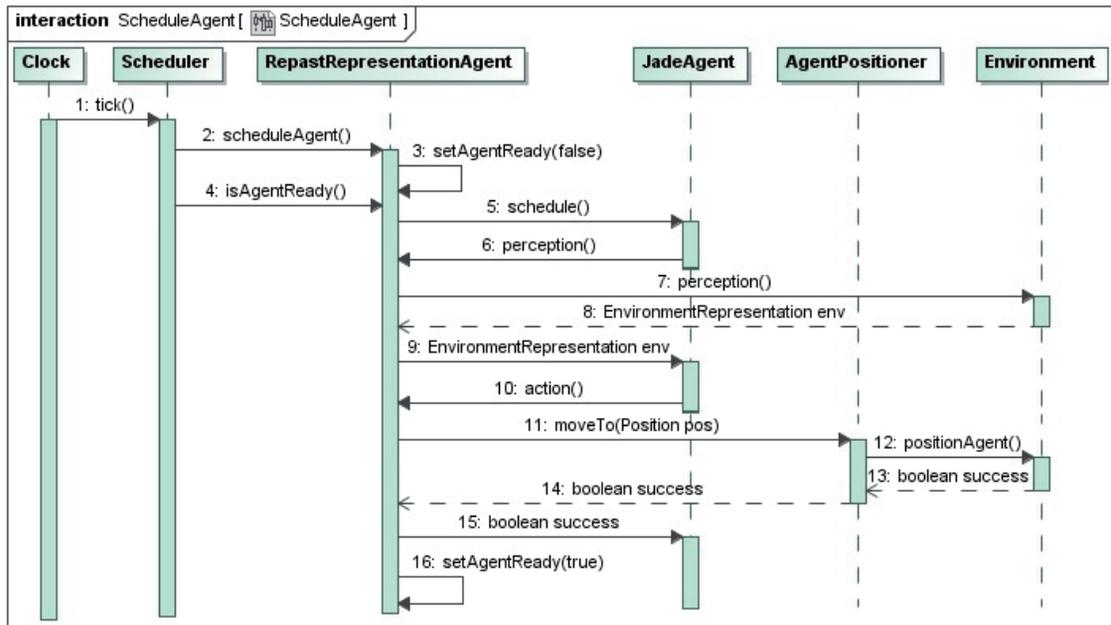


Figure 2: Scheduling agents in JRep

perform an action (in order to keep the agents synchronous to simulation). After the scheduler informs the agent (2.) it asks repeatedly to sets its flag (4.) and informs the next agent only if the flag is approved. The JADE agent performs perception (6.) only after receiving the information about the new Tick (5.) from its *RepastAgentRepresentation* object. This is achieved in the environment by the *RepastAgentRepresentation* object (7.) and it gives the information about the environment to the agent (8. and 9.). Now the agent can react or deliberate and can make an action in the environment (11.). Note that we also define “waiting” as an action. Then the *RepastAgentRepresentation* object tries to execute the movement (11.) by activating the agent positioner (12.). The agent receives information concluding whether the movement was successful (13., 14. and 15.).

Open platform

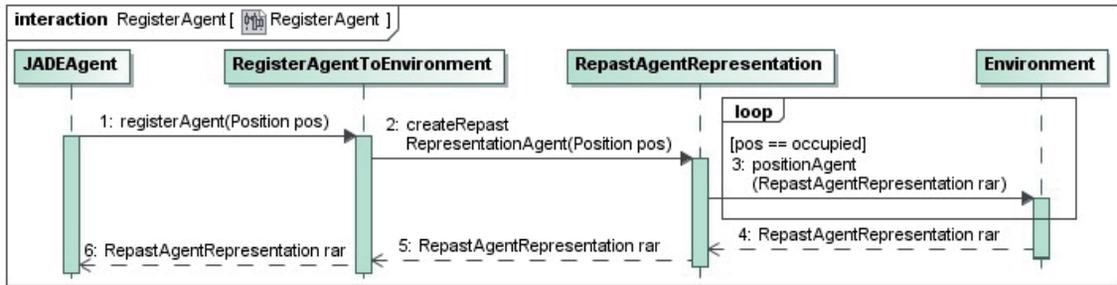
In our platform we implement the agents as JADEAgents, but it is also possible to connect other agent programming (AP) platforms (i.e. Jason, Jack) to JRep. A first approach to realize this is to enable Repast S to have a reference to the JADEAgent and

vice versa. This is achieved by representing an agent in JRep by a *RepastAgentRepresentation* wrapper class and by the fact that the *JADEAgents* implement the *AgentRepresentationInterface* Interface. A more general view of the approach considers that each Java agent can implement the *AgentRepresentationInterface* Interface. Thus, both Repast S and the Java agent can call methods using references from each other, because both are based on Java. Another approach for connecting other AP platforms with Repast S is to use Remote Procedure Calls (RPCs). In this case the *RepastAgentRepresentation* class must allow an agent to call appropriate RPC methods from remote (i.e. the methods must be available). RPCs connect any AP platform independent from its programming languages to Repast S. For instance one can implement XML RPCs for information exchange (i.e. perception and action) between Repast S and an agent. However, this approach can slow down the overall system scalability, because the communication between an agent and the simulation environment can take more time than a simple call by reference. For our purpose we have implemented JRep with *JADEAgents* that are connected to Repast S by references.

Agent perspective

While in Section 4.1 we present how to combine a *JADEAgent* with Repast S from an overall perspective, in this section we focus on the single agent perspective. First of all we describe the connecting process to the simulation platform:

A *JADEAgent* can be added to the simulation environment at the simulation run-time. The class *RegisterAgentToEnvironment* is implemented in Repast S which makes the connection between Repast S and JADE. It ensures that new agents are positioned initially. In Figure 3, we illustrate the procedure for connecting a new agent to the simulation platform. After instantiation of an agent, it must receive a reference to the appropriate *RegisterAgentToEnvironment* object (e.g., by delivery in the constructor or by web services). Now, the method *registerAgent(Position pos)* in the *RegisterAgentToEnvironment* class (1.) is called. The agent gives the preferred coordinates to start in the environment. Then the *RegisterAgentToEnvironment* generates a *RepastAgentRepresentation* object for that agent (2.). This places the agent in the desired position if that space is



available (e.g. not occupied by another agent) (3.). If the agent is placed, the reference of the *RepastAgentRepresentation* object will be handed over (4., 5. and 6.).

Figure 3: Register a new agent to the simulation platform

Although for our airport scenario we have implemented our own agent architecture [4], our platform allows also the implementation of different kind of agents: i.e. simple reactive agents or agents with complex architectures, due to the use of JADE which comes with no specification about an agent architecture, but provides some features that simplifies the implementation of agent.

Agent interaction

In MAS, agents can interact with each other either through the communication or by reacting to the behavior of the other agents. We have chosen the first approach. However, in order to communicate agent must be able to understand each other. This can be achieved by using a common language. In our approach we use FIPA ACL, because it is a scientifically recognized standard that fits our requirements. Figure 4 shows two agents that implement a communication module based on FIPA ACL.

For a common language FIPA ACL provides ontologies on basis of vocabularies. As a feature we implemented the appropriate ontology in Repast S, so that each agent connected to the environment "speaks" the same language. For communication the agents exchange ACL messages.

Evaluation

A smart airport demonstrator, currently under development,

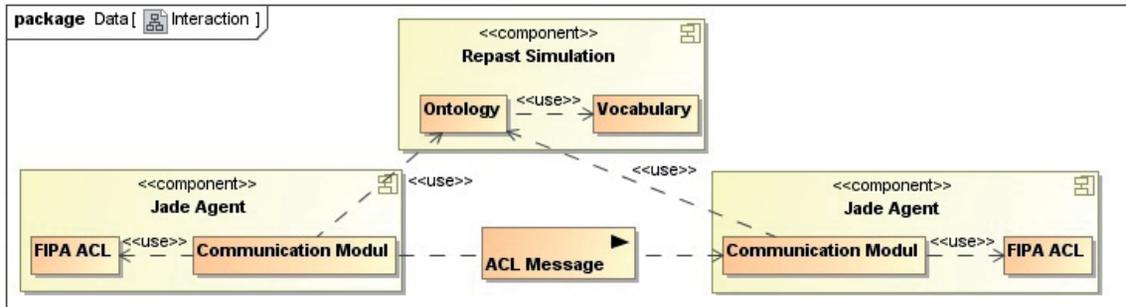


Figure 4: Communication between agents

built on the JREP platform, will present the results of our research, in the case of autonomous transportation services at an airport's departure area [3]. Here, we will present two scenarios to demonstrate the capabilities of the JREP platform. First, a study regarding JREP's scalability and performance is made on a coin flip scenario in Section 5.1. Then, we show details of JREP on our airport example in Section 5.2.

Coin Flip Scenario

In this section a derivation of a coin flip scenario is described. It is based on the assumption that autonomous agents work in two teams. Each team is allocated one side of a coin. Thus, the task of the teams becomes flipping the coins such that their allocated side of the coin shows on top. One important aim of this scenario is to show that for certain parameters the count of both sides of the coins is in equilibrium.

In the implementation of the scenario the agents are located in a grid based environment. The teams are of equal size. The coins are represented by blocks (colored red or blue) which can be flipped by the agents. The scenario has been first tested for 100 agents working on an 30 by 30 grid and 200 blocks. The agents look for the nearest block not being of its team color and move (by A* search algorithm) to the block to flip it.

Scalability means that a system or platform can handle growing amounts of work load, whereas performance is defined as the amount of work that is finished by a system compared to the time and resources used. Thus, we have tested the coin flip scenario with different starting configurations: A 500 by 500

grid and 100/500/1000/5000/10000 agents/blocks. In order to validate the scalability we compared the memory footprints of each start configuration. The performance is analyzed by counting the simulation steps (ticks) within the first 10 seconds of simulation. Table 1 shows the results from these tests. It can be seen that the ticks decrease linear, whereas the memory footprint increases super linear. We interpret these results positively, because JREP still scales and performs well for 10000 acting agents.

Configuration (Grid;Agents;Blocks)	Ticks in 10 seconds	Memory Footprint
500x500;100;100	725	2 MB
500x500;500;5000	553	14 MB
500x500;1000;1000	335	37 MB
500x500;5000;5000	261	169 MB
500x500;10000;10000	148	332 MB

Table 1: Scalability and performance of JRep

Airport Scenario

Figure 5 shows a first implementation of our scenario. The airport infrastructure is represented by a grid defined in Repast S. It is defined by a set of environment elements (e.g., roads) that are arranged at design time. In order to sign in to the environment the agents (e.g. TAs, charging stations) have to connect to a registry (cf. Fig. 3) at runtime. This registry stores references to agents for accessing their properties (e.g., positions). Manipulations of the environment are based on interactions among agents which requires communication. Based on JADE framework, we implemented an ontology to support ACL-based communication. The regions (subsystems) are realized in the registry as mappings of agents to sets of norms and infrastructural restrictions. Institutions are realized as static agents, but not visible in the infrastructure. It defines norms for its region. It has unrestricted access to the agent references in registry in order to observe behavior, enforce norms, and give rewards.

Conclusion and Future Work

As one example of a complex MAS we demonstrated on a smart airport the integration of Repast S and JADE in a common platform called JRep, which provides the machinery to comfortably

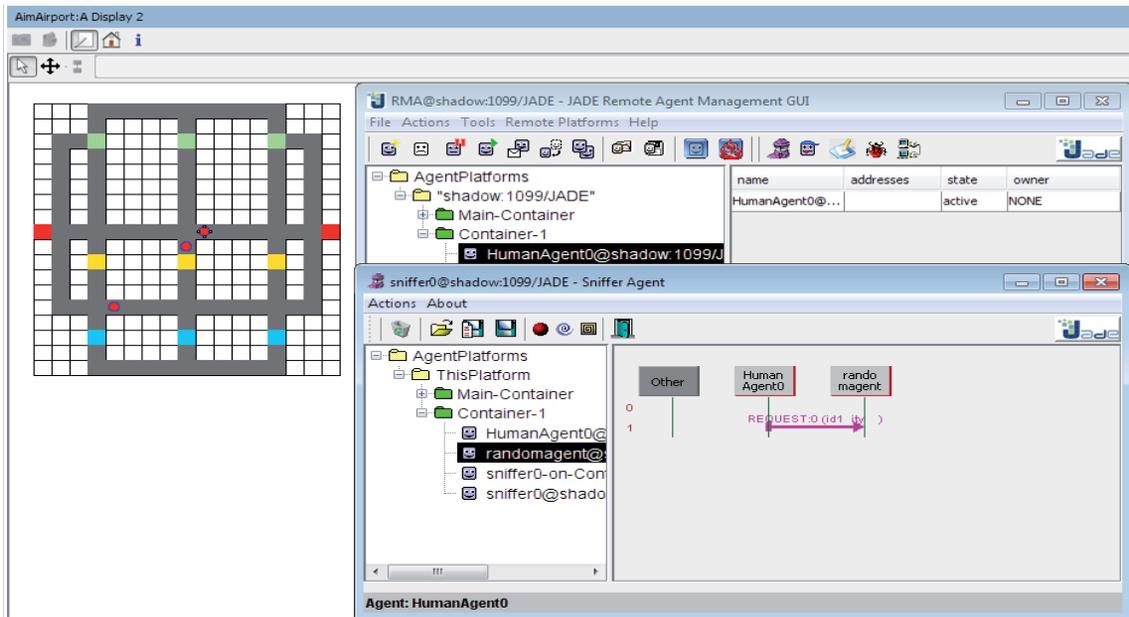


Figure 5: Communication between two agents

model the behavior and interactions on the agent level (micro level), as well as the instruments to investigate the emerging effects on the overall system (macro level). JRep uses the strength of both platforms, by efficiently combining their complementary features. Previous work [14] has also addressed the combination of JADE and Repast, but their approach has some drawbacks. Our JRep simulation platform is an open platform that enables the connection with other agent programming platforms. It is sufficiently generic that it can be used for different application fields. Importantly, we have developed generic concepts and an architecture which provides a novel contribution in itself. We showed how our platform handles (time) synchronization with the environment, scheduling of agents and registering of new agents. Two case studies demonstrated the flexibility and scalability of our approach. In the near future we will consider the usage of a standardized interface between the agents' behavior and the environment as it has been proposed in [1]. A standardized interface to the environment will not only ease the integration to alternative simulation and execution backends, but also improve separation the concerns as the environment is decoupled from system level issues like agents' registry and scheduling.

References

- [1] F. L. Bellifemine, G. Caire, and D. Greenwood. *Developing Multi-Agent Systems with JADE* (Wiley Series in Agent Technology). Wiley, April 2007.
- [2] K. Chmiel, D. Tomiak, M. Gawinecki, P. Karczmarek, M. Szymczak, and M. Paprzycki. Testing the efficiency of jade agent platform. In *Proc. of the 3rd Intern. Symposium on Parallel and Distributed Computing, ISPDC '04*, pages 49–56. IEEE Computer Society, 2004.
- [3] C. Deiters, M. Köster, S. Lange, S. Lützel, B. Mokbel, C. Mumme, and D. N. (eds.). *Demsey – a scenario for an integrated demonstrator in a smartcity*. Technical Report 2010/01, NTH Research School for IT Ecosystems, Clausthal University of Technology, 2010.
- [4] M. Huhn, A. Duraslan, G. Ganceanu, J. Görmer, J. Hähner, J. Müller, C. Müller-Schloer, C. Mumme, and C. Schulz. *AIM – Adaptive Interaction Mechanisms: Meta-model and architecture*. Technical Report, NTH Research School for IT Ecosystems, Clausthal University of Technology, 2010.
- [5] J. P. Müller. *The Design of Intelligent Agents – A Layered Approach*, volume 1177 of *Lecture Notes in Computer Science*. Springer, 1996.
- [6] C. Nikolai and G. Madey. Tools of the trade: A survey of various agent based modeling platforms. *Journal of Artificial Societies and Social Simulation*, 12, 2009.
- [7] M. J. North, T. R. Howe, N. T. Collier, and J. R. Vos. *The Repast Symphony Runtime System*. In *Proc. of the Agent 2005 Conf. on Generative Social Processes, Models, and Mechanisms*, 2005.
- [8] L. Northrop, P. Feiler, R. P. Gabriel, J. Goodenough, R. Linger, T. Longstaff, R. Kazman, M. Klein, D. Schmidt, K. Sullivan, and K. Wallnau. *Ultra-Large-Scale Systems – The Software Challenge of the Future*. Technical report, Software Engineering Institute, Carnegie Mellon, June 2006.
- [9] L. Panait and S. Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434, 2005.
- [10] A. R. Plummer. Model-in-the-loop testing. In *Proc. of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, pages 183–199, 2006.

- [11] S. F. Railsback, S. L. Lytinen, and S. K. Jackson. Agent-based Simulation Platforms: Review and Development Recommendations. *SIMULATION*, 82(9):609–623, September 2006.
- [12] E. Shakshuki and Y. Jun. Multi-agent development toolkits: An evaluation. In Proc. of the 17th Intern. Conf. on Innovations in Applied Artificial Intelligence, IEA/AIE'2004, pages 209–218. Springer Verlag, 2004.
- [13] M. Wooldridge. An Introduction to Multi Agent Systems. Wiley, 2nd edition, 2009.
- [14] M.-J. Yoo and R. Glardon. Combining JADE and Repast for the Complex Simulation of Enterprise Value-Adding Networks. In *AOSE*, pages 243–256, 2008.

5.5 Simulation und Realität bei der Chipsynthese mit VHDL am Beispiel von endlichen und unendlichen Zählern

Harald Richter – Institut für Informatik

Einleitung

VHDL ist eine sog. Hardware-Beschreibungssprache, die es erlaubt, digitale Schaltungen formal zu spezifizieren, zu simulieren und in realen Chips zu synthetisieren. Ihr aktueller Sprachdefinition wurde in [1] festgelegt. In vorliegendem Beitrag wurde untersucht, ob es einen Unterschied zwischen Simulation und Realität bei der Chipsynthese mit VHDL am Beispiel von endlichen und unendlichen Zählern gibt.

VHDL-Programme spezifizieren Hardware-Funktionen inkl. deren Zeitverhalten. Sie werden üblicherweise auf zwei verschiedene Arten ausgeführt: zum einem in einem Software-Simulator, zum anderen in einem programmierbaren Logikbaustein aus der Kategorie der sog. FPGAs (Field Programmable Gate Array). Der normale Weg bei der VHDL-Programmentwicklung ist, dass das Programm zuerst simuliert und danach die Chipsynthese begonnen wird, sofern der Simulator das Programm fehlerfrei übersetzen und ausführen konnte, und ein Funktionstest erfolgreich war. Bei der Ausführung im Simulator wird im Gegensatz zur Chipsynthese der gesamte Sprachumfang von VHDL unterstützt, und es können auch sehr große und komplexe Programme simuliert werden. Ein zweiter Vorteil bei der Simulation ist, dass VHDL-Programme auf einem Rechner mit Betriebssystem und nicht auf der blanken Hardware eines FPGA-Chips ausgeführt werden. Durch diese Betriebssystemunterstützung wird das Programmtesten erleichtert und verkürzt. Ein Beispiel für diese Unterstützung ist die Prozedur „fprint“ aus der öffentlich erhältlichen Bibliothek PCK_FIO [2]. Damit können die Werte aller interessierenden VHDL-Programmvariablen und -signale auf dem Bildschirm des Rechners ausgegeben werden. Die Bildschirmausgabe ist die klassische Art des Programmtestens, wie sie auch in C/C++ mit „printf“ praktiziert wird. Ebenso können mit Betriebssystemunterstützung Eingaben von der Tastatur von dem zu testenden VHDL-

Programm eingelesen werden. In einem FPGA hingegen gibt es kein Betriebssystem, keinen Bildschirm und keine Tastatur. Alle oben genannten Testhilfen fallen deshalb weg.

Aus diesen Gründen spielt die simulative Programmausführung beim VHDL-Programmtest eine wichtige Rolle. Die Grundidee ist hierbei, dass ein Programmier- oder Denkfehler vorliegen muss, wenn die Simulation nicht korrekt abläuft. Die Chipsynthese braucht dann nicht versucht zu werden. Allerdings sind gemäß [3] auch Fälle bekannt, bei denen eine Simulation abbricht oder ein falsches Ergebnis zeigt, die Chipsynthese aber erfolgreich ist. Das bedeutet, es gibt VHDL-Programme, die zwar korrekt synthetisiert aber nicht richtig simuliert werden können. Umgekehrt führt eine erfolgreiche Simulation nicht automatisch zu einer erfolgreichen Chipsynthese. Der Grund dafür ist, dass die Erstellung von synthesefähigem Code für ein FPGA aus einem VHDL-Programm deutlich komplexer ist als dessen reine Simulation ist. Im streng mathematischen Sinne ist eine Simulation für die Chipsynthese weder notwendig noch hinreichend, aber dennoch nützlich.

Stand der Technik

Es gibt zahlreiche Simulatoren für VHDL, die sowohl kommerziell erhältlich als auch frei verfügbar sind. Beispiele kommerzieller Simulatoren sind ModelSim von Mentor Graphics [4], ISE Simulator (ISim) von Xilinx [5], Portunus von Adapted Solutions [6], SimPlover von ANSYS [7], SMASH von Dolphin Integration [8] und Quartus II von Altera [9]. Ein Beispiel für einen frei erhältlichen Simulator ist GHDL [10] von T. Gingold. Von den genannten Simulatoren wurden im Rahmen dieses Projekts ModelSim und ISim ausführlich getestet. Dabei erwies sich ModelSim als deutlich besser. ISim ist beispielsweise nur in der Lage, Signale auszugeben aber keine Variable. Darüber hinaus ist ModelSim weit verbreitet und unterstützt vollständig den aktuellen Sprachumfang gemäß [1]. Alle nachfolgenden Simulationen im Projekt wurden deshalb mit ModelSim durchgeführt.

Simulative Programmausführung

Die simulative Abarbeitung der VHDL-Befehle soll den Pro-

grammentwurf wirksam unterstützen und bereits vor der Chipsynthese auf Fehler hinweisen. Dazu muss jeder VHDL-Simulator möglichst denselben VHDL-Ausführungsregeln gehorchen. Diese Regeln sind so spezifiziert, dass sie zum einen die Effizienz bei der Simulation erhöhen, und zum anderen die Ausführung auf dem Chip möglichst präzise nachahmen [11]. Da die beiden Zielsetzungen verschieden sind, kommt es vor, dass Simulation und Realität auf dem FPGA-Chip voneinander abweichen. Die Abweichungen sind in beiden Richtungen möglich, d.h. es gibt Programme, die sich zwar simulieren aber nicht synthetisieren lassen und umgekehrt. Der erste Fall ist der bei weitem häufiger, der zweite Fall ist die Ausnahme. Die Regeln [12], nach denen alle VHDL-Simulatoren arbeiten, sind:

- 1.) Prozesse werden nur aktiviert, wenn sich in der sog. sensitivity list des Prozesses etwas ändert. Diese Vorgehensweise wird als diskrete Ereignissimulation bezeichnet und ist der Grund, warum es die sensitivity list überhaupt gibt. Diese Liste enthält alle potentiellen Triggersignale, die den Prozess aktivieren können und sorgt dadurch für Effizienz bei der Ausführung der Simulation, da ein Simulationslauf nur nach Veränderung einer Eingabegröße durchgeführt wird und nicht permanent. Es wird also nicht ununterbrochen simuliert, sondern nur bei Bedarf. Handelt es sich um die Simulation taktsynchroner Prozesse, wird jeder Prozess mit der Änderung des Taktsignals aktiviert, das dazu in der sensitivity list aufgeführt sein muss. Die Prozessaktivierung geschieht häufig durch Abprüfen auf eine ansteigende Taktflanke über das VHSL-statement „*if clk'event and clk='1'*“ bzw. „*wait until clk'event and clk='1'*“.
- 2.) Zuweisungen an Signale werden vom Simulator gesammelt und erst am Ende des Prozesses ausgeführt, so dass deren Ergebnisse bei der nächsten Prozessausführung zur Verfügung stehen. Dies ahmt die Tatsache nach, dass Signale stets als speichernde Elemente, d.h. als D-Flip Flops, Register, Zähler, Akkumulator oder Schieberegister synthetisiert werden, und dass solche Elemente ihren Eingangswert in der Regel erst mit der nächsten Taktflanke speichern können. Die nächste Taktflanke aktiviert zugleich den nächsten Simulationsdurchlauf, so dass mit Beginn der nächsten Prozessaktivierung auch das Registerausgangssignal zur Verfü-

gung steht. Dies ahmt die Wirklichkeit tatsächlich sehr gut nach, vorausgesetzt, dass auf dem Chip tatsächlich speichernde Elemente verwendet werden, die wie D-Flip Flops angesteuert werden und nicht wie Latches. Auf die Verwendung von Latches, die nicht taktflanken- sondern taktpegelgesteuert sind, sollte auf dem FPGA deshalb verzichtet werden, wenn die Simulation genau sein soll.

- 3.) Von mehreren Zuweisungen im selben Prozess an dasselbe Signal wird nur die letzte ausgeführt. D.h. im Programmtext vorangehende Zuweisungen an ein Signal werden durch die letzte Zuweisung überschrieben. Dies entspricht der Tatsache, dass speichernde Elemente in der Regel den Eingangswert mit einer Taktflanke übernehmen. Signale, die vor einer Taktflanke anliegen, werden nicht übernommen. Bei dieser Vorgehensweise wird die Wirklichkeit nur näherungsweise nachgebildet, da die setup- und hold-Zeiten der speichernde Elemente vernachlässigt werden.
- 4.) Zuweisungen an Variable werden dann sofort ausgeführt, wenn sie nicht als speichernde Elemente simuliert oder synthetisiert werden. Solche Variable sind reine Compiler-interne Zwischengrößen. Sie haben auf dem FPGA nur eine Verbindungsleitung als Hardware-Gegenstück. Eine binäre Variable wird durch eine einzige Verbindung repräsentiert, alle mehrstelligen Variablen durch ein Bündel von Verbindungen. Werden Variable hingegen als speichernde Elemente simuliert oder synthetisiert, stehen deren Ergebnisse wie bei Signalen erst bei der nächsten Prozessaktivierung zur Verfügung. Dies ahmt die Wirklichkeit ebenfalls sehr gut nach.

Inkompatibilitäten der Synthese- und Simulatorwerkzeuge

Stand der Technik bei der Chipsynthese sind Inkompatibilitäten sowohl zwischen den Synthesewerkzeugen untereinander als auch zwischen Simulation und Realität, sobald es um komplexere Schaltungen geht [13]. In der Regel sind diese Inkompatibilitäten aber nicht genau genug in der Literatur beschrieben, so dass es dem VHDL-Programmierer selbst überlassen bleibt, herauszufinden, was im konkreten Einzelfall möglich ist und was nicht. Grundsätzlich gilt: das, was in der VHDL-Simulation

funktioniert, kann bei der Synthese je nach verwendetem Synthesewerkzeug entweder zu verschiedenen oder zu keinem Resultat führen. Denn im Gegensatz zur Synthese kann der Simulator fast jedes VHDL-Programm richtig ausführen, sofern dessen Syntax korrekt ist. Bei der Chipsynthese können hingegen nur weniger komplexe und kürzere Programme erfolgreich synthetisiert werden. Insgesamt ist die Chipsynthese ein nicht-trivialer Vorgang mit nicht-linearer Zeitkomplexität. Sie kann hohe Rechenzeit und viel Speicher erfordern.

Ein Beispiel für die genannten Inkompatibilitäten zwischen Synthese- und Simulatorwerkzeugen ist die Verwendung einer *while*-Schleife zur Ausgabe von Werten auf einem Ausgangsport, die nur in der Simulation, aber nicht auf dem Chip funktioniert [3]. Bei der Ausführung des „*while*“-Befehls, aber auch beim Befehl „*for*“ gibt große prinzipielle Unterschiede zwischen der Simulation und der Realität. Auch sind die erzielten Syntheseergebnisse je nach verwendetem Synthesewerkzeug verschieden. Das führt zu einer insgesamt schwierigen Situation. Der Grund dafür ist, dass VHDL ursprünglich nur zur Hardware-Beschreibung und Simulation entwickelt wurde. Erst später wurden Werkzeuge geschaffen, mit denen VHDL zu einer Chipsynthesprache desselben Namens erweitert wurde.

Der Stand der Technik bei der VHDL-Programmierung ist deshalb der, dass sich viele VHDL-Programmierer nach ersten schlechten Erfahrungen mit der Chipsynthese auf die Simulation von VHDL-Programmen beschränken.

Forschungsaufgaben

In [3] wurde ein sog. „Software-orientierter Programmierstil“ beschrieben, der die Chipsynthese wesentlich vereinfacht. Dieser neue Programmierstil wurde im Rahmen des von der DFG geförderten Projekts „CarRing II“ entwickelt. Wichtige Fragestellungen im Projekt waren: 1) Was lässt sich überhaupt synthetisieren? 2) Worin liegen die Unterschiede zwischen Simulation und Synthese? 3) Worin unterscheiden sich die Synthesewerkzeuge im Hinblick auf die erzeugten Netzlisten voneinander?

Um diese Fragen zu beantworten, wurden der VHDL-Simulator Modelsim von Mentor Graphics und die Synthesewerkzeuge xst von Xilinx und Synplify von Synopsys mit Hilfe von Testprogrammen miteinander verglichen. Die Testprogramme implementieren endliche und unendliche Zähler. Die Zähler wurden für das Spartan 3-FPGA der Fa. Xilinx synthetisiert. Insgesamt wurden vier verschiedene Szenarien untersucht:

- 1.) Endloszähler mit Zählvariable
- 2.) Endloszähler mit Zählsignal
- 3.) Endlicher Zähler mit Zählvariable
- 4.) Endlicher Zähler mit Zählsignal

Endloszähler mit Zählvariable

In Code 1 ist das VHDL-Programm für den Fall eines unendlichen Zählers mit Zählvariable gezeigt. Das Programm wird zuerst von Modelsim simuliert und dann von Synplify und von xst für das Spartan 3-FPGA der Fa. Xilinx synthetisiert. Danach wird das Zeitverhalten des Programms am FPGA gemessen und mit der Simulatorprognose verglichen.

```
H. Richter 02.04.11
-- Endloszaehler mit Zaehlvariable library
IEEE;
use IEEE.STD_LOGIC_1164.all;
use work.Konstanten.all;

package Konstanten is
-- Deklariere Konstanten
constant HoechsterWertVoni: integer:= 3;
constant HoechsterWertAmAusgabeport:
integer:= HoechsterWertVoni;
-- Begrenzung auf 2 Bit-Variable in diesem Bei-
spiel
end Konstanten;
entity UnbegrenzterZaehler is
Port (clk: in STD_LOGIC;
      NotwendigerAusgabeport: inout integer
      range 0 to HoechsterWertAmAusgabeport);
```

```

end UnbegrenzterZaehler;

architecture One of UnbegrenzterZaehler is
begin
  process (clk)
    variable i: integer range 0 to Hoechster
    WertVoni := 1;
    -- Zaehlvariable; starte mit 1
  begin
    if clk'event and clk = '1' then
      i := i+1;
      -- Register => wird einen Takt spaeter
      gueltig
      NotwendigerAusgabeport <= i;
      -- Register. Wird einen Takt spaeter
      gueltig.
    end if; -- clk'event and clk = '1' ohne else
  end process;
end One;

```

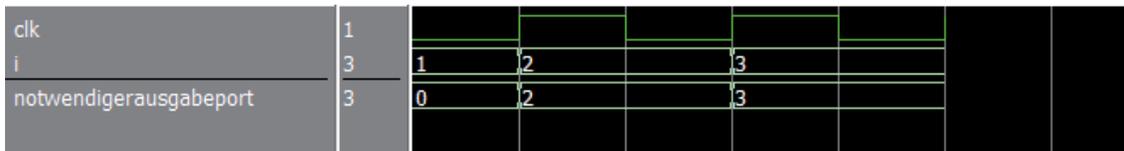
Endloszähler mit Zählvariable.

Programmcode 1: Endloszähler mit Zählvariable.

Modelsim

Das von Modelsim vorhergesagte Zeitverhalten für die Zählvariable i und das Ausgangssignal „notwendigerAusgangsport“ sind in Abbildung 1 dargestellt. Man sieht, dass die Anfangswertinitialisierung von i wie erwartet mit 1 vorhergesagt wird, und dass nach dem ersten Taktimpuls i und das Ausgangssignal identisch sind. Da beim Simulationslauf die Bereichsgrenzenüberwachung der Variablen (range check) eingeschaltet war, bricht die Simulation nach drei Takten ab, da i dann überläuft.

Abbildung 1: Durch Modelsim vorhergesagtes Zeitverhalten von Code 1



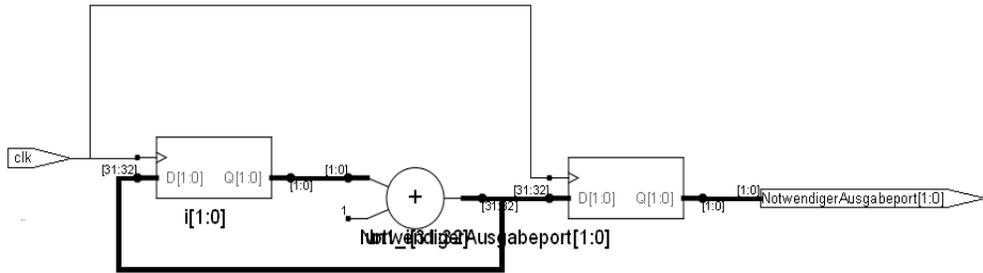


Abbildung 2: Von Synplify erzeugtes Kompilat für den in Code 2 dargestellten Endloszähler

Synplify

In Abbildung 2 ist das von Synplify erzeugte Syntheseresultat für den in Code 1 dargestellten Endloszähler mit Zählvariable gezeigt. Die Zählfunktion wird über einen rückgekoppelten Addierer am Ausgang des Variablenregisters implementiert. Das Ausgangssignal erhält ein zweites Register, das denselben Takt und denselben Input wie die Zählvariable verwendet, d.h. es wird noch im selben Takt gültig und ändert sich synchron mit der Zählvariablen. Ein extra Register für das Signal wäre deshalb nicht notwendig. Der von Synopsys erzeugte Code ist aus diesem Grunde nicht optimal. Das Verhalten des Endloszählers ist in Abbildung 3 gezeigt. Bis auf die Anfangswerte verhalten sich i und das notwendige Ausgabesignal wie erwartet gleich. Es ist noch bemerkenswert, dass dieselbe Hardware synthetisiert wird, wenn statt $i := i+1$; der Befehl $i := i-1$; oder $i := i$ -inkrement; eingesetzt wird, nur die Konstante, die vom Addierer hinzugezählt wird, ändert sich.

Das durch Modelsim simulierte Verhalten und die Messung verhalten sich genau gleich.

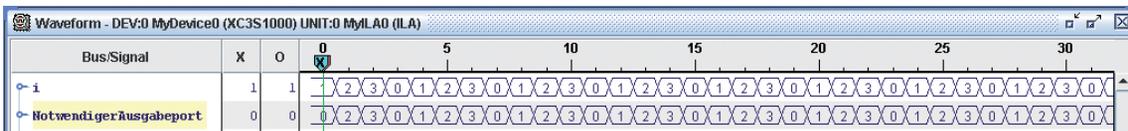


Abbildung 3: Am FPGA gemessenes Verhalten des Endloszählers von Code 1

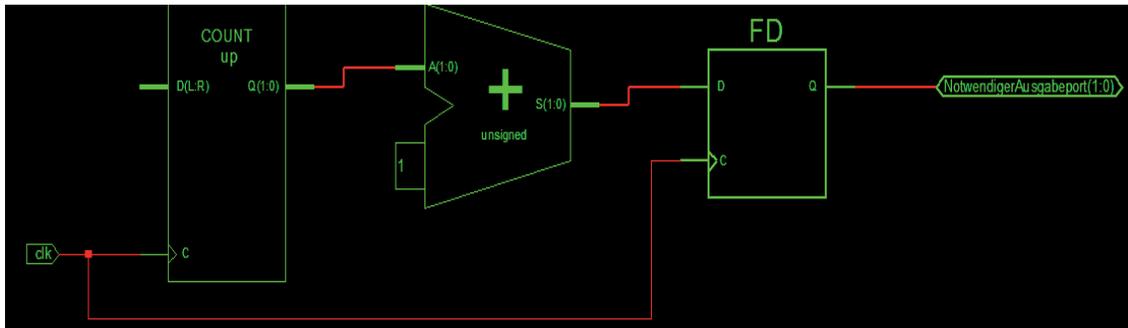


Abbildung 4: Von xst erzeugtes Kompilat für den in Code 2 dargestellten Endloszähler

XST

Im Vergleich dazu folgen die Syntheserergebnisse von xst für dieselben Programmbeispiele. In Abbildung 4 ist das von xst erzeugte Kompilat gezeigt. Die Zählerfunktion wird über einen Aufwärtszähler und einen Addierer, der zum Zählerwert +1 hinzuzählt, implementiert. Der Addierer hat bis auf die Bereitstellung des Anfangswertes von 1 keine sinnvolle Funktion. Das Ausgangssignal verfügt wie bereits zuvor über ein eigenes Register. Das zeitliche Verhalten ist genau wie in Abbildung 1, d.h., auch hier stimmen Simulation und Realität überein. Insgesamt ist der von xst erzeugte Code in diesem Falle schlechter als der von Synplify, da der Aufwand für einen Zähler mit Addierer über dem eines Registers mit Addierer liegt.

Endloszähler mit Zählsignal

Ersetzt man die Zählvariable durch ein Zählsignal erhält man den Code 2.

```
H. Richter 02.04.11
-- Endloszaehler mit Zaehlsignal library IEEE;
use IEEE.STD_LOGIC_1164.all;
use work.Konstanten.all;
package Konstanten is
-- Deklariere Konstanten
constant HoechsterWertVoni: integer:= 3;
constant HoechsterWertAmAusgabeport:
integer:= HoechsterWertVoni;
-- Begrenzung auf 2 Bit-Variable in diesem Bei-
```

```

spiel end Konstanten;
entity UnbegrenzterZaehler is
Port (clk: in STD_LOGIC;
      NotwendigerAusgabeport: inout integer
      range 0 to HoechsterWertAmAusgabeport);
end UnbegrenzterZaehler;

architecture One of UnbegrenzterZaehler is
  signal i: integer range 0 to HoechsterWert
  Voni := 1;
  -- Zaehlsignal; starte mit 1
begin
  process (clk)
  begin
    if clk'event and clk = '1' then
      i <= i+1;
      -- Register => wird einen Takt
      spaeter gueltig
      NotwendigerAusgabeport <= i;
      -- Register. Wird einen Takt spaeter
      gueltig. end if; -- clk'event and clk
      = '1' ohne else
    end process;
  end One;

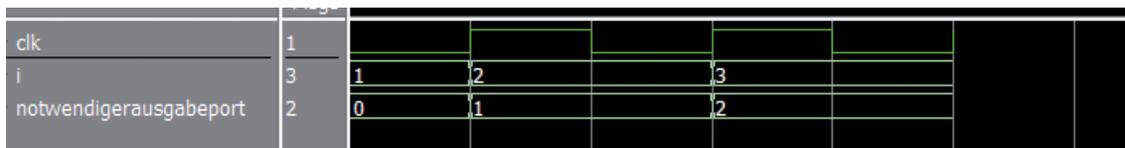
```

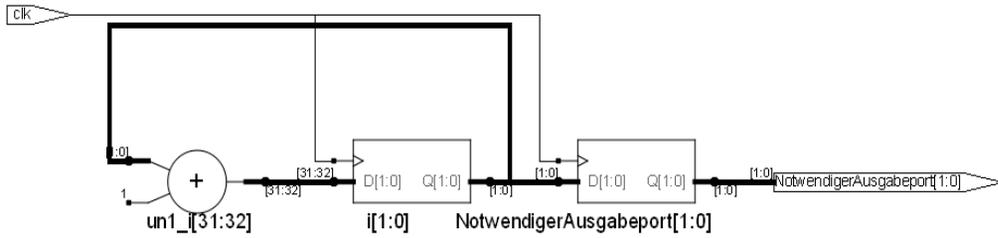
Programmcode 2: Endloszähler mit Zählsignal.

Modelsim

Das von Modelsim vorhergesagte Verhalten von Code 2 ist in Abbildung 2 dargestellt. Wie erwartet hinkt die Ausgabe dem Zählsignal um einen Takt hinterher.

Abbildung 5: Durch Modelsim vorhergesagtes Zeitverhalten von Code 2





Synplify

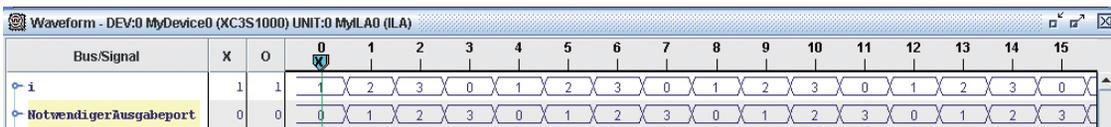
Wird anstelle einer Zählvariable ein Zählsignal im Programmtext verwendet, wobei alles andere gleich bleibt, ändert sich das Compilationsergebnis in signifikanter Weise. In Bild 6 ist das von Synplify erzeugte Kompilat dargestellt. Die Zählfunktion wird jetzt über einen rückgekoppelten Addierer am Eingang des Registers implementiert, d.h. die Position des Addierers hat sich vom Ausgang des Registers zum Eingang verschoben. Das Ausgangssignal erhält, wie immer, ein eigenes Register. Dieses bildet mit dem Zählregister eine Pipeline, d.h. sein Ausgangswert wird erst einen Takt später als der Zählerwert gültig. Das neue zeitliche Verhalten ist in Abbildung 7 gezeigt. Es stimmt mit dem simulierten Verhalten überein.

Abbildung 6: Von Synplify erzeugtes Kompilat für den in Code 2 dargestellten Endloszähler mit Zählsignal anstelle einer Zählvariablen

XST

In Abbildung 7 ist das von xst erzeugte Kompilat für den Endloszähler mit Zählsignal anstelle einer Zählvariablen gezeigt. Die Zählfunktion wird jetzt nur über einen Aufwärtszähler implementiert. Der Aufwand ist dadurch geringer, und der erzeugte Code trifft genau den beabsichtigten Zweck, so dass dieses Ergebnis besser als bei Synplify ist. Allerdings ist das Register für das Ausgangssignal ohne Funktion, da bereits der Zähler den Ausgabewert bis zum jeweils nächsten Takt speichert. D.h., auch das xst-Kompilat ist nicht optimal.

Abbildung 7: Verhalten des in Code 2 dargestellten Endloszählers auf dem FPGA mit Zählsignal anstelle einer Zählvariablen



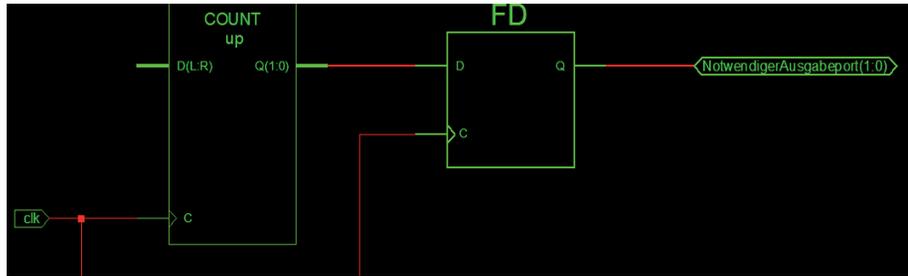


Abbildung 8: Von xst erzeugtes Kompilat für den in Code 2 dargestellten Endloszähler

Endlicher Zähler mit Zählvariable

Für einen endlichen Zähler mit Zählvariable muss Code 1 dahingehend modifiziert werden, dass ein sog. Abschnittszähler nach einer endlichen Zahl von Zählerinkrementierungen das Zählen stoppt. Das ist in Code 3 gezeigt.

```
H. Richter 02.04.11
-- Endlicher Zaehler mit Zaehlvariable library
IEEE;
use IEEE.STD_LOGIC_1164.all;
use work.Konstanten.all;

package Konstanten is
-- Deklariere Konstanten
constant HoechsterWertVoni: integer:= 7;
constant HoechsterWertAmAusgabeport:
integer:= HoechsterWertVoni;
-- Begrenzung auf 3 Bit-Variable in diesem Bei-
spiel

type Abschnitte is (Abschnitt0, Stop); -- enu-
meration type
-- Programm hat nur einen Abschnitt + Stop
end Konstanten;

entity ifSchleife is
Port (clk: in STD_LOGIC;
      NotwendigerAusgabeport: inout integer
      range 0 to HoechsterWertAmAusgabeport);
end ifSchleife;
```

```

architecture One of ifSchleife is
  signal Abschnittszaehler:
  Abschnitte:= Abschnitt0;
  -- starte mit Abschnitt0 begin
  process(clk)
    variable i: integer range 0 to HoechsterWert
    Voni := 1;
    -- Zaehlvariable; starte mit 1
    begin
      if clk'event and clk = '1' then
        if Abschnittszaehler = Abschnitt0 then
          -- Beginn der if-Schleife
          if i<HoechsterWertVoni then
            i := i+1;
            -- Register => wird einen Takt spaeter
            gueltig
            NotwendigerAusgabeport <= i;
            -- Register => wird einen Takt spaeter
            gueltig
          else -- i>=HoechsterWertVoni
            Abschnittszaehler <= Stop;
            -- Register. Wird einen Takt spaeter
            gueltig.
          end if; -- if-Schleife
        else -- Abschnittszaehler <= Stop
          -- Programmende erreicht
          null;
        end if; -- Abschnittszaehler = Abschnitt0
      end if; -- clk'event and clk = '1' ohne else
    end process;
  end One;

```

Programmcode 3: Endlicher Zähler mit Zählvariable.

Modelsim

Das von Modelsim vorhergesagte Zeitverhalten ist in Abbildung 9 dargestellt. Man sieht, dass die Anfangswertinitialisierung von i wie gewünscht mit 1 vorhergesagt wird, und dass nach dem ersten Taktimpuls i und das Ausgangssignal identisch sind.

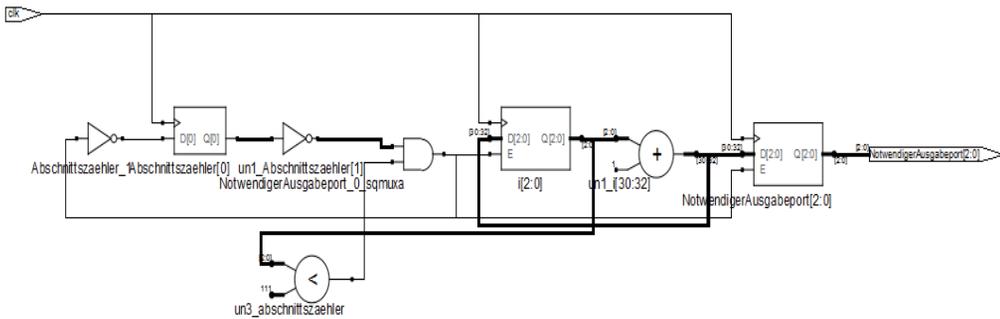
clk	1					
i	3	1	2		3	
notwendigerausgabeport	3	0	2		3	

Abbildung 9: Durch Modelsim vorhergesagtes Zeitverhalten von Code 3

Synplify

In Abbildung 10 ist das von Synplify erzeugte Kompilat für den in Code 3 dargestellten endlichen Zähler mit Zählvariable gezeigt. Die Zählfunktion wird genau wie im Falle der Endloschleife mit Variable über einen rückgekoppelten Addierer am Ausgang des Zählregisters implementiert. Synplify macht an dieser Stelle keinen Unterschied zwischen endlosem und endlichem Zähler. Das Ausgangssignal erhält ein eigenes Register das redundant ist. Die endliche Zählfunktion wird über einen endlichen Automaten realisiert, der den Zähler nach Erreichen der Schleifenobergrenze über dessen Enable-Eingang stilllegt. Der Automat benützt einen Vergleicher (Komparator), der den Zählerwert mit der Schleifenobergrenze vergleicht. Insgesamt ist der erzeugte Schaltplan nicht optimal im Sinne minimalen Aufwandes, aber er ist verständlich, da aus einfacheren Grund-

Abbildung 10: Von Synplify erzeugtes Kompilat für den in Code 3 dargestellten endlichen Zähler



Bus/Signal	X	O	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
i	1	1	1	2	3	4	5	6	X						7			
NotwendigerAusgabeport	0	0	0	2	3	4	5	6	X						7			
Abschnittszaehler	0	0																

Abbildung 11: Gemessenes Zeitverhalten bei Synplify und Code 3

elementen aufgebaut. Das daraus resultierende zeitliche Verhalten ist in Abbildung 11 gezeigt. Bis auf die Anfangswerte sind i und das notwendige Ausgabesignal identisch und stimmen mit der Simulation überein. Zur Messung des zeitlichen Verhaltens des FPGAs mit Hilfe des Logikanalysators „Chipscope“ von der Fa. Xilinx ist es erforderlich, ein Startsignal als Input in die Portliste von Code 3 hinzuzufügen, das dann von Chipscope als Trigger verwendet werden kann.

XST

Abbildung 12 zeigt das von xst erzeugte Kompilat für einen endlichen Zähler mit Zählvariable. Der erzeugte Schaltplan ist unübersichtlicher als bei Synplify. Die Zählfunktion wird über einen Addierer realisiert, der zum Ausgabesignal solange +1 addiert, wie ein endlicher Automat (realisiert im unteren Flip Flop) angibt. Ein Register zum Speichern des Zählerwertes gibt es nicht. Diese Funktion wird vom Ausgangsregister (Flip Flop oben rechts) mit übernommen, was eine Ressourcenoptimierung darstellt. Die Schleifenobergrenze wird über einen „Less Than“-Komparator und einen zusätzlichen „Greater or Equal“-Komparator abgeprüft. Das bedeutet, dass xst den if-

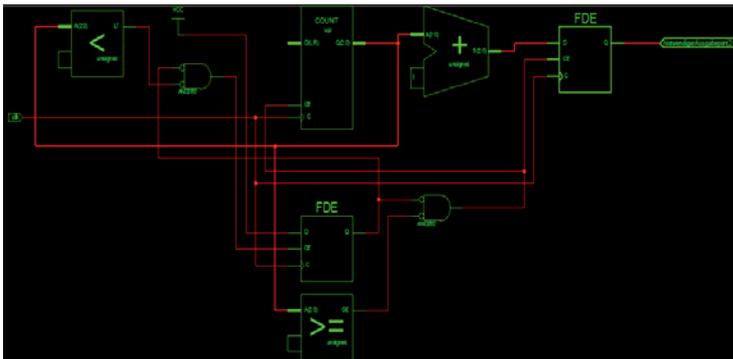
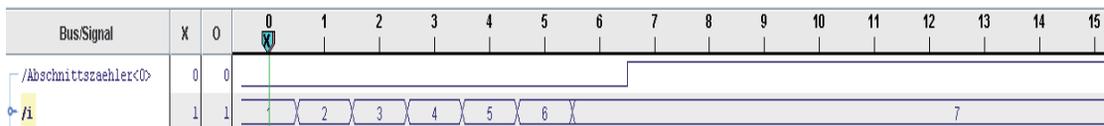


Abbildung 12: Von xst erzeugtes Kompilat für den in Code 3 dargestellten endlichen Zähler

Abbildung 13: Gemessenes Zeitverhalten bei xst und Code 3



Zweig und den elsezweig im Programm separat synthetisiert. Ist die Schleifenobergrenze erreicht, wird das Register des Ausgangssignals über seinen Clock Enable-Eingang deaktiviert. Das resultierende Verhalten ist in Abbildung 13 dargestellt. Es stimmt mit der Simulation überein.

Endlicher Zähler mit Zählsignal

Wird im endlichen Zähler die Zählvariable durch ein Zählsignal ersetzt, entsteht Code 4.

```
H. Richter 02.04.11
-- Endlicher Zaehler mit Zaehlsignal
-- Deklariere Konstanten
constant HoechsterWertVoni: integer:= 7;
constant HoechsterWertAmAusgabeport:
integer:= HoechsterWertVoni;
-- Begrenzung auf 3 Bit-Variable in diesem
Beispiel
type Abschnitte is (Abschnitt0,
Stop); -- enumeration type
-- Programm hat nur einen Abschnitt + Stop end
Konstanten;
entity ifSchleife is
Port (clk: in STD_LOGIC;
      NotwendigerAusgabeport: inout integer
      range 0 to HoechsterWertAmAusgabeport);
end ifSchleife;

architecture One of ifSchleife is
signal Abschnittszaehler:
Abschnitte:= Abschnitt0;
-- starte mit Abschnitt0
signal i: integer range 0 to Hoechster
WertVoni := 1;
-- Zaehlsignal; starte mit 1
begin
process (clk)
begin
if clk'event and clk = '1' then
if Abschnittszaehler = Abschnitt0 then
-- Beginn der if-Schleife
```

```

    if i<HoechsterWertVoni then i <= i+1;
    -- Register => wird einen Takt
    spaeter gueltig
    NotwendigerAusgabeport <= i;
    -- Register => wird einen Takt
    spaeter gueltig
    else -- i>=HoechsterWertVoni
        Abschnittszaehler <= Stop;
        -- Register. Wird einen Takt
        spaeter gueltig.
    end if; -- if-Schleife
else -- Abschnittszaehler <= Stop
    -- Programmende erreicht
    null;
end if; -- Abschnittszaehler = Abschnitt0
end if; -- clk'event and clk = '1' ohne else
end process;
end One;

```

Programmcode 4: Endlicher Zähler mit Zählsignal.

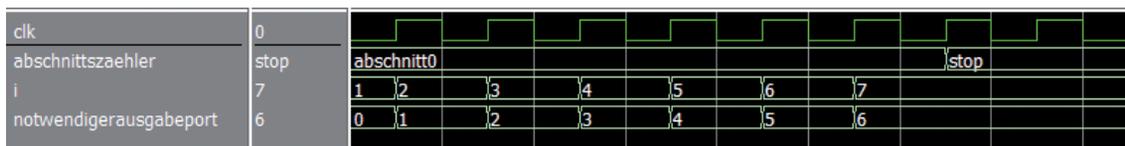
Modelsim

Modelsim sagt das in Abbildung 14 dargestellte Zeitverhalten voraus. Auch hier hinkt wie erwartet die Ausgabe dem Zählsignal um einen Takt hinterher.

Synplify

Wird anstelle einer Zählvariable ein Zählsignal erzeugt, wobei alles andere gleich bleibt, ändert sich das Kompilationsergebnis im Vergleich zur Zählvariablen nur beim Zähler. Wie bereits beim Endloszähler von Code 2 wird die Zählfunktion über einen rückgekoppelten Addierer am Eingang des Registers implementiert. Das Ausgangssignal erhält ein eigenes Register und bildet eine Pipeline. Das Ergebnis ist in Abbildung 15 gezeigt.

Abbildung 14: Durch Modelsim vorhergesagtes Zeitverhalten von Code 4



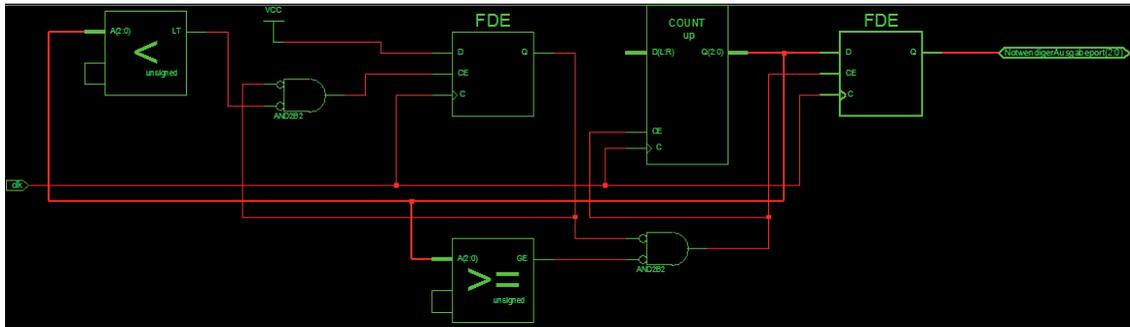
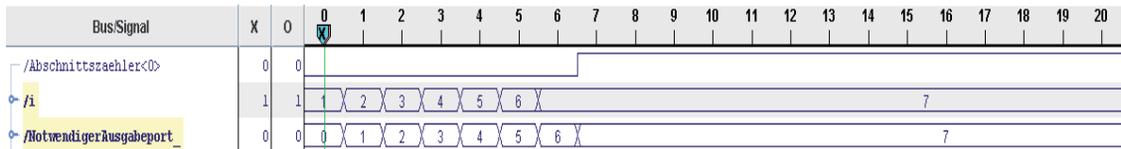


Abbildung 17: Von xst erzeugtes Kompilat für den in Code 4 dargestellten endlichen Zähler mit Zählsignal anstelle einer Zählvariablen



Testhilfsmittel zur Simulation empfohlen, jedoch besteht ein Unterschied zwischen dem, was grundsätzlich simuliert werden kann und zwischen dem, was tatsächlich synthetisierbar ist. In Zukunft sollen auch komplexere Szenarien wie z.B. das eines Rechnernetzprotokolls für ein Automobil im Hinblick auf Inkompatibilitäten der Synthese- und Simulatorwerkzeuge untersucht werden.

Abbildung 18: Gemessenes Zeitverhalten bei xst und Code 4

Literaturhinweise

- [1] IEEE, Standard VHDL Language Reference Manual - Std 1076™-2008, IEEE Computer Society, Park Avenue, New York, NY 10016-5997, USA, 26 January 2009.
- [2] Easics NV, http://www.easics.com/webtools/freesics_pck_fio_1_16_manual, (10. April 2011).
- [3] H. Richter, Software-Orientierter Programmierstil bei VHDL, Ifl Technical Report Ifl-10-07, in Ifl Technical Report Series ISSN 1860-8477, <http://www.in.tu-clausthal.de/>

- fi-leadadmin/homes/techreports/ifi1007richter.pdf, editor: Department of Computer Science, Clausthal University of Technology, Germany, pp. 105, Nov. 2010.
- [4] Mentor Graphics, <http://www.mentor.com/products/fv/modelsim/>, (10. April 2011).
 - [5] Xilinx, http://www.xilinx.com/support/documentation/sw_manuals/xilinx12_4/ug682.pdf, (10. April 2011).
 - [6] Adapted Solutions, <http://www.adapted-solutions.com/web/ASProduktPortunusUeberblick.html>, (10. April 2011).
 - [7] ANSYS, <http://www.ansoft.com/products/em/simplorer/>, (10. April 2011).
 - [8] Dolphin Integration, http://www.dolphin.fr/medal/smash/smash_overview.php, (10. April 2011).
 - [9] Quartus II von Altera, <http://www.altera.com/products/software/quartus-ii/subscriptionedition/qts-se-index.html>, (10. April 2011).
 - [10] T. Gingold, <http://ghdl.free.fr/download.html>, (10. April 2011).
 - [11] P. Molitor, J. Ritter, VHDL Eine Einführung, Pearson-Studium, 2004.
 - [12] G. Lehmann, B. Wunder, M. Selz, Schaltungsdesign mit VHDL, Franzis Verlag, Poing, ISBN: 3-772-36163-3, 1994.
 - [13] Andreas Mäder, Vergleichende Untersuchungen zum effizienten Einsatz von VHDL in Simulation und Synthese, Der Andere Verlag, 2004.

Simulationswissenschaftliches Zentrum

Vorstand

Prof. Dr. Gunther Brenner	Institut für Technische Mechanik
Prof. Dr. Thomas Hanschke (Sprecher)	Institut für Angewandte Stochastik und Operations Research
Prof. Dr. Michael Kolonko	Institut für Angewandte Stochastik und Operations Research

Mitglieder

Prof. Dr. Hans-Peter Beck	Institut für Elektrische Energie- technik
Prof. Dr. Uwe Bracht	Institut für Maschinelle Anlagen- technik und Betriebsfestigkeit
Prof. Dr. Wolfgang Busch	Institut für Geotechnik und Markscheidewesen
Prof. Dr. Otto Carlowitz	Institut für Umweltwissenschaften
Prof. Dr. Jürgen Dix	Institut für Informatik
Prof. Dr. Eberhard Gock	Institut für Aufbereitung, Depo- nietechnik und Geomechanik
Prof. Dr. Stefan Hartmann	Institut für Technische Mechanik
Prof. Dr. Karl-Heinz Lux	Institut für Aufbereitung, Depo- nietechnik und Geomechanik
Prof. Dr. Jörg P. Müller	Institut für Informatik
Prof. Dr. Heinz Palkowski	Institut für Metallurgie
Prof. Dr. Günter Pusch	Institut für Erdöl- und Erdgas- technik

Prof. Dr. Harald Richter	Institut für Informatik
Prof. Dr. Roman Weber	Institut für Energieverfahrens- technik und Brennstofftechnik
Prof. Dr. Volker Wesling	Institut für Schweißtechnik und Trennende Fertigungsverfahren
Prof. Dr. Gabriel Zachmann	Institut für Informatik
Prof. Dr. Gerhard Ziegmann	Institut für Polymerwerkstoffe und Kunststofftechnik

