Marc Deichmann - Diplomarbeit

Reengineering, Modularisierung und Erweiterung einer Simulationssoftware für die dreidimensionale Walzenerwärmung unter Einsatz objektorientierter Techniken

Technische Universität Clausthal

Clausthal, den 31. Mai 2004

Diplomarbeit

Reengineering, Modularisierung und Erweiterung einer Simulationssoftware
für die dreidimensionale Walzenerwärmung
unter Einsatz objektorientierter Techniken

Marc Deichmann

Institut für Informatik
Technische Universität Clausthal

im Auftrag und in Zusammenarbeit mit
Dr.-Ing. Ingolf Jäckel
Dr. rer. nat. Horst W. Tamler
ThyssenKrupp Stahl AG, Duisburg

In beratender Position:
Prof. Dr.-Ing. Palkowski
Institut für Metallurgie
AG Werkstoffumformung
Technische Universität Clausthal

Erstgutachter:
Dr.-Ing. habil. Günter Kemnitz
Institut für Informatik
Technische Universität Clausthal

Zweitgutachter:
Prof. Dr. Michael H. Breitner
Institut für Wirtschaftsinformatik
Universität Hannover

Betreuer:
Dipl.-Math. Torsten Sander
Institut für Mathematik
Technische Universität Clausthal

Die vorliegende Diplomarbeit wurde gefördert durch die ThyssenKrupp Stahl AG, Duisburg. Jedwede Verwertung dieser Diplomarbeit bedarf der schriftlichen Zustimmung.

Ich erkläre hiermit, die vorliegende Arbeit selbstständig angefertigt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben.

Clausthal-Zellerfeld, den 6. Juni 2004.

Hiermit möchte ich einigen Leuten meinen Dank für ihre Hilfe an dieser Diplomarbeit aussprechen.

Als erstes möchte ich Herrn Dr.-Ing. Ingolf Jäckel und Dr. rer. nat. Horst Tamler von der ThyssenKrupp Stahl AG, Duisburg danken. Die finanzielle Unterstützung war sehr hilfreich und ihre Geduld führte zu einem sehr angenehmen Arbeitsklima.

Danken möchte ich auch Herrn Dr.-Ing. habil. Günter Kemnitz und Herrn Prof. Dr. Michael H. Breitner als Gutachter. Auch sie zeigten Geduld und haben meine Verlängerungsanträge bereitwillig bestätigt.

Herrn Dipl.-Math Torsten Sander danke ich für die Betreuung. Seine unnachahmlichen Art direkte Frage oft nur durch Hinweise zu beantworten, brachte mich dazu Antworten selber zu suchen. Die Befriedigung, ein Problem selber zu lösen, ist ungemein höher einzuschätzen, als die einfache Beantwortung einer Frage.

Meiner langjährigen Freundin Beate Winkelmann möchte ich einen besonderen Dank aussprechen. Als ruhender Punkt in meiner hektischen Art stand sie immer zur Verfügung, wenn ich ein paar aufmunternde Worte brauchte. Auch über 8000 Kilometer hinweg.

Mein Dank gilt ebenfalls Herrn Dipl.-Ing. Christian Hopp. Als professioneller Programmierer stand er mir immer mit Rat und Tat zur Seite und verkörperte oft den Advokat des Teufels, der die unangenehmen Fragen stellt und Anworten fordert.

Frau Veronika Lenk sei speziell gedankt. Sie hat mir geholfen die zahlreichen bürokratischen Hürden zu überwinden.

Inhaltsverzeichnis

Kapitel 1

Grundlagen

„Adding manpower to a late software project makes it later.“
– F. Brooks, The Mythical Man-Month

Bei der Warmumformung von Stahl ist die genaue Kenntnis des Temperaturgeschehens sowohl im Umformgut als auch in den Umformwerkzeugen von entscheidender Bedeutung für den Produktionsprozeß.

1.1 Warmwalzen

Das Warmwalzen ist ein Vorgang, bei dem ein erhitztes Walzgut in einem Walzgerüst geformt wird. Ein Walzgerüst besteht aus zwei Arbeitswalzen (Abbildung 1.1) und bis zu sechs Stützwalzen, die symmetrisch übereinander angeordnet sind. Das Walzgut, in Form einer Bramme, wird in den sogenannten Walzspalt zwischen den Arbeitswalzen geschoben und dort druckgeformt.

Bis zu neun Walzgerüste bilden eine Walzstraße. In der Walzstraße wird die Bramme zu Stahlblech geformt, das dann, zum Beispiel in der Automobilindustrie, weiterverarbeitet wird.

Der Walzvorgang läuft folgendermaßen ab: Die Bramme wird in einem Ofen auf circa 1000°C erhitzt. Danach wird sie von einem Transportband zur Walzstraße befördert. Bevor die Bramme am ersten Walzgerüst angelangt, wird sie vom sogenannten Zunder befreit.

Die Bramme duchläuft nun nacheinander alle Walzgerüste. Dabei wird sie von jedem nachfolgenden Walzgerüst immer flacher gedrückt, bis am Ende ein langes

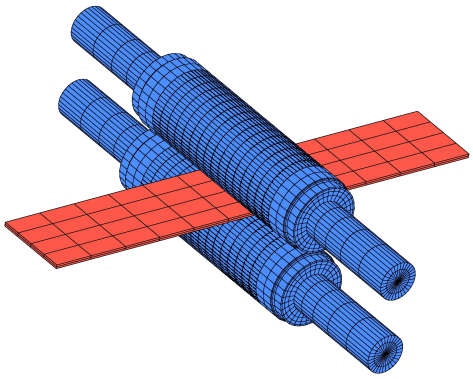

Abbildung 1.1: Darstellung von zwei Arbeitswalzen und einem Band. Die Stützwalzen sind über bzw. unter den Arbeitswalzen angeordnet.

Stück Stahlblech vorliegt. Das resultierende Stahlblech wird abgekühlt und aufgerollt. Zu diesem Zeitpunkt hat das Stahlblech immer noch eine Temperatur von circa 600°C.

Beim Warmwalzen dringt fortlaufend Wärme vom heißen Walzgut in die Arbeitswalzen ein. Als Reaktion darauf verändert sich die Walzengeometrie infolge thermischer Ausdehnung, so dass die Form des Walzspaltes und damit die Form des auslaufenden Walzgutes verändert wird. Dieser Effekt ist fester Bestandteil des Formgebungsverfahrens und kann zur Verbesserung der Form des fertigen Walzproduktes ausgenutzt werden. Hierfür ist eine gezielte Kontrolle des Wärmehaushaltes der Arbeitswalzen über die Steuerung der Walzenkühlung notwendig. Die Walzenkühlung der Arbeitswalzen findet mit Hilfe von zwei so genannten Kühlbalken statt, die die Walzen mit Wasser besprühen [Lan02].

1.2 RollHeat-3D

Um das Zusammenspiel der erwärmenden und abkühlenden Einflüsse für eine Arbeitswalze beschreiben zu können, ist in der AG Werkstoffumformung des Instituts für Metallurgie der Technischen Universität Clausthal die computergestützte Simulationssoftware „Rollheat-3D“ entwickelt worden. Diese Software berechnet die dreidimensionale Temperaturverteilung einer Arbeitswalze über mehrere Walzzyklen.

Die Software modelliert die dreidimensionale Wärmeleitungsgleichung mit temperaturkonstanten Materialparametern in Zylinderkoordinaten. In radialer Richtung findet eine Gitterverdichtung zur Walzenoberfläche statt. Zusätzlich wird das Gitter auf die reale Geometrie einer ausgesuchten Arbeitswalze transformiert, das so genannte „Body-Fitting“. Die Randbedingungen sind von NEUMANNscher Art und zeitabhängig. Die Diskretisierung erfolgt durch Finite Differenzen. Es steht ein explizites Verfahren *„Forward Time Central Space“* bzw. `FTCS` zur Lösung der Wärmeleitungsgleichungen zur Verfügung.

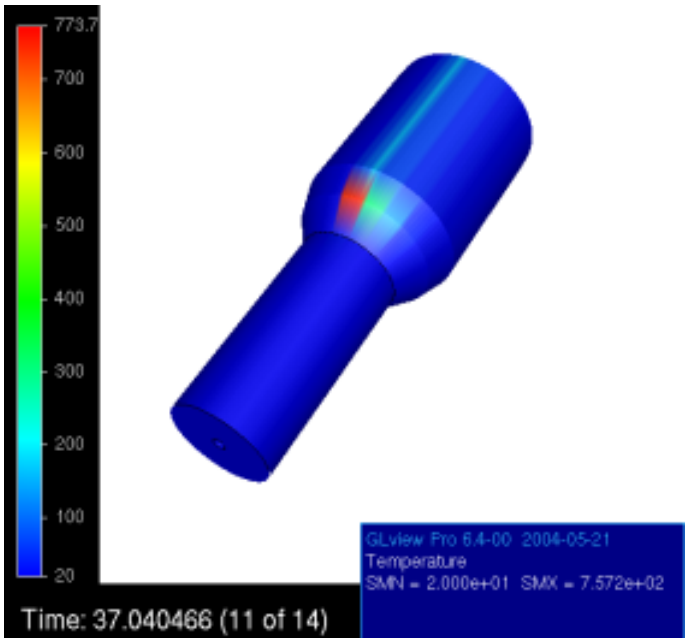

Abbildung 1.2: Modell einer berechneten Walze, dargestellt mit GLView

Abbildung 1.2 zeigt das berechnete Modell einer Walze. Die graphische Ausgabe wurde mit Hilfe von GLView realisiert. Der Löser berechnet nur eine halbe Walze, da davon ausgegangen, dass sich die andere Hälfte der Walze symmetrisch verhält. In Abbildung 1.3 ist die oben erwähnte Walze dargestellt, komplettiert durch die andere Hälfte, die an der Mittelebene gespiegelt wurde.

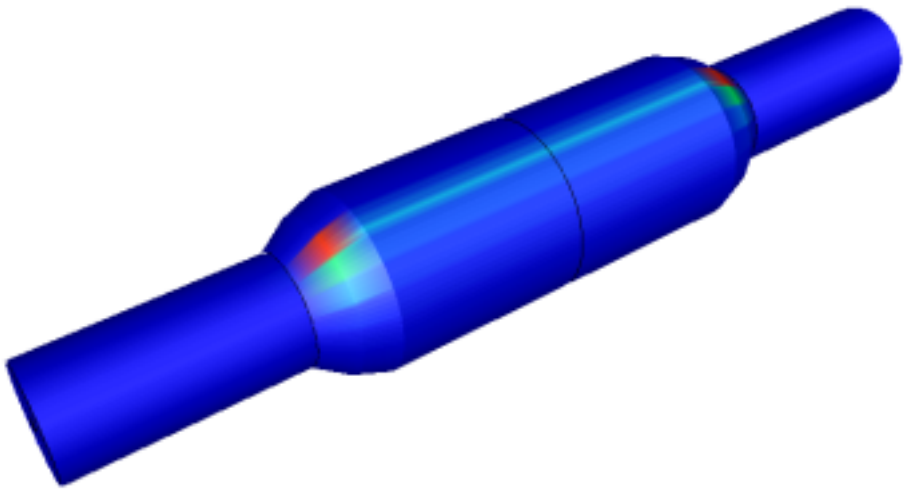

Abbildung 1.3: Modell einer kompletten Walze, gespiegelt an der z-Achse.

1.3 Segmentierte Walzenkühlung

Die Kühlung der Arbeitswalze wurde im ursprünglich vorliegenden Modell als konstant angenommen. Das ist eine unangemessene Vereinfachung der Realität, war für die ersten Modellversuche aber ausreichend.

Um das Modell realitätsgetreuer zu gestalten, wurde die Kühlung so angeordnet wie es auch in der Realität bei ThyssenKrupp Stahl der Fall ist: Der Kühlbalken ist in Segmente aufgeteilt. Diese Segmente sind in Längsrichtung (z-Achse) der Walze verteilt. Zusätzlich gibt es drei unterschiedliche Düsenreihen, die Grund-, Mitten- und Randkühlung. Die Düsenreihen lassen sich mit einer Gewichtung von 0% bis 100% zuschalten. In Abbildung 1.4 ist die segmentierte Kühlung beispielhaft dargestellt.

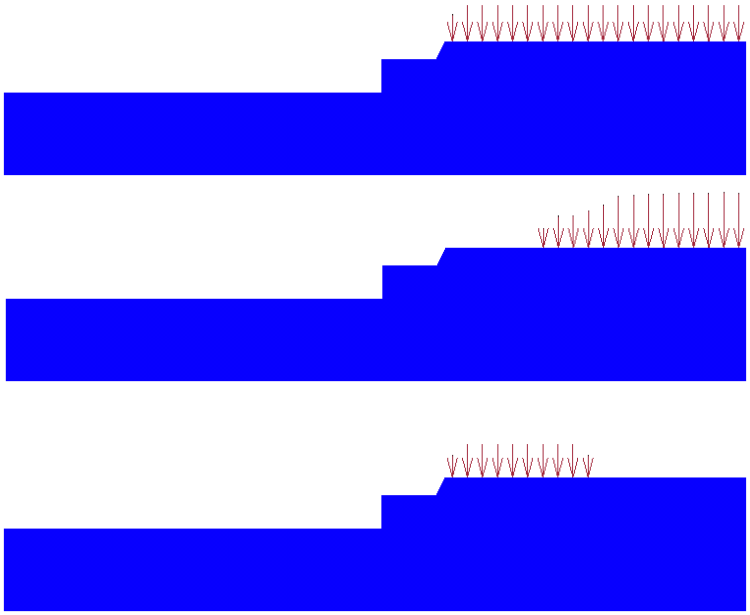

Abbildung 1.4: Von oben nach unten, die Grund-, Mitten- und Randkühlung in einer schematischen Darstellung.

Kapitel 2

Aufgabenstellung

„A plan never survives first contact with reality.“
– Murphy’s Laws

Im Rahmen dieser Diplomarbeit sollte für den bisherigen Programmcode des Modells, der prozedural in C++ geschrieben war, ein objektorientierter Neuentwurf erfolgen. Hierfür war geplant nach der Methode des Top-Down-Designs vorzugehen. Zunächst war mittels der UML-Beschreibungssprache [BW99] ein rein formaler Entwurf auszuarbeiten. Dabei sollte Wert auf weitgehende Modularisierbarkeit gelegt werden.

Insbesondere sollte die Modularisierung hinsichtlich der Austauschbarkeit der konkreten Modellbeschreibung und -parametrisierung vorgenommen werden (z.B. für die Geometrie, die zeitabhängigen Randbedingungen und die Materialparameter). Außerdem sollte die Erweiterung durch weitere Rechenkerne und verschiedene Ausgaberoutinen möglich gemacht werden.

Die mathematische Theorie [Roh03, EBY99] ist nicht Gegenstand dieser Arbeit. Vielmehr sollte das Reengineering die Programmvorlage weitestgehend als Blackbox auffassen. Die Theorie ist nur zur Erkennung der strukturellen Komponenten und zur Bewertung der Anforderungen notwendig.

Die vorhandene Simulationssoftware wurde geschrieben von Herrn Dipl.-Math. Torsten Sander. Die mathematischen Modelle und die Programmierung in C++ sind Ergebnisse seiner Arbeit. Auf diesen Grundlagen baute diese Diplomarbeit auf. Die Simulationssoftware war bereits vorher lauffähig und hat gute Ergebnisse geliefert.

Für die neue Implementierung sollte entschieden werden, ob die Software in C++ fortzuführen war, oder ob sie zuerst in eine andere Programmiersprache übertragen und dort erweitert werden sollte. Besonderes Augenmerk war auf die Portabilität und Effizienz des Codes zu legen. Die Geschwindigkeit des neuen Codes war durch Laufzeitvergleiche mit dem bisherigen Code zu vergleichen. Um dieses Ziel zu erreichen, musste der C++ Code analysiert und nach objektorientierten Gesichtspunkten neu gestaltet werden.

Um die Leistungsfähigkeit des gewählten Designs bzw. der konkreten Umsetzung zu demonstrieren, waren folgende Punkte zu erfüllen:

1. Die Modellbeschreibung sollte es möglich machen, beliebige Abfolgen von Walz- und Ruhezyklen bei unterschiedlichen Walzguttemperaturen und Kühlungsbedingungen zu simulieren. Ebenso war eine flexible segmentierte Walzenkühlung umzusetzen. Es sollte die Modellierung von komplexen Radialgeometrien ermöglicht werden.

2. Der Löser war dahingehend zu erweitern, dass eine radiale Koordinatentransformation zum Zwecke der Gitterverdichtung bzw. des Body-Fittings vorhanden war.

3. Die Ausgaberoutinen sollten neben den Modellergebnissen auch die interne Modellbeschreibung exportieren können. Diese war in Form einer ASCII-Modelldatei des Finite-Elemente-Paketes MSC.Marc/Mentat 2000 auszugeben.

4. Der Quelltext sollte sowohl unter Microsoft Windows™ als auch unter verschiedenen Unices kompilierbar sein.

Kapitel 7

Fazit und Ausblick

„Ich denke niemals an die Zukunft. Sie kommt früh genug.“
– Albert Einstein

7.1 Fazit

In der fertigen Simulationssoftware sind folgende Ziele erreicht worden: Die Umsetzung der ursprünglichen C++ Simulation nach Java wurde erfolgreich vollzogen. Die Aufteilung des monolithischen Ursprungsprogramms in eine übersichtlichere Struktur ist geglückt.

Außerdem sind objektorientierte Ansätze wie Vererbung, Implementierung und Kapselung von Daten erfolgreich eingesetzt worden, um die spätere Erweiterung und den Zugriff auf die Klassen zu erleichtern. Das Programm is dynamisch erweiterbar durch Nutzung von javaspezifischen Techniken. Neue Module können mit geringem Aufwand hinzugefügt werden.

Parameteränderungen können, im Gegensatz zum C++ Modell, ohne Eingriffe in den Programmcode erfolgen.

Die Ergebnisse der Simulation können in Dateien ausgegeben werden, die von MSC.Marc/Mentat 2000 und GLView gelesen und dargestellt werden können. Zusätzlich wurden neue Berechnungsunterklassen erstellt, die im C++ Modell nicht vorhanden waren.

Ein grafische Oberfläche wurde geschaffen, mit dem man einfach und schnell Walzpläne erstellen kann. Trotzdem ist es immer noch möglich die Simulation über

Konfigurationsdateien zu steuern, da die Werte in den Dateien von Hand veränderbar sind.

Die grafische Oberfläche kann die nachfolgenden Eigenschaften erfüllen: Die Simulation kann vollständig an der Benutzerschnittstelle konfiguriert werden. Es ist möglich komplette Walzpläne in Form von Projekten zu erstellen und abzuspeichern.

Genauso wie im Löser wurde auch in der grafische Oberfläche auf dynamische Erweiterbarkeit Wert gelegt. Die Benutzerführung ist internationalisiert und kann um weitere Sprachen erweitert werden.

Die vollständige Dokumentation des Quelltextes mit Hilfe von JavaDoc gewährt eine übersichtliche Struktur der Kommentare und erlaubt die Umwandlung und Ausgabe in verschiedene Formate.

Schließlich wurden intensive Programmtests durchgeführt um die Funktionsfähigkeit und die Korrektheit der Simulation zu garantieren.

7.2 Ausblick

Während der Ausarbeitung der Software sind einige mögliche Optionen unrealisiert geblieben, die hier nun als eventuelle zukünftige Erweiterungen erwähnt werden sollen.

Eine Anbindung an die Datenbank mit Walzplänen bei ThyssenKrupp Stahl könnte realisiert werden. Diese Anbindung sollte es ermöglichen direkt Daten aus realen Walzplänen auszulesen und sie in die grafische Oberfläche einzufügen. Danach können sie dort weiter verarbeitet werden um anschließend Simulationen zu berechnen.

Die Konfiguration der Software könnte auch über eine Netzwerkfernsteuerung erfolgen, um die Berechnungen auf leistungsfähigen Rechnern stattfinden zu lassen, die meistens nicht in Büroumgebungen stehen.

Es sollte möglich sein, eine Hilfefunktion, die direkt in die grafische Oberfläche implementiert ist, zu erstellen. Diese Hilfe sollte die Suche nach Parametern und ihren Erklärungen ermöglichen.

Wie in 6.2 bereits erwähnt und näher erklärt, sollte eine Auslagerung der MSC. Marc/Mentat Ausgabe erhebliche Geschwindigkeitsvorteile bringen. Dazu sollte die Klasse `T19IO` in einen Thread umgewandelt werden, der neben dem Löser gleichzeitig laufen kann.