

**Holger Hartmann**

Entwicklung von Handelssystemen mittels  
Genetischer Programmierung anhand eines  
Fallbeispiels

**Diplomarbeit**

# BEI GRIN MACHT SICH IHR WISSEN BEZAHLT



- Wir veröffentlichen Ihre Hausarbeit, Bachelor- und Masterarbeit
- Ihr eigenes eBook und Buch - weltweit in allen wichtigen Shops
- Verdienen Sie an jedem Verkauf

Jetzt bei [www.GRIN.com](http://www.GRIN.com) hochladen  
und kostenlos publizieren



## **Bibliografische Information der Deutschen Nationalbibliothek:**

Bibliografische Information der Deutschen Nationalbibliothek: Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de/> abrufbar.

Dieses Werk sowie alle darin enthaltenen einzelnen Beiträge und Abbildungen sind urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsschutz zugelassen ist, bedarf der vorherigen Zustimmung des Verlanges. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen, Auswertungen durch Datenbanken und für die Einspeicherung und Verarbeitung in elektronische Systeme. Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

Copyright © 2007 Examicus Verlag  
ISBN: 9783869437408

**Holger Hartmann**

# **Entwicklung von Handelssystemen mittels Genetischer Programmierung anhand eines Fallbeispiels**

Examicus - Verlag für akademische Texte

Der Examicus Verlag mit Sitz in München hat sich auf die Veröffentlichung akademischer Texte spezialisiert.

Die Verlagswebseite [www.examicus.de](http://www.examicus.de) ist für Studenten, Hochschullehrer und andere Akademiker die ideale Plattform, ihre Fachtexte, Studienarbeiten, Abschlussarbeiten oder Dissertationen einem breiten Publikum zu präsentieren.

# Entwicklung von Handelssystemen mittels Genetischer Programmierung anhand eines Fallbeispiels



UNIVERSITÄT HAMBURG  
Department Informatik  
Arbeitsbereich Kognitive Systeme

Diplomarbeit  
von  
Holger Hartmann

*Gutachter*  
Dr. Andreas Günter  
Prof. Dr. Stefan Voss

Juni 2007



# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Ziel und Aufbau . . . . .	2
<b>2</b>	<b>Grundlagen und Stand der Technik</b>	<b>5</b>
2.1	Genetische Programmierung . . . . .	5
2.1.1	Aufbau eines Programms . . . . .	5
2.1.2	Initialisierung der GP Population . . . . .	6
2.1.3	Die Genetischen Operatoren . . . . .	7
2.1.4	Fitness Funktion . . . . .	8
2.1.5	Selektion . . . . .	9
2.1.6	Ablauf des GP Algorithmus . . . . .	10
2.1.7	Crossover, Building Blocks und Schemata . . . . .	11
2.1.8	Ansätze gegen Makromutation . . . . .	12
2.1.9	Modularisierung . . . . .	14
2.1.10	Weitere Verbesserungsansätze . . . . .	16
2.2	Künstliche Neuronale Netze . . . . .	16
2.2.1	Bestandteile neuronaler Netze . . . . .	17
2.2.2	Netztopologien . . . . .	18
2.2.3	Lernmethoden . . . . .	18
2.3	Handelssysteme . . . . .	20
2.3.1	Tape Reader . . . . .	21
2.3.2	Market Timing . . . . .	22
2.3.3	Position Sizing . . . . .	24
2.3.4	Vergleich von Handelssystemen . . . . .	27
2.3.5	Fundamentale versus Technische Analyse . . . . .	30
2.3.6	Der Währungsmarkt . . . . .	34
2.3.7	Entwicklungsansätze für Handelssysteme in der Literatur	35



<b>3 Entwurf</b>	<b>39</b>
3.1 Überblick . . . . .	39
3.2 Anforderungen an die Software . . . . .	39
3.3 Konzeption der Software . . . . .	42
3.3.1 Der Evolutionäre Algorithmus . . . . .	42
3.3.2 Die Fitness-Funktion . . . . .	45
<b>4 Implementierung</b>	<b>49</b>
4.1 Komponenten der entwickelten Software . . . . .	49
4.2 Klassen des Kursdatenservers . . . . .	50
4.3 Klassen des Evolutionären Algorithmus . . . . .	51
4.4 Übersicht über das Framework ECJ . . . . .	54
4.5 Probleme während der Experimente . . . . .	55
<b>5 Versuchsergebnisse</b>	<b>57</b>
5.1 Resultate mit Knotengewichten . . . . .	57
5.1.1 Resultate des Trainingszeitraums . . . . .	58
5.1.2 Resultate des Validierungszeitraums . . . . .	58
5.1.3 Resultate des Testzeitraums . . . . .	59
5.1.4 Resultate als Monatsrenditen . . . . .	60
5.1.5 Erzeugte Handelsregeln . . . . .	61
5.2 Resultate ohne Knotengewichte . . . . .	64
5.2.1 Resultate des Trainingszeitraums . . . . .	64
5.2.2 Resultate des Validierungszeitraums . . . . .	64
5.2.3 Resultate des Testzeitraums . . . . .	65
5.2.4 Resultate als Monatsrenditen . . . . .	66
5.2.5 Erzeugte Handelsregeln . . . . .	67
5.3 Bestimmung und Anwendung von Optimal f . . . . .	67
<b>6 Diskussion und Bewertung</b>	<b>77</b>
6.1 Ausblick . . . . .	78
<b>7 Zusammenfassung</b>	<b>81</b>
<b>Abbildungsverzeichnis</b>	<b>86</b>
<b>Literaturverzeichnis</b>	<b>92</b>
<b>Glossar</b>	<b>93</b>

# Kapitel 1

## Einführung

### 1.1 Motivation

Die natürliche Evolution hat sich zur Erzeugung und Anpassung von Lebewesen an eine sich ändernde Umgebung als höchst erfolgreicher Mechanismus herausgestellt. Ohne bestimmte Anweisungen oder auch nur genaue Zieldefinitionen zu erhalten, ist es ihr gelungen, raffinierte Lösungen für Probleme der realen Welt zu finden.

Ein Ansatz, die in der natürlichen Evolution steckende kreative Kraft zur automatischen Entwicklung von Computer-Programmen zu verwenden, ist die Genetische Programmierung (GP) (vgl. (Koz92, Kapitel 1-6)). Mit ihr wird versucht, Mechanismen der natürlichen Evolution nachzuahmen, um automatisch Programme zu erzeugen, die ein gegebenes Problem lösen. GP ist in einer Reihe von Anwendungen sowohl zum Lösen mathematischer Probleme als auch zum Beheben von Problemen der realen Welt erfolgreich angewandt worden. Hierzu zählen z.B. Symbolische Regression (Koz92, S. Kapitel 10), Klassifikation (Koz92, Kapitel 17), Synthese Künstlicher Neuronaler Netze (Gru94, Kapitel 2f), Muster Erkennung (Tac93, S. 2-10), Roboter Steuerung (BNO97, S. 2-10) und Generierung von Bildern (GH97, S. 2-7).

Das maschinelle Lernen mittels GP kann als ein heuristischer Suchalgorithmus interpretiert werden, der in der Menge aller möglichen Programme diejenigen sucht, die das gegebene Problem am besten lösen. Da der Suchraum je nach gegebenem Problem sehr groß und oft weder stetig noch differenzierbar ist, eignet sich der Suchraum aller möglichen Programme schlecht für klassische Suchalgorithmen (vgl. (LP02, S. 2f)).

Das Anwendungsgebiet der GP in dieser Arbeit ist die Erzeugung von Handelssystemen für den Finanzmarkt, insbesondere für den Währungsmarkt. An den Finanzmärkten handeln erfolgreiche spekulative Händler gewöhnlich aufgrund gewisser Regelwerke. Diese Regelwerke sind jedoch relativ starker individueller Interpretation unterworfen. Bei genauer Betrachtung fällt auf, dass Händler die Regeln, nach denen sie zu handeln meinen, in entscheidenden Situationen beugen und gewissermaßen nach ihrem „Bauchgefühl“ handeln. Evtl. unterscheidet dieser Anteil an intuitivem Handeln einen erfahrenen profitablen Händler von einem unerfahrenen unprofitablen Händler, auch

wenn beide meinen, nach dem gleichen Regelwerk zu arbeiten. Das Definieren eines Handelssystems durch einen Menschen ist mit Schwierigkeiten verbunden, da er nicht alle Regeln eindeutig wiedergeben kann. Daher hat sich die Übertragung von Regelwerken zum Handeln auf einen Rechner als nicht erfolgreich herausgestellt.

Ein anderer Ansatz ist, den Rechner die Handelsregeln selbst lernen zu lassen. Hierzu werden z.B. Künstliche Neuronale Netze (KNN) erfolgreich eingesetzt (vgl. (Ska01, S. 2-5), (MK, S. 2-7)). Es ist jedoch nicht ohne weiteres möglich, die einzelnen Regeln in einfach deutbarer Weise aus dem Netz zu extrahieren. Diese „Black-Box“ Eigenschaft von KNN wird von Anwendern kritisiert.

GP bietet sich als Alternative zu KNN an, da sie direkt Regeln erzeugen kann und sich diese trotz einiger Komplexität besser deuten lassen (vgl. (YCK05, S. S. 23f)). Was die Fähigkeit zur Lösung schwieriger Probleme angeht, sind beide Ansätze vergleichbar (vgl. (BB98, S. 13)).

## 1.2 Ziel und Aufbau

Ziel dieser Arbeit ist es, GP anzuwenden, um Handelssysteme zu erzeugen und auf Profitabilität im Rahmen einer historischen Simulation zu untersuchen. Ein Softwaresystem, das diese Aufgabe löst, wird entworfen und interessante Implementierungsaspekte dargestellt.

Um Handelssysteme mit GP entwickeln zu können, muss die zu entwickelnde Software einer Reihe von Anforderungen genügen. Die Entwicklung der Handelssysteme soll basierend auf historischen Kurszeitreihen vorgenommen werden. Es wird angenommen, dass sich der Markt mit der Zeit ändert und daher ehemals profitable Handelssysteme an Profitabilität verlieren. Daher ist es nötig, das Entwicklungssystem der Handelssysteme so zu konzipieren, dass im Laufe der Zeit neue, an die geänderten Marktbedingungen angepasste Handelssysteme erzeugt werden können. Um die Versorgung mit aktuellen Kursdaten sicherzustellen, müssen die relevanten Marktdaten kontinuierlich erfasst und dem Entwicklungssystem zur Verfügung gestellt werden. Die Entwicklung profitabler Handelssysteme wird unterstützt, indem Vorverarbeitungen der Kursdaten, die bei Wertpapierhändlern verbreitet sind, dem System zur Verfügung gestellt werden. Zur visuellen Überprüfung sollen die Kursdaten, die Vorverarbeitungen sowie die Transaktionen der Handelssysteme grafisch dargestellt werden. Überoptimierung bei der Entwicklung der Handelssysteme soll verhindert werden, indem die vorhandene Kurshistorie in Trainings-, Validierungs- und Testzeitraum unterteilt wird. Um ein Handelssystem für den Testzeitraum zu erhalten, werden die besten Handelssysteme des Trainingszeitraums auf den Validierungszeitraum angewandt. Das beste Handelssystem während der Validierung wird für den Handel im Testzeitraum ausgewählt. Der Entwicklungsprozess soll reproduzierbar und durch Log-Dateien der Zwischenergebnisse transparent sein. Handelssysteme zu großer Komplexität erwecken bei Anwendern Mißtrauen, da sich die Entscheidungen des Systems schwer nachvollziehen lassen. Auch wenn sich mit komplexeren Handelssystemen eine höhere Rendite erzielen ließe, ist eine gewisse