



# Vergleich technischer Trading-Tools anhand der Ergebnisse genetischer Optimierungen

DIPLOMARBEIT

zur Erlangung des akademischen Grades

**Diplom-Ingenieur/in**

im Rahmen des Studiums

**Wirtschaftsinformatik**

eingereicht von

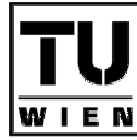
**Lei Zhu**

Matrikelnummer 0150825

an der  
Fakultät für Informatik der Technischen Universität Wien

Betreuung:  
Betreuer/Betreuerin: **Thomas Grechenig**

Wien, 05.01.2010 \_\_\_\_\_  
(Unterschrift Verfasser/in) (Unterschrift Betreuer/in)



# Vergleich technischer Trading-Tools anhand der Ergebnisse genetischer Optimierungen

## DIPLOMARBEIT

zur Erlangung des akademischen Grades

**Diplom-Ingenieur/in**

im Rahmen des Studiums

**Wirtschaftsinformatik**

eingereicht von

**Lei Zhu**

0150825

ausgeführt am

Institut für Rechnergestützte Automation

Forschungsgruppe Industrial Software

der Fakultät für Informatik der Technischen Universität Wien

### **Betreuung:**

Betreuer: Thomas Grechenig

Mitwirkung: Andreas Mauczka

Wien, 05.01.2010

# Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benützt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Wien, am 24. Februar 2010

.....

Name

*„An der Börse ist alles möglich, auch das Gegenteil.“*

André Kostolany

# Danksagung

Besonderer Dank gilt meiner Familie für ihre langjährige Unterstützung in jeder Hinsicht. Für die Betreuung meiner Arbeit möchte ich Prof. Grechenig herzlich danken. Mit wertvollen Ideen und Ansätzen zur Lösung von Anfangsschwierigkeiten stand er immer zu meiner Verfügung. Darüber hinaus möchte ich auch meinem Assistenzbetreuer Andreas Mauczka vielen Dank aussprechen. Er war immer zur Unterstützung meiner Arbeit bereit und stand mir immer mit kompetentem fachlichen Rat zur Seite.

# Kurzfassung

Diese Diplomarbeit befasst sich mit dem Einsatz genetischer Algorithmen im Rahmen von Trading Tools. Genetische Algorithmen sind stochastische Optimierungsverfahren, die von John H. Holland entwickelt wurden. In der modernen Investment Finanz werden genetische Algorithmen in Trading Tools integriert um die Profitmaximierung zu unterstützen. Ziel dieser Arbeit ist es, die Arbeitsweise der genetischen Algorithmen im Detail zu beschreiben und die technisch-strategischen Vorteile der genetischen Algorithmen im Rahmen von Trading Tools zu beleuchten. Zwei Fallbeispiele - „Merchant of Venice“ und „MultiCharts“ werden herangezogen und auf allgemeine Eigenschaften wie Funktionsumfang, Lizenz, Kostenpunkt und Systemanforderungen untersucht und miteinander verglichen. In Merchant of Venice werden genetische Algorithmen zur Parameter Optimierung in Kauf/Verkauf Regel und zur Kauf/Verkauf Regel Generierung eingesetzt. In MultiCharts werden genetische Algorithmen zur Parameter Optimierung und zur Portfolio Optimierung eingesetzt. Es werden empirische Versuche und Analysen mit unterschiedlichen Parameterkonfigurationen durchgeführt, um die Performanz des Einsatzes von genetischen Algorithmen in Trading Tools zu untersuchen. Von speziellem Interesse ist die Performanz der genetischen Algorithmen in unterschiedlichen Wirtschaftsbedingungen wie zum Beispiel in einem wachsenden bzw. in einem turbulenten Markt. Die Ergebnisse aus den Untersuchungen zeigen signifikante Potentiale der genetischen Algorithmen.

**Keywords:** *Genetische Algorithmen, Evolutionäre Algorithmen, Trading Tools, Merchant of Venice, MultiCharts*

# Abstract

This master thesis deals with the application of genetic algorithms within the frameworks of trading tools. Genetic algorithms are stochastic optimization techniques developed by John H. Holland. There is a recent development in investment finance to integrate genetic algorithms in trading tools to maximize profits. The goal of this thesis is to describe the principles of genetic algorithms and to show the technical and strategic advantages of using genetic algorithms in trading tools. Two trading tools - „Merchant of Venice“ and „MultiCharts“ - have been chosen for this purpose. They will be examined and compared in regards to common characteristics such as features, license details, costs and system requirements. Merchant of Venice uses genetic algorithms for input parameter optimization and buy/sell rules generation. MultiCharts uses genetic algorithms for input parameter optimization and portfolio optimization. Empirical experiments and analysis with different parameters will be conducted to evaluate the performance of the application of genetic algorithms in trading tools. The performance of genetic algorithms in different economical market conditions, such as growing and instable markets is specifically evaluated. The results indicate significant potentials of the application of genetic algorithms within the frameworks of trading tools.

**Keywords:** *Genetic algorithm, Evolutionary algorithm, Trading tools, Merchant of Venice, MultiCharts*

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Problemstellung . . . . .	1
1.2	Zielsetzung . . . . .	2
1.3	Methodik und Aufbau der Arbeit . . . . .	2
<b>2</b>	<b>Grundlagen</b>	<b>4</b>
2.1	Grundlagen der genetischen Algorithmen . . . . .	5
2.1.1	Was sind genetische Algorithmen . . . . .	5
2.1.1.1	Geschichtliche Entwicklung der GA . . . . .	6
2.1.1.2	Biologische Evolution . . . . .	6
2.1.2	Funktionsweise von genetischen Algorithmen . . . . .	7
2.1.2.1	Terminologie . . . . .	7
2.1.2.2	Ablaufschema . . . . .	10
2.2	Grundlagen der Trading-Tools . . . . .	15
2.2.1	Begriffsdefinition . . . . .	15
2.2.2	Klassifizierung von Trading-Tools . . . . .	15
2.3	Grundbegriffe der Finanz . . . . .	17
2.3.1	Wertpapierarten . . . . .	17
2.3.2	Traditionelle Finanzanalyse im Überblick . . . . .	19
2.3.2.1	Fundamentalanalyse . . . . .	20
2.3.2.2	Technische Analyse . . . . .	20
2.3.2.3	Random-Walk-Hypothese . . . . .	26
2.3.3	Elektronischer Handel . . . . .	26
2.3.4	Der Kostenfaktor . . . . .	26
2.3.5	Kundenprofil . . . . .	29
2.3.6	Handelssysteme . . . . .	29
2.4	Einsatz der genetischen Algorithmen in der Finanz . . . . .	31
2.4.1	Allgemeine Anwendungsgebiete . . . . .	31
2.4.2	Einsatz der GA in Trading-Tools . . . . .	32
2.4.2.1	Kauf/Verkauf Regel Generierung . . . . .	33
2.4.2.2	Kauf/Verkauf Regel Optimierung . . . . .	34



2.4.2.3	Portfolio-Optimierung . . . . .	35
2.4.2.4	Kursprognose . . . . .	35
<b>3</b>	<b>Fallbeispiel Merchant of Venice</b>	<b>37</b>
3.1	Allgemeine Produkt Beschreibung . . . . .	38
3.1.1	Kurzbeschreibung . . . . .	38
3.1.2	Software Packages und Programmaufbau . . . . .	38
3.1.3	Grundfunktionalitäten . . . . .	39
3.1.4	Lizenz und Kostenpunkt . . . . .	48
3.1.5	Systemanforderung . . . . .	48
3.2	Einsatz der GA . . . . .	49
3.2.1	Kauf/Verkauf Regeln Optimierung mittels GA . . . . .	49
3.2.1.1	Parameter Einstellung . . . . .	49
3.2.1.2	Verlauf der Optimierung . . . . .	52
3.2.1.3	Empirische Versuche . . . . .	53
3.2.2	Kauf/Verkauf Regeln Generierung mittels GP . . . . .	62
3.2.2.1	Parameter Einstellung . . . . .	62
3.2.2.2	Ablauf der Kauf/Verkauf Regel Generierung . . . . .	62
3.2.2.3	Empirische Versuche . . . . .	64
3.3	Performanz-Analyse der GA/GP Funktionalität unter ver- schiedenen Marktbedingungen . . . . .	67
3.3.1	Studie der GA Funktionalität im wachsenden Markt . . . . .	67
3.3.1.1	Parameter Konfiguration . . . . .	67
3.3.1.2	Das Ergebnis . . . . .	69
3.3.2	Studie der GA Funktionalität im turbulenten Markt . . . . .	71
3.3.2.1	Parameter Konfiguration . . . . .	71
3.3.2.2	Das Ergebnis . . . . .	72
3.3.3	Studie der GP Funktionalität im wachsenden Markt . . . . .	73
3.3.3.1	Parameter Konfiguration . . . . .	73
3.3.3.2	Das Ergebnis . . . . .	73
3.3.4	Studie der GP Funktionalität im turbulenten Markt . . . . .	74
3.3.4.1	Parameter Konfiguration . . . . .	74
3.3.4.2	Das Ergebnis . . . . .	74
<b>4</b>	<b>Fallbeispiel MultiCharts</b>	<b>76</b>
4.1	Allgemeine Produkt Beschreibung . . . . .	77
4.1.1	Kurzbeschreibung . . . . .	77
4.1.2	Software Packages und Programmaufbau . . . . .	77
4.1.3	Grundfunktionalitäten . . . . .	78
4.1.4	Lizenz und Kostenpunkt . . . . .	86
4.1.5	Systemanforderung . . . . .	87
4.2	Einsatz der GA . . . . .	89

4.2.1	Strategie-Optimierung mittels GA . . . . .	89
4.2.1.1	Parameter Einstellung . . . . .	89
4.2.1.2	Ablauf der Optimierung . . . . .	92
4.2.1.3	Empirische Versuche . . . . .	93
4.2.2	Portfolio-Optimierung mittels GA . . . . .	97
4.3	Performanz-Analyse der GA-basierte Portfolio-Optimierung unter verschiedenen Marktbedingungen . . . . .	99
4.3.1	Studie der GA-basierten Portfolio-Optimierung im wachsenden Markt . . . . .	99
4.3.1.1	Parameter-Konfiguration: Setting 1 . . . . .	100
4.3.1.2	Das Ergebnis . . . . .	100
4.3.1.3	Parameter-Konfiguration: Setting 2 . . . . .	101
4.3.1.4	Das Ergebnis . . . . .	101
4.3.2	Studie der GA-basierten Portfolio-Optimierung im turbulenten Markt . . . . .	101
4.3.2.1	Parameter-Konfiguration: Setting 1 . . . . .	101
4.3.2.2	Das Ergebnis . . . . .	103
4.3.2.3	Parameter-Konfiguration: Setting 2 . . . . .	103
4.3.2.4	Das Ergebnis . . . . .	103
<b>5</b>	<b>Diskussion der technisch-strategischen Aspekte von Trading-Tools und genetischen Algorithmen</b>	<b>104</b>
5.1	Vergleich der Funktionalitäten . . . . .	105
5.1.1	Kursdatenimport/-export und Management . . . . .	105
5.1.2	Grafische Oberfläche . . . . .	105
5.1.3	Analysetechniken . . . . .	106
5.1.4	Handelstrategien: Erzeugung, Optimierung und Testen	106
5.1.5	Portfolio-Management und Backtesting . . . . .	107
5.2	Diskussion der Vor- und Nachteile . . . . .	108
5.2.1	Vor-/Nachteile von Merchant of Venice . . . . .	108
5.2.2	Vor-/Nachteile von MultiCharts . . . . .	109
5.3	Vergleich der Ergebnisse aus der Performanz-Analyse . . . . .	111
5.3.1	Kauf/Verkauf Regel Optimierung mittels GA . . . . .	111
5.3.2	Kauf/Verkauf Regel Generierung mittels GP . . . . .	115
5.3.3	Analyse der Ergebnissen zu den empirischen Versuchen	116
5.3.3.1	Versuch in Sektion 3.2.1.3 . . . . .	116
5.3.3.2	Versuch in Sektion 3.2.2.3 . . . . .	117
<b>6</b>	<b>Conclusio</b>	<b>119</b>
6.1	Interpretation der Untersuchungsergebnisse . . . . .	120
6.2	Die Grenzen der genetischen Algorithmen . . . . .	123

# Abbildungsverzeichnis

2.1	Beispiel einer Roulette-Wheel Selektion mit 5 Individuen . . .	8
2.2	Beispiel einer gleichverteilten Mutation mit dem mutierten Gen $X_4$ . . . . .	9
2.3	Beispiel eines 1-Punkt-Crossovers mit Trennstelle $p = 2$ . . .	10
2.4	Ablaufschema von genetischen Algorithmen . . . . .	11
2.5	GD Verlauf in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien . . . . .	21
2.6	Momentum Verlauf in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien . . . . .	22
2.7	RSI Verlauf in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien . . . . .	24
2.8	MACD Linie und Signallinie in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien . . . . .	25
2.9	Transaktionskosten unter dem Mindest-Ordervolumen . . . .	27
2.10	Transaktionskosten bei einer Ordervolumen von 5000 und 50000 Euro . . . . .	28
3.1	Screenshot des Import von End-Of-Day Kursdaten über die online Yahoo Finance Anbindung . . . . .	40
3.2	Screenshot des Datei Import von Kursdaten . . . . .	40
3.3	Screenshot der Intra-Day Kursdaten Import Funktion . . . .	40
3.4	Screenshot der Kursdaten Export Funktion . . . . .	41
3.5	Screenshot der Graphenfunktionalität . . . . .	42
3.6	Screenshot der Tabellen Darstellung der SAP Aktie . . . . .	43
3.7	Screenshot der Portfolio Ansicht . . . . .	44
3.8	Screenshot der Funktionalität „New Transaction“ . . . . .	45
3.9	Screenshot der Transaktionshistorie . . . . .	45
3.10	Screenshot vom Reiter „Range“ und „Rules“ der Paper Trade Funktionalität . . . . .	46
3.11	Screenshot der Ergebnisse einer Real-Simulation mit Paper Trade anhand von SAP.DE . . . . .	46
3.12	Screenshot der Reiter „Range“ und „Rules“ im GA Fenster .	50
3.13	Screenshot der Reiter „Portfolio“ und „Trade Value“ im GA Fenster . . . . .	50

3.14	Screenshot der Reiter „GA Params“ im GA Fenster . . . . .	51
3.15	Kursentwicklung der SAP Aktie (XETRA) im Zeitraum 2003 bis 2009 . . . . .	54
3.16	Kursentwicklung der Siemens Aktie (XETRA) im Zeitraum 2003 bis 2009 . . . . .	55
3.17	Screenshot der Top 5 Ergebnisse der GA-Parameter Optimierung . . . . .	57
3.18	Screenshot der Paper-Trading Transaktionen des besten Ergebnis . . . . .	57
3.19	Screenshot der Top 5 Ergebnisse der GA-Parameter Optimierung . . . . .	58
3.20	Kursentwicklung der SAP Aktie (XETRA) im Zeitraum 01/01/2009 bis 30/04/2009 . . . . .	60
3.21	Kursentwicklung der Siemens Aktie (XETRA) im Zeitraum 01/01/2009 bis 30/04/2009 . . . . .	61
3.22	Screenshot der Einstellungsmöglichkeit für alle Elemente der Kauf/Verkauf Regel im Reiter „GP Language“ . . . . .	63
3.23	Screenshot der Ergebnisse der GP Kauf/Verkauf Regel Generierung . . . . .	64
3.24	Performanz der DAX in den letzten 5 Jahren . . . . .	68
4.1	Screenshot des QuoteManager in MultiCharts . . . . .	78
4.2	Screenshot der Kursdatenquellen in MultiCharts . . . . .	79
4.3	Beispiel Abbildung eines Chartwindow in MultiCharts . . . . .	81
4.4	Screenshot der Reiter „Signals“ und „Indicator“ . . . . .	83
4.5	Screenshot des Sub-Programms „3D Optimization Charts“ in MultiCharts . . . . .	85
4.6	Screenshot der Einstellungen für auf GA-basierende Strategie-Optimierung in MultiCharts . . . . .	90
4.7	Screenshot der Einstellungen für auf GA-basierende Strategie-Optimierung in MultiCharts . . . . .	91
4.8	Kursentwicklung der SAP Aktie (XETRA) im Zeitraum 2006 bis 2009 . . . . .	94
4.9	Screenshot des Subprogramms „MultiCharts Portfolio-Backtester“ . . . . .	98
4.10	Screenshot vom Reiter „Portfolio Settings“ im Subprogramm „MultiCharts Portfolio-Backtester“ . . . . .	98
5.1	DAX im Zeitraum 2006 bis 2008 . . . . .	114
5.2	DAX im Zeitraum 2008 bis 2009 . . . . .	114

# Tabellenverzeichnis

2.1	Bekannte Aktienindizes . . . . .	18
2.2	Ratingsymbole der Ratingagenturen Moody und Standard & Poor's . . . . .	19
3.1	Kauf/Verkauf Regel . . . . .	54
3.2	Intervallbereiche der Kauf/Verkauf Regel Parameter . . . . .	54
3.3	GA spezifische Parameter - Setting 1 . . . . .	56
3.4	GA spezifische Parameter - Setting 2 . . . . .	56
3.5	Ergebnisse der Paper Trade mit Kauf/Verkauf Regel aus Versuch 1 und 2 . . . . .	59
3.6	Ergebnisse der Paper Trade mit Kauf/Verkauf Regel aus Versuch 3 und 4 . . . . .	59
3.7	GP spezifische Parameter . . . . .	64
3.8	10 ausgewählte Aktientitel aus dem DAX . . . . .	68
3.9	Ausführungsspezifische Parameter . . . . .	69
3.10	GA spezifische Parameter . . . . .	69
3.11	Ergebnisse der Parameter Optimierung im Zeitraum 2006 bis 2008 . . . . .	70
3.12	GA spezifische Parameter . . . . .	70
3.13	10 ausgewählte Aktientitel aus dem DAX . . . . .	71
3.14	Ausführungsspezifische Parameter . . . . .	71
3.15	GA spezifische Parameter . . . . .	72
3.16	Ergebnisse der Parameter Optimierung im Zeitraum 2006 bis 2008 . . . . .	72
3.17	GA spezifische Parameter . . . . .	73
3.18	GP spezifische Parameter . . . . .	73
4.1	GA-Parameter: Setting 1 . . . . .	94
4.2	GA-Parameter: Setting 2 . . . . .	95
4.3	GA-Parameter: Setting 3 . . . . .	95
4.4	Kauf/Verkauf Signale . . . . .	95
4.5	Ergebnistabelle mit verschiedenen Settings . . . . .	96
4.6	10 ausgewählte Aktientitel aus dem DAX . . . . .	99
4.7	Ausführungsspezifische Parameter . . . . .	100

4.8	10 ausgewählte Aktientitel aus dem DAX . . . . .	101
4.9	Ausführungsspezifische Parameter . . . . .	102
5.1	10 ausgewählte Aktientitel aus dem DAX für den wachsenden Markt . . . . .	112
5.2	10 ausgewählte Aktientitel aus dem DAX für den turbulenten Markt . . . . .	112
5.3	In Merchant of Venice und in MultiCharts optimierte Kauf/-Verkauf Regeln . . . . .	112
5.4	Vergleich der Ergebnisse im wachsenden Markt . . . . .	113
5.5	Vergleich der Ergebnisse im turbulenten Markt . . . . .	113
5.6	Ergebnisse der Kauf/Verkauf Regel Generierung im wachsenden und im turbulenten Markt . . . . .	115
5.7	Vergleich der GA-Parameter-Settings . . . . .	116
5.8	Ergebnisse der empirischen Versuche im Zeitraum 01/01/2008 bis 01/01/2009 und im Zeitraum 01/01/2009 bis 30/04/2009 mit unbekanntem Daten . . . . .	117
5.9	Ergebnisse der empirischen Versuche zur GA-basierten Kauf/Verkauf Regel Optimierung im Zeitraum 01/04/2003 bis 01/01/2009 und im Zeitraum 01/01/2009 bis 30/04/2009 mit unbekanntem Daten . . . . .	117
5.10	Ergebnisse der empirischen Versuche zur GP basierten Kauf/-Verkauf Regel Generierung in den Zeiträumen 01/04/2003 bis 01/01/2009 und 01/01/2008 bis 01/01/2009 und im Zeitraum 01/01/2009 bis 30/04/2009 mit unbekanntem Daten . . . . .	118
6.1	Vergleich der Funktionalitäten von Merchant of Venice und MultiCharts . . . . .	120
6.2	Vor- und Nachteile von Merchant of Venice und MultiCharts	121

# Kapitel 1

## Einleitung

### 1.1 Problemstellung

In den Jahren vor der Wirtschaftskrise ist das Interesse der Privatanleger an kurzfristigen und ertragreichen Aktieninvestments durch das hohe Wirtschaftswachstum und die ausgezeichneten Aktienkurse stark gestiegen. Immer mehr Investitionsgelder von Privatanlegern flossen in den Aktienmarkt. Profitmaximierung ist das Hauptziel jedes getätigten Investments. In der Zeit der Wirtschaftskrise gewinnt neben Profitmaximierung auch Risikominimierung immer mehr an Bedeutung. Aus diesem Anlass ist auch das Interesse an neuen und wirkungsvollen Instrumenten bzw. Verfahren gestiegen, die versprechen, gezielt solche Probleme zu lösen. Wissenschaftlich gesehen, sprechen wir hier von Problemlösungsverfahren oder auch Optimierungsverfahren. Bekannte traditionelle Verfahren sind zum Beispiel Greedy Algorithmen und A\* Suchverfahren. Diese Arten von Optimierungsverfahren sind komplex, kostenintensiv und zeitaufwändig in der Implementierung und Anwendung. Andere statistische und strategische Methoden wie zum Beispiel technische Analysen oder „Buy-And-Hold“ Strategie sind auch keine Garantie für ein sicheres Return on Investment (ROI).

Als ein neuer und innovativer Impuls wurde die Forschung im Bereich der künstlichen Intelligenz und der evolutionären Algorithmen in die Problemlösungsmöglichkeiten integriert. Evolutionäre Algorithmen, vor allem genetische Algorithmen (GA), fanden breite Einsatzgebiete in der Finanz und Ökonomie (siehe [BS03], [AK99], [KSS06], [Szp02], [Kop92], [Wri91], [CLP98], [GKK04] und [PB00]).

Man findet im Web viele proprietäre und Open Source Trading Tools, die gezielt für Privatanleger entwickelt wurden. Eine kleine Menge darunter verwendet die Technologien der evolutionären Strategien, neuronaler Net-

ze, sowie genetischer Algorithmen zur Unterstützung oder Optimierung von Investment Entscheidungen<sup>1</sup>.

Viele Anbieter bewerben auf ihren Homepages ihre Software Pakete mit außergewöhnlichen Funktionen, innovativen Strategien und zutreffenden Kurs-Prognosen<sup>2</sup>. Der Kaufpreis solcher Software ist oftmals hoch angesetzt. Gerade für Privatanleger ohne umfassende Technologiekenntnisse ist die Qualität und Praxistauglichkeit der Produkte nicht transparent. Daraus ergibt sich die Frage, ob die Anwendungen den Erwartungen von Privatanlegern in Anbetracht spezifischer Qualitätskriterien und Erwartungen technisch bzw. strategisch gerecht werden können. Die Problematik ist vielfältig und wird in Rahmen dieser Arbeit anhand von zwei Fallbeispielen analysiert und diskutiert.

## 1.2 Zielsetzung

Ziel der Arbeit ist es, Trading-Tools zu veranschaulichen, die auf der Technologie der genetischen Algorithmen basieren. Es sollen anhand von zwei Fallbeispielen die GA-Funktionen und deren Performanz analysiert, dokumentiert und präsentiert werden. Es soll mitunter untersucht werden, ob und inwiefern die Technologie der genetischen Algorithmen Privatanleger in Investmententscheidungen unterstützt. Weiters sollen Aussagen getroffen werden, wie beide Tools im direkten Vergleich in Hinblick auf ausgewählte Kriterien zu einander abschneiden.

## 1.3 Methodik und Aufbau der Arbeit

Die Arbeit gliedert sich in sechs Kapitel. Nach der Einleitung folgt eine kompakte Einführung in die Thematik. Themen wie die Entstehung und Arbeitsweise der genetischen Algorithmen, die geschichtlichen und biologischen Hintergründe, Definition des Terminus „Trading-Tools“, Grundbegriffe der Investment-Finanz und Anwendungsmöglichkeiten der GA werden erläutert und miteinander in Zusammenhang gebracht. Nach der Einführung folgen

---

<sup>1</sup>Im Internet findet sich nach einfacher Suche folgende auf der Technologie der genetischen Algorithmen basierende Trading Software: Investox [Knö09], Trading Solutions [Neu09], Wave59 [Tec09], Merchant of Venice [LM09], NeuroShell [Gro09] und Tradecision [Res09]

<sup>2</sup>Siehe Produktbeschreibung Neuroshell: <http://www.neuroshell.com/geewhiz.asp>, Sample Performance NeuroDimension: <http://www.tradingsolutions.com/products/sample.html> und Produkt Information Tradecision [http://www.tradecision.com/product/trading\\_software.htm](http://www.tradecision.com/product/trading_software.htm)



zwei Fallbeispiele. Es existieren verschiedene auf GA basierende Trading-Tools, die als Fallbeispiele in Frage kämen. Tools wie Investox der Firma Knöpfel Software Entwicklung [Knö09], TradingSolutions der NeuroDimension inc. [Neu09], Wave59 von Wave59 Technologies int'l. Inc. [Tec09], MultiCharts von TS Support [Sup09] und die Open-Source Software - Merchant of Venice [LM09] wurden für die Vorentscheidung untersucht. Leider sind viele dieser Programmen nicht bzw. nur beschränkt evaluierbar<sup>3</sup>. Aus diesem Grund werden nur zwei Tools als Fallbeispiele in dieser Arbeit herangezogen. Beide Tools werden im Kapitel 3 und im Kapitel 4 zuerst auf allgemeine Eigenschaften wie Funktionsumfang, Lizenz, Kostenpunkt und Systemanforderungen untersucht. Anschließend wird der Einsatz der genetischen Algorithmen im Programm im Detail erläutert. Weiters wird die Performanz der GA-Funktionen analysiert und dokumentiert. Von Interesse ist vor allem die Performanz in speziellen Wirtschaftsbedingungen. Es werden die Performanz sowohl in einem wachsenden als auch in einem turbulenten Markt untersucht. Als Performanz Kriterium wird Return of Initial Capital herangezogen. Return of Initial Capital ist die prozentuelle Relation zwischen dem Initialkapital und dem Nettogewinn. Die Ergebnisse der Performanzanalysen werden als Abschluss des dritten und vierten Kapitels präsentiert.

Im fünften Kapitel werden Merchant of Venice und MultiCharts in direkten Vergleich gesetzt. Sowohl die Funktionalitäten als auch die Performanz beider Fallbeispiele werden einander gegenübergestellt. Anhand der Untersuchungsergebnisse werden weiters die Vor- bzw. Nachteile beider Tools beschrieben. Als Abschluss der Arbeit wird im letzten Kapitel die Untersuchungsergebnisse interpretiert und über die Grenzen der genetischen Algorithmen diskutiert.

---

<sup>3</sup>Diverse Anbieter wie zum Beispiel NeuroShell und Tradecision sind nicht bereit zeitlimitierte Testversionen den Interessenten zur Verfügung zu stellen. Investox bietet online eine zeitlich und funktional beschränkte Evaluationsversion an. Die Funktionen der genetischen Algorithmen wurden in der Testversion deaktiviert. Auch TradingSolutions bietet eine zeitlich und funktional beschränkte Evaluationsversion an.

## Kapitel 2

# Grundlagen

Dieses Kapitel ist ein Grundlagenkapitel und gliedert sich in mehrere Thematiken. Die erste gibt eine Einführung in die Grundlagen und Arbeitsweisen der genetischen Algorithmen (GA). Ausgehend von der Definition genetischer Algorithmen und einer kurzen Beschreibung der geschichtlichen Entwicklung und biologischen Hintergründe folgt eine detaillierte Beschreibung der Arbeitsweise der genetischen Algorithmen. Zum Schluss werden die Anwendungsbereiche der Algorithmen beleuchtet.

Die zweite Thematik befasst sich mit dem Begriff “Trading-Tool“. In Folge der Begriffsdefinition werden unterschiedliche Funktionen von den so genannten Trading-Tools erläutert.

Die dritte Thematik befasst sich mit den Begrifflichkeiten der Investment Finanz. Die Zusammenhänge zwischen den wichtigsten Begriffen werden erklärt. Anschließend werden die bekanntesten Finanzanalysemethoden präsentiert.

Zum Schluss werden verschiedene Anwendungsmöglichkeiten von genetischen Algorithmen in der Finanz und insbesondere in Trading-Tools erläutert.

Mit diesem Kapitel wird dem Leser das Grundwissen über die genetischen Algorithmen, der Trading-Tools und der Finanz vermittelt um den Bezug zur Hauptproblematik dieser Arbeit herzustellen.

## 2.1 Grundlagen der genetischen Algorithmen

### 2.1.1 Was sind genetische Algorithmen

Genetische Algorithmen sind stochastische Optimierungsverfahren, die von John H. Holland [Hol75] im Jahr 1975 entwickelt und später von David E. Goldberg [Gol89] in das Ingenieurwesen eingeführt wurden. Die Grundidee der GA stammt aus der Natur, inspiriert durch die Evolutionstheorie von Charles Darwin. Darwins Werk "On the Origin of Species by Means of Natural Selection" [Dar02] beschreibt die Fähigkeit der Lebewesen und Organismen in der Natur, durch gewisse Veränderungen des eigenen Erbguts, sich an die Umwelt und Lebensbedingungen anzupassen. Diese Mechanismen, die die Änderungen bewerkstelligen und somit die Evolutionsprozesse auslösen, dienen als Basis der genetischen Algorithmen.

Michalewicz beschreibt die Grundidee der GA am Beispiel von Hasen:

„The idea behind genetic algorithms is to do what nature does. Let us take rabbits as an example: at any given time there is a population of rabbits. Some of them are faster and smarter than other rabbits. These faster, smarter rabbits are less likely to be eaten by foxes, and therefore more of them will survive to do what rabbits do best: make more rabbits. ... The resulting baby rabbits will (on average) be faster and smarter than those in the original population because more faster, smarter parents survived the foxes.“

*Genetic Algorithms + Data Structures = Evolution Programs [Mic96]*

Die Anwendung der biologischen Evolution lässt sich bei einem Standard-GA so vereinfachen: Genetische Algorithmen simulieren ein künstliches Ökosystem, das aus einer Population von Individuen besteht. Die Überlebenswahrscheinlichkeit einzelner Individuen wird als "Fitness" abgebildet und mittels einer Fitness-Funktion ausgewertet. Die für das Problem am optimalsten erscheinenden Individuen werden selektiert zur Reproduktion. Im Reproduktionsvorgang werden die Eigenschaften der selektierten Kandidaten durch verschiedene, verschieden wahrscheinlich auftretende genetische Operationen verändert. Die daraus entstehende Nachkommengeneration kann ähnliche Eigenschaften der Elterngeneration, aber auch sehr stark mutierte, um vielfaches verbesserte Eigenschaften aufweisen. Auf die Nachkommengeneration wird das selbe Verfahren mehrmals angewendet bis eine optimale bzw. Bereichs-optimale Lösung gefunden wird, oder bis ein Abbruchkriterium erfüllt wird.

### 2.1.1.1 Geschichtliche Entwicklung der GA

Die Anwendung der Evolutionstheorie im Bereich der Optimierungsverfahren und Optimierungssysteme wurde von zwei Forschern fast zeitgleich und unabhängig von einander untersucht.

Ab Mitte der sechziger Jahre entwickelten Ingo Rechenberg und Hans-Paul Schwefel an der TU Berlin eine Optimierungsmethode für die Reduzierung des Strömungswiderstandes bei verschiedenen Körperformen [Sch65]. Aus dieser Forschung resultierten später die Evolutionsstrategien [Rec73] (ES). John H. Holland erforschte in den siebziger Jahren an der University of Michigan die Anwendung der Evolutionstheorie im Bereich der künstlichen Intelligenz (siehe [Hol75]). Aus seinen Forschungen resultierte das Fachgebiet der genetischen Algorithmen.

Das Grundschema der ES und der GA ist fast identisch. Die Unterschiede liegen in den Details der Datenstrukturen zur Darstellung der Lösungen und der Realisierung der Evolutionszyklen. Sowohl ES als auch GA bedienen sich Optimierungsproblemen und werden heute zu den Gruppen der evolutionären Algorithmen zusammengefasst.

### 2.1.1.2 Biologische Evolution

Genetische Algorithmen haben ihren Ursprung in der Biologie und in der Evolutionstheorie von Darwin [Hol75]. In diesem Absatz werden die Aspekte der biologischen Evolution beleuchtet. Die wichtigen Elemente der Vererbungsvorgänge finden im Zellkern jedes Lebewesen statt.

Jedes Lebewesen trägt im Kernplasma des Zellkerns Chromosomen. Chromosomen sind die Träger der Gene und bestehen aus Nukleinsäuren und Proteinen. Die bekannteste Nukleinsäure ist die Desoxyribonukleinsäure (DNS). Die DNS besteht wiederum aus vier Basen: Adenin, Guanin, Cytosin und Thymin. Die Reihenfolge der Basen ist insofern wichtig, da die Basensequenzen der DNS und die Anzahl der Basen die genetischen Erbinformationen kodieren. Erbinformationen sagen zum Beispiel aus, welche Augenfarbe ein Mensch hat, oder gegen welche Krankheiten ein Lebewesen anfällig ist.

Bei einer Zellteilung verdrillen sich die Chromosomen zu einer Helix. Hier werden die Erbinformationen über die Chromosomen auf die neu entstehende Zelle verteilt. So enthalten die neugebildeten Zellen wieder die vollständigen Erbinformationen. Bei einer geschlechtlichen Zellteilung werden Chromosomen miteinander rekombiniert und es kommt zu einer Vermischung oder auch Veränderung der Erbinformationen. Hier können verschiedene Mechanismen auftreten, die die Veränderungen der Gene beeinflussen. Die drei Mechanismen Selektion, Mutation und Rekombination sind maßgeblich für

die Artenvielfalt und die Veränderungen in der Evolution aller Lebewesen. Die Funktionsweise der Mechanismen werden in Kapitel 2.1.2 in Zusammenhang mit der Terminologie der GA im Detail beschrieben.

## 2.1.2 Funktionsweise von genetischen Algorithmen

### 2.1.2.1 Terminologie

Die Grundprinzipien der genetischen Algorithmen basieren auf dem Modell eines natürlichen Ökosystems. Ausgehend von diesem Modell wurden die Elemente und Prozesse abstrahiert und eine Terminologie dazu definiert (vgl. [Hol75]).

**Individuum:** Ein Individuum ist eine Darstellung eines Chromosoms. Bei Optimierungsproblemen ist ein Individuum eine mögliche Lösung des Problems. Individuen können aus Zeichenketten, Zahlenfolgen oder Variablen bestehen und werden anschließend binär kodiert. Das heißt, Individuen bestehen nach der Codierung ausschließlich aus Nullen und Einsen. Jede Eins oder Null bildet ein Gen ab. Ein Beispiel Individuum:  $x = 1,1,0,0,1,1,0$

**Population:** Eine Population ist die Gesamtmenge von Individuen. Mathematisch gesehen repräsentiert eine Population den gesamten Suchraum eines Problems. Eine Population kann aus hundert oder auch hunderttausend Individuen bestehen. Ihre Größe ist von der Problemstellung abhängig und kann somit sehr stark variieren.

Die Startpopulation ist die erste Population. Die Startpopulation wird in den meisten Fällen zufallsbedingt erzeugt.

**Genetische Operation oder Operator:** Eine genetische Operation ist ein Verfahren zur Generierung eines modifizierten neuen Individuums. Beispiele von genetischen Operationen sind Crossover und Mutation.

**Fitness:** Man kann jedem Individuum einen Fitnesswert zuweisen. Fitnesswerte werden mit Fitnessfunktionen ermittelt und stellen die Überlebenswahrscheinlichkeiten eines Individuums dar. In der biologischen Evolution überleben die Individuen mit den höchsten Fitnesswerten. Bei Optimierungsproblemen im mathematischen Sinn geben die Fitnesswerte an, wie gut ein Individuum als Lösungskandidat in Frage kommen würde. Die Fitnessfunktion und die Gütekriterien werden dem Problem bzw. der Anwendung entsprechend definiert. Anders als in der Natur werden nicht immer die fittesten Individuen selektiert. Die Kriterien zur Selektion werden auch anwendungsspezifisch definiert.

**Selektion:** Die Selektion dient der Auswahl von Individuen zur Generierung neuer Individuen. Grundlegend für die Selektion sind die Selektionskriterien bzw. der Selektionsalgorithmus. Selektionskriterien besagen, wie die Auslese stattfindet und welche Individuen von der Population ausgewählt werden sollen.

Beispiele von Selektionsalgorithmen sind: Roulette-Wheel Selektion, Rang-basierte Selektion, elitäre Selektion, Nachbarschafts-Selektion und Wettbewerbs-Selektion. Selektionsalgorithmen unterscheiden sich voneinander im Gegenstand der Bewertung.

Die Roulette-Wheel Selektion hat diesen Namen, weil der Selektionsalgorithmus wie ein Roulette Spiel funktioniert. Die Fitness jedes Individuums ist abgebildet auf dem Roulette. Je grösser ein Anteil, desto grösser ist der Fitnesswert des dazugehörigen Individuums. Die Selektion der Elterngeneration ist vergleichbar mit dem Drehen des Roulettes und geschieht zufällig. Dennoch haben Individuen mit größeren Fitnesswerten eine größeren Chance selektiert zu werden, da ihre Anteile im Roulette grösser sind. Abbildung 2.1 zeigt eine Darstellung mit fünf Individuen und den dazugehörigen Fitnesswerten in Prozent.

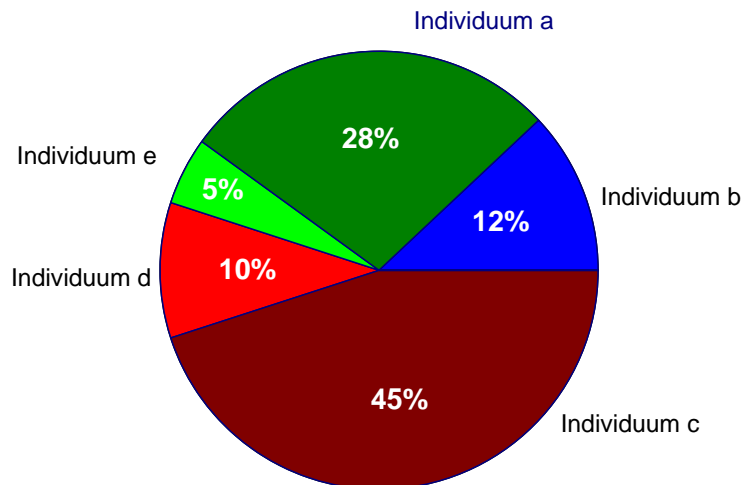


Abbildung 2.1: Beispiel einer Roulette-Wheel Selektion mit 5 Individuen

Bei einem anderen Selektionsalgorithmus - der Wettbewerbs-Selektion - werden Paare ausgesucht, die direkt gegeneinander antreten. Das fittere Individuum der beiden Wettbewerber wird selektiert. So werden

die schwächsten Individuen nie selektiert.

Bei der elitären Selektion werden ausschließlich Individuen mit dem höchsten Fitnesswert selektiert. Das Prinzip der elitären Selektion ist auch bekannt unter der Bezeichnung - „Survival of the Fittest“.

Selektionsalgorithmen dieser Art haben den Nachteil, dass die Diversität der Individuen durch einseitige Selektion in Grenzen gehalten wird und somit die Lösungen suboptimal konvergieren.

**Mutation:** Mutation ist ein genetischer Operator, bei dem die Erbinformationen eines einzelnen Individuums modifiziert werden. Die Modifikation kann ein oder auch mehrere Gene betreffen. Durch viele Einflüsse aus der Umwelt können Gene mutieren. Die Wahrscheinlichkeit einer Mutation wird durch mutationauslösende Stoffe, die so genannten Mutagene, erhöht. Mutationen können die Fitnesswerte eines Individuums steigern, aber auch drastisch senken. Beispiele für sich negativ auswirkende Mutationen sind Mutationen in Folge von Radioaktivität. Im Rahmen eines GAs wird die Mutationswahrscheinlichkeit problemspezifisch definiert. Wichtige Varianten des Mutations-Operators sind: Gleichverteilte Mutation und Normalverteilte Mutation.

Anhand der Abbildung 2.2 sieht man ein einfaches Beispiel einer Gleichverteilten Mutation.  $p$  ist eine Zufallszahl und liegt in dem Beispiel in einem Wertebereich von 0 bis 6. Die Zufallszahl  $p$  ist gleichverteilt. In der Abbildung 2.2 enthält  $p$  einen Integerwert von 4, somit wird das Gen an der fünften Stelle, also  $X_4$  mutiert. Bei der Normalverteilten Mutation ist die Zufallszahl  $p$  normalverteilt.

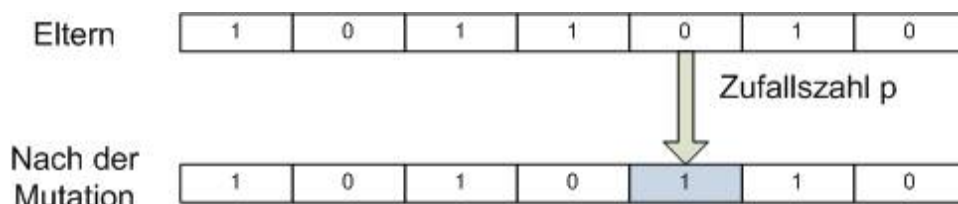


Abbildung 2.2: Beispiel einer gleichverteilten Mutation mit dem mutierten Gen  $X_4$

**Rekombination:** Die Rekombination, auch Crossover genannt, simuliert die geschlechtliche Fortpflanzung zweier Individuen. Bei einer Rekombination werden Gene zweier Eltern-Individuen mit einander vermischt. Crossover-Mechanismen können problemspezifisch angepasst zum Einsatz kommen. Variationen der Crossover-Mechanismen sind

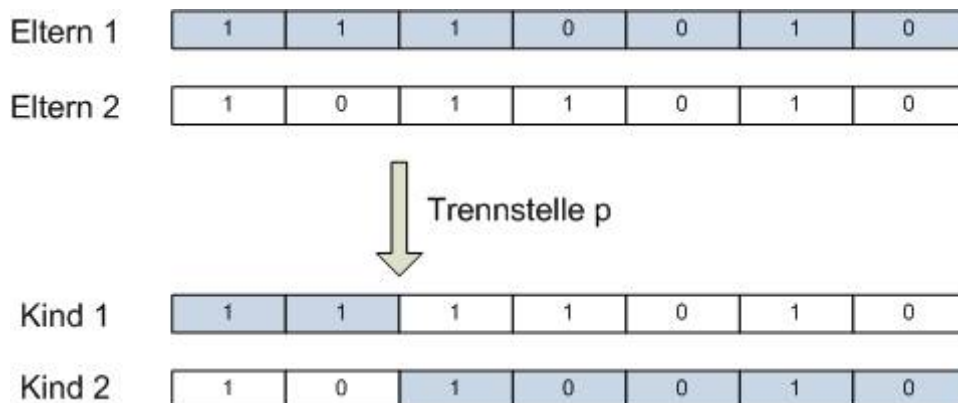


Abbildung 2.3: Beispiel eines 1-Punkt-Crossovers mit Trennstelle  $p = 2$

1-Punkt-Crossover, N-Punkt-Crossover, Uniform-Crossover, Shuffle-Crossover, Intermediärer Crossover, Linearer Crossover und Tausch-Crossover.

Die Abbildung 2.3 zeigt einen 1-Punkt-Crossover mit der Trennstelle  $p$ . Die Trennstelle  $p$  in dem Beispiel ist eine Zufallszahl mit einem Wertebereich von 0 bis 6. Es entstehen zwei Kinder nach der Operation. Kind 1 übernimmt bis zur Trennstelle  $p$  die Werte des ersten Elternteils und nach der Trennstelle die Werte des zweiten Elternteils. Kind 2 übernimmt jedoch bis zur Trennstelle  $p$  die Werte des zweiten Elternteils und nach der Trennstelle die Werte des ersten Elternteils. Rekombinationsmechanismen wie zum Beispiel der N-Punkt-Crossover funktionieren nach einem ähnlichen Prinzip, besitzen jedoch mehrere Trennstelle.

### 2.1.2.2 Ablaufschema

Der nachfolgend dargestellte Pseudocode beschreibt den Ablauf eines Standard GA in neun Zeilen:

```

Class GenetischerAlgorithmus {
  initialisieren()
  do {
    auswertenFitness()
    Selektion()
    abarbeitenGenetischerOperator()
  }
  while Abbruchkriterien nicht erfuehlt

```



}

Ein Standard GA besitzt ein festes Ablaufschema und lässt sich in drei Hauptphasen einteilen:

1. Initialisierungsphase
2. Evolutionszyklus
3. Bestimmung des optimalen Individuums

Die Abbildung 2.4 zeigt die Phasen und die Unterprozesse einzelner Phase.

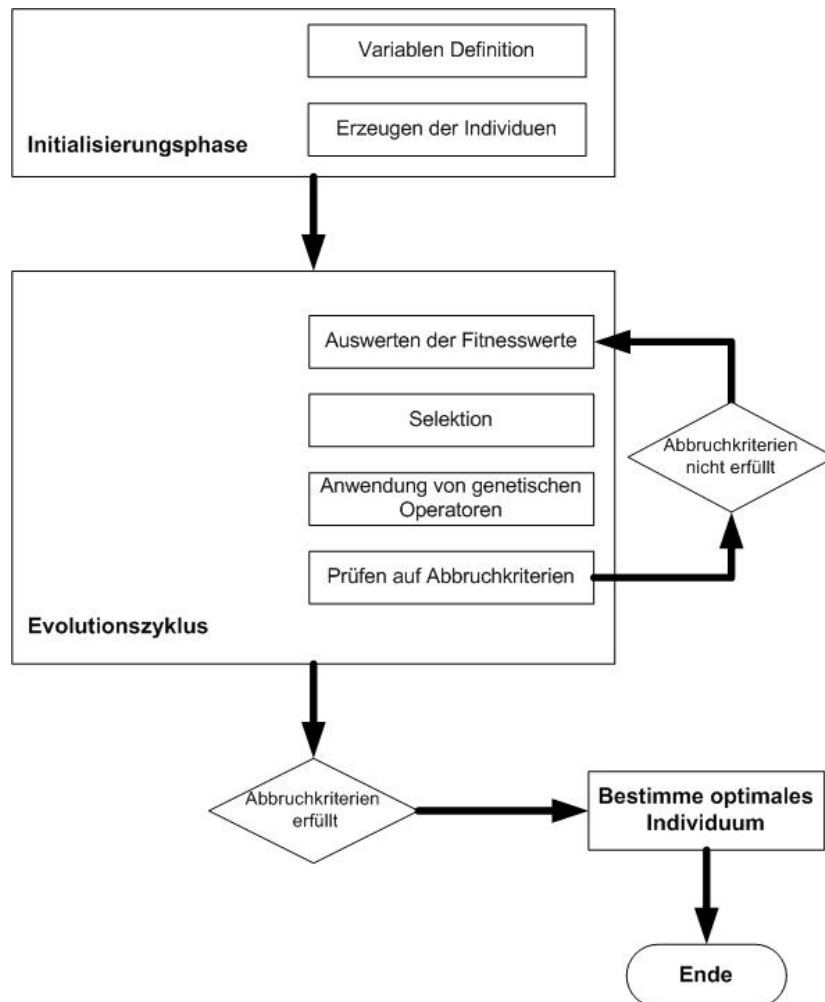


Abbildung 2.4: Ablaufschema von genetischen Algorithmen

## Initialisierungsphase

In der Initialisierungsphase werden problemspezifische Variablen und Funktionen definiert. Definiert werden:

- problemspezifisches Kodierungsformat für Individuen und deren Eigenschaften
- Suchraum des Optimierungsproblems
- Fitnessfunktion
- Selektionskriterien
- Abbruchkriterien.

Der Suchraum des Optimierungsproblems ist die Menge aller Individuen, die als Lösung in Frage kommen. Jedes Individuum im Suchraum wird in einem spezifizierten Kodierungsformat abgebildet und mit Hilfe der Fitnessfunktion die Optimalität der Lösungen berechnet. Die Berechnung kann mathematisch erfolgen, bei manchen Problemstellungen kann diese aber auch durch Beobachtungen oder Simulation erfolgen. Selektionskriterien bestimmen, wie Individuen ausgewählt werden. Die bekannteste, natürliche Auslese geschieht nach dem „Survival of the Fittest“ Prinzip. Hierbei werden von einer Population immer Individuen mit den höchsten Fitnesswerten selektiert. Neben den Selektionskriterien müssen auch die Abbruchkriterien definiert werden. Abbruchkriterien abstrahieren einen oder mehrere Zustände, die den Algorithmus beenden.

Nachdem alle Werte definiert wurden, wird die Initialisierungsfunktion `initialisieren()` aufgerufen. Die Initialisierungsfunktion erzeugt Individuen und kodiert sie in einem entsprechenden binären Format.

## Evolutionszyklus

Der Evolutionszyklus ist ein sich wiederholender Prozess. Er simuliert das Entstehen und das Sterben einer Spezies, einer Generation einer Spezies oder einzelner Individuen. Neue Spezies entstehen häufig durch verschiedene Anwendungen von genetische Operatoren wie z.B. die Mutationen. Das Sterben bzw. das Aussterben von Individuen ist auf schlechte Fitnesswerte zurückzuführen. Die wichtigsten genetische Operatoren, die das Entstehen und das Sterben von Individuen maßgeblich beeinflussen, sind Selektion, Mutation und Rekombination. Weitere genetische Operatoren, die in einem GA implementiert werden können, sind: Umkehrung (engl.: Inversion), Löschung

(engl.: Deletion) und Verdopplung (engl.: Duplication).

Der Ablauf eines typischen Evolutionszyklus kann folgenderweise abgebildet werden:

1. Am Anfang jedes Evolutionszyklus findet die Selektion statt. Durch diese werden Individuen ausgesucht, die der Fortpflanzung einer Kinder-Generation dienen. Durch verschiedenen Selektionsalgorithmen und die unterschiedlichen Selektionskriterien kann eine problemspezifische Auswahl der Elternpaare getroffen werden. Somit können die Qualität und die Vielfalt der Kinder-Generation stark beeinflusst werden.
2. Nach der Selektion werden verschiedene genetische Operationen auf die selektierten Individuen angewendet. Je nach Anforderung und konkretem Problem wird ein Mix von Gen verändernden Operationen implementiert. Diese sind zum Beispiel Mutation und Crossover. Durch die Wiederholung dieser zwei Schritte werden iterativ neue Generationen erzeugt, die optimalere Lösungen darstellen.
3. Der letzte Schritt des Evolutionszyklus ist die Überprüfung, ob die definierten Abbruchkriterien erfüllt sind. Wenn ein Kriterium erfüllt wurde, wird der Algorithmus abgebrochen. Wenn nicht kehrt der Algorithmus wieder zur Schritt 1 des Zyklus - der Selektion - zurück.

### **Abbruchkriterium**

Die Wahl von Abbruchkriterien beeinflusst die Qualität der Ergebnisse. Der Evolutionszyklus wird solange wiederholt, bis ein Zustand erreicht ist, der einem oder mehreren der definierten Abbruchkriterien entspricht. Übliche Abbruchkriterien sind:

- Eine Lösung wurde gefunden (die der Optimalitätserwartung entspricht)
- Die definierte Anzahl von Generationen wurde überschritten
- Die definierte Ausführungszeit wurde überschritten
- Die neu generierte Generation unterscheidet sich nicht oder kaum mehr von der vorherigen Generation. Somit kann die Optimalität der Lösung nicht mehr verbessert werden.

Wenn Kriterien unzureichend definiert werden, kann es dazu führen, dass die Algorithmen verfrüht enden. Konsequenzen sind somit weniger optimale

Lösungen.

Nach der Überprüfung wird das optimalste Individuum aus der letzten Generation von Individuen bestimmt.

## 2.2 Grundlagen der Trading-Tools

### 2.2.1 Begriffsdefinition

Der Begriff „Trading-Tools“ ist ein Sammelbegriff aus dem Bereich der Finanz, der jedoch in verschiedenen Nachschlagewerken<sup>4</sup> nicht vertreten ist.

Wir definieren den Begriff „Trading-Tool“ im Rahmen dieser Arbeit so: Ein computergestütztes Werkzeug, das Privatanleger bei Investitionen auf dem Finanzmarkt weitgehend unterstützt.

In der Finanz Fachliteratur [ER06] [Spa02] [Arn06] [Gei03] [May08] [Sch02] werden für den Begriff „Trading-Tool“ Synonyme wie „Börsensoftware“ oder „Trading-Software“ verwendet.

### 2.2.2 Klassifizierung von Trading-Tools

Trading-Tools dienen als Werkzeug und unterstützen Privatanleger durch Bereitstellung unterschiedlicher Funktionen bzw. Informationen. Da es nicht möglich ist, auf eine wissenschaftliche Klassifikation von Trading-Tools zurückzugreifen, wird hier eine merkmalsorientierte Klassifikation aufgestellt. Merkmale sind in dem Fall Funktionen der Tools gleichzusetzen.

Eine Dokumentation-basierte Recherche<sup>5</sup> über die Funktionen bekannter Trading-Tools ergibt folgende Resultate:

- Datenschnittstelle zum Importieren von Aktienkurs-Informationen
- Datenschnittstelle zum Exportieren von programm-spezifische Daten
- Funktion zur Darstellung von Kursinformation (Charting)
- Funktion zur Analyse von Kursdaten
- Funktion zur Kursprognose (Mit Hilfe von Indikatoren, Muster-Erkennung oder Einsatz von künstlicher Intelligenz und neuronalen Netzen)
- Generierung von manuellen und automatisierten Handelssystemen (Mit Hilfe von traditionellen Optimierungsstrategien oder Einsatz von genetischen Algorithmen und neuronalen Netzen)

---

<sup>4</sup>Gesucht wurde in Wikipedia, Britannica encyclopaedia, Enzyklopädie der Wirtschaftsinformatik, Meyers Lexikon Online, Wiktionary, Leo Wörterbuch. Suchwörter waren „Trading-Tool“, „Trading Tool“.

<sup>5</sup>Informationen wurden aus Internet Präsentationen und veröffentlichten Dokumentationen folgender Tools entnommen: Investox [Knö09], Trading Solutions [Neu09], Wave59 [Tec09], Merchant of Venice [LM09], NeuroShell [Gro09], Tradecision [Res09] AmiBroker [AMI09]

- Automatische Orderabgabe zu ausgewählten Online-Brokern
- Optimierung von Handelssystemen
- Portfolio Management
- Funktion zur Risikobewertung

Kursinformation bilden die Basis für Trading-Tools. Grundsätzlich müssen alle Trading-Tools die Fähigkeit zur Darstellung von Kursinformation besitzen. Die Darstellung kann in Listen-Form oder auch in Variationen von Diagramm-Formen (wie z.B. Balkendiagramm, Stabdiagramm) erfolgen. Die Bereitstellung von Kursinformation an sich ist optional. Viele Programme stellen Kursinformation dem Kunden nicht direkt zur Verfügung, sondern vielmehr Möglichkeiten Kursinformation durch interne oder fremde Schnittstellen zu laden. Je nach Schnittstelle und Anbieter ist der Service kostenpflichtig oder kostenfrei. Die Kosten sind umso höher, je akkurater die Echtzeit-Kursdaten (Ticks) sind. Sobald man die Kursinformation geladen und dargestellt hat, hat der Kunde die Möglichkeit, die Daten auf Risiken, Trends und Profitabilität zu analysieren. Viele der in der Finanz bekannten Analysemethoden (wie z.B. die technische Analyse, Fundamentalanalyse) werden hierfür herangezogen. Die Anzahl der zur Verfügung gestellten Analysemethoden unterscheidet sich von Tool zu Tool. Zusätzlich zu der Kursanalyse werden auch Kursprognosen durchgeführt. Viele Tools prognostizieren zukünftige Kurse basierend auf Trends und Indikatoren. Innovativere Tools setzen hier künstliche Intelligenz, wie z.B. neuronale Netze oder evolutionäre Algorithmen ein um möglichst treffsichere Prognosen zu erzielen. Die professionellen Tools generieren zu den prognostizierten Kursen zusätzlich noch Kauf/Verkauf Signale, die man anschließend als Tradingstrategie (Handelssystem) zusammenfassen, automatisieren oder auch individuell erweitern kann.

Viele der untersuchten Tools bieten auch noch weitaus spezialisiertere Funktionalitäten wie z.B. die Möglichkeit individuelle Handelssysteme mit einer Skriptsprache zu programmieren. Diese zählen nicht zu den Grundfunktionalitäten eines Trading-Tools, sondern verbessern die Qualität bzw. die Erweiterbarkeit des Tools und befriedigen spezielle Kundenwünsche. Daher sind diese nicht in der Auflistung zu finden.

Die Aufstellung der allgemeinen Funktionen eines Trading-Tools ist insofern wichtig, da die Anwendung der genetischen Algorithmen in den allgemeinen Funktionen integriert sein können. Die detaillierte Darstellung der Anwendungen von GA in Trading-Tools wird in Kapitel 2.4 erläutert.

## 2.3 Grundbegriffe der Finanz

In diesem Kapitel werden die Grundbegriffe des Finanzmarktes und ihre Zusammenhänge erklärt.

### 2.3.1 Wertpapierarten

#### Aktien

Auf dem Finanzmarkt können diverse Finanzinstrumente gehandelt werden. Die bekannteste Form ist die Aktie. Aktien sind Wertpapiere und stellen einen Anteil an einer Aktiengesellschaft dar [Mül07]. Die Aktionäre werden am Unternehmensgewinn im Form von Dividenden beteiligt. Der Preis der Aktie wird einerseits durch den Erfolg des Unternehmens, andererseits durch das Angebot und die Nachfrage bestimmt. Durch die schnelle Änderung von Angebot und Nachfrage, schwankt der Kurs der Aktien in einem kleinen Zeitraum<sup>6</sup>. Der Kauf bei niedrigen Kursen und Verkauf bei hohen Kursen bringt den Aktionären noch weitere Gewinne [Gsc91]. Den spekulativen Aspekt des Finanzmarktes nennt man Trading und er spielt eine sehr wichtige Rolle [Sch08]. Dieses Grundprinzip der Profitgenerierung ist essentiell für das Verständnis der Finanzanalyse. Jedes Finanzinstrument besitzt Vor- und Nachteile. Die Vorteile sind zusammenfassend die Gewinne, die erzielt werden können. Nachteile der Aktien sind einerseits wegen der großen Vielfalt die Auswahlrisiken und andererseits das Kursrisiko [Gei03]. Worst-Case Szenario des Kursrisikos ist die Insolvenz des Unternehmens, an dem man Aktienanteile besitzt.

#### Aktienindizes

Ein weiterer Begriff im Zusammenhang von Aktien sind die Aktienindizes. Ein Aktienindex enthält eine bestimmte Anzahl von Aktien. Der Indexwert ist der Gesamtwert aller anteiligen Aktien im Index. Ein exzessiver prozentueller Anstieg eines Aktienindex deutet auf einen durchschnittlichen, prozentuellen Anstieg der beinhalteten Aktientitel. Tabelle 2.1 zeigt bekannte Aktienindizes.

#### Anleihen

Anleihen sind verzinsliche Wertpapiere[Gei03]. Anleihen können von einem Staat, einem Bundesland oder auch einem Unternehmen herausgegeben wer-

---

<sup>6</sup>Je nach Anzahl der Interessenten kann der Kurs in einer Sekunde vielfach sich verändern.

Name	Beschreibung
DAX 30	30 größte Deutsche Unternehmen
TexDax	30 größte Deutsche Technologie Unternehmen
Euro-Stoxx 50	50 größte europäische Unternehmen
Nasdaq 100	100 umsatzstärkste US Technologie Unternehmen
Dow Jones	30 größte US Unternehmen
S&P 500	500 größte US Unternehmen
Nikkei	225 größte japanische Unternehmen

Tabelle 2.1: Bekannte Aktienindizes

den. Die Herausgeber benötigen Kapital und suchen Anleger, die ihnen das benötigte Kapital gegen Zinszahlungen verleihen. Bekannte Anleihenarten sind Staatsanleihen und Bankanleihen. Anleihen sind zwar risikoärmer verglichen mit Aktien, jedoch sind sie nicht risikofrei. Sowohl Staaten als auch Banken können in Zahlungsschwierigkeit geraten oder gar insolvent gehen (zum Beispiel die US-amerikanische Investmentbank Lehman Brothers). Es gilt, je höher das Risiko für den Anleger, desto höher sind die Zinsen (vgl. [Göt01]). Die Bonität der Anleihen bzw. der Emittenten werden von unabhängigen Ratingagenturen bewertet und veröffentlicht. In der Tabelle 2.2 sieht man die Ratingsymbole der zwei bekanntesten Ratingagenturen - Moody's und Standard & Poor's.

### Investmentfonds

Investmentfonds sind Bündel aus ausgewählten Wertpapieren. Diese Anlageform werden von Börsenexperten - den Fondsmanagern - zusammengestellt [Gei03]. Die Bündelung bringt den Vorteil, dass das Auswahlrisiko und das Kursrisiko verringert bzw. gestreut werden. Bekannte Formen von Investmentfond sind zum Beispiel Dachfonds, Aktienfonds, Immobilienfonds und Rentenfonds. Der Name sagt aus, wie und woraus dieser Fond zusammengestellt wurde. Aktienfonds sind zusammengestellt aus Aktientiteln von Unternehmen aus verschiedenen Sektoren. Immobilienfonds zum Beispiel bestehen ausschließlich aus Aktien von Bau- und Immobilienunternehmen.

### Devisen

Devisen sind Forderungen auf Fremdwährungen [Lip67]. Wenn man im Besitz von Devisen ist, so hat man Anspruch auf die Auszahlung des Wertes der Devisen.



Bonitätsbewertung	Ratingsymbole von Moody	Ratingsymbole von Standard & Poor's
Sehr gute Anleihen Beste Qualität, geringstes Ausfallrisiko	Aaa Aa1 Aa2 Aa3	AAA AA+ AA AA-
Gute Anleihen Gute Qualität, viele gute Attribute, aber auch Risiken im Falle einer veränderten Wirtschaftsentwicklung	A1 A2 A3	A+ A A-
Mittlere Qualität, aber mangelnder Schutz gegen veränderte Wirtschaftsentwicklung	Baa1 Baa2 Baa3	BBB+ BBB BB-
Spekulative Anleihe	Ba1 Ba2 Ba3	BB+ BB BB-
Sehr spekulative Anleihe	B1 B2 B3	B+ B B-
Junk Bonds, niedrigste Qualität	Caa Ca C	CCC CC C

Tabelle 2.2: Ratingsymbole der Ratingagenturen Moody und Standard & Poor's

### Andere Anlageformen

Andere, riskantere Anlageformen sind zum Beispiel Optionen und Futures. Beim Erwerb von Optionen bzw. Optionsscheinen erkaufte man sich die Rechte ein Wertpapier in einem vereinbarten Zeitraum zu einem bestimmten Preis zu erwerben [Ell96] [SB00]. Bei Optionen erwirbt man sich ein Kaufrecht, jedoch kein Kaufpflicht. Eine grundlegende Einführung zu derivativen Wertpapieren ist aus [SB00] zu entnehmen.

### 2.3.2 Traditionelle Finanzanalyse im Überblick

Das Ziel der Aktienanalyse ist es, eine effiziente Proportion<sup>7</sup> zwischen Profit und Risiko einer Anlage zu finden und eine möglichst zutreffende Prognose

<sup>7</sup>Effizient in dem Sinne, dass eine Maximierung des Profits bei einer entsprechenden Risikozahl stattfindet.

zu geben [Mül07]. Die Wertpapieranalyse lässt sich in mehrere Formen einteilen und wird in vielen Trading-Tools eingesetzt. Diese Analysemethoden werden in Folge näher erläutert.

### 2.3.2.1 Fundamentalanalyse

„Fundamental analysis tries to forecast future level of economy, industry sales and earnings, company sales and earnings, and so on.“ *Investments [SAB85]*

Die am weitesten verbreitete Methode der Wertpapieranalyse ist die Fundamentalanalyse [Sch02]. Sie geht von der Annahme aus, dass der Kurs eines Wertpapiers um seinen „inneren Wert“ schwankt. Der innere Wert wird bestimmt durch Beobachtung und Beurteilung wirtschaftlicher Entwicklungen und Tendenzen. Diese Beobachtungen und Beurteilungen werden in interne und externe Einflußgrößen eingeteilt. Interne Einflußgrößen sind zum Beispiel Intention des Managements, Innovationskraft, technisches Know-how und längerfristige Gewinnprognosen. Externe Einflußgrößen sind zum Beispiel Wirtschaftswachstum, Zinsentwicklung, Rohstoffpreise und Export/Importentwicklung. Das Ziel der Fundamentalanalyse ist es den inneren Wert zu ermitteln [SB00]. Gelingt es, den inneren Wert eines Wertpapiers zu ermitteln, wird dieser mit dem tatsächlichen Börsenkurs verglichen. Der Vergleich resultiert in zwei möglichen Handlungsempfehlungen:

1. Der innere Wert ist höher als der Börsenkurs = das Papier ist unterbewertet. Also lautet die Empfehlung „Kaufen“.
2. Der innere Wert ist niedriger als der Börsenkurs = das Papier ist überbewertet. Also lautet die Empfehlung „Verkaufen“.

### 2.3.2.2 Technische Analyse

Die bedeutendsten Teilgebiete der Technischen Analyse sind unter anderem Chartanalyse und Indikatorenanalyse [UN01]. Die Chartanalyse beschäftigt sich mit dem graphischen Kursverlauf eines Wertpapiers. Die Indikatorenanalyse generiert die Handelsentscheidungen nicht aus dem Kursverlauf, sondern direkt aus den Kursdaten der Vergangenheit. Beispiele verschiedener technischer Analyse Methoden werden hier erläutert:

**Gleitender Durchschnitt (GD):** Beim gleitenden Durchschnitt wird ein Durchschnitt aus historischen Kurswerten berechnet. Dabei variiert die Anzahl der Tage, die für die Bewertung herangezogen wird. Eine

Beispiel Empfehlung: „Anlage kaufen, wenn der Kurs die N-Tage-Linie GD(X,N) von unten nach oben schneidet.“

„Anlage verkaufen, wenn der Kurs die N-Tage-Linie von oben nach unten durchbricht.“

Nachteile des GD sind die Trägheit der Methode und zu spätes Einsetzen der Handlungsempfehlungen (vgl. [Sch08]). Abbildung 2.5 zeigt den GD Verlauf der SAP Aktien (Orange Linie) im Vergleich zum Kursverlauf der Aktien vom April 2008 bis April 2009<sup>8</sup>.



Abbildung 2.5: GD Verlauf in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien

**Momentum:** Das Momentum gibt Auskunft in welcher Phase sich der Markt befindet. Die Berechnung des Momentums basiert auf dem in Verhältnis setzen von aktuellem Kurs und von Kursen der vergangenen Tage. Die Anzahl der vergangenen Tage kann variieren. Ist das Verhältnis grösser 1, befindet sich der Markt im Aufwärtstrend. Ist das Verhältnis kleiner 1, ist der Markt in einem Abwärtstrend. Ein Kaufsignal wird dann generiert, wenn diese Marke von unten nach oben durchstossen wird und umgekehrt ein Verkaufsignal generiert, wenn dieser Marke von oben nach unten durchbrochen wird [Sch08]. Die Abbildung 2.6 zeigt die Darstellung des Momentum Verlaufs in Gegenüberstellung zum Kursverlauf der SAP Aktien vom April 2008

<sup>8</sup>Quelle: generiert durch das Comdirect.de Online Broker Portal

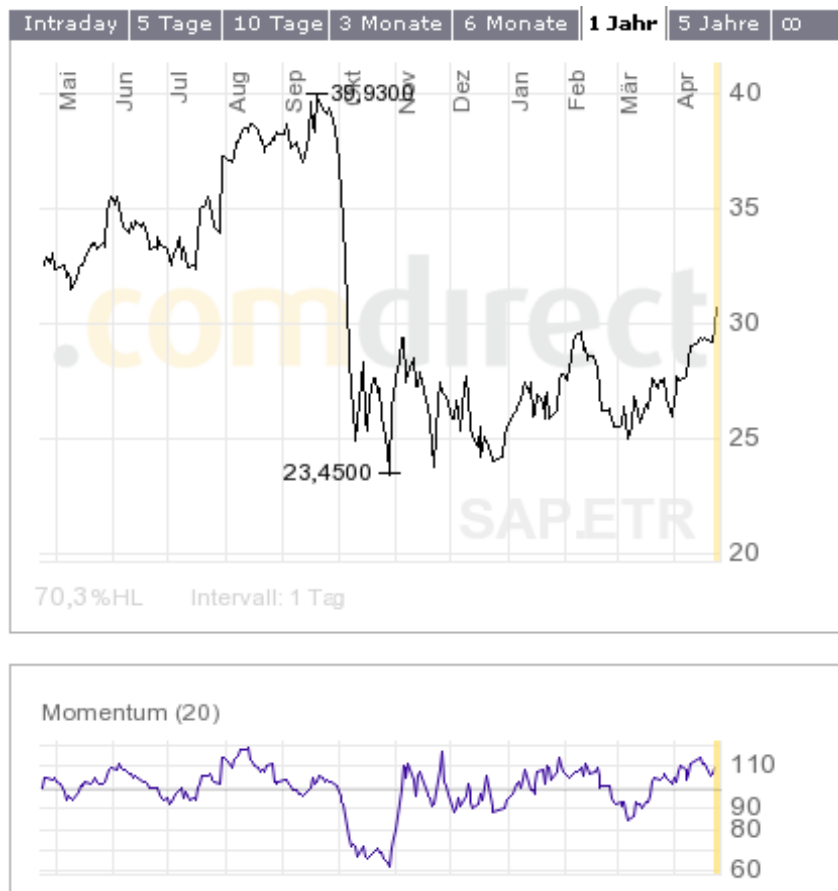


Abbildung 2.6: Momentum Verlauf in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien

bis April 2009<sup>9</sup>.

**Trendoszillator (TO):** Der Trendoszillator funktioniert ähnlich wie das Momentum mit einem Unterschied: bei dieser Methode werden nicht einzelne Kurse miteinander verglichen, sondern mehrere gleitende Durchschnitte. Dadurch verschafft sich diese Methode den Vorteil, dass sie gegen kleine Marktausreisser resistent ist. Der Wert des TOs ist das Verhältnis zwischen kurzfristigen und langfristigen GDs. Bei einem Wert kleiner 1 liegt eine technische Schwäche des Kurses vor. Umgekehrt sind bei einem Wert grösser 1 weitere Kursanstiege zu vermuten. Ein Kauf-/Verkaufsignal wird generiert, wenn die 1er Linie durchstossen wird.

**Relative Stärke (RS):** Die Relative Stärke gibt die relative Performanz eines Wertpapiers zu einem vergleichbaren Wertpapier an. Diese Methodik wird am häufigsten in einem Performanzvergleich verschiedener Aktien mit einem Aktienindex eingesetzt [MN98]. Wenn eine Aktie eine hohe relative Stärke zum Vergleichsindex aufweist, bedeutet dies, dass die Aktie größere Kursgewinne und geringere Kursverluste verzeichnet als der Vergleichsindex.

**Relative Stärke Index (RSI):** Der relative Stärke Index bringt die Anzahl der Kursgewinne und Kursverluste in einer bestimmten Zeitspanne miteinander in Beziehung. RSI zeigt die innere Stärke einer Kursbewegung an. Bei einem kleinen RSI ist das Wertpapier überverkauft und somit ist ein Kursanstieg zu erwarten. Umgekehrt sollte bei einem großen RSI das Wertpapier verkauft werden, weil mit einem Kursrückgang zu rechnen ist [Sch08]. Die Abbildung 2.7 zeigt den RSI Indikator in Gegenüberstellung zum Kursverlauf der SAP Aktien<sup>10</sup>.

**Moving Average Convergence/Divergence Trading Methode (MACD):**

Der Moving Average Convergence Divergence Indikator ist einer der zuverlässigsten technischen Indikatoren [Gey08]. Der MACD Indikator berechnet sich aus der Differenz von zwei oder mehr gleitenden Durchschnitten. Der MACD Indikator wird in dem Chart als MACD Linie abgebildet. Zusätzlich zur MACD Linie existiert noch eine Signallinie. Die Signallinie wird berechnet aus dem gleitenden Durchschnitt der MACD Linien [Sch08]. Durchstößt die MACD Linie die Signallinie von unten nach oben, wird ein Kaufsignal generiert. Umgekehrt wird

---

<sup>9</sup>Quelle: generiert durch das Comdirect.de Online-Broker Portal

<sup>10</sup>Quelle: generiert durch das Comdirect.de Online-Broker Portal



Abbildung 2.7: RSI Verlauf in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien

ein Verkaufssignal generiert [MN98]. Die Abbildung 2.8 zeigt die Darstellung der MACD Linie und die Signallinie in Gegenüberstellung zum Kursverlauf der SAP Aktien von April 2008 bis April 2009<sup>11</sup>.

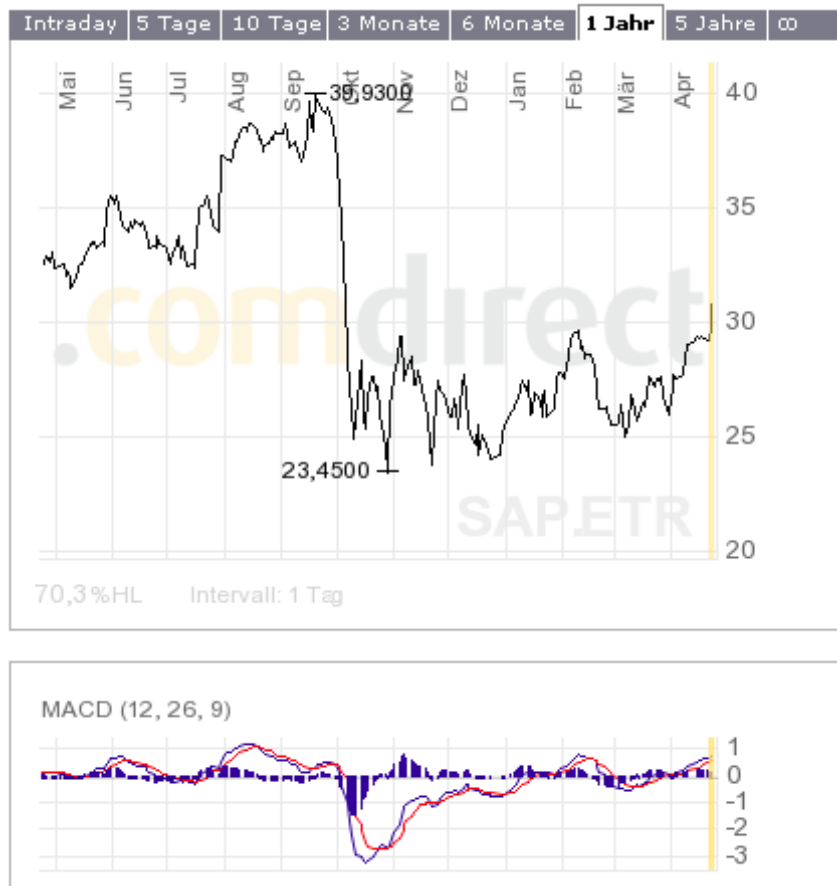


Abbildung 2.8: MACD Linie und Signallinie in Gegenüberstellung mit dem 1-jährigen Kursverlauf der SAP Aktien

Eine andere umstrittene Form der technische Wertpapieranalyse ist die Formationsanalyse. Bei dieser Analyse Methodik wird versucht, verborgene Kursmuster zu identifizieren. Diese Methode basiert auf der Annahme, dass sich Wertpapierkurse wiederholen [Rei96]. Grundsätzlich unterscheidet man bei der Formationsanalyse zwischen der Trendbestätigungs- und Trendumkehrformation. Bei der Trendbestätigungsformation entwickeln sich Kurse für eine kurze Zeit seitlich oder leicht entgegen der Trendrichtung [Jam00]. Eine Trendumkehrformation sagt einen Bruch des bestehenden

<sup>11</sup>Quelle: generiert durch das Comdirect.de Online Broker Portal

Trends voraus. Die Kurse entwickeln sich in die entgegengesetzte Richtung des bestehenden Trends. Die Formationsanalyse ist deswegen umstritten, da die Analyse auf subjektiven Interpretationen basiert.

### **2.3.2.3 Random-Walk-Hypothese**

Die Random-Walk-Hypothese stellt die Sinnhaftigkeit der Fundamentalanalyse und der technischen Analyse infrage. Sie geht davon aus, dass der Aktienkursverlauf einen Zufallspfad folgt und nicht prognostizierbar ist. Die Hypothese geht auch von der Annahme aus, dass im Aktienkurs schon alle relevanten Informationen enthalten sind. Aus diesem Grund haben ausschließlich zukünftige Informationen Einfluss auf den Verlauf der Kurse (vgl. [Mül07]).

### **2.3.3 Elektronischer Handel**

Das bisher bekannte Präsenzgeschäft<sup>12</sup> wird immer mehr durch den elektronischen Handel verdrängt. Als Privatanleger kann man heute das Handeln mit Wertpapieren online von jedem Computer aus durchführen. Online-Broker ermöglichen den Handel mit Wertpapieren über das Internet [Rim03]. Durch den Internetboom werden immer mehr Finanzgeschäfte, insbesondere Wertpapiergeschäfte, online abgewickelt [Wei01]. Vorteile von Online-Brokern sind die schnelle und kostengünstige Abwicklung von Kauf/Verkauf Ordnern für Wertpapiere (vgl. [BB02] und [BK09]). Durch das Fehlen eines flächendeckenden Filialnetzes, verzichten die Online-Broker Kunden jedoch auf eine individuelle, persönliche Beratung und Betreuung.

### **2.3.4 Der Kostenfaktor**

Der Handel mit Wertpapieren ist kostenpflichtig. Für den Kunden eines Online-Brokers setzen sich die Kosten zusammen aus: Depotgebühren plus Transaktionskosten. Depotgebühren sind eine fixe Grösse, die der Kunde quartalsweise/halbjährig/jährlich entrichtet. Die Kosten für die Abwicklung von Kauf/Verkauf Ordnern nennt man Transaktionskosten oder Ordergebühren. Es wird im Normalfall pro Transaktion, also pro aufgegebenem Order verrechnet. Für Transaktionen unter der Mindesthöhe wird ein fixer Betrag an Transaktionskosten verrechnet. Über der Mindesthöhe wird sie prozentual verrechnet. Sprich, je höher das Ordervolumen, desto mehr Gebühr muss der Kunde entrichten. Das Online Portal [www.broker-test.de](http://www.broker-test.de)

---

<sup>12</sup>Definition von Präsenzgeschäft nach Bauert [BK09]: Handel von Finanzmittel auf dem Börsenparkett



bietet einen Transaktionsrechner [BT09] an, der je nach Ordervolumen die Transaktionskosten von zwanzig unterschiedlichen Brokern über Xetra<sup>13</sup> anzeigt.

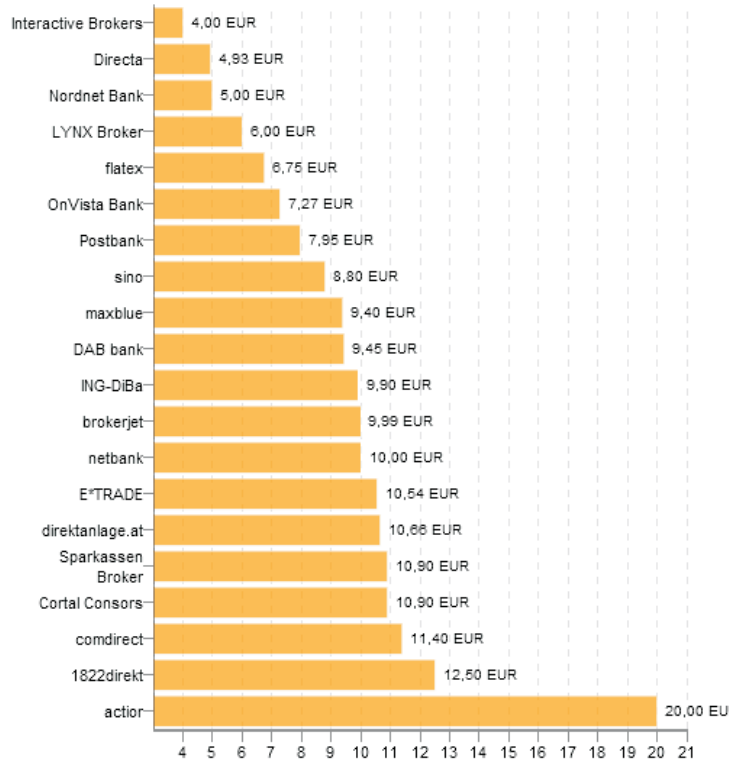


Abbildung 2.9: Transaktionskosten unter dem Mindest-Ordervolumen

Abbildung 2.9 zeigt die Transaktionskosten unter dem Mindest-Ordervolumen. Die Transaktionskosten variieren in dem Fall zwischen 4 Euro bis 20 Euro. Abbildung 2.10 zeigt die Transaktionskosten für eine Order von 5000 und 50000 Euro an, hier variieren Transaktionskosten sehr stark. Der Broker Directa verlangt für eine Transaktion mit einem Ordervolumen von 50000 Euro nur 7.57 Euro, wogegen der Broker 1822direct schon knapp bei 100 Euro ist. In allen Grafiken wurde nach dem Broker mit den günstigsten Transaktionskosten für das jeweilige Ordervolumen sortiert. Im direkten Vergleich sieht man, dass die Transaktionskosten von Broker zu Broker sehr unterschiedlich ausfallen können.

<sup>13</sup>Xetra steht für die vollelektronische pan-europäische Handelsplattform der Deutschen Börse AG.

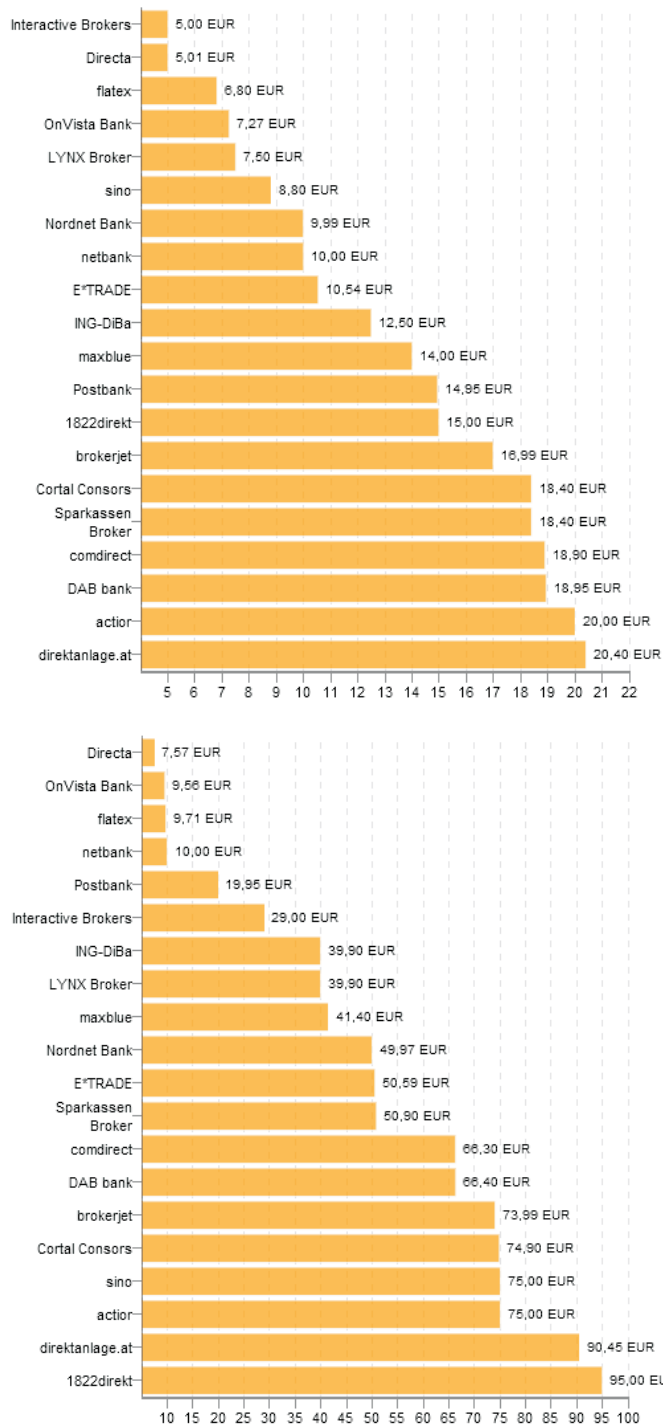


Abbildung 2.10: Transaktionskosten bei einer Ordervolumen von 5000 und 50000 Euro

### 2.3.5 Kundenprofil

Pischulti beschreibt in [Pis97] die Online-Brokerage Kunden mitunter so:

- Sie besitzen zeitgemäße Kommunikationsmedien wie z.B. Personal Computer, Handy und Faxgerät. Diese sind die Grundlagen für das elektronische Trading.
- Sie sind daran interessiert, zu jeder Stunde die Bankgeschäfte auf elektronischem Wege durchzuführen.
- Sie besitzen ein überdurchschnittliches Bildungsniveau.
- Sie sind allgemein karriere- und erfolgsorientiert.
- Sie sind unzufrieden mit Filialgeschäften durch die beschränkten Öffnungszeiten, Parkplatzprobleme und mögliche Wartezeiten.

Da viele Trading-Tools automatische Orderabgabe durch ihre Online-Brokerage Schnittstelle anbieten, ist es anzunehmen, dass das Kundenprofil eines Online-Brokerage Kunden dem Anwender eines Trading-Tools gleicht.

### 2.3.6 Handelssysteme

Automatische Ordergenerierung basiert auf Handelssystemen. Harris definiert ein Handelssystem als eine Menge von Kauf/Verkauf Regeln für den Handel mit Wertpapieren [Har03]. Handelssysteme erzeugen bei Eintritt der definierten Regeln Kauf/Verkauf Signale. Man unterscheidet zwischen manuellen und mechanischen Handelssystemen.

Manuelle Handelssysteme bestehen aus einem einfachen Konstrukt und können vom Trader selbst (per Hand) ausgeführt werden. Mechanische Handelssysteme sind in der Regel sehr komplex und für einen Menschen schwer durchschaubar. Aus diesem Grund werden sie im Normalfall von einem Computer ausgeführt. Trading-Tools mit einer Online-Broker Anbindung wie z.B. Investox [Knö09] generieren aus den Handelssystemen Kauf/Verkauf Signale, die dann vom Tool automatisiert als Kauf/Verkauf Order an den Broker geleitet werden.

Vor dem tatsächlichen Einsatz können die Handelssysteme mit historischen Daten auf Robustheit, Fehlerfreiheit und Performanz getestet werden. Dieser Schritt wird auch Backtesting genannt. Backtesting mit realen und exakten Bedingungen nennt man „Paper Trading“. Paper Trading simuliert Käufe und Verkäufe von Wertpapieren in einem fiktiven Depot mit Spielgeld. Durch Kursschwankungen und Marktveränderungen ändern sich die Aktualität bzw. die Qualität eines Handelssystems. Daher müssen sie ständig

optimiert und an die Marktsituation angepasst werden. Hier können genetische Algorithmen eingesetzt werden.

## 2.4 Einsatz der genetischen Algorithmen in der Finanz

### 2.4.1 Allgemeine Anwendungsgebiete

Evolutionäre Algorithmen, im Speziellen die genetischen Algorithmen, wurden auf viele Probleme verschiedenster Fachbereiche angewendet (vgl. [AK99, p. 249], [Szp02]).

Typische Anwendungen in der Betriebswirtschaft sind: Optimierung von Geschäftsprozessen, Optimierung der Produktionsplanung (Ressourcenallokations-Probleme), Optimierung von Zulieferungsplanung (Travelling Salesman Problem) und vieles mehr (siehe [PF07], [JTM07], [KSSG96], [LS05], [LL95]).

Beispiele der Einsatzmöglichkeiten in der Ökonomie sind in [Ari95], [Ari96], [Szp97], [Art91] zu finden. Auch in der Finanzwissenschaft haben viele Forscher genetische Algorithmen in ihre Problemfelder eingebettet und haben die Profitabilität des Einsatzes bewiesen (siehe [LCWZ04], [Vac97], [FGLG06], [Col00], [NWD97], [RSC07], [Kab00], [KCM05], [KSS06], [BLL92], [PSV04], [FRGMSR01] [PB00], [AK99], [Szp02], [Sha92], [BS03] und [KF95]).

Beispiele für mögliche und erforschte Anwendungen in der Finanz sind:

- Regelbasierte Insolvenzvorhersage: genetische Algorithmen werden hier eingesetzt um Regeln herzuleiten. Als Input werden Bilanz-Aufstellungen und andere Finanzgrößen, die erste Anzeichen einer möglichen Insolvenz indizierten, herangezogen. Diese Methodik liefert einfache Regeln, die bessere Ergebnisse vorwiesen als traditionelle „multiple Diskriminanzanalyse“ [KF95]. Die Verbesserungen der Prognosequalität durch den Einsatz der genetischen Algorithmen lagen bei über 10 Prozent.
- Beurteilung der Kreditwürdigkeit: Genetische Algorithmen wurden auch eingesetzt zur Beurteilung der Kreditwürdigkeit. Die erzielten Ergebnisse überzeugen mehr als andere Methoden (wie z.B. Neuronale Netze) durch ihre Durchsichtigkeit [Dav94].

Weitere Anwendungen sind:

- Data Mining
- Trainieren von Neuronalen Netzwerken (als Parameter Generator)

- Finanz-Kalkulationen
- Versicherungsapplikationen
- Investment Portfolio-Selektion und Management
- Generierung und Optimierung von Kauf/Verkauf Regeln für Aktien- und Devisenhandel
- Risikobewertung und Risikomanagement von Optionen

#### 2.4.2 Einsatz der GA in Trading-Tools

Folgende Trading-Tools mit GA-Funktionen wurden durch eine Internet Recherche ermittelt:

- Investox [Knö09]
- TradingSolutions [Neu09]
- Merchant of Venice [LM09]
- MultiCharts [Sup09]
- NeuroShell [Gro09]
- Tradecision [Res09]

Pereira fasst den Einsatz der GA in Finanzmarkt Applikationen so zusammen [PB00]:

**f1:** Kauf/Verkauf Regel Generierung

**f2:** Kauf/Verkauf Regel Optimierung

**f3:** Portfolio-Optimierung

**f4:** Kursprognose

Anhand dieser Klassifikation wurden diverse GA-basierte Trading-Tools auf ihre GA Funktionalitäten untersucht. Untersuchungsergebnisse von zwei Fallbeispielen werden im Kapitel 3.2 und 4.2 präsentiert. Die Anwendungsmöglichkeiten der genetischen Algorithmen in Trading-Tools werden in Folge besprochen.

### 2.4.2.1 Kauf/Verkauf Regel Generierung

Kauf/Verkauf Regeln sind Funktionen mit historischen Kursdaten als Input, die als Output entweder ein Kauf oder Verkauf Signal generieren [AK99]. Sie können sowohl auf Methoden der Fundamentalanalyse als auch der technischen Analyse basieren.

Genetische Algorithmen können zur Generierung der Kauf/Verkauf Regeln eingesetzt werden (siehe [BLL92] [BS03] [PSV04], [AK99], [AK94], [Col00], [NWD97] und [PB00]). Die Parameter Konfiguration der Regeln ist ausschlaggebend für die Profitabilität der Regeln. Nur optimale Parameter führen zur maximalen Performanz [PB99]. Genetische Algorithmen sind effizient und effektiv als Ansatz zur Lösung solcher Optimierungsprobleme [PB00] [Gol89]. Die Effizienz dieses Ansatzes ist bezogen auf die Berechnungsgeschwindigkeit des Algorithmus. Die Effektivität der genetischen Algorithmus zeigt sich in den Ergebnissen verglichen mit bisherigen Optimierungsmethoden. Zwei Schwerpunkte dominieren in diesem Optimierungsproblem. Der Erste ist die Vielzahl der möglichen Parameterkonfigurationen. Der zweite ist die Tatsache, dass mehrere Optima existieren können mit unterschiedlicher Profitabilität. Der erste Schwerpunkt fordert den Rekombinationsoperator als Lösungsansatz. Der Rekombinationsoperator kann eine große Menge an Lösungskandidaten generieren, rekombinieren und auf Eigenschaften wie Profitabilität untersuchen. Das zweite Problem kann durch den Mutations-Operator gelöst werden. Der Mutations-Operator erzeugt zufällige Änderungen in der Lösungsmenge. Der Operator kann so die Wahrscheinlichkeit, dass der Algorithmus an einem lokalen Optimum hängt und das globale Optimum nicht findet, mindern. Diese Eigenschaft der genetischen Algorithmen erhöht die Effektivität des Ansatzes (vgl. [PB99]). Allen und Karajalainen verwenden in [AK94] genetische Programmierung um profitable Kauf/Verkauf Signale zu erzeugen. Als Testdaten wurde der S&P 500 Aktienindex<sup>14</sup> herangezogen. In [AK94] berichten Allen und Karajalainen von signifikanten Ergebnissen in der Profitabilität. Spätere Forschungen von Allen und Karajalainen ergeben jedoch, dass die Kauf/Verkauf Regel Generierung mittels GA keine signifikant beweisbare Profitabilität ergibt (vgl. [AK99]). Das liegt an den hohen Transaktionskosten, die in die Performanzbewertung mit einbezogen wurden. Ausgehend von den Forschungsergebnissen von Allen und Karajalainen, setzten Colin und Neely et al in [Col00] und [NWD97] genetische Programmierung zur Generierung von Kauf/Verkauf Regeln in Modellen des Devisenhandel ein. Auf dem Devisenmarkt sind die

---

<sup>14</sup>S&P 500 steht für Standard & Poor's und ist ein Aktienindex, der die 500 größten börsennotierten US-amerikanischen Unternehmen zusammenfasst.

Transaktionskosten wesentlich geringer. Daher stellt dieser Ausgangspunkt grössere Chancen auf mögliche Profitabilität dar. Neely et al belegten in ihrem Paper [NWD97], dass die Generierung von Kauf/Verkauf Regeln mittels GA signifikante Gewinne mit sich bringen kann.

Tatsächlicher Einsatz der Kauf/Verkauf Generierung in Trading-Tools wurde im Programm Merchant of Venice nachgewiesen und untersucht. Details dazu siehe Kapitel 3.2.

#### 2.4.2.2 Kauf/Verkauf Regel Optimierung

„A genetic algorithm is an appropriate method to select the parameter values for trading rules because of its property of robustness in the presence of multiple equilibria and non-linearity of the profit surface, and the property of efficiency in searching across very large parameter spaces.“

*Genetic Algorithm Optimisation for Finance and Investment [PB00]*

Anders als bei der Kauf/Verkauf Regel Generierung, muss bei der Optimierung keine neue profitable Regel von der Menge der möglichen Lösungskandidaten ausgesucht werden. Hier ist die Problematik allein die der optimalen Parameter Konfiguration [FRGMSR01]. Drake beleuchtete in seinem Paper [DM02] das Prinzip der GA-basierten Kauf/Verkauf Regel Optimierung anhand eines konkreten Beispiels. Der gleitende Durchschnitt ist sehr simpel und basiert auf dem Durchschnitt der letzten N Tage. Parameter N steht hier für die Anzahl der Tage, woraus der gleitende Durchschnitt berechnet wurde. Die Parameter Optimierung mittels GA wird im Falle des gleitenden Durchschnitts versuchen einen optimalen Wert für den Parameter N zu finden.

Pereira setzte die GA-basierte Parameter Optimierung im Devisenhandel ein (siehe [Per96]). Es sollte für drei unterschiedliche Kauf/Verkauf Regeln eine Parameter Optimierung durchgeführt werden. Pereira implementierte für die Regeln unterschiedliche genetische Algorithmen mit unterschiedlichen Konfigurationen. Die Unterschiede in der Implementierung stammen von den unterschiedlichen Anforderungen der Regeln. Die Gemeinsamkeit der drei Algorithmen waren die binäre Darstellung mit gleicher Fixlängen, der rangfolge-basierte Selektionsalgorithmus und die gleichen profit-maximierenden Kriterien zur Fitnessbewertung. Die Eintrittswahrscheinlichkeit des Rekombinationsoperators betrug 0,6. Die des Mutations-Operators betrug 0.001. Die Resultate der Optimierung von Pereira (siehe [Per96]) ergaben im Rahmen der eingesetzten historischen Daten profitable Gewinne. Ein Vergleich in der Studie ergab weiters, dass die Optimierung



von komplexen Regeln mittels genetischen Algorithmen um das 50-fache schneller war als mit einem erschöpfenden Raster-Suchverfahren.

GA-basierte Kauf/Verkauf Regel Optimierung wurde in den Trading-Tools „Merchant of Venice“ und „MultiCharts“ eingesetzt. Details dazu siehe Kapitel 3.2 und 4.2

### 2.4.2.3 Portfolio-Optimierung

Ein Portfolio ist die Aufteilung des Vermögens in unterschiedliche Anlagemöglichkeiten. Portfolio-Optimierung (auch Portfolio-Selektion) ist die Bestimmung der optimalen Investmentstrategie. Sprich, der Anleger entscheidet wieviele Anteile welcher Wertanlagen (z.B. Aktien, Fonds usw.) er wann halten soll, um seinen Nutzen und sein erwartetes Endvermögen zu maximieren [KF02]. Das Problem der Portfolio-Optimierung ist ein dynamisches Optimierungsproblem. Technisch gesehen besteht das Problem darin, den unterschiedlichen Titeln im Portfolio risiken-minimierende Gewichtungen zu vergeben. Praktische, GA-basierte Umsetzungen, legte Loraschi et Al in [LTT<sup>+</sup>95] und [LT96] vor.

Das Trading-Tool MultiCharts bietet seinen Benutzern die Funktionalität der GA-basierten Portfolio-Optimierung an. Details dazu siehe 4.2.

### 2.4.2.4 Kursprognose

Es existieren viele Ansätze zur Kursprognose für Wertpapiere und andere handelbare Güter. In der Statistik ist das Problem bekannt unter dem Namen „Zeitreihen-Prognose“. Das Ziel ist es, mittels statistischen Berechnungen die Werte einer Zeitreihe  $n$  Zeiteinheiten später zu bestimmen. Forscher wie Schlitter kombinierten die statistische Berechnungen der Zeitreihen mit Technologien der künstlichen Intelligenz zur Lösung des Problems [Sch08]. Die technische Analyse hingegen berechnet durch Informationen in den historischen Kursdaten die möglichen Gewinne. Insbesondere wird in den historischen Daten nach wiederkehrenden Mustern gesucht um eine Wiederkehr derselben Zustände vorrauszusagen [DHLN99].

Genetische Algorithmen können zur Prognose von komplexen Daten angewendet werden [Pac90] [KCM05] [Kab00]. Insbesondere im Bereich des Aktien- und Devisenhandels wurden sie zur Prognose von Zeitreihen und Erträgen eingesetzt. Dunis et al setzten in [DHLN99] genetische Algorithmen zur Ermittlung von technischen Indikatoren ein. Diese sollen verborgene Muster und Beziehungen in Zeitreihen ermitteln, welche der Profitmaximierung dienen.

Mahfoud, Mani und Reigel präsentierten in [MM96] den Einsatz der GA zur Ermittlung des relativen Gewinns einzelner Wertpapiertitel. Sie haben bewiesen, dass genetische Algorithmen erhebliche Prognosepotentiale aufweisen.

Sexton et Al untersuchten den Einsatz der GA als Optimierungsmethode für künstlichen neuronalen Netzen (KNN). Sie führten einen Performancevergleich zwischen dem GA-basierten Ansatz und der traditionellen Backpropagationsmethodik durch [SDJ98]. Der Vergleich ergab dass genetische Algorithmen eine durchaus robuste Lösung darstellen. Schlitter präsentierte in [Sch08] eine unterstützende Funktion der GA in Kombination mit künstlichen neuronalen Netzen. Schlitter setzte genetische Algorithmen zur Generierung von Parametern für die Berechnung von statistischen Zeitreihen mittels KNN ein. Im Trading-Tool TradingSolutions [Neu09] ist diese Einsatzmöglichkeit eingebettet worden.

## Kapitel 3

# Fallbeispiel Merchant of Venice

## 3.1 Allgemeine Produkt Beschreibung

### 3.1.1 Kurzbeschreibung

„Merchant of Venice“ ist ein Trading-Tool mit umfassenden Funktionen wie Portfoliomanagement, Graphenfunktionen, Bereitstellung von technischen Analysen, Paper-Trading und genetischen Algorithmen bzw. genetischer Programmierung zur Generierung und Optimierung von Kauf/Verkauf Regeln. Merchant of Venice wurde im Jahr 2001 von Andrew Leppard entwickelt. Ursprünglich war das Projekt als reine Implementierung von genetischen Algorithmen und genetische Programmierung angedacht. Jedoch entwickelte sich das Programm weiter, es kamen mehr und mehr Funktionen hinzu.

Eine Besonderheit des Programms ist die Portabilität. Die Mehrheit der Trading-Tools sind für reine Windows Benutzer gedacht. Nur sehr wenige unterstützen andere Betriebssysteme. Merchant of Venice läuft sowohl auf UNIX Derivaten (mitunter Mac OS X), als auch auf Windows.

### 3.1.2 Software Packages und Programmaufbau

Das aktuelle Release von Merchant of Venice ist 0.71beta und wurde im Februar 2007 herausgegeben. Das Programm braucht zur Ausführung eine JAVA Runtime Umgebung (minimale Anforderung ist J2SE version 1.4). Kursdaten und mathematische Gleichungen werden in einer SQL Datenbank gespeichert. In der Installation ist die Anbindung an die interne Datenbank HSQLDB schon integriert. Wenn man jedoch andere Datenbanken bevorzugt, gibt es die Möglichkeit MySQL oder PostgreSQL anzubinden. Für die Anbindung muss man jedoch entsprechende Datenbanken und die zugehörigen Treiber eigenständig installieren. Weitere im Programm integrierte Technologien sind:

- HSQLDB V1.8.0.4<sup>15</sup>: HSQL ist eine in JAVA programmierte relationale SQL Datenbank und steht unter einer Open-Source-Lizenz [ST04]. Die Standard-Version der HSQLDB ist nur 600 Kilobytes groß. Aufgrund der Kompaktheit und der Flexibilität, wurde HSQLDB in vielen Applikationen (wie z.B. OpenOffice.org<sup>16</sup> [Mah08] und Mathematica<sup>17</sup> [WJ01]) eingebunden.
- MySQL Connector/J V3.1.12<sup>18</sup>: MySQL Connector/J ist ein in Ja-

---

<sup>15</sup><http://hsqldb.sourceforge.net>

<sup>16</sup><http://www.openoffice.org>

<sup>17</sup><http://www.wolfram.com/products/mathematica/index.html>

<sup>18</sup><http://www.mysql.com>

va programmierter JDBC-Treiber. MySQL Connector/J ermöglicht das Verbinden von Java Client-Applikationen zur MySQL Datenbank [MyS06].

- Jython V2.1<sup>19</sup>: Jython ist eine reine Java-Implementierung der Skriptsprache Python. Jython ermöglicht die Ausführung von Python Programmen auf Java Plattformen [PR02] [Bil02].
- Joone V1.2.1<sup>20</sup>: Joone ist ein Java basiertes Framework für Neuronale Netzwerke. Joone ermöglicht das Entwerfen, das Trainieren und das Testen von künstlichen Neuronalen Netzwerken [Mar07].
- XStream V1.1.2<sup>21</sup>: XStream ist eine Bibliothek zum Serialisieren und De-Serialisieren von XML Objekten [XSt09] [WCL04].

### 3.1.3 Grundfunktionalitäten

Das aktuelle Release 0.71beta enthält folgende Funktionen:

#### **Kursdaten Import/Export und Management**

Gängige Kursdaten Formate wie EzyChart, InsightTrader oder MetaStock können importiert werden. Merchant of Venice bietet eine direkte Anbindung an das Yahoo Finance<sup>22</sup> Portal zum importieren von End-Of-Day Kursdaten. In der Abbildung 3.1 sieht man Screenshots der online Anbindung an Yahoo Finance zum importieren von End-Of-Day Kursdaten. Die Abbildung 3.2 zeigt die Kursdaten Import Funktion aus Dateien. Auch Intra-Day Kursdaten können direkt durch das Portal importiert und in Sync gehalten werden (siehe Abbildung 3.3). Kursdaten können wahlweise in der integrierten HS-QLDB oder auch in einer installierten MySQL- oder PostgreSQL Datenbank gespeichert und darauf zugegriffen werden.

Das Exportieren von Kursdaten ist recht eingeschränkt. Unterstützte Formate sind EzyChart, InsightTrader oder MetaStock. Jedoch können jedesmal nur Daten von einem einzigen Tag exportiert werden. Abbildung 3.4 zeigt einen Screenshot des Kursdaten Export Fensters.

#### **Graphen und Tabellendarstellung von Kursdaten**

Folgende Diagramm-Darstellungsformen werden unterstützt:

---

<sup>19</sup><http://www.jython.org>

<sup>20</sup><http://www.jooneworld.com>

<sup>21</sup><http://xstream.codehaus.org>

<sup>22</sup><http://finance.yahoo.com/>

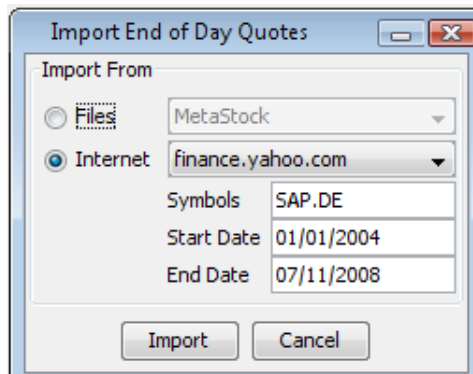


Abbildung 3.1: Screenshot des Import von End-Of-Day Kursdaten über die online Yahoo Finance Anbindung

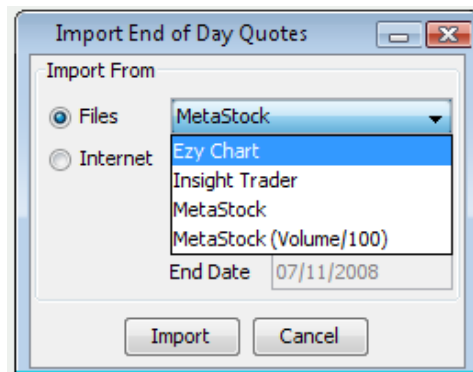


Abbildung 3.2: Screenshot des Datei Import von Kursdaten

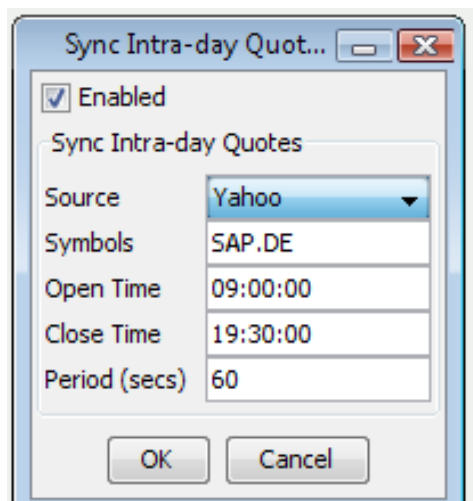


Abbildung 3.3: Screenshot der Intra-Day Kursdaten Import Funktion

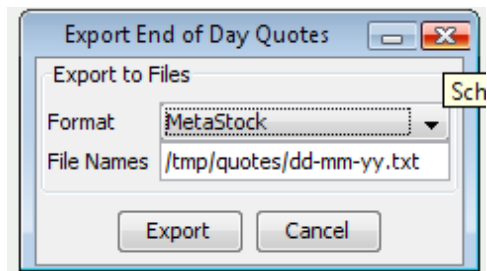


Abbildung 3.4: Screenshot der Kursdaten Export Funktion

- Liniendiagramm
- Balkendiagramm
- Kerzen-Graph
- Hoch-Tief Balkendiagramm
- Punkt-und-Zeichen-Graph

Folgende Finanzgrößen können in Diagrammform dargestellt werden:

- Bollinger Bands
- Exponentieller gleitender Durchschnitt (EGD)
- Multiple gleitende Durchschnitte
- Moving Average Convergence/Divergence (MACD)
- Momentum
- Gleitender Durchschnitt (GD)
- On Balance Volume (OBV)
- Relative Strength Index (RSI)
- Standardabweichung
- Tageshoch/Tagestief
- Eröffnung-/Schlusskurse
- Handelsvolumen
- Support-Resistance Profil

Abbildung 3.5 zeigt einen Screenshot mit allen unterstützten Diagramm Arten und Finanzgrößen der Graphenfunktion. Außerdem kann der Benutzer weitere Kenngrößen mittels der zur Verfügung gestellten Programmiersprache Gondola eigenständig implementieren und in einem Diagramm darstellen lassen. Die Tabellendarstellung unterstützt das Auflisten aller Kursdaten.

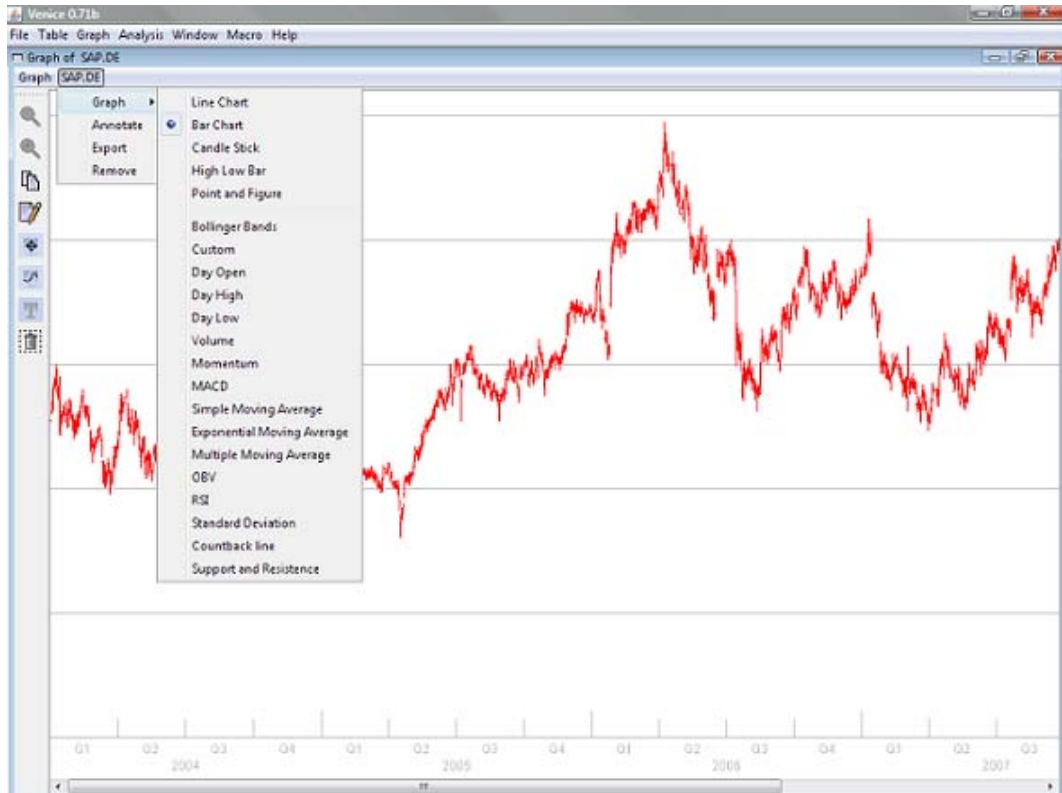


Abbildung 3.5: Screenshot der Graphenfunktionalität

Ein Beispiel einer tabellarischen Darstellung der SAP.DE Aktie sieht man in der Abbildung 3.6. Aus den tabellarisch dargestellten Daten können spezielle Informationen wie z.B. Verkaufsvolumen, Veränderungen, Tageshoch bzw. Tagestief gefiltert und sortiert werden.



Table Symbols						
Date	Volume	Day Low	Day High	Day Open	Day Close	Change
6/10/2008	24160500	27,860	33,720	33,000	28,840	-16,41% ▼
7/10/2008	31874500	24,700	29,190	28,100	26,740	-7,28% ▼
8/10/2008	24226100	25,940	28,980	26,000	26,970	+0,86% ▲
9/10/2008	13755400	25,660	27,560	27,240	26,420	-2,04% ▼
10/10/2008	21305200	23,300	25,500	24,200	24,970	-5,49% ▼
13/10/2008	15598600	25,810	27,480	27,000	27,480	+10,05% ▲
14/10/2008	13693600	27,650	29,740	27,950	28,370	+3,24% ▲
15/10/2008	13065200	25,330	28,400	28,400	25,830	-8,95% ▼
16/10/2008	14078300	23,610	26,390	24,000	25,420	-1,59% ▼
17/10/2008	13781000	26,330	27,810	26,330	26,750	+5,23% ▲
20/10/2008	7840200	27,330	28,090	27,530	27,650	+3,36% ▲
21/10/2008	8189700	26,250	28,060	27,840	27,660	+0,04% ▲
22/10/2008	7501300	26,330	27,180	26,600	27,020	-2,31% ▼
23/10/2008	7566000	26,190	27,850	27,000	27,250	+0,85% ▲

Abbildung 3.6: Screenshot der Tabellen Darstellung der SAP Aktie

## Portfoliomanagement

Die Portfoliomanagementfunktion unterstützt das Anlegen und Verwalten von mehreren Portfolios. In jedem Portfolio können 1 bis n Wertpapierdepots oder Geld-Konten mit unterschiedlicher Höhe an Geldsummen oder Anzahl an Wertpapieren angelegt werden. In der Abbildung 3.7 sieht man das Portfoliomanagement Fenster mit einem angelegten Geld-Konto „Show Cash Account“ und einem Wertpapier Depot „Show Depot“. Das Wertpapier Depot enthält zur Zeit 1,045 Anteile des Aktien Titels „SAP.DE“. In jedem



The screenshot shows a window titled 'Show Portfolio' with tabs for 'Portfolio', 'Account', and 'Transaction'. The main content is divided into two sections:

**Show Depot (EUR)**

Symbol	Shares	Mkt Price	Mkt Value	Change
SAP.DE	1045	31,110	32.509,95 €	+2,17% ▲

**Summary (EUR)**

Account	Mkt Value	Change
Show Cash Account	50.344,10 €	+0,69% ▲
Show Depot	32.509,95 €	+6,77% ▲
Total	82.854,05 €	+2,99% ▲

Abbildung 3.7: Screenshot der Portfolio Ansicht

der angelegten Geld-Konten und Wertpapierdepots ist das Durchführen von Transaktionen möglich. Folgende Arten von Transaktionen sind unterstützt:

- Kauforder und Verkauforder
- Deposit Transaktion
- Dividenden Transaktion
- Dividenden Re-investment Transaktion
- Transaktion zur Begleichung von Spesen und Gebühren
- Zinszahlungen
- Geldtransfer zwischen den Konten
- Transaktion zur Abhebung von Geldsummen

Die Abbildung 3.8 zeigt einen Screenshot der Funktion „New Transaction“. Alle durchgeführten Transaktionen werden in der Transaktionshistorie gespeichert. Ein Beispiel einer Transaktionshistorie sieht man in der Abbildung 3.9. Portfolios und alle dazugehörigen Informationen wie der Umsatz/-Verlust einzelner Depots, der Wert einzelner Geld-Konten usw. können in Diagramm- oder in Tabellenform dargestellt werden.

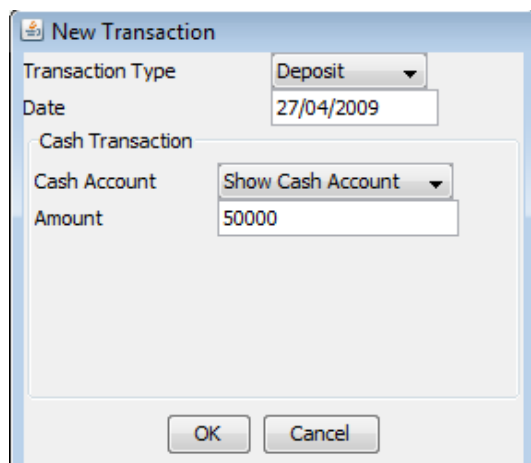


Abbildung 3.8: Screenshot der Funktionalität „New Transaction“

Date	Transaction	Credit	Debit
1/1/2009	Deposit	50.000,00 €	€ 0,00
2/1/2009	Accumulate 1500 SAP....	€ 0,00	0,00 €
27/3/2009	Reduce 500 SAP.DE @ ...	€ 0,00	0,00 €
27/4/2009	Accumulate 100 SAP.D...	€ 0,00	0,00 €
27/4/2009	Dividend SAP.DE	350,00 €	€ 0,00
27/4/2009	Fee	€ 0,00	5,90 €
27/4/2009	Reduce 55 SAP.DE @ 0...	€ 0,00	0,00 €

Abbildung 3.9: Screenshot der Transaktionshistorie

## Paper-Trading

Merchant of Venice ermöglicht weiters die Real-Simulation durch Paper-Trading. Abbildung 3.10 zeigt die beiden Reiter „Range“ und „Rules“ der Paper Trade Funktionalität. Sowohl einzelne Kauf/Verkauf Regeln als auch ein spezieller Mix von Regeln können automatisiert und manuell getestet werden.

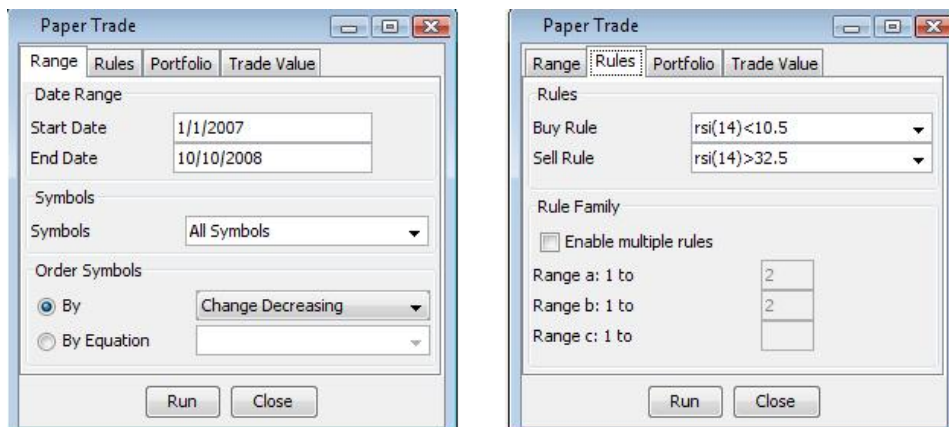
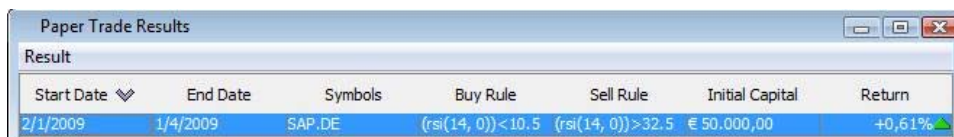


Abbildung 3.10: Screenshot vom Reiter „Range“ und „Rules“ der Paper Trade Funktionalität

Abbildung 3.11 zeigt die Ergebnisse einer Simulation anhand von SAP.DE im Zeitraum 02.01.2009 bis 01.04.2009. Als Kauf Regel wurde  $RSI(14) < 10.5$  und als Verkauf Regel wurde  $RSI(14) > 32.5$  herangezogen. Aus der Abbildung kann man einen Gewinnzuwachs von +0.61 Prozent ablesen. Hier muss angemerkt werden, dass nach der Hinzunahme von Transaktionskosten der Gewinn einen negativen Wert annehmen wird. Aus diesem Grund wäre von einer Investition mit dieser Risikoklasse abzuraten. Eine risikolose Veranlagung wie z.B. ein Sparbuch wäre sinnvoller und ertragreicher.

Veranlagung wie z.B. ein Sparbuch.



Start Date	End Date	Symbols	Buy Rule	Sell Rule	Initial Capital	Return
2/1/2009	1/4/2009	SAP.DE	(rsi(14, 0)) < 10.5	(rsi(14, 0)) > 32.5	€ 50.000,00	+0,61%

Abbildung 3.11: Screenshot der Ergebnisse einer Real-Simulation mit Paper Trade anhand von SAP.DE

## **Genetische Algorithmen und genetische Programmierung**

Merchant of Venice bietet genetische Algorithmen und genetische Programmierung zum Generieren und zum Optimieren von Kauf/Verkauf Regeln. Die Funktionen werden in der nachfolgenden Sektion 3.2 im Detail erläutert.

## **Gondola und Python Unterstützung**

„The Gondola Language is a language for analysing stock market data. The language allows you to do a variety of tasks from listing stocks in tables that only match your criteria to creating automated buy/sell paper trade rules. The language is closely modelled after the C programming language [...]“  
*Merchant of Venice - Manual [Lep07]*

Für Benutzer, die das System um ihre eigenen Funktionen und Regeln erweitern wollen, stellt Merchant of Venice die Gondola Programmiersprache zur Verfügung. Mit Hilfe der Gondola Sprach Unterstützung kann der Benutzer eigene Kauf/Verkauf Regeln aufstellen und diese mittels Paper Trade testen. Der Benutzer kann mit Gondola auch technische Indikatoren erstellen und die Werte als Graph anzeigen lassen. Weiters kann man mit Gondola Filterregeln implementieren und ein Diagramm oder eine Tabelle danach filtern lassen.

Seit Version 0.5 unterstützt Merchant of Venice die Python Skriptsprache. Mittels Jython kann der Benutzer automatisierte Ausführungen von Programmteilen initiieren.

## **User Interface**

Das User Interface kann in sieben Sprachen dargestellt werden: Katalan, Englisch, Französisch, Italienisch, Chinesisch, Schwedisch und Polnisch. Das User Interface ist sehr kompakt und einfach gehalten.

## **Dokumentation und Online Help**

Dokumentation ist minimal vorhanden. Merchant of Venice stellt dem Benutzer ein Manual [Lep07] zur Verfügung, das sowohl online verfügbar als auch offline im Programm integriert ist. Installationsanleitung ist direkt im Installationspaket als auch online verfügbar.

### 3.1.4 Lizenz und Kostenpunkt

Das Programm ist kostenlos erhältlich und steht unter der GNU General Public Licence (Version 2, 1991). GNU General Public Licence (GPL) (siehe [S<sup>+</sup>05]) ist eine von der Free Software Foundation herausgegebene Lizenz für die Lizenzierung freier Software. Programme unter GPL dürfen von Lizenznehmern nur dann verbreitet werden, wenn sie von diesen ebenfalls zu den Bedingungen der GPL lizenziert werden.

### 3.1.5 Systemanforderung

Dezierte Systemanforderungen für Merchant of Venice sind nicht bekannt und nicht dokumentiert. Allgemein kann man folgende Anforderung aus dem Manual und der Installationsanleitung ableiten:

- Prozessor: alle gängigen Prozessor Versionen
- Betriebssystem: Unix Derivate, MacOS X, alle gängige Windows Versionen (z.B. Windows 2003, Windows XP, Windows Vista 32bit/64bit)
- Speicherverbrauch: Minimum 3.7MB, Maximum offen
- Software: JAVA Runtime Umgebung JAVASE2 1.4

## 3.2 Einsatz der GA

Das Trading-Tool Merchant of Venice setzt genetische Algorithmen, insbesondere genetische Programmierung (GP), zur Generierung und Optimierung von Kauf/Verkauf Regeln ein. Durch die GPL Lizenzierung steht uns der Quellcode des Tools zur Verfügung. Dieser wurde anschließend analysiert. Im Quellcode wird zwischen zwei Implementierungen unterschieden: genetische Algorithmen und genetische Programmierung<sup>23</sup>. Der Einsatz der GA wird vom Einsatz der GP in den folgenden Unterkapiteln getrennt beleuchtet.

### 3.2.1 Kauf/Verkauf Regeln Optimierung mittels GA

Die Implementierung der GA in Merchant of Venice erlaubt es dem Benutzer Kauf und Verkauf Regeln zu optimieren. Die Ausführung der Optimierung kann erfolgreich sein oder auch fehlschlagen. Der Grund dafür ist das Wesen des Optimierungsproblems. Es kann eine, mehrere oder aber auch keine Lösung zu einem Problem existieren. Vorallem garantieren die GA nicht, dass immer eine Lösung gefunden werden kann, auch wenn eine existiert.

#### 3.2.1.1 Parameter Einstellung

Vor der Optimierung müssen alle Parameter vollständig und korrekt ausgefüllt werden. Die Abbildungen 3.12 und 3.13 zeigen das GA Fenster mit den Reitern „Range“, „Rules“, „Portfolio“ und „Trade Value“. In diesen Reitern muss der Benutzer die investment-spezifischen Parameter zur Ausführung der Optimierung angeben. Die Parameter sind:

- Start Date/End Date: der Zeitraum, in welchem das Backtesting durchgeführt werden soll.
- Symbols: Wertpapier, Aktienindex oder andere Anlageformen, auf welche die Optimierung angewendet werden soll.
- Buy Rule: Kauf Regel, nach welchen das betreffende Wertpapier gekauft werden soll.

---

<sup>23</sup>Genetische Programmierung wurde von John Koza im Jahr 1992 vorgestellt und ist eine Erweiterung zu genetischen Algorithmen [KR92]. GP unterscheidet sich von GA mit einem Hauptmerkmal. Die Repräsentation der Individuen ist anders als bei GA nicht in binär codierten Strings, sondern wird in Baum-Struktur abgebildet. Eine Einführung zur genetischen Programmierung ist [BKR<sup>+</sup>00] zu entnehmen.

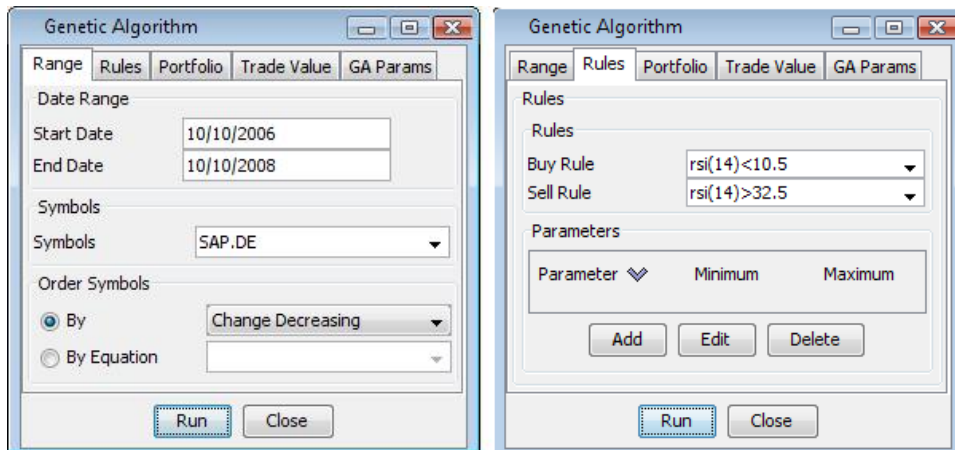


Abbildung 3.12: Screenshot der Reiter „Range“ und „Rules“ im GA Fenster

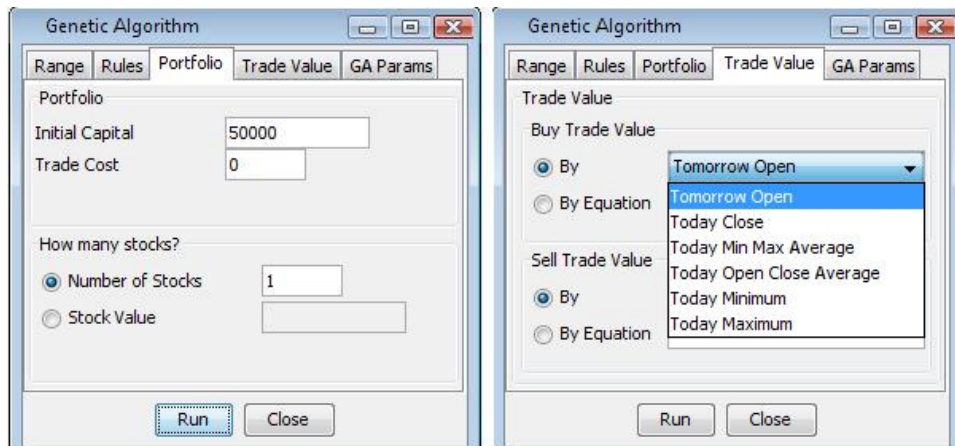


Abbildung 3.13: Screenshot der Reiter „Portfolio“ und „Trade Value“ im GA Fenster



- Sell Rule: Verkauf Regeln, nach welchen das betreffende Wertpapier verkauft werden soll.
- Parameters: die Parameter in den Kauf/Verkauf Regeln, die optimiert werden sollen.
- Initial Capital: die Initial Startsumme im Portfolio, die für den Kauf angewendet werden kann.
- Number of Stocks: Anzahl der Wertpapiere bzw. Wert der Wertpapiere, die sich im Portfolio befinden sollen.
- Buy Trade Value/Sell Trade Value: Diese Parameter bestimmen zu welchen Preisen die Wertpapiere beim Backtesting gekauft und verkauft werden sollen<sup>24</sup>. Bei vorhandenen End-of-Day Kursinformationen kann der Benutzer zwischen Schlusskurs, Eröffnungskurs, Tageshoch, Tagestief und anderen Möglichkeiten wählen.

Weiters benötigt das Programm die GA spezifischen Ausführungsparameter. Diese sind Anzahl der Generationen, Größe der Population, Größe des Mating Pools und Mutationswahrscheinlichkeit (siehe Abbildung 3.14). Der

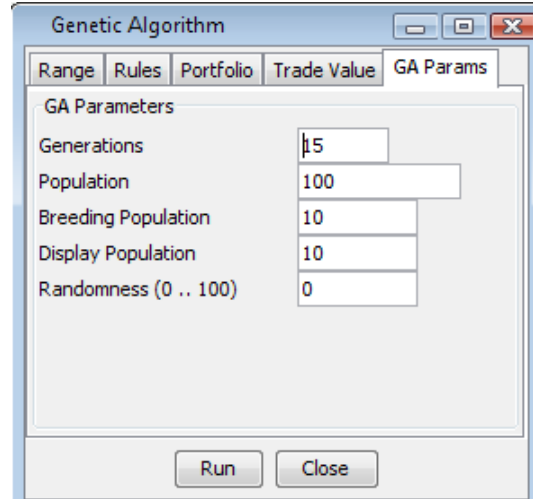


Abbildung 3.14: Screenshot der Reiter „GA Params“ im GA Fenster

Parameter „Display Population“ gibt an wieviele von den optimalsten Individuen pro Generation in der Ergebnistabelle angezeigt werden sollen. Nach

<sup>24</sup>Nehmen wir die Anlageform Aktien als Beispiel zur Veranschaulichung. Aktienkurse unterliegen im Laufe des Tages einer großen Schwankung. Ein Ankauf bzw Verkauf zu Eröffnungskurs oder zum Schlusskurs liefern somit unterschiedliche Profit Performanz.

der Betätigung des Run Buttons, startet das Programm die Ausführung der GA mit den angegebenen Parametern.

### 3.2.1.2 Verlauf der Optimierung

Die Implementierung des genetischen Algorithmus arbeitet so:

**Schritt 0 - Initialisierung der Startpopulation:** In dieser Phase wird die erste Generation an Individuen erzeugt und deren Fitnesswerte berechnet. Jedes Individuum entspricht einem Paar von Kauf/Verkauf Regeln. Da eine Kauf/Verkauf Regel schon bekannt ist, unterscheiden sich die Individuen von den Parameter Werten, die der User eingetragen hat. Der Vorgang der Initialisierung wird anhand einer beliebigen RSI Kaufregel verdeutlicht. Die Kaufregel:  $rsi(14) < 10.5 + ParamBuy$ .  $ParamBuy$  ist an dieser Stelle der einzige Parameter, der optimiert werden kann. Wenn der Intervallbereich von diesem einzigen Parameter zwischen 0 und 1 liegt, können in diesem Fall nur 2 Individuen erzeugt werden. Nämlich  $rsi(14) < 10.5 + 0$  und  $rsi(14) < 10.5 + 1$ . Daher müssen die Intervalle der zu optimierenden Parameter mit Bedacht gewählt werden.

**Schritt 1 - Selektion:** Aus der Population werden eine definierte Anzahl an Individuen selektiert, die fit genug sind. Diese Individuen erzeugen die nächste Generation von Individuen.

**Schritt 2 - Rekombination und Mutation:** Aus den ausgewählten Elternpaaren werden Kinder erzeugt (rekombiniert) und gegebenenfalls mutiert.

**Schritt 3 - Bewertung der Fitness:** Jedes der erzeugten Kinder wird einer Fitnessbewertung unterzogen. Fitnesswerte jedes Individuums werden durch Backtesting ermittelt. Als Methode zum Backtesting im Tool Merchant of Venice wurde Paper-Trading herangezogen. Für jeden einzelnen Tag im angegebenen Zeitraum wird geprüft, ob anhand der Regel ein Kauf/Verkauf Signal erzeugt wird. Wenn ein Kauf/Verkauf Signal erzeugt wurde, dann wird ein virtueller Kauf/Verkauf durchgeführt und die Wertveränderung des Portfolios gespeichert. Wenn alle Tage in dem Zeitraum abgearbeitet wurden, werden alle Wertveränderungen summiert. Dieses ergibt den Fitnesswert des Individuums, was der Performanz der Regel entspricht. Die besten Individuen werden in der Ergebnisliste angezeigt.

**Schritt 4 - Iteration:** Der Algorithmus kehrt zurück zur Schritt 1.

Die Iteration wird solange wiederholt, bis ein Abbruchkriterium erfüllt wurde. Zwei Abbruchkriterien sind im Tool implementiert:

1. Anzahl der Generationen wurde erreicht
2. Der Benutzer betätigt den Abbruch Button in der Benutzeroberfläche

### 3.2.1.3 Empirische Versuche

Die empirische Versuche werden unter folgender Testumgebung durchgeführt:

**Prozessor:** Intel Core 2 Duo P8600 2.40GHz

**RAM:** 4 GB

**Betriebssystem:** Windows Vista 32 Bit

Die Durchführung der Optimierungsfunktion wird anhand von 2 Aktien aus dem DAX beleuchtet. Für diesen Zweck wurden zwei Technologie Aktien der Firmen SAP und Siemens ausgesucht. Die historischen Kursdaten sind Auszüge aus dem Zeitraum 2003 bis 2009. Abbildungen 3.15 und 3.16 zeigen den Kursverlauf der SAP und der Siemens Aktie im Zeitraum 2003 bis 2009. Anhand dieser Grafiken sieht man bei beiden Aktien ein langsames fortlaufendes Wachstum ab 2003 bis hin zum Jahr 2008. Ab 2008 fallen bei beiden Aktien die Kurse rapide herab, wobei sich die SAP Aktie in der Jahresmitte kurzzeitig erholt.

Zur Generierung von Kauf/Verkauf Signalen wird eine auf RSI basierte Funktion benutzt. Die Kauf Regel sieht so aus:  $rsi(bx) < 22.0 + ParamBuy$ .  $bx$  und  $ParamBuy$  sind die zwei zu optimierenden Parameter in der Funktion. Diese Regel heißt soviel wie: Wenn der RSI Wert der letzten  $bx$  Zeitperioden kleiner als 22.0 plus  $ParamBuy$  ist, dann wird ein Kaufsignal generiert. Je größer die Intervalle der Parameter  $ParamBuy$  und  $bx$  sind, desto mehr Individuen muss der GA erzeugen und bewerten. Die Verkauf Regel ist so ähnlich aufgebaut wie die Kauf Regel. Sie sieht folgenderweise aus:  $rsi(sx) > 42.0 + ParamSell$ .  $sx$  und  $ParamSell$  sind auch zu optimieren. Diese Regel besagt: Wenn der RSI Wert der letzten  $sx$  Zeitperioden größer als 42.0 plus  $ParamSell$  ist, dann wird ein Verkaufsignal generiert. In der Tabelle 3.2 sind die Intervallbereiche der Parameter  $bx$ ,  $sx$ ,  $ParamBuy$  und  $ParamSell$  aufgelistet. Die Kauf/Verkauf Regeln sieht man in der Tabelle 3.1 auf einem Blick.

Für die GA-Parameter werden zwei verschiedene Settings definiert. Setting 1 besteht aus 10 Generationen und 100 Individuen pro Generation.



Abbildung 3.15: Kursentwicklung der SAP Aktie (XETRA) im Zeitraum 2003 bis 2009

Regel	Formel
Kauf	$rsi(bx) < 22.0 + ParamBuy$
Verkauf	$rsi(sx) > 42.0 + ParamSell$

Tabelle 3.1: Kauf/Verkauf Regel

Parameter	Minimum	Maximum
bx	7	14
sx	10	14
ParamBuy	0	5
ParamSell	0	20

Tabelle 3.2: Intervallbereiche der Kauf/Verkauf Regel Parameter



Abbildung 3.16: Kursentwicklung der Siemens Aktie (XETRA) im Zeitraum 2003 bis 2009

Tausend Individuen werden bei diesem Setting erzeugt<sup>25</sup>. Setting 2 besteht aus 50 Generationen und 250 Individuen pro Generation. Bei Setting 2 werden 12,500 Individuen erzeugt. Daher dauert die Berechnungszeit bei Setting 2 um ein vielfaches länger als bei Setting 1. Tabelle 3.3 und 3.4 zeigen alle GA spezifischen Parameter auf einem Blick.

Parameter	Wert
Generations	10
Population	100
Breeding Population	10
Randomness	0

Tabelle 3.3: GA spezifische Parameter - Setting 1

Parameter	Wert
Generations	50
Population	250
Breeding Population	30
Randomness	55

Tabelle 3.4: GA spezifische Parameter - Setting 2

**Versuch 1:** Versuch 1 wurde mit dem GA-Parameter Setting 1 zehn mal durchgeführt. Für den Zeitraum 01/01/2008 bis 01/01/2009 wurde mittels Paper-Trading eine Performanz<sup>26</sup> von 11.45 Prozent berechnet. Die beste Kauf/Verkauf Regel ist folgende:

**Kauf Regel:**  $(rsi(14, 0)) < (22.0+0)$

**Verkauf Regel:**  $(rsi(14, 0)) > (42.0+0)$

Bei einem Startkapital von 50,000 Euro und einer Performanz von 11.45 Prozent ergäbe es ein Endkapital von 55,726.53 Euro. Abbildung 3.17 zeigt die Ergebnistabelle der GA Optimierung. Abbildung 3.18 zeigt die Transaktionen, die im Rahmen des Paper-Tradings durchgeführt wurden zur Fitnessbewertung der Regeln.

<sup>25</sup>10 Generationen x 100 Individuen = 1,000 Individuen

<sup>26</sup>Die prozentuelle Performanz der Ergebnisse ist das Verhältnis zwischen dem Initial Kapital und dem erzielten Netto Profit.

Genetic Algorithm Results				
Result				
Symbols	Buy Rule	Sell Rule	Generation	Return
SAP.DE, SIE.DE	(rsi(14, 0)) < (22.0+0)	(rsi(14, 0)) > (42.0+0)	3	+11,45% ▲
SAP.DE, SIE.DE	(rsi(14, 0)) < (22.0+3)	(rsi(14, 0)) > (42.0+0)	3	+10,81% ▲
SAP.DE, SIE.DE	(rsi(14, 0)) < (22.0+2)	(rsi(14, 0)) > (42.0+1)	7	+9,53% ▲
SAP.DE, SIE.DE	(rsi(14, 0)) < (22.0+4)	(rsi(14, 0)) > (42.0+2)	2	+7,75% ▲
SAP.DE, SIE.DE	(rsi(14, 0)) < (22.0+3)	(rsi(11, 0)) > (42.0+3)	7	+6,94% ▲

Abbildung 3.17: Screenshot der Top 5 Ergebnisse der GA-Parameter Optimierung

Genetic Algorithm Portfolio Transactions			
Transaction			
Date	Transaction	Credit	Debit
2/1/2008	Deposit	€ 50.000,00	€ 0,00
9/1/2008	Accumulate 1515 SAP.DE @ € 33,00	€ 0,00	€ 0,00
16/1/2008	Reduce 1515 SAP.DE @ € 32,55	€ 0,00	€ 0,00
24/1/2008	Accumulate 1590 SAP.DE @ € 31,00	€ 0,00	€ 0,00
12/2/2008	Reduce 1590 SAP.DE @ € 33,40	€ 0,00	€ 0,00
22/8/2008	Accumulate 719 SIE.DE @ € 73,88	€ 0,00	€ 0,00
4/9/2008	Reduce 719 SIE.DE @ € 74,32	€ 0,00	€ 0,00
9/10/2008	Accumulate 1962 SAP.DE @ € 27,24	€ 0,00	€ 0,00
15/10/2008	Reduce 1962 SAP.DE @ € 28,40	€ 0,00	€ 0,00

Abbildung 3.18: Screenshot der Paper-Trading Transaktionen des besten Ergebnis

**Versuch 2:** Versuch 2 wurde mit dem GA-Parameter Setting 2 ebenfalls zehn mal durchgeführt. Für den Zeitraum 01/01/2008 bis 01/01/2009 ist das Endergebnis dem vom Versuch 1 gleich.

**Versuch 3:** Versuch 3 wurde mit dem GA-Parameter Setting 1 zehn mal durchgeführt. Dieses mal wurde jedoch ein größerer Zeitraum gewählt. Die beste Performanz im Zeitraum 01/04/2003 bis 01/01/2009 liegt bei 251.51 Prozent. Die Kauf/Verkauf Regel mit der besten Performanz sieht folgenderweise aus:

**Kauf Regel:**  $(rsi(7, 0)) < (22.0+4)$

**Verkauf Regel:**  $(rsi(11, 0)) > (42.0+12)$

Bei einem Startkapital von 50,000 Euro und einer Performanz von 251.51 Prozent ergäbe es einen Portfolio Wert von 175,754.73 Euro. Abbildung 3.19 zeigt die Ergebnistabelle der GA Optimierung.

Symbols	Buy Rule	Sell Rule	Generation	Final Capital	Return
SAP.DE, SIE.DE	$(rsi(7, 0)) < (22.0+4)$	$(rsi(11, 0)) > (42.0+12)$	12	€ 175.754,73	+251,51% ▲
SAP.DE, SIE.DE	$(rsi(7, 0)) < (22.0+4)$	$(rsi(11, 0)) > (42.0+13)$	2	€ 172.865,21	+245,73% ▲
SAP.DE, SIE.DE	$(rsi(7, 0)) < (22.0+4)$	$(rsi(11, 0)) > (42.0+14)$	43	€ 172.083,45	+244,17% ▲
SAP.DE, SIE.DE	$(rsi(7, 0)) < (22.0+5)$	$(rsi(11, 0)) > (42.0+13)$	32	€ 171.948,37	+243,90% ▲
SAP.DE, SIE.DE	$(rsi(7, 0)) < (22.0+5)$	$(rsi(11, 0)) > (42.0+14)$	13	€ 171.163,57	+242,33% ▲

Abbildung 3.19: Screenshot der Top 5 Ergebnisse der GA-Parameter Optimierung

**Versuch 4:** Versuch 4 wurde wieder mit dem GA-Parameter Setting 2 zehn mal durchgeführt. Für den Zeitraum 01/04/2003 bis 01/01/2009 ist das Endergebnis dem vom Versuch 3 gleich.

Alle durchgeführten Versuche ergeben eine signifikante Performanz in der Vergangenheit. Die Frage, die bleibt ist: Wie gut oder wie schlecht schneiden die optimierten Regeln bei unbekanntem Testdaten ab? Um nähere Kenntnisse über die Qualität der optimierten Regel zu gewinnen, wurden noch zwei Versuche durchgeführt. Die generierten Regeln werden einem Paper Trade Backtesting unterzogen. Der Testing Zeitraum liegt zwischen 01/01/2009 und 30/04/2009, da die Kursdaten aus diesem Zeitraum für die optimierten Regeln unbekannt sind. Als Kauf/Verkauf Regel wurden die Top 5 Ergebnisse aus Versuch 1 bis 4 verwendet. Die Ergebnisse der Paper Trade Versuche werden in den Tabellen 3.5 und 3.6 den Performanz Ergebnisse der Versuche 1 bis 4 gegenübergestellt. In diesen vier Monaten weisen beide Aktien



Versuch	Kauf/Verkauf Regel	Performanz in % 01/01/2008 bis 01/01/2009	Performanz in % 01/01/2009 bis 30/04/2009
Paper Trade 1	(rsi(14, 0)) <(22.0+0), (rsi(14, 0)) >(42.0+0)	+11.45	-0.66
Paper Trade 2	(rsi(14, 0)) <(22.0+3), (rsi(14, 0)) >(42.0+0)	+10.81	-2.89
Paper Trade 3	(rsi(14, 0)) <(22.0+2), (rsi(14, 0)) >(42.0+1)	+9.63	-2.89
Paper Trade 4	(rsi(14, 0)) <(22.0+4), (rsi(14, 0)) >(42.0+2)	+7.75	-2.62
Paper Trade 5	(rsi(14, 0)) <(22.0+3), (rsi(14, 0)) >(42.0+3)	+6.94	-2.62

Tabelle 3.5: Ergebnisse der Paper Trade mit Kauf/Verkauf Regel aus Versuch 1 und 2

Versuch	Kauf/Verkauf Regel	Performanz in % 01/04/2003 bis 01/01/2009	Performanz in % 01/01/2009 bis 30/04/2009
Paper Trade 6	(rsi(7, 0)) <(22.0+4), (rsi(11, 0)) >(42.0+12)	+251.51	+7.71
Paper Trade 7	(rsi(7, 0)) <(22.0+4), (rsi(11, 0)) >(42.0+13)	+245.73	+7.71
Paper Trade 8	(rsi(7, 0)) <(22.0+4), (rsi(11, 0)) >(42.0+14)	+244.17	+7.87
Paper Trade 9	(rsi(7, 0)) <(22.0+5), (rsi(11, 0)) >(42.0+13)	+243.90	+7.71
Paper Trade 10	(rsi(7, 0)) <(22.0+5), (rsi(11, 0)) >(42.0+14)	+242.33	+7.87

Tabelle 3.6: Ergebnisse der Paper Trade mit Kauf/Verkauf Regel aus Versuch 3 und 4

eine sehr starke Kursschwankung auf. Die Siemens Aktie erleidet mitunter sogar einen Kursverlust von 30 Prozent in den ersten drei Monaten. Im vierten Monat erholt sie sich und beträgt 93 Prozent des Wertes vom Anfang Jänner 2009. Die SAP Aktie erzielt einen Tiefstand in März 2009 und gewinnt gegen Ende April 2009 wieder an Wert. Die Abbildungen 3.20 und 3.21 zeigen die Kursverläufe der SAP und der Siemens Aktien von 01/01/2009 bis 30/04/2009 grafisch. Der Vergleich der Ergebnisse von Merchant of Venice im Zeitraum 01/01/2009 bis 30/04/2009 mit der Kursentwicklung beider Aktien im selben Zeitraum zeigt, dass die Ergebnisse aus dem Versuch an unbekannte Daten keine signifikante Gewinnzuwachs aufweisen.

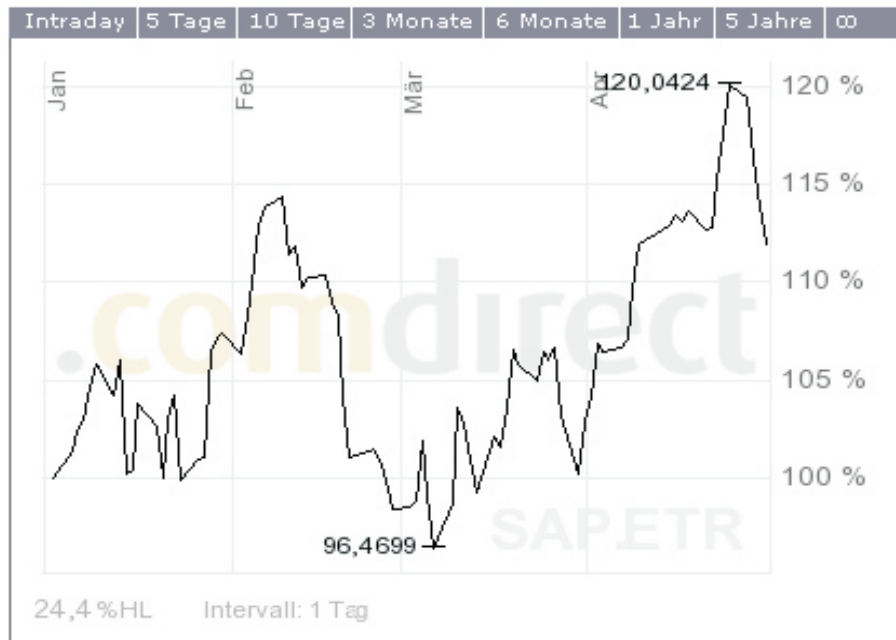


Abbildung 3.20: Kursentwicklung der SAP Aktie (XETRA) im Zeitraum 01/01/2009 bis 30/04/2009

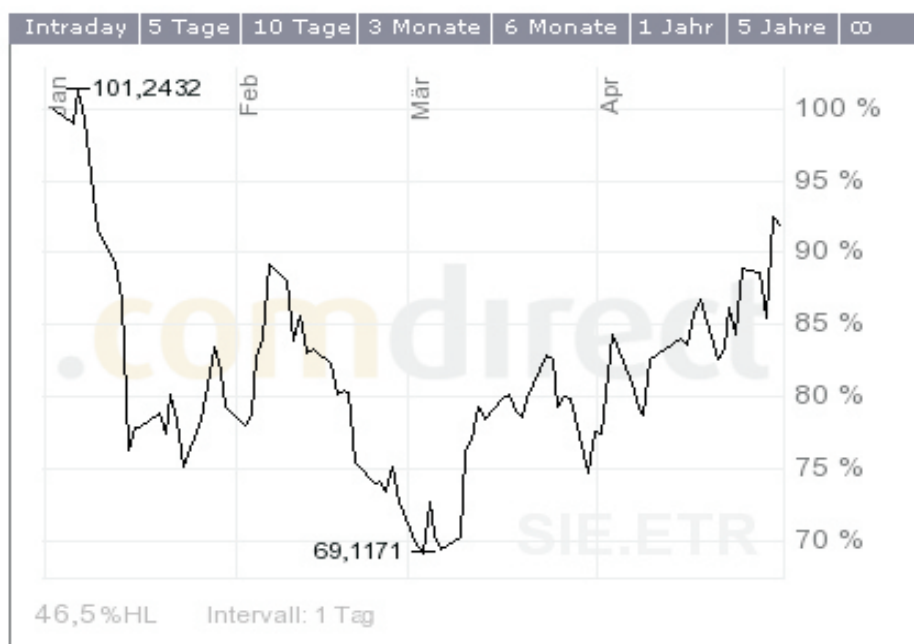


Abbildung 3.21: Kursentwicklung der Siemens Aktie (XETRA) im Zeitraum 01/01/2009 bis 30/04/2009

### 3.2.2 Kauf/Verkauf Regeln Generierung mittels GP

Die Implementierung der GP in Merchant of Venice bietet Funktionen zur Generierung von Kauf/Verkauf Regeln. Die Parameter Einstellungen und der Ablauf des Algorithmus ist dem der GA ähnlich. Der Benutzer kann hier zusätzliche Einstellungen treffen.

#### 3.2.2.1 Parameter Einstellung

Die investment-spezifischen Parameter wie der Zeitraum für das Backtesting, Portfolio Startkapital usw. bleiben gleich. Auch die GA spezifischen Parameter sind fast die gleichen. Statt der Mutationswahrscheinlichkeit „Randomness“, kommt hier ein neuer Parameter hinzu. Dieser heißt „Window Size (Days)“ und steht für Anzahl der Tage in der Vergangenheit, auf die die Kauf/Verkauf Regeln Zugriff haben dürfen.

Parameter Einstellungen, die neu hinzukommen, sind Informationen zur Startpopulation und Wahrscheinlichkeitbestimmung der Elemente einer Kauf/Verkauf Regel. Anders als beim GA Fenster hat der Benutzer hier die Möglichkeit verschiedene Kauf/Verkauf Regeln mit entsprechender Eintrittswahrscheinlichkeit einzutragen. Die Eintrittswahrscheinlichkeit bestimmt, wie wahrscheinlich diese Regel in die Startpopulation ausgewählt wird. Das hat den Zweck, erfolgversprechende Regeln in die Generierung von neuen Regeln miteinzubeziehen. Zusätzlich zur Bestimmung der Startpopulation hat der Benutzer die Möglichkeit alle Elemente der Kauf/Verkauf Regeln mit einem Wahrscheinlichkeitswert zu belegen. Elemente einer Kauf/Verkauf Regel können aus Rechenoperatoren wie +/- oder auch aus Funktionen der Finanzanalyse wie z.B. `rsi()` und `momentum()` bestehen. Die Wahrscheinlichkeitbestimmung hat zur Folge, dass der Benutzer bestimmen kann welche Elemente öfter zur Bildung neuer Regeln in Betracht gezogen werden sollen. Im Tool sind default Werte eingetragen. Diese schreiben eine gleichmäßige Wahrscheinlichkeit aller Elemente vor. In der Abbildung 3.22 sieht man Beispiel-Elemente einer Kauf/Verkauf Regel.

#### 3.2.2.2 Ablauf der Kauf/Verkauf Regel Generierung

Der Ablauf ist ähnlich wie bei der GA Funktion. Der Unterschied liegt bei der Initialisierung der Startpopulation. Der Ablauf wird schrittweise erläutert:

**Schritt 0 - Initialisierung der Startpopulation:** In dieser Phase stellt der GP Algorithmus eine Startpopulation zusammen. Der Algorithmus wählt zwischen einem nach Zufall generierten Individuum und

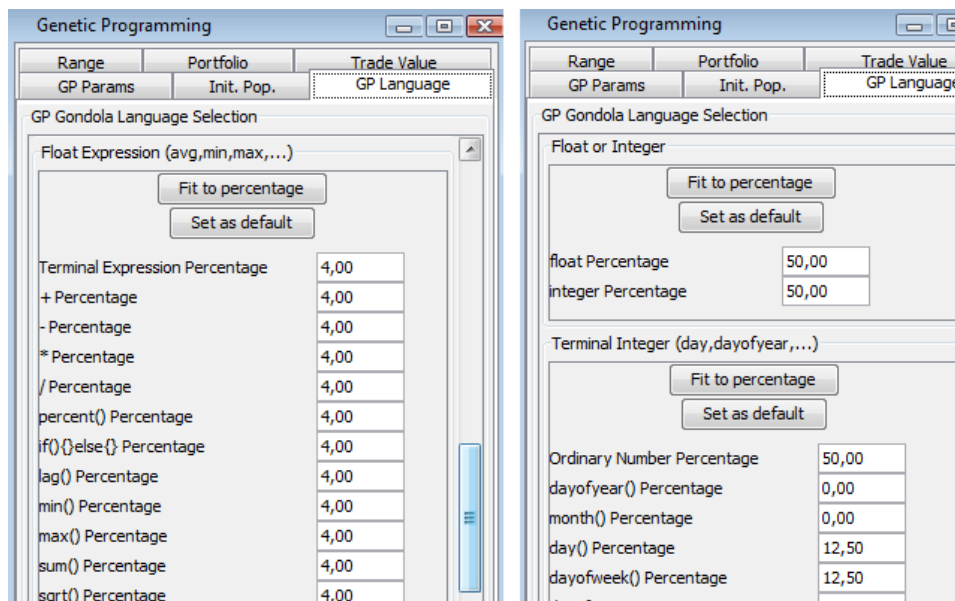


Abbildung 3.22: Screenshot der Einstellungsmöglichkeit für alle Elemente der Kauf/Verkauf Regel im Reiter „GP Language“

einem vom Benutzer eingetragenen Individuum. Wenn ein vom Benutzer bestimmtes Individuum gewählt wurde, wird dieses der Mutationswahrscheinlichkeit entsprechend verändert und bewertet. Die Mutation wird solange fortgeführt, bis der Fitnesswert des Individuums ansprechend ist. Nur fitte Individuen werden in die Startpopulation eingetragen.

**Schritt 1 - Selektion:** Aus der Startpopulation werden fitte Elternindividuen selektiert, die der Fortpflanzung dienen sollen.

**Schritt 2 - Rekombination und Mutation:** Durch Rekombination und Mutation wird eine neue Generation gezeugt.

**Schritt 3 - Bewertung der Fitness:** Die Fitnesswerte werden wieder anhand der Performanz der Regeln bewertet. Die besten Individuen werden in die Ergebnistabelle eingetragen.

**Schritt 4 - Iteration:** Der Algorithmus kehrt zurück zur Schritt 1.

Der Algorithmus wird solange fortgeführt bis eines der zwei Abbruchkriterien erreicht ist. Die Abbruchkriterien sind:

1. Anzahl der Generationen wurde erreicht

2. Der Benutzer betätigt den Abbruch Button in der Benutzeroberfläche

### 3.2.2.3 Empirische Versuche

Die Durchführung der Optimierungsfunktion wird anhand von 2 Aktien aus dem DAX gezeigt. Für diesen Zweck wurden zwei Technologie Aktien der Firmen SAP und Siemens ausgesucht. Die historischen Kursdaten sind Auszüge aus dem Zeitraum 2003 bis 2009. Die Abbildungen 3.15 und 3.16 zeigen den Kursverlauf der SAP und der Siemens Aktie im Zeitraum 2003 bis 2009.

Für die GP Parameter wird ein Setting definiert. Dieses besteht aus 50 Generationen und 250 Individuen pro Generation. Der Wert der Window Size ist 30 Tage. Tabelle 3.7 zeigt alle GP spezifischen Parameter auf einem Blick.

Parameter	Wert
Generations	50
Population	250
Breeding Population	30
Window Size	30

Tabelle 3.7: GP spezifische Parameter

Result	Buy Rule	Sell Rule	Generation	Return
SAP.DE, SIE.DE	((min(volume, dayofweek(), percent(... ((if(order <3) transactions+3else held)/29)! = ...	((if(order <3) transactions+3else held)/29)! = ...	49	+206,68%
SAP.DE, SIE.DE	((min(volume, dayofweek(), percent(... ((if(order <3) transactions+3else held)/3)! = d...)	((if(order <3) transactions+3else held)/3)! = d...)	49	+199,41%
SAP.DE, SIE.DE	((min(volume, dayofweek(), percent(... ((if(order <3) transactions+heldelse held)/29)...)	((if(order <3) transactions+heldelse held)/29)...)	37	+199,05%
SAP.DE, SIE.DE	((min(volume, dayofweek(), percent(... ((if(order <3) transactions+3else held)/3)! = d...)	((if(order <3) transactions+3else held)/3)! = d...)	41	+198,06%
SAP.DE, SIE.DE	((min(volume, dayofweek(), percent(... ((if(order <3) transactions/heldelse -35)/29)! = ...)	((if(order <3) transactions/heldelse -35)/29)! = ...)	42	+197,97%

Abbildung 3.23: Screenshot der Ergebnisse der GP Kauf/Verkauf Regel Generierung

**Versuch 1:** Der Versuch wurde zehn mal durchgeführt. Für den Zeitraum 01/01/2008 bis 01/01/2009 wurde mittels Paper-Trading eine Performanz von 206.68 Prozent berechnet. Abbildung 3.23 zeigt die Top 5 Ergebnisse der GP Kauf/Verkauf Regel Generierung. Bei einem Startkapital von 50,000 Euro und einer Performanz von 206.68 Prozent ergäbe es ein Endkapital von 153,339.26 Euro.

Die dazugehörige Kauf Regel sieht folgenderweise aus:

```
((min(volume, dayofweek(),
```

```

percent(percent(-13, daysfromstart), 3))
>=(lag(volume, percent(-22, day()))))
or (((momentum(volume, sqrt(day()), -18))
>=(min(volume, dayofweek(), -13)))
or ((momentum(volume, sqrt(day()), -18))
>=(min(volume, sqrt(day()),
percent(percent(percent(percent(-13, daysfromstart),
33), daysfromstart), 33))))))

```

Die Verkauf Regel lautet:

```

((if(order < 3)
  transactions+3
  else
  held)/29)
!=dayofweek()

```

**Versuch 2:** Der Versuch wurde ebenfalls zehn mal durchgeführt. Für diesen Versuch wird ein größerer Zeitraum als Optimierungsgrundlage verwendet. Für den Zeitraum 01/04/2003 bis 01/01/2009 wurde mittels Paper-Trading eine Performanz von 593.79 Prozent berechnet. Bei einem Startkapital von 50,000 Euro und einer Performanz von 593.79 Prozent ergäbe es ein Endkapital von 346,893.75 Euro. Die dazugehörige Kauf Regel sieht folgenderweise aus:

```

(((transactions<49) and (day()<=dayofweek()))
or (19<=day()))
or ((((((momentum(low, 18, 0)) < -3.142278)
and (day()<=dayofweek()))
or (transactions<49))
or ((momentum(close, dayofweek(), 0))
>(momentum(close, 3, 0))))
and (30<(min(volume, 18, -2))))

```

Die Verkauf Regel hat diesen Form:

```

((1202604.284165==stockcapital)
and ((lag(volume, daysfromstart))==0))
or (day()<=17)

```

**Versuch 3:** Beim Versuch 3 werden die besten Kauf/Verkauf Regeln aus dem ersten und zweiten Versuch in die Menge der Startpopulation hinzugefügt. Diese Regeln werden mit einer 90 prozentigen Wahrscheinlichkeit in die Startpopulation aufgenommen. Der Versuch wurde ebenfalls zehn mal durchgeführt. Für den Zeitraum 01/01/2008 bis 01/01/2009 wurde mittels Paper-Trading keine Performanz Verbesserung ermittelt. Die Profit Generierung liegt bei 243.07 Prozent. Bei einem Startkapital von 50,000 Euro und einer Performanz von 243.07 Prozent ergäbe es ein Endkapital von 171,533.20 Euro. Die Kauf Regel mit der besten Performanz sieht so aus:

```
((min(volume, dayofweek(), 0))
>=(lag(volume, percent(-22, day()))))
or (((momentum(volume, sqrt(day()), -18))
>=(min(volume, dayofweek(), -13)))
or (day())>=(bol_lower(volume, 5, -22)))
or ((momentum(volume, 3, -18))
>=(min(volume, 1, percent(percent(-18, daysfromstart),
day())))) or (day()==15))
```

Die dazugehörige Verkauf Regel sieht so aus:

```
(stockcapital+(sum(volume, 3, -12)))
>=(stockcapital+(day()- (-40.350957)))
```

**Versuch 4:** Die Ergebnisse der letzten drei Versuche zeigen signifikante Hinweise auf Erfolgchancen. Die generierten Regeln werden wieder an unbekanntem Datensätze getestet, um zu sehen wie gut oder wie schlecht die Regeln in der Wirklichkeit abschneiden. Der Testing Zeitraum liegt zwischen 01/01/2009 und 30/04/2009. Die Ergebnisse:

**Paper Trade - Regel 1** Mit der abgeleiteten Regel aus Versuch 1 ergab sich mittels Paper Trade im Zeitraum 01/01/2009 bis 30/04/2009 kein Ergebnis. Aufgrund eines Bugs war das Auswerten der Regel im Tool nicht möglich.

**Paper Trade - Regel 2** Mit der abgeleiteten Regel aus Versuch 2 ergab sich mittels Paper Trade im Zeitraum 01/01/2009 bis 30/04/2009 eine negative Performanz von 0.58 Prozent.

**Paper Trade - Regel 3** Mit der abgeleiteten Regel aus Versuch 3 ergab sich mittels Paper Trade im Zeitraum 01/01/2009 bis 30/04/2009 eine Portfolio Wertsteigerung von 7.25 Prozent.



Auch hier kann man erkennen, dass Kauf/Verkauf Regeln, obwohl sie in der Vergangenheit vielversprechende Ergebnisse brachten, in der Zukunft und bei veränderter Marktsituation wenige bis keine Erfolge versprechen. Genau aus diesem Grund müssen Kauf/Verkauf Regeln besonders bei geänderter Marktsituation neu aktualisiert und neu optimiert werden, um Gewinneinbüßen zu verhindern.

### **3.3 Performanz-Analyse der GA/GP Funktionalität unter verschiedenen Marktbedingungen**

In dieser Sektion wird die Performanz der GA bzw. GP Funktionalität bei unterschiedlichen Marktsituationen untersucht. Es soll betrachtet werden, wie sich die Ergebnisse der GA/GP bei einem wachsenden Aktienmarkt als auch bei einem turbulenten Markt verhalten. Der Untersuchungszeitraum für den wachsenden Markt ist von 2006 bis 2008. Der Untersuchungszeitraum für den turbulenten Markt ist von 2008 bis 2009. Die Performanz-Analyse wird anhand von 10 Aktien aus dem deutschen Aktien Index DAX durchgeführt. Anhand der Kursgrafik des DAX (siehe Abb. 3.24) erkennt man deutlich die zwei unterschiedlichen Phasen vor 2008 und nach 2008. Im Zeitraum 2006 bis 2008 erreichte der DAX einen Zuwachs von 3000 Punkten. Im Zeitraum 2008 bis 2009 hingegen stürzte der deutsche Aktienindex massiv ab, nämlich von 8000 Punkten auf die ursprünglichen 5000 Punkte.

#### **3.3.1 Studie der GA Funktionalität im wachsenden Markt**

Die Bewertungsgrundlagen sind die Kursdaten von 10 Aktientiteln, die nach einem Stabilitätsfaktor ausgesucht werden. Zur Bewertung im wachsenden Markt werden aus verschiedenen Wirtschaftssektoren Aktien entnommen, die einen kleinen Volatilitätswert aufweisen (sprich ein stabileres Kursverhalten aufzeigen).

In der Tabelle 3.8 sind die ausgewählten Aktien dargestellt. Die historischen Kursdaten der Aktien wurden anschließend in Merchant of Venice importiert.

##### **3.3.1.1 Parameter Konfiguration**

In der Tabelle 3.9 und 3.10 sind alle Parameter der Untersuchung aufgelistet. Diese Parameter Aufstellung besagt, dass das virtuelle Depot mit einem Initial Kapital von 100,000 Euro startet. Im Depot können sich maximal zehn Aktien gleichzeitig befinden. Für die Kauf/Verkauf Regeln werden wieder



Abbildung 3.24: Performanz der DAX in den letzten 5 Jahren

Rang	Name	Branche	Volatilität
1	Fresenius Medical Care	Pharma u. Gesundheitswesen	34.17
2	Beiersdorf AG	Pharma u. Gesundheitswesen	35.50
3	Henkel AG	Haushalts Produkte	39.88
4	Linde AG	Maschinenbau	45.17
5	RWE AG	Energieversorgung	46.47
6	Deutsche Telekom AG	Telekommunikation	46.69
7	Muenchner Rueck	Versicherung	47.18
8	SAP AG	Software	49.15
9	Adidas AG	Textilien und Bekleidung	49.41
10	Deutsche Lufthansa AG	Fluggesellschaft	52.99

Tabelle 3.8: 10 ausgewählte Aktientitel aus dem DAX

Parameter	Wert
Kauf Regel	$\text{rsi}(\text{bx}) < 22.0 + \text{ParamBuy}$
bx	7 - 14
ParamBuy	0 - 5
Verkauf Regel	$\text{rsi}(\text{sx}) > 42.0 + \text{ParamSell}$
sx	10 - 14
ParamSell	0 - 20
Anwendungszeitraum	01/01/2006 - 01/01/2008
Initial Kapital	100,000
Transaktionskosten	0
Anzahl der Aktien im Depot	10

Tabelle 3.9: Ausführungsspezifische Parameter

Parameter	Wert
Generations	10
Population	100
Breeding Population	10
Randomness	0

Tabelle 3.10: GA spezifische Parameter

die RSI Werte der letzten  $x$  Tage einer Aktie angewendet. Hier werden bewusst dieselben Regel aus dem vorherigen Unterkapitel 3.2.1.3 verwendet, da schon ein Vergleichswert mit zwei Aktien von den empirischen Versuchen existiert. Weiters ist bekannt, dass der gewählte Zeitraum von Kurswachstum geprägt ist. Es wird daher erwartet, dass dieselben Regeln sehr gute Performanz in Form von Kursgewinnen mit sich bringen.

### 3.3.1.2 Das Ergebnis

Nach 10 Durchführungen der Parameter Optimierung mittels GA im Merchant of Venice gibt es folgende Ergebnisse:

Die Top 5 Kauf/Verkauf Regeln bringen nach der Optimierung im Durchschnitt nur 30 Prozent Gewinn. Dieser Satz ist extrem gering verglichen mit der Wachstum Quote des DAX von etwa 3000 Punkten in diesem zwei Jahres Zeitraum. Es muss auch vermerkt werden, dass bei dieser Analyse der Faktor der Transaktionskosten nicht berücksichtigt wurde. Sprich, mit Berücksichtigung der Transaktionskosten, wäre der Gewinn beinahe nicht vorhanden. Die Frage, die sich nun stellt: Hat die GA-basierte Optimierung versagt oder hat die schlechte Performanz andere Gründe? Die Performanz der GA liegt

Paper Trade Nr.	Kauf/Verkauf Regel	Performanz im Zeitraum von 01/01/2006 bis 01/01/2008
Paper Trade 1	(rsi(8, 0)) <(22.0+5), (rsi(14, 0)) >(42.0+19)	+32.08
Paper Trade 2	(rsi(7, 0)) <(22.0+4), (rsi(14, 0)) <(42.0+19)	+31.44
Paper Trade 3	(rsi(8, 0)) <(22.0+5), (rsi(13, 0)) >(42.0+20)	+31.30
Paper Trade 4	(rsi(8, 0)) <(22.0+4), (rsi(13, 0)) >(42.0+20)	+30.67
Paper Trade 5	(rsi(7, 0)) <(22.0+4), (rsi(14, 0)) >(42.0+12)	+30.32

Tabelle 3.11: Ergebnisse der Parameter Optimierung im Zeitraum 2006 bis 2008

nicht allein in der Qualität der Optimierungsfunktion. Ein Faktor, der direkt auf die Ergebnisse der GA wirkt, sind die GA-Parameter wie z.B. die Anzahl der Generationen, die Größe der Population und der Zuchtpopulation. Für einen erneuten Versuch werden die GA-Parameter verändert. Die GA spezifischen Parameter wurden in der Tabelle 3.12 aufgelistet. Wie man sieht, wurden die Anzahl der Generationen und die Größe der Population eindeutig erhöht, mit dem Ziel, dass möglichst viele Lösungen bewertet werden in möglichst vielen Durchläufen.

Parameter	Wert
Generations	100
Population	1000
Breeding Population	100
Randomness	50

Tabelle 3.12: GA spezifische Parameter

Die Versuche mit einer größeren Population, mit mehr Generationen und mit einer größeren Zuchtpopulation ergeben die gleichen Ergebnisse wie in der Tabelle 3.11. Ein weiterer sehr wichtiger Faktor der Performanz liegt auch in der Qualität der verwendeten Kauf/Verkauf Regeln. Sind die Regeln von Anfang an unpassend gewählt worden, kann ein Optimierungsalgorithmus, sei es ein genetischer Algorithmus oder die erschöpfende Suche, die Gewinne auch nicht massiv verbessern.

### 3.3.2 Studie der GA Funktionalität im turbulenten Markt

Die Bewertungsgrundlagen bilden wieder 10 ausgewählte Aktientitel aus dem DAX. Zur Bewertung der GA Funktionalität werden aus verschiedenen Wirtschaftssektoren Aktien entnommen, die eine hohe Volatilitätsrate aufweisen. Die ausgesuchten Aktien wurden in der Tabelle 3.13 aufgelistet und in Merchant of Venice importiert.

Rang	Name	Branche	Volatilität
1	K+S Aktiengesellschaft	Rohstoffe	167.41
2	Volkswagen AG	Automobil Industrie	120.77
3	Commerzbank AG	Bankensektor	107.80
4	Deutsche Bank AG	Bankensektor	100.67
5	Salzgitter	Maschinenbau	77.82
6	Daimler AG	Automobil Industrie	72.44
7	Allianz SE	Versicherung	71.65
8	Thyssenkrupp AG	Metall- und Bergbau	71.00
9	Metro AG	Einzelhandel	66.17
10	Deutsche Post AG	Luftfracht und Kurierdienste	64.83

Tabelle 3.13: 10 ausgewählte Aktientitel aus dem DAX

#### 3.3.2.1 Parameter Konfiguration

Parameter Konfiguration sind in den Tabellen 3.14 und 3.15 aufgelistet.

Parameter	Wert
Kauf Regel	$rsi(bx) < 22.0 + ParamBuy$
bx	7 - 14
ParamBuy	0 - 5
Verkauf Regel	$rsi(sx) > 42.0 + ParamSell$
sx	10 - 14
ParamSell	0 - 20
Anwendungszeitraum	01/01/2008 - 01/01/2009
Initial Kapital	100,000
Transaktionskosten	0
Anzahl der Aktien im Depot	10

Tabelle 3.14: Ausführungsspezifische Parameter

Parameter	Wert
Generations	10
Population	100
Breeding Population	10
Randomness	0

Tabelle 3.15: GA spezifische Parameter

### 3.3.2.2 Das Ergebnis

Nach 10 Versuche erreichte die am besten optimierte Kauf/Verkauf Regel eine Performanz von 7.97 Prozent im Jahr 2008 bis 2009. Der Zeitraum 01/01/2008 bis 01/01/2009 war ein sehr turbulentes Jahr. Der DAX fiel in diesem Zeitraum von ursprünglich 8000 Punkte auf 5000 Punkte, was einen beträchtlichen Marktabsturz darstellt. So gesehen ist die positive Performanz von 7.97 Prozent Gewinn innerhalb eines Jahres ein mittelmäßiges Ergebnis, die die GA-Parameter Optimierung hier berechnen konnte. Alle 5 Top Kauf/Verkauf Regeln sind in der Tabelle 3.16 aufgelistet.

Paper Trade Nr.	Kauf/Verkauf Regel	Performanz im Zeitraum von 01/01/2008 bis 01/01/2009
Paper Trade 1	(rsi(10, 0)) <(22.0+5), (rsi(11, 0)) >(42.0+1)	+7.97
Paper Trade 2	(rsi(10, 0)) <(22.0+5), (rsi(12, 0)) <(42.0+2)	+7.26
Paper Trade 3	(rsi(10, 0)) <(22.0+5), (rsi(11, 0)) >(42.0+0)	+7.11
Paper Trade 4	(rsi(10, 0)) <(22.0+5), (rsi(12, 0)) >(42.0+0)	+5.97
Paper Trade 5	(rsi(10, 0)) <(22.0+4), (rsi(10, 0)) >(42.0+0)	+5.49

Tabelle 3.16: Ergebnisse der Parameter Optimierung im Zeitraum 2006 bis 2008

Die GA spezifischen Parameter werden wieder verändert um zu untersuchen, ob die Performanz durch eine größere Population und Anzahl an durchlaufenen Generationen noch verbesserbar ist. Die neuen GA-Parameter sind in der Tabelle 3.17 zu finden. Nach 10 Durchläufen ergab die Berechnung wieder die gleichen Werte wie in der Tabelle 3.16 angegeben.

Parameter	Wert
Generations	100
Population	1,000
Breeding Population	50
Randomness	50

Tabelle 3.17: GA spezifische Parameter

### 3.3.3 Studie der GP Funktionalität im wachsenden Markt

Die Untersuchungen zuvor haben ergeben, dass die Funktionalität der GA bei der Parameter Optimierung maßgeblich von der Qualität der Kauf/Verkauf Regeln abhängt. In diesem Unterkapitel wird die GP Funktionalität des Programms untersucht. Bei der Kauf/Verkauf Regel Generierung existiert keine Abhängigkeit zwischen der Qualität der Kauf/Verkauf Regel und der Qualität der Ergebnisse.

Parameter	Wert
Generations	100
Population	1,000
Breeding Population	100
Window Size	30 Tagen

Tabelle 3.18: GP spezifische Parameter

#### 3.3.3.1 Parameter Konfiguration

In den nachfolgenden Untersuchungen werden jeweils die Kursdaten der letzten 30 Tage (Window Size = 30) betrachtet. Es sollen 100 Generationen mit 1,000 Individuen pro Generation generiert werden. Alle GP spezifischen Parameter sind aus der Tabelle 3.18 zu entnehmen. Es werden die selben zehn Aktien wie in der Tabelle 3.8 angegeben zur Untersuchung herangezogen. Initial Kapital beträgt wieder 100,000 Euro. Die maximale Anzahl an Aktien im Depot liegt bei 10 verschiedenen Aktien. Anwendungszeitraum liegt zwischen 01/01/2006 und 01/01/2008.

#### 3.3.3.2 Das Ergebnis

Der GP Algorithmus ergab nach der Berechnung folgende Ergebnisse:  
Kauf Regel:

```

((if(9>=day())
  lag(high, -16)
else
  -47.368033
)<(avg(open, 30, 0)))
and (((avg(low, 26, 0))<(lag(high, -16)))
and (((min(volume, 5, -16))>58862)
and ((avg(low, 30, 0))<(lag(high, -15))))
and ((avg(low, 31, 0))<(lag(high,
-16))))))

```

Verkauf Regel:

```

(day())>=(min(volume, 1, -1)) or (((dayofweek()!=day())
and (((0.958666>=cos(daysfromstart))
or (36>=(min(volume, dayofweek(), -25))))
and ((0.958666>=cos(avg(volume, dayofweek(),
percent(-19, day()/dayofweek()))))
or (36>=(min(volume, 9, -22)))))) and (day()<26))

```

Der Gewinnzuwachs beträgt für den Zeitraum der Berechnung +96.97 Prozent. Ein Papertest im Zeitraum 01/01/2009 bis 30/04/2009 ergibt einen Gewinnzuwachs von +1.46 Prozent.

### 3.3.4 Studie der GP Funktionalität im turbulenten Markt

#### 3.3.4.1 Parameter Konfiguration

Der Anwendungszeitraum der nachfolgenden Untersuchungen liegt zwischen 01/01/2008 und 01/01/2009. Alle anderen Parameter bleiben gleich wie in der Sektion 3.3.3.1.

#### 3.3.4.2 Das Ergebnis

Kauf Regel:

```

((bol_upper(volume, dayofweek(), -6))<(lag(volume, -1)))
or (((min(volume, 3, -20))< order)
or (((((obv(12, -7, 199041))>(momentum(volume, 25, -5)))
and ((momentum(volume, 25, -4))<(lag(volume, -13))))
and (242477<(lag(volume, -2))))
and (dayofweek()<(lag(volume, -13))))))

```



Verkauf Regel:

```
(transactions<=65) or (((3<=dayofweek()  
or ((min(volume, 6, -21))<=dayofweek()))  
or (((min(volume, 9, -7))<=(held*daysfromstart))  
or ((transactions<=47)  
or ((bol_upper(volume, 3, 0))<=(sd(volume, day(), -1))))))
```

Der Gewinnzuwachs beträgt für den turbulenten Kursabschnitt +152.20 Prozent. Ein Papertest im Zeitraum 01/01/2009 bis 30/04/2009 ergibt einen Verlust von -17.83 Prozent.

## Kapitel 4

# Fallbeispiel MultiCharts

## 4.1 Allgemeine Produkt Beschreibung

### 4.1.1 Kurzbeschreibung

MultiCharts ist ein professionelles Trading-Tool für End-of-Day und Intra-Day Trader. Entwickelt wurde das Tool von der Firma TS Support. TS Support hat ihren Sitz in den USA und setzt den Fokus auf die Entwicklung von professionellen Trading-Tools. Ihr Ziel ist es ihren Kunden Trading-Tools mit der neusten Technologie und der höchsten Qualität sowie dem besten Service zu bieten. Das Tool MultiCharts bietet neben üblichen Funktionen wie Portfoliomanagement, Finanzanalyse und Handelssystemgenerierung noch weitaus fortgeschrittenere Funktionen an. Das Tool erlaubt automatisierte Orderabgabe, unterstützt das Handeln mit Futures, Aktien und Forex<sup>27</sup> und multidimensionale Graphen-Analyse-Methodiken. Weiters unterstützt das Programm die Programmiersprache TradeStation EasyLanguage<sup>28</sup>. MultiCharts wurde von diversen Finanzmagazine höchst positiv bewertet. Im Jahr 2008 gewann das Tool die Member's Choice Award Auszeichnung als beste professionelle Trading Plattform, beste Software für UK/US End-of-Day Traders und beste Software für US Intra-Day Traders.

### 4.1.2 Software Packages und Programmaufbau

MultiCharts hat einen dezentralen Aufbau. Zur MultiCharts Plattform gehören noch fünf unabhängig ausführbare Dateien:

- MultiCharts
- QuoteManager
- Portfolio-Backtester
- PowerLanguage Editor
- 3D Optimization Charts

MultiCharts ist die zentrale Komponente der Plattform. Diverse Funktionen wurden aus der zentralen Komponente ausgelagert, um das Hauptprogramm modular und übersichtlich zu halten. Der Benutzer kann im Hauptprogramm auf die Funktionalitäten aller Komponenten zugreifen. Die Komponenten sind auch getrennt vom Hauptprogramm ausführbar.

---

<sup>27</sup>Forex steht für „Foreign Exchange market“ (Devisenmarkt).

<sup>28</sup>TradeStation EasyLanguage ist die defacto-standard Programmiersprache zur kundenspezifischen Erstellung, Anpassung, Test und Automatisierung von Indikatoren und Handelstrategien [PH03]

Das Sub-Programm QuoteManager ist ein Tool zum Importieren, Exportieren und Verwalten von Kursinformationen. Portfolio-Backtester ermöglicht das Verwalten, das Backtesting und die Optimierung von Portfolios. PowerLanguage Editor ist eine Programmier Umgebung zur Erstellung und Anpassung von technischen Indikatoren und Trading Strategien. Die vom Benutzer programmierten Indikatoren und Strategien können im Hauptprogramm eingesetzt werden. Das Sub-Programm 3D Optimization Charts ist eine drei-dimensionale Chartingumgebung zur Anzeige von Optimierungsergebnissen.

### 4.1.3 Grundfunktionalitäten

#### Kursdatenimport/-export und Management

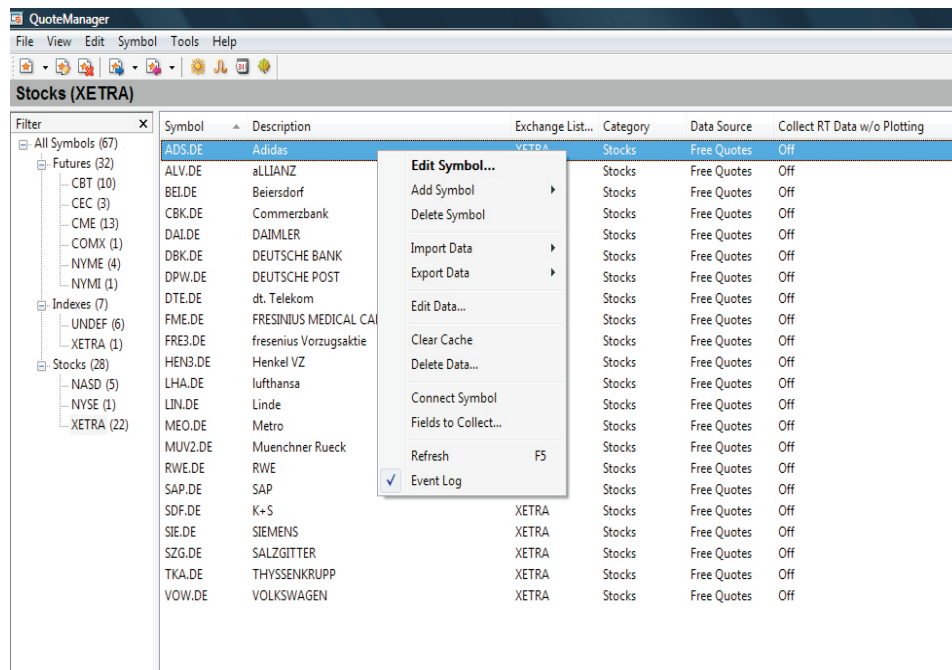


Abbildung 4.1: Screenshot des QuoteManager in MultiCharts

Alle Kursdatenimport/-export Funktionen werden im Sub-Programm QuoteManager den Benutzern zur Verfügung gestellt. Abbildung 4.1 zeigt einen Screenshot von QuoteManager mit einer Liste von importierten Kursdaten. MultiCharts bietet eine direkte Anbindung an diverse kostenpflichtige und kostenfreie Kursdatenquellen zum automatischen Kursdatenimport. Folgende Portale werden unterstützt: AGN Futures, Bloomberg, eSignal,

GlobalServer, IQFeed, Interactive Brokers, MarketCast, MetaStock, OpenE-Cry, OpenTick, Patsystems, QFeed, TradeStation8, Tenfore, TransAct, FreeQuotes (Yahoo Finance, MSN MoneyCentral, ProphetFinance.com und Google Finance). Die Unterstützung für online Anbindungen zu Datenquellen ist erweiterbar. Benutzer haben die Möglichkeit, Kursdaten von neuen Portalen zu importieren. Abbildung 4.2 zeigt einen Screenshot des „Data Source“ Fensters mit einem kleinen Ausschnitt der verfügbaren Kursdatenquellen.

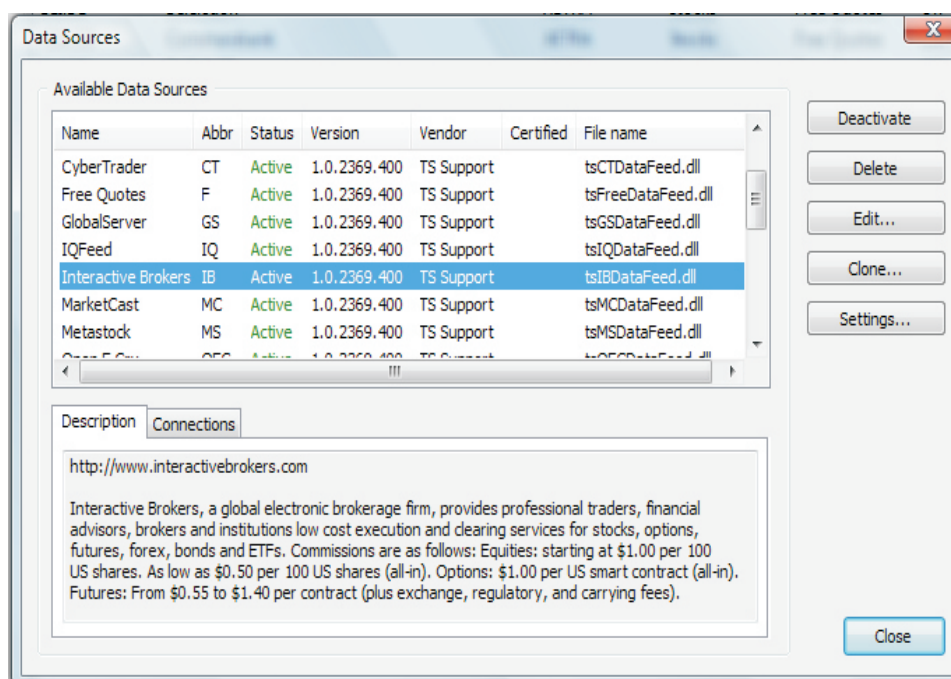


Abbildung 4.2: Screenshot der Kursdatenquellen in MultiCharts

Manuelle Datenimports über Dynamic Data Exchange Schnittstelle (DDE)<sup>29</sup> und ASCII Files sind ebenfalls möglich. Folgende DDE Schnittstellen werden unterstützt: Bloomberg, CAIT, eSignal, FAINEX, FXCM, GL Trade, Hi SAT, Infobolsa.de, IQLink, MetaTrader, QLink, QuoteSpeed, Reuters, Taipan und Tradeexpert.

Die in MultiCharts integrierte Datenbank hat eine Speicherkapazität von über 2 Milliarden Tick Daten. Um die Performanz zu verbessern werden Kursdaten in mehreren Granularitätsstufen gespeichert, nämlich Tick basiert, minütlich und täglich. Weiters besitzt die Datenbank einen Tick Ca-

<sup>29</sup>DDE ist ein Daten-Übertragungsprotokoll zwischen MS Windows Applikationen [Pez90].

che. Dort werden die aktuellsten Tick Daten gespeichert, um einen schnelleren Zugriff auf neueste Daten sicherzustellen. Gespeicherte Tick Daten und historische Kursdaten können im Falle von fehlerhaften Daten editiert und gelöscht werden. Weiters können alle gespeicherten Kursdaten als Comma Separated Value (CSV) exportiert und als ASCII Dokument gespeichert werden.

Eine weitere Besonderheit der Datenbank von MultiCharts ist die Fähigkeit Daten zusammenzuführen. Das System erlaubt es Kursdaten eines Wertpapierstitels von verschiedenen Datenquellen zu beziehen. Die bezogenen Daten werden je nach Auflösungsstufen zusammengeführt und abgespeichert.

### **Innovative Graphische Oberfläche**

MultiCharts vermarktet auf ihrer Homepage (siehe [Sup09]) ihre graphische Oberfläche als innovativ. Die Innovation besteht in der objekt-orientierten Benutzer Oberfläche. Jedes Element ist ein Objekt mit dem der Benutzer interagieren kann. Durch die intuitive Gestaltung kann der Benutzer das Arbeiten mit dem Programm einfacher und rascher erlernen.

Die Benutzer Oberfläche besteht aus einer Menü Leiste, zwei Symbolleisten, und der Arbeitsfläche. Die Toolbar stellt wichtige Funktionen mit einem Mausklick dem Benutzer zur Verfügung. Das zentrale Element der Benutzeroberfläche ist jedoch die Arbeitsfläche (Workspace). In dieser kann der Benutzer beliebig viele Chartwindows erzeugen, aktivieren, de-aktivieren, kopieren, maximieren, minimieren, löschen usw. Chartwindows dienen vorrangig der Darstellung von Kursdaten, Indikatoren und anderen zusammenhängenden Daten. Zur Verdeutlichung und Darstellung des Chartwindows wurde eine Beispiel aus der mitgelieferten MultiCharts Dokumentation entnommen (Abbildung 4.3).

Die Nummerierungen in der Abbildung 4.3 haben folgende Bedeutungen:

1. Status Linie - zeigt Informationen des dargestellten Diagramms wie z.B. Titelname, gehandelte Börse, angewendete Indikatoren an.
2. Chart Bereich - Bereich wo die Kursdaten in Form eines Diagramms dargestellt werden.
3. Trennlinie
4. Subchart Bereich - Subcharts sind Vergleichsdiagramme zu Nummer 2.
5. Zeit Achse

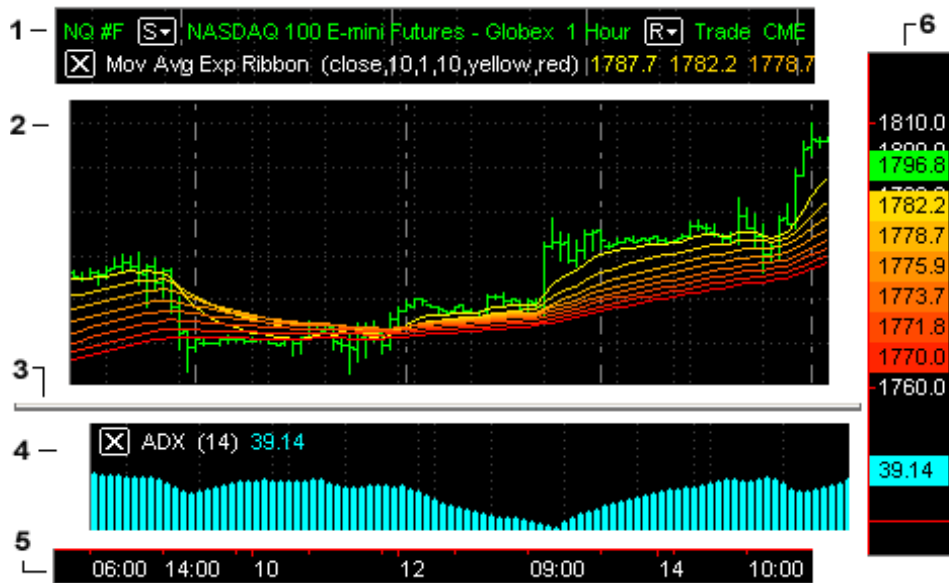


Abbildung 4.3: Beispiel Abbildung eines Chartwindow in MultiCharts

## 6. Preis Achse

MultiCharts bietet eine ausgiebige Kursdaten Darstellungs- und Analyse Funktionalität.

Der Benutzer hat große Einstellmöglichkeiten zum Konfigurieren von Aussehen und Verhalten der Chart Fenster. Es können jederzeit Screenshots von einem Graph erstellt und als PNG oder BMP File abgespeichert werden. Screenshots können auch per Email verschickt werden. Kursdaten werden in mehreren Granularitätsstufen in der Datenbank gespeichert. Aus diesem Grund können sie auch in verschiedenen Auflösungen bildlich dargestellt werden. MultiCharts unterstützt Kursdaten Darstellungen in folgenden Detailstufen:

- Darstellung von unüberarbeiteten Ticks
- Darstellung von N-Ticks in einem Balken (bis zu 1 mio. Ticks per Balken möglich)
- Darstellung der Intervalle in einem Balken
- Darstellung der Kursänderungen in einem Balken (bis zu 1 mio. Kursänderungen per Balken möglich)
- Darstellung der Sekunden/Minuten/Stunden/Tage/Wochen/Monate/Quartale in einem Balken

- Darstellung der Handelsvolumen

Ein Graph kann eine beliebige Anzahl an Wertpapiertiteln oder technischen Indikatoren in beliebiger Detailstufe darstellen. Jedes Element im Graph kann selektiert und analysiert werden, sodass auch Informationen aus einem Chart zur Weiterverarbeitung rausgezogen werden können. In einem Chart können bis zu 50 sichtbare oder auch unsichtbare Subcharts erstellt werden. Die Darstellung der Kursdaten passiert in Echtzeit. Sprich, alle Objekte auf einem Graph werden zeitgleich aktualisiert, sobald neue Kursdaten übertragen worden sind. Die Navigation im Graph ist ebenfalls sehr mächtig. Sowohl horizontale als auch vertikale Scrollingfunktionen sind vorhanden, weiters kann man in einen Graph beliebig hinein- und herauszoomen. Ein Beispiel zur Verdeutlichung: der Benutzer kann ausgehend von den End-of-Day Daten zu den stündlichen Kursveränderungen bis hin zu den einzelnen Ticks hineinzoomen. Voraussetzung ist natürlich, dass eine entsprechende Datenquelle Anbindung besteht, oder die Daten bereits ins System importiert wurden. Eine weitere nützliche Funktion vom MultiCharts ist das Warnsystem. Beim Eintreffen bestimmter Ereignisse können Warnsignale ausgegeben werden. Der Benutzer hat die Wahl zwischen audio-visuellen Alarmmeldungen und Email Warnungen. Zusammenfassend ist die Chartingfunktion des Tools sehr mächtig und funktional ausgestattet, sodass fast alle Interaktionen mit und im Graph möglich sind.

### **Analysetechniken**

MultiCharts unterstützt alle gängigen technischen Indikatoren und Signale zur Analyse von Aktienkursdaten. Alle implementierten Techniken sind zugänglich über sogenannte „Studien“ (Studies). Jede Studie besteht aus modular gestalteten Einzelteilen und kann sowohl über die GUI als auch über den Sourcecode modifiziert werden. Man zählt bis zu hundert verschiedene traditionelle und moderne Analysetechniken. Wenn die zur Verfügung gestellten Studien jedoch nicht spezielle Kundenwünsche befriedigen, können die Benutzer über Studien-Templates eigene Analysetechniken erstellen und als Studien-Objekt abspeichern. Abbildung 4.4 zeigt Screenshots der Reiter „Signals“ und „Indicator“ mit Ausschnitt an verfügbaren Signalen und Indikatoren für die Analyse.

### **Handelsstrategien erzeugen, optimieren und testen**

Die Erstellung von Handelsstrategien stellt eine Hauptkomponente des Programms dar. Der Benutzer kann seine Strategie aus einem oder mehreren



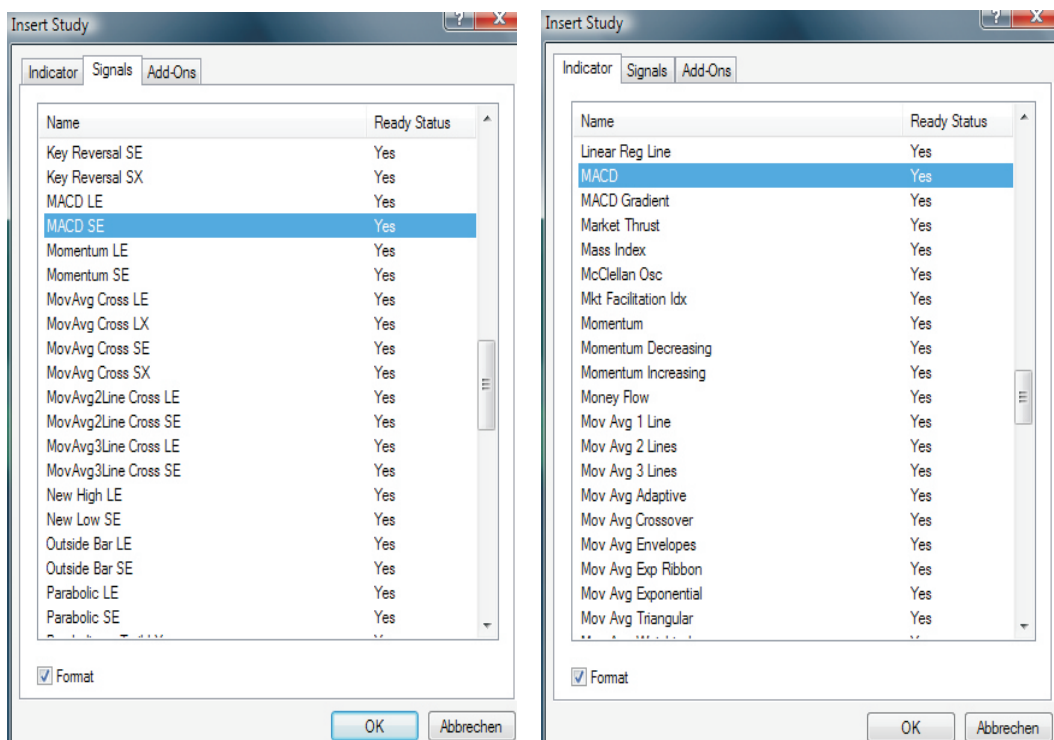


Abbildung 4.4: Screenshot der Reiter „Signals“ und „Indicator“

vorgefertigten Signalen zusammenstellen oder auch Strategien mittels der zur Verfügung gestellten Programmiersprache PowerLanguage implementieren. Eine Strategie kann aus einzelnen, multiplen oder kombinierten Kauf/-Verkauf Signalen bestehen und mit einer sekundären Strategie versehen werden. Der Benutzer hat weiters die Möglichkeit die Darstellung der Strategien in Diagrammen nach seinen Vorstellungen und Vorlieben anzupassen. Das Programm kann auf Wunsch dem Benutzer beim Eintreten verschiedener Ereignisse audio/visuelle Warnungen schicken. Aus den Strategien generierte Kauf/Verkauf Signale können automatisiert an Interactive Broker geleitet werden, um dazugehörige Kauf/Verkauf Order abzugeben.

Vor dem tatsächlichen Einsatz einer Strategie muss diese ausführlich auf Fehler, Performanz und Robustheit getestet werden. Tests dieser Art, auch Backtesting genannt, werden mittels historischer Kursdaten durchgeführt. Weiters kann die Performanz einer Strategie optimiert werden. Bei der Performanz Optimierung wird mittels historischer Kursdaten die Performanz einer Strategie mit unterschiedlichen Sets von Parametern untersucht. MultiCharts bietet dem Kunden zwei Optimierungsansätze:

1. Optimierung mittels Erschöpfender Suche (Brute Force/exhaustive search): Die erschöpfende Suche geht jede potentielle Parameter-Konfiguration durch. Die Rechenzeit ist somit proportional zur Anzahl der möglichen Lösungen. Bei einem sehr großen Lösungsraum wird die Rechenzeit enorm lang. Daher ist dieser Optimierungsansatz nur für Optimierungsproblemen mit kleiner Lösungsmenge geeignet.
2. Optimierung mittels genetischer Algorithmen: Die Optimierung mittels genetischer Algorithmen bedarf nur einen Bruchteil der Rechenzeit verglichen mit der erschöpfende Suche. Genetische Algorithmen können bis zu hundert Parameter in einem akzeptablen Zeitraum optimieren. MultiCharts bewirbt diesen Optimierungsansatz mit einem Vorteil, dass über-optimierte Lösungen schon in frühen Stadien aussortiert werden.

Ergebnisse von Backtesting und Performanz Optimierung werden in Performanz Reports umgewandelt. Der Benutzer kann selbst entscheiden, welche Messgrößen bzw. Performanz Kriterien im Report angezeigt werden sollen. Eine andere Art der Darstellung der Optimierungsergebnisse ist der 3D Optimierungsgraph. Dieser Graph zeigt an, wie sich die Parameter einer Strategie auf die Performanz auswirken. Weiters werden die robustesten Parameter Zonen im Graph angezeigt um eine Überoptimierung zu verhindern. Abbildung 4.5 zeigt einen Screenshot des Sub-Programms „3D Optimization Charts“ mit einem Beispiel 3D Optimierungsgraph.

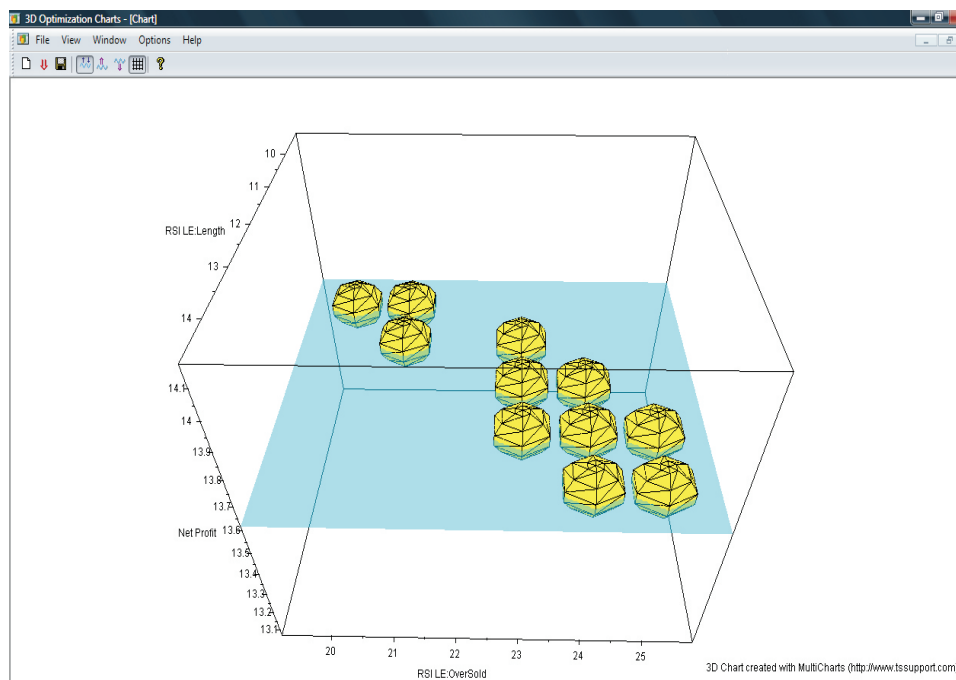


Abbildung 4.5: Screenshot des Sub-Programms „3D Optimization Charts“ in MultiCharts

## Portfoliomanagement und Backtesting

MultiCharts bietet umfassende Portfolio Funktionen wie z.B. Erstellung, Verwaltungen, Testen und Optimieren von 1 bis n Portfolios. Der Benutzer kann mehrere Portfolios mit exakt denselben Titeln und selben Werten erstellen und in jedem Portfolio eine andere Handelstrategie anwenden. Der Benutzer kann aber auch in jedem Portfolio eine andere Anlageform wählen und auf allen Portfolios dieselbe Handelstrategie anwenden. Dieses Prozedere wird im Rahmen von MultiCharts Portfolio-Backtesting genannt. Den Vorteil des Portfolio-Backtesting gegenüber der normalen Backtesting Prozedur erklärt MultiCharts so:

„Backtesting a strategy on a single instrument may not produce enough trades to distinguish a pattern from a coincidence. Backtesting a strategy on a number of instruments is more likely to reveal any shortcomings and helps to avoid over-optimization. A robust strategy is likely to demonstrate consistent profitability across various instruments. By backtesting on a diverse portfolio, the instruments most suitable for the particular trading strategy can be selected.“ *MultiCharts [Sup09]*

## **Interface zum Online-Broker zur automatisierten Orderabgabe**

Automatisierte Orderabgabe ermöglicht das Erstellen und Weiterleiten der generierten Order an den Broker zur Ausführung. Verglichen mit der manuellen Orderabgabe ist die Automatisierung zeitlich effizienter und erlaubt somit einen hochfrequenten Handelsrhythmus. MultiCharts kann an folgende Online-Broker-Schnittstellen angebunden werden:

- **Interactive Brokers:** Um die Orderausführung über die Interactive Brokers Schnittstelle zu automatisieren, muss ein separates Programm installiert werden. Das Programm dient als Gateway. Generierte Order werden über dieses Gateway an den Ziel Server geroutet, wo die Order vom Broker entgegengenommen und ausgeführt wird.
- **Zen-Fire Trading Engine:** Die Zen-Fire Trading Engine ist eine weitverbreitete Live-Trading Engine und wird von vielen Brokern unterstützt.

MultiCharts unterstützt zwei Arten von automatisierter Orderabgabe. Die erstere arbeitet im synchronen Modus und die zweite im asynchronen Modus. Synchroner Modus bedeutet, dass die Order bezogenen Informationen wie Kauf/Verkauf Preise bzw. Kauf/Verkauf Zeitpunkt nur dann historisiert und auf dem Graph angezeigt werden, wenn die Order tatsächlich vom Broker durchgeführt wurde. Bei dem asynchronen Modus ist die Order Durchführung nicht garantiert. Das kann zu Konflikten in der Datenaufzeichnung führen.

## **Genetische Algorithmen**

MultiCharts bündelt die Funktionen der genetischen Algorithmen in zwei Bereichen ein. Der Erste ist die Optimierung von Handelsstrategien. Der Zweite ist die Optimierung von Portfolios. Die Funktionen werden in der Sektion 4.2 näher erläutert.

### **4.1.4 Lizenz und Kostenpunkt**

TS Support bietet fünf Typen von Lizenzen an, die man als Privatperson oder Unternehmen erwerben kann. Das Minimumpaket für drei Monate fängt bei 236 Euro an, sprich der Benutzer zahlt pro Monat 78.60 Euro für die Benutzung des Programms. Wenn man jedoch das Programm längerfristig verwenden will, hat man die Möglichkeit eine Halb- oder Jahreslizenz zu erwerben. So würde man für die Halbjahres Lizenz 410 Euro und für die Jahreslizenz 658.58 Euro zahlen. Für Firmen oder Institutionen bietet TS

Support auch eine zeitlich unlimitierte Lizenz um 1,425 Euro an. Die Lizenz für MultiCharts PRO kostet 9,735 Euro.

#### **4.1.5 Systemanforderung**

MultiCharts teilt die Systemanforderung in drei Gruppen:

1. Minimale Anforderung
2. Empfohlene Anforderung
3. Anforderung für Power User

##### **Minimale Anforderung**

Die minimale Systemanforderung umfasst folgende Komponenten:

**CPU** : 1 GHz

**RAM** : 256 MB

**Speicherplatz** : 200 MB

**Monitor** : 1024 x 768

**OS** : Windows 2000, Windows XP SP2, Windows XP x64, Windows Vista, Windows Vista x64

##### **Empfohlene Anforderung**

Die empfohlene Systemanforderung umfasst folgende Komponenten:

**CPU** : Pentium 3 GHz oder Dual Core

**RAM** : 1 GB

**Speicherplatz** : 1 GB

**Monitor** : multiple Monitor Lösungen

**OS** : Windows 2000, Windows XP SP2, Windows XP x64, Windows Vista, Windows Vista x64

### **Anforderungen für Power User**

Multicharts definiert Power User als Benutzer mit übermäßigen Anforderungen wie z.B. die gleichzeitige Verwendung von 5 oder mehr Tick-basierten Charts, die gleichzeitige Verwendung von 10 oder mehr Intervall Charts, die Verwendung von 15 oder mehr Indikatoren bzw. Signalen auf einem einzigen Chart (siehe [Sup09]).

Die Anforderung für Power User sieht folgendermaßen aus:

**CPU** : Quad Core

**RAM** : 3 GB

**Speicherplatz** : 1 GB

**Monitor** : multiple Monitor Lösungen

**OS** : Windows 2000, Windows XP SP2, Windows XP x64, Windows Vista,  
Windows Vista x64

## 4.2 Einsatz der GA

MultiCharts ist ein kostenpflichtiges Programm mit einer kommerziellen Lizenz. Aus diesem Grund steht der Quellcode des Programms nicht zur Verfügung, weshalb die Arbeitsweise der genetischen Algorithmen in diesem Programm nur annähernd beschrieben werden kann. Die Beschreibung basiert auf den Funktionen, die das Programm in der Testversion dem Benutzer zur Verfügung stellt und der dazugehörigen Dokumentation. Generell unterscheidet MultiCharts zwischen zwei Einsatzbereichen der genetischen Algorithmen. Der Erste dient der Optimierung von Handelsstrategien. Der Zweite dient der Optimierung von Portfolios.

### 4.2.1 Strategie-Optimierung mittels GA

Eine Strategie basiert auf einer Menge von einfachen und komplexen Kauf/-Verkauf Signalen<sup>30</sup>, Finanz-Investment-Konzepten und Beobachtungen aus historischen Kursdaten. Das Strategie-basierte Handeln soll den Prozess der Analyse von Kursdaten, Generierung von Kauf/Verkauf Signalen und Anwendung von Finanz Konzepten weitgehend automatisieren. Die Optimierung einer Strategie ist somit die Suche nach den geeigneten Parametern in einer vorgegebenen Umgebung. Im Prozess der Strategie-Optimierung werden die möglichen Parameter Werte mit Hilfe von historischen Kursdaten und anderen wichtigen Finanzgrößen bewertet und die Performanz der Strategie gemessen. Die optimierte Strategie kann dann in der Anwendung zum Erfolg führen oder auch versagen.

In MultiCharts kann eine Strategie-Optimierung nur dann durchgeführt werden, nachdem die Strategie zum tatsächlichen Einsatz gekommen ist. Auch hier wird der Benutzer darauf hingewiesen, dass der Einsatz der genetischen Algorithmen keinen Erfolg garantiert. Es kann eine, mehrere, oder auch keine Lösung zu einem Optimierungsproblem existieren. Durch die Implementierung der Abbruchkriterien wird der Optimierungsalgorithmus auf jedenfall beendet.

#### 4.2.1.1 Parameter Einstellung

Vor der Optimierung muss der Benutzer alle notwendigen Parameter vollständig und korrekt ausfüllen. Das Programm teilt die Parameter in zwei Kategorien ein: die optimierbaren Parameter und die GA spezifischen

---

<sup>30</sup>Signale bestimmen den Markteintritt und Marktaustritt basierend auf verschiedenen Einflussfaktoren wie etwa Transaktionskosten, Performanzentwicklungen und Resultaten aus Backtesting.

Parameter. Ein Beispiel der optimierbaren Parameter zeigt der Screenshot des Reiters „Optimizable Inputs“ in der Abbildung 4.6. Die GA spezifischen Parameter sind in der Abbildung 4.7 zu sehen.

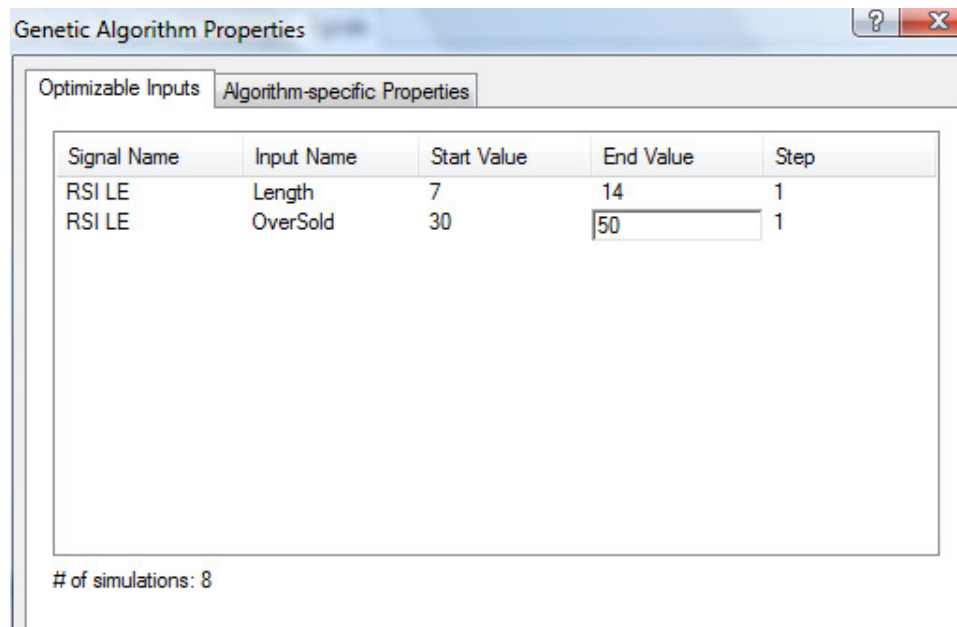


Abbildung 4.6: Screenshot der Einstellungen für auf GA-basierende Strategie-Optimierung in MultiCharts

In der Beispiel Abbildung 4.6 wird die Optimierung von einer RSI basierten Kauf/Verkauf Regel durchgeführt. Der Benutzer muss hier die Intervalle aller betroffenen Parameter angeben. Das Feld „# of Simulations“ zeigt an, wieviele mögliche Parameterkonfigurationen mit den eingetragenen Werten existieren und für die Optimierung relevant sind.

Die Einstellung der GA ist im Reiter „Algorithm-specific Properties“ durchzuführen. Der Benutzer kann diese Punkte selbst bestimmen:

- Nach welchen Kenngrößen die Bewertung der Performanz stattfinden soll
- Wie groß die Population sein soll
- Wie groß die Wahrscheinlichkeit des Crossover- bzw. Mutations-Operators sein soll
- Wieviele Generationen der Algorithmus durchlaufen soll
- Die Abbruchkriterien



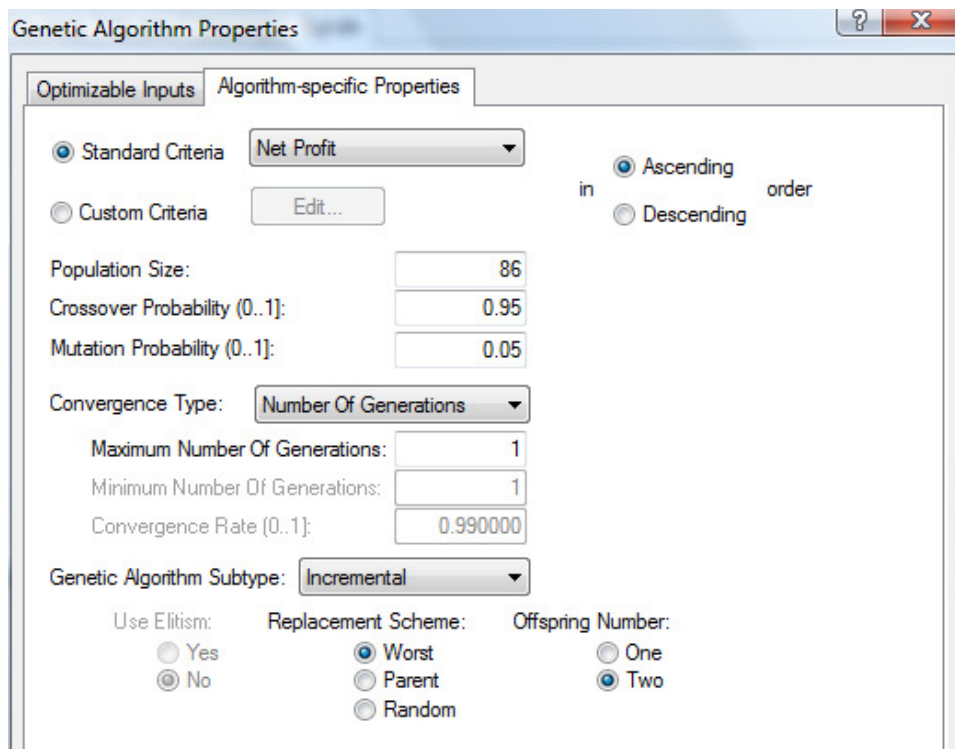


Abbildung 4.7: Screenshot der Einstellungen für auf GA-basierende Strategie-Optimierung in MultiCharts

- Den GA Subtyp und die Ersetzungs Regeln

#### 4.2.1.2 Ablauf der Optimierung

Der Ablauf des GA-basierten Optimierungsalgorithmus verläuft laut der Dokumentation folgenderweise:

1. Generierung der gesamten Lösungsmenge mit Hilfe der vorhandenen Benutzereinstellungen.
2. Selektion der Startpopulation aus der generierten Lösungsmenge. Die Selektion passiert zufällig bis die optimale Größe der Population erreicht ist.
3. Bewertung der Fitnesswerte der Individuen aus der Startpopulation. Anschließend erfolgt die Selektion der fittesten Individuen bzw. Verwerfung der unfittesten Individuen.
4. Anwendung der Crossover- und Mutations-Operatoren zur Generierung der Kindergeneration aus der selektierten Elterngeneration. Hier kann der Benutzer einstellen, wie wahrscheinlich die Crossover und Mutations-Operatoren auftreten. Die Eintrittswahrscheinlichkeit des Crossover Operators liegt zwischen 0.95 und 0.99 (entspricht 95 und 99 Prozent). Die Eintrittswahrscheinlichkeit des Mutations-Operators liegt zwischen 0.01 und 0.05 (entspricht 1 und 5 Prozent).

Je nach eingestelltem GA Subtyp und Ersetzungsregel arbeitet der Algorithmus anders. GA Subtyp ist die Art wie der GA neue Individuen erzeugt und die Vorgänger Individuen ersetzt. Das Programm unterscheidet zwischen zwei auswählbaren Arten: Basic und Inkrementell. Bei dem „Basic“ Subtyp werden vollkommen neue Individuen erzeugt. Bei dem „Inkrementellen“ Subtyp werden jedoch bei jeder Generation ein oder zwei neue Kinderindividuen zu der Elterngeneration hinzugefügt. Sprich in diesem Fall werden die Generationen inkrementell aufgebaut und keine vollkommen neuen Individuen in jeder Generation erstellt. Die neuen Individuen ersetzen auch gleichzeitig dieselbe Anzahl der Individuen der Elterngeneration. Der Benutzer kann auch hier einstellen, wie die Ersetzung der Vorgänger Individuen passieren soll. Das Programm unterscheidet zwischen drei Typen von Ersetzungsregeln: Ersetzung der unfittesten Individuen, zufallsbedingte Ersetzung oder Ersetzung der Elternindividuen.

5. Bewertung der Fitnesswerte der Kindergeneration und Verwerfung der unfittesten Individuen.

6. Der Algorithmus wird solange wiederholt bis Abbruchkriterien eintreten.

Die Ergebnisse der Optimierung werden in einer Tabelle präsentiert, die verschiedene Einflussfaktoren und Parameter Werte der Strategie anzeigen. Diese Tabelle kann in einer CSV Datei exportiert werden.

#### 4.2.1.3 Empirische Versuche

Die empirischen Versuche werden unter folgender Testumgebung durchgeführt:

**Prozessor:** Intel Core 2 Duo P8600 2.40GHz

**RAM:** 4 GB

**Betriebssystem:** Windows Vista 32 Bit

Die Durchführung der GA-basierten Strategie-Optimierung wird anhand einer Technologieaktie aus dem DAX gezeigt. Für diesen Zweck wurde die Aktie der Firma SAP gewählt. Die historischen Kursdaten sind Auszüge aus dem Zeitraum 2006 bis 2009. Abbildung 4.8 zeigt den Kursverlauf der SAP Aktie im Zeitraum 2006 bis 2009. Anhand dieser Grafik kann man einen langsamen Abwärtstrend mit kurzfristigen Erholungen im Kursverlauf erkennen. Ende 2008 fiel die SAP Aktie auf 23,45 Euro pro Anteil herab und erreicht einen historischen Tiefstand.

Als Initialkapital wurde eine Höhe von 100,000 Dollar angenommen. Transaktionskosten werden hier aus Komplexitätsgründen vernachlässigt. Für die GA-Parameter werden verschiedene Parameter-Settings definiert, die in Folge aufgelistet werden. Die Konfigurationen unterscheiden sich hauptsächlich in der Eintrittswahrscheinlichkeit der Crossover und Mutations-Operatoren und in der Wahl der GA Subtypen bzw. der Ersetzungsregeln.

Als Kauf/Verkauf Signal wird wieder RSI verwendet. In der Tabelle 4.4 sind die Formeln der Kauf/Verkauf Signale aufgelistet. Die Intervalle für die Parameter „length“, „OverSold“ und „OverBought“ sind frei wählbar. Für die Strategie-Optimierung werden folgende Parameter benutzt:

**RSI LE length:** 7 bis 14

**RSI SE length:** 10 bis 14

**OverSold:** 22 bis 27

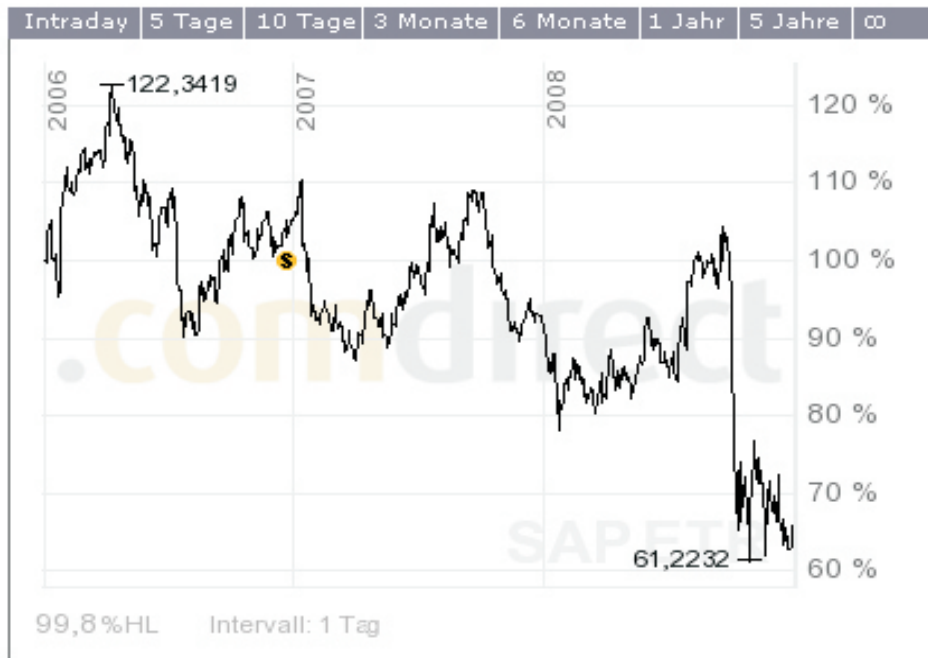


Abbildung 4.8: Kursentwicklung der SAP Aktie (XETRA) im Zeitraum 2006 bis 2009

Parameter	Wert
Standard Criteria	Net Profit
Population Size	100
Crossover Probability	0.95
Mutation Probability	0.01
Convergence Type (Abbruchkriterium)	Number of Generations
Maximum of Generations	1,000
GA subtyp	Basic
Use Elitism	yes

Tabelle 4.1: GA-Parameter: Setting 1

Parameter	Wert
Standard Criteria	Net Profit
Population Size	100
Crossover Probability	0.95
Mutation Probability	0.01
Convergence Type (Abbruchkriterium)	Number of Generations
Generations	1,000
GA subtyp	Incremental
Replacement Scheme	Worst
Offspring Number	2

Tabelle 4.2: GA-Parameter: Setting 2

Parameter	Wert
Standard Criteria	Net Profit
Population Size	10
Crossover Probability	0.95
Mutation Probability	0.05
Convergence Type (Abbruchkriterium)	Number of Generations
Generations	10
GA subtyp	Incremental
Replacement Scheme	Random
Offspring Number	2

Tabelle 4.3: GA-Parameter: Setting 3

Regel	Signal Name	Formel
Kauf	RSI LE	$\text{rsi}(\text{length}) < \text{OverSold}$
Verkauf	RSI SE	$\text{rsi}(\text{length}) > \text{OverBought}$

Tabelle 4.4: Kauf/Verkauf Signale

## OverBought: 42 bis 62

Die Versuche ergaben folgende Ergebnisse:

**Versuch 1:** Versuch 1 wurde mit dem Setting 1 durchgeführt. Das beste Ergebnis lieferte einen Netto Profit von 45,356 USD. Das ergibt einen „Return on initial capital“<sup>31</sup> von +45.35 Prozent.

Die dazugehörige Kauf/Verkauf Regel:  $rsi(12) < 24$  bzw.  $rsi(13) > 57$   
Das bedeutet, wenn die RSI Werte der SAP Aktie von den letzten zwölf Tagen unter 24 liegt, ein Kauf Signal erzeugt wird. Umgekehrt, wenn der RSI Wert von den letzten dreizehn Tagen über 57 liegt, wird ein Verkauf Signal erzeugt.

**Versuch 2:** Versuch 2 wurde mit dem Setting 2 durchgeführt. Versuch 2 lieferte exakt das gleiche Ergebnis wie beim ersten Versuch. Das beste Ergebnis lieferte einen Netto Profit von 45,356 USD. Das ergibt einen „Return on initial capital“ von +45.35 Prozent.

**Versuch 3:** Versuch 3 wurde mit dem Setting 3 durchgeführt. Hier werden pro Generation nur 10 Individuen erzeugt. Es werden insgesamt 10 Generationen erzeugt. Das beste Ergebnis lieferte einen Netto Profit von 29,589 USD. Das ergibt einen „Return on initial capital“ von +29,58 Prozent.

Die dazugehörige Kauf/Verkauf Regel:  $rsi(13) < 25$  bzw.  $rsi(11) > 54$

Die Tabelle 4.5 fasst die Ergebnisse der drei Versuche auf einem Blick zusammen.

Versuch	Regel	Performanz in Prozent
Versuch 1	$rsi(12) < 24; rsi(13) > 57$	+45.35
Versuch 2	$rsi(12) < 24; rsi(13) > 57$	+45.35
Versuch 3	$rsi(13) < 25; rsi(11) > 54$	+29.58

Tabelle 4.5: Ergebnistabelle mit verschiedenen Settings

---

<sup>31</sup>Return on initial capital = Net Profit / Initial Capital

## 4.2.2 Portfolio-Optimierung mittels GA

Die zweite GA Funktion, die das Programm bietet ist die Portfolio-Optimierung. Die Strategie-Optimierung kann nur auf eine einzige Anlage im Rahmen eines Diagramms durchgeführt werden. Die Einschränkung ist bei der Portfolio-Optimierung aufgehoben. Das Handeln im Rahmen eines Portfolios erlaubt dem Benutzer verschiedene Anlagen in einem Portfolio zusammenzuführen und gebündelt eine oder mehrere Handelsstrategien auf das Portfolio anzuwenden. Bei der Optimierung des Portfolios wird in der Lösungsmenge nach der optimalsten Strategie gesucht.

Die Portfolio-Optimierung ist als Funktion in dem separaten Unterprogramm „MultiCharts Portfolio-Backtester“ verfügbar. Die Abbildung 4.9 zeigt einen Screenshot des Programms. Bevor man die Optimierung durchführen kann, muss man in diesem Subprogramm Anagentitel und Signale in das Portfolio hinzufügen. Im Portfolio-Baum und in der Symbolliste sieht man alle Titel, die einem Portfolio hinzugefügt wurde. In der Abbildung 4.9 sieht man 10 verschiedene Aktien aus dem DAX. Auf der rechten Seite des Programms kann der Benutzer portfolio-abhängige Einstellungen vornehmen. Der Benutzer kann einen Zeitraum angeben, der beim Portfolio-Backtesting bzw. bei der Portfolio-Optimierung als Grundlage dienen soll. Weiters kann er die Höhe des Initialkapitals angeben und Risiko Management Einstellungen vornehmen. Abbildung 4.10 zeigt die weiteren Parameter des Portfolio-Backtesters.

Der Ablauf der auf GA-basierenden Portfolio-Optimierung ist der gleiche wie bei der Strategie-Optimierung, der in der Sektion 4.2.1.2 beschrieben wurde. Daher wird dieser hier nicht nochmals erklärt. In den nachfolgenden Sektionen wird die Performanz der Portfolio-Optimierung bei unterschiedlichen Marktsituationen untersucht.

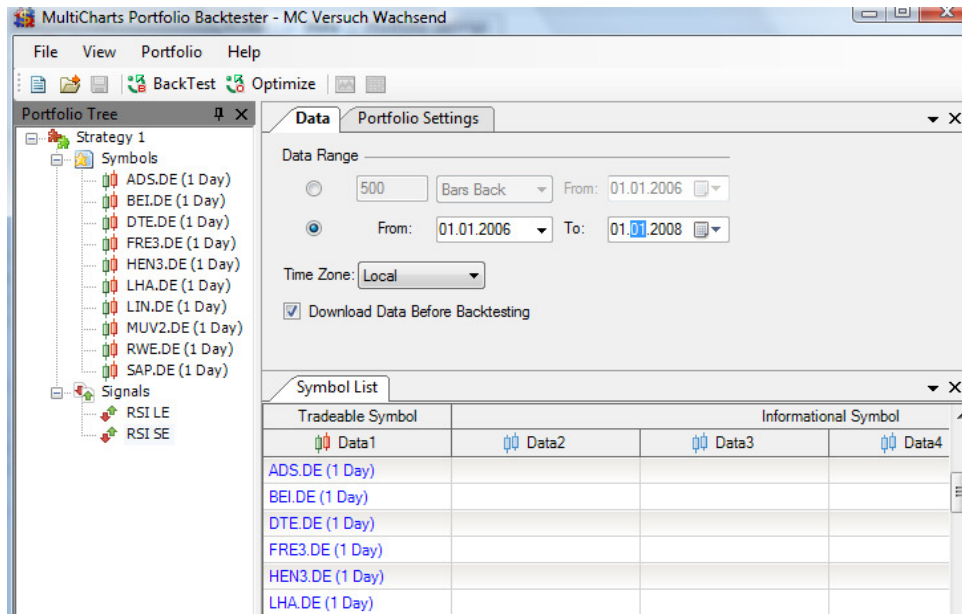


Abbildung 4.9: Screenshot des Subprogramms „MultiCharts Portfolio-Backtester“

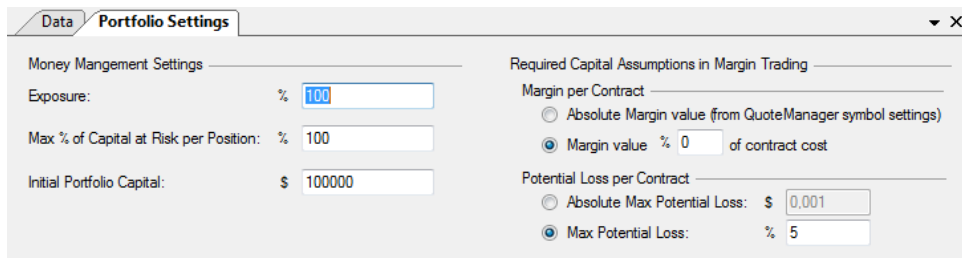


Abbildung 4.10: Screenshot vom Reiter „Portfolio Settings“ im Subprogramm „MultiCharts Portfolio-Backtester“



## 4.3 Performanz-Analyse der GA-basierte Portfolio-Optimierung unter verschiedenen Marktbedingungen

In dieser Sektion wird die Performanz der GA-basierten Portfolio-Optimierung bei unterschiedlichen Marktsituationen untersucht. Es soll betrachtet werden, wie sich die Ergebnisse der Portfolio-Optimierung bei wachsendem und bei turbulentem Aktienmarkt verhalten, analog zu Kapitel 3.3. Untersuchungszeitraum für den wachsenden Markt ist von 2006 bis 2008. Untersuchungszeitraum für den turbulenten Markt ist von 2008 bis 2009. Die Performanz-Analyse wird wie im Kapitel 3.3 anhand von 10 Aktien aus dem DAX durchgeführt.

### 4.3.1 Studie der GA-basierten Portfolio-Optimierung im wachsenden Markt

Die Bewertungsgrundlagen bilden wieder 10 ausgewählte Aktientitel aus dem DAX. In der Tabelle 4.6 sind die ausgewählten Aktien dargestellt. Die historischen Kursdaten wurden dem Yahoo Finanz Portal entnommen und in MultiCharts als ein Portfolio importiert. Nachfolgend werden verschiedene Settings definiert und anschließend Untersuchungen mit den gesetzten Parameter-Konfigurationen durchgeführt.

Rang	Name	Branche	Volatilität
1	Fresenius Medical Care	Pharma u. Gesundheitswesen	34.17
2	Beiersdorf AG	Pharma u. Gesundheitswesen	35.50
3	Henkel AG	Haushalts Produkte	39.88
4	Linde AG	Maschinenbau	45.17
5	RWE AG	Energieversorgung	46.47
6	Deutsche Telekom AG	Telekommunikation	46.69
7	Muenchner Rueck	Versicherung	47.18
8	SAP AG	Software	49.15
9	Adidas AG	Textilien und Bekleidung	49.41
10	Deutsche Lufthansa AG	Fluggesellschaft	52.99

Tabelle 4.6: 10 ausgewählte Aktientitel aus dem DAX

#### 4.3.1.1 Parameter-Konfiguration: Setting 1

In der Tabelle 4.7 sind alle Parameter für die Untersuchung aufgelistet. Als Initialkapital werden wieder 100,000 USD verwendet und Transaktionskosten werden aus Komplexitätsgründen vernachlässigt. Als Kauf/Verkauf Regel wird die RSI Formel verwendet. Die maximale Transaktionshöhe ist der maximale Geldbetrag, der pro Kauf/Verkauf Transaktion aufgewendet werden darf. Dieser Parameter ist insofern wichtig, da bei einer hohen Limitierung auch hohe Verluste entstehen können, aber natürlich auch hohe Gewinne möglich sind. In diesem Setting wird die maximale Transaktionshöhe auf 100,000 USD gesetzt. Das entspricht dem gesamten zur Verfügung stehenden Initialkapital.

Parameter	Wert
Kauf (RSI LE)	rsi(length) < OverSold
RSI LE length	7 - 14
OverSold	22 - 27
Verkauf (RSI SE)	rsi(length) > OverBought
RSI SE length	10 - 14
OverBought	42 - 62
Anwendungszeitraum	01/01/2006 - 01/01/2008
Initial Kapital	100,000 USD
Transaktionskosten	0
max. Transaktionshöhe	100,000 USD
Standard Criteria	Net Profit
Population Size	100
Crossover Probability	0.95
Mutation Probability	0.05
Convergence Type (Abbruchkriterium)	Number of Generations
Generations	1,000
GA subtyp	Incremental
Replacement Scheme	Worst
Offspring Number	2

Tabelle 4.7: Ausführungsspezifische Parameter

#### 4.3.1.2 Das Ergebnis

Das Ergebnis dieser Analyse ergibt einen Netto Profit von 209,777 USD. Das entspricht einem „Return on initial capital“<sup>32</sup> von +209.78 Prozent.

<sup>32</sup>Return on initial capital = Net Profit / Initial Capital

Die dazugehörigen Kauf/Verkauf Regeln lauten:  $rsi(8) < 26$  bzw.  $rsi(14) > 62$

#### 4.3.1.3 Parameter-Konfiguration: Setting 2

Beim Setting 2 wird lediglich ein Parameter geändert. Die maximale Transaktionshöhe wird auf 10,000 USD gesetzt.

#### 4.3.1.4 Das Ergebnis

Das Ergebnis dieser Analyse ergibt eine Netto Profit von 13,752 USD. Das entspricht einem „Return on initial capital“ von +13.75 Prozent. Das ist ein markanter Unterschied verglichen mit dem Ergebnis der vorherigen Analyse.

### 4.3.2 Studie der GA-basierten Portfolio-Optimierung im turbulenten Markt

Die Bewertungsgrundlagen bilden wieder 10 ausgewählte Aktientitel aus dem DAX. Die ausgesuchten Aktien wurden in der Tabelle 4.8 aufgelistet. Die historischen Kursdaten wurden dem Yahoo Finanz Portal entnommen und in MultiCharts importiert.

Rang	Name	Branche	Volatilität
1	K+S Aktiengesellschaft	Rohstoffe	167.41
2	Volkswagen AG	Automobil Industrie	120.77
3	Commerzbank AG	Bankensektor	107.80
4	Deutsche Bank AG	Bankensektor	100.67
5	Salzgitter	Maschinenbau	77.82
6	Daimler AG	Automobil Industrie	72.44
7	Allianz SE	Versicherung	71.65
8	Thyssenkrupp AG	Metall- und Bergbau	71.00
9	Metro AG	Einzelhandel	66.17
10	Deutsche Post AG	Luftfracht und Kurierdienste	64.83

Tabelle 4.8: 10 ausgewählte Aktientitel aus dem DAX

#### 4.3.2.1 Parameter-Konfiguration: Setting 1

In der Tabelle 4.9 sind alle Parameter für die Untersuchung aufgelistet.

Parameter	Wert
Kauf (RSI LE)	rsi(length) < OverSold
RSI LE length	7 - 14
OverSold	22 - 27
Verkauf (RSI SE)	rsi(length) > OverBought
RSI SE length	10 - 14
OverBought	42 - 62
Anwendungszeitraum	01/01/2008 - 01/01/2009
Initial Kapitel	100,000 USD
Transaktionskosten	0
max. Transaktionshöhe	100,000 USD
Standard Criteria	Net Profit
Population Size	100
Crossover Probability	0.95
Mutation Probability	0.05
Convergence Type (Abbruchkriterium)	Number of Generations
Generations	1,000
GA subtyp	Incremental
Replacement Scheme	Worst
Offspring Number	2

Tabelle 4.9: Ausführungsspezifische Parameter

#### **4.3.2.2 Das Ergebnis**

Das Ergebnis dieser Analyse ergibt einen Netto Profit von 246,542 USD. Das entspricht einen „Return on initial capital“ von +245.54 Prozent. Die dazugehörigen Kauf/Verkauf Regeln lauten:  $rsi(7) < 27$  bzw.  $rsi(14) > 44$

#### **4.3.2.3 Parameter-Konfiguration: Setting 2**

Beim Setting 2 die maximale Transaktionshöhe auf 10,000 USD gesetzt.

#### **4.3.2.4 Das Ergebnis**

Das Ergebnis dieser Analyse ergibt eine Netto Profit von 21,223 USD. Das entspricht einen „Return on initial capital“ von +21.22 Prozent. Auch hier ist eine markante Differenz mit den Ergebnissen zuvor zu vermerken.

## Kapitel 5

# Diskussion der technisch-strategischen Aspekte von Trading-Tools und genetischen Algorithmen

In diesem Kapitel werden die Funktionalitäten und die Ergebnisse der Performanz-Analysen der vorgestellten Trading-Tools miteinander verglichen. Darüber hinaus werden die Vor- und Nachteile des jeweiligen Tools diskutiert.

## 5.1 Vergleich der Funktionalitäten

In dieser Sektion werden die Funktionalitäten von „Merchant of Venice“ denen von „MultiCharts“ gegenübergestellt. Ein direkter Vergleich beider Tools ist insofern schwierig, da die bereitgestellten Funktionen beider Tools variieren. Aus diesem Grund werden die wichtigsten Funktionen beider Tools entnommen und in Folge miteinander verglichen. Die Grundfunktionen sind: Kursdatenimport/-export bzw. Management, Grafische Benutzeroberfläche, Analysetechniken, Erzeugen/Optimieren/Testen von Handelstrategien, Portfolio-Management und Backtesting. In dieser Sektion wird ein besonderes Augenmerk auf den Einsatz der genetischen Algorithmen beider Tools gelegt.

### 5.1.1 Kursdatenimport/-export und Management

Der Import von Kursdaten ist essentiell in einem Trading-Tool. Merchant of Venice unterstützt nur eine kleine Palette an Kursdatenformaten. Die direkte Anbindung an das Yahoo Finance Portal ist einfach gestaltet und sehr nützlich, da dieser Service kostenlos ist. Mit nur einem Klick werden End-Of-Day Kursdaten des gewünschten Zeitraums auf dem Rechner gespeichert.

In MultiCharts kann der Benutzer die Kursdaten über den QuoteManger importieren, exportieren und editieren. MultiCharts unterstützt mehr Kursdatenformate und Datenquellen als Merchant of Venice. Die Mehrheit davon sind jedoch kostenpflichtig. Die Anbindung an kostenlose Kursdaten Portale wie Yahoo geschieht aufwendiger. Der Benutzer muss auf der Homepage des Yahoo Finance Portals die Kursdaten in Form von einem Comma Separated File runterladen. Der QuoteManager importiert anschließend die Kursdaten aus dem File und speichert Sie in einer internen Datenbank. MultiCharts richtet sich an einen anderen Kundenkreis als Merchant of Venice. Hier wurde besonderes Augenmerk auf das Importieren von Real-Time Kursdaten, auf die Speicherkapazität der Datenbank und die Granularität der Daten gerichtet.

### 5.1.2 Grafische Oberfläche

Beide Trading-Tools bieten eine grafische Benutzeroberfläche. Die grafische Benutzeroberfläche in Merchant of Venice ist rudimentär einfach gestaltet und realisiert. Sie konzentriert sich auf die Darstellung der Kursdaten und Analyseergebnissen in Diagramm oder in Tabellenform. MultiCharts hingegen bietet eine sehr innovative und hochwertige Oberfläche. Durch die vielen Steuerungsmöglichkeiten können nicht alle Interaktionen intuitiv ge-

halten werden. Es ist ratsam das mitgelieferte Tutorial durchzuarbeiten um das Maximum der zur Verfügung gestellten Steuerungsmöglichkeiten auszuschöpfen.

### 5.1.3 Analysetechniken

Ähnlich wie bei der Gestaltung der GUI kann Merchant of Venice nur eine kleine Anzahl an Analysetechniken vorweisen. Implementierte Techniken umfassen Bollinger Bands, gleitender Durchschnitt, Moving Average Convergence Divergence, Momentum, On Balance Volume und Relative Strength Index. MultiCharts bietet über hundert fertig implementierte Analysetechniken, die miteinander kombiniert, editiert und erweitert werden können. Ein klarer Vorteil in Hinblick auf die funktionale Erwartung professioneller Trader.

### 5.1.4 Handelstrategien: Erzeugung, Optimierung und Testen

Die Qualität einer Handelstrategie ist eine wichtige Grundlage für den Erfolg beim Handeln. Sowohl Merchant of Venice als auch MultiCharts unterstützen das Erstellen, Optimieren und Testen von Handelstrategien. In Merchant of Venice können fortgeschrittene Anwender eigene Strategien in Form mathematischer Formeln erstellen und diese mittels historischer Kursdaten testen. Für Trader, die noch über keine Strategie verfügen, kann das Tool mittels genetischer Algorithmen mögliche Handelstrategien erzeugen. Die Qualität einzelner Strategien kann anhand historischer Kursdaten ausgewertet werden. Die Optimierung von Handelstrategien dient der Profitmaximierung. Genetische Algorithmen finden auch hier Einsatz. Merchant of Venice generiert mittels GA mögliche Parameter in einem Optimierungsproblem und wertet diese ebenfalls mit historischen Kursdaten aus. Hier muss man jedoch erwähnen, dass die Rechenzeit bei einem komplexen Problem mit einem großen Lösungsraum extrem lang ist. Aus diesem Grund stellt sich hier die Frage ob Merchant of Venice als Trading-Tool für Privat Investoren mit aktuellen Heim PCs hinreichend benutzbar ist.

MultiCharts bietet ebenfalls die Funktionalität zum Erstellen, Optimieren und Testen von Handelstrategien. Anders als bei Merchant of Venice hat der Benutzer nicht die Möglichkeit zufällige Strategien vom Tool erzeugen zu lassen. Der Benutzer kann seine gewünschten Handelstrategien aus einer Liste der im System enthaltenen Strategien auswählen und diese bei Bedarf zusammenführen. MultiCharts bietet zwei Ansätze zum Optimieren von Handelstrategien. Bei der erschöpfenden Suche wird der gesamte Lösungsraum nach einem bzw. mehreren Optima abgesucht. Die Optimierung mittels



GA ist eine neuere und innovativere Optimierungsmethode, die bei einem größeren Lösungsraum vorteilhafter ist. Das Testen von erzeugten Handelstrategien wird ebenfalls mittels Backtesting durchgeführt. Jedoch bietet MultiCharts ein umfangreicheres und somit komplexeres Reporting von Testergebnissen. Eine Extrafunktionalität des Tools ist die 3D Darstellung von Testergebnissen. Der erzeugte 3D-Graph soll laut MultiCharts eine einfache und übersichtliche Darstellung der optimalen Parameter-Konfiguration abbilden.

### **5.1.5 Portfolio-Management und Backtesting**

Sowohl Merchant of Venice als auch MultiCharts unterstützen das Anlegen und Verwalten von Wertpapierdepots. Merchant of Venice trennt zwischen Geld-Konten und Wertpapierdepots. Transaktionen können sowohl in den Geld-Konten als auch Wertpapierdepots simuliert werden. Unterstützte Transaktionsformen sind Kauf/Verkauf Order, Deposit Transaktion, Dividenden Transaktion, Dividenden Re-investment Transaktion, Spesen bzw. Gebührenabrechnungen, Zinszahlungen, Geldtransfer zwischen Geld-Konten und Geldabhebungen. Getätigte Transaktionen werden in einer Transaktionshistorie gespeichert und können vom Benutzer jederzeit in Tabellenform sowie in Diagrammform angezeigt werden.

MultiCharts bietet neben dem Anlegen und Verwalten von Portfolios zusätzliche Portfolio-Management Funktionalitäten. Eine Besonderheit des Tools ist das Portfolio-Backtesting. Hier hat der Benutzer die Möglichkeit die Ergebnisse einer Handelstrategie auf unterschiedlichen Portfolios zu vergleichen. Umgekehrt ist es ebenso möglich, die Resultate unterschiedlicher Handelstrategien auf exakt denselben Portfolios gegenüber zu stellen. MultiCharts betont die Wichtigkeit von Backtesting mit unterschiedlichem Inputs (Anlagen) und unterschiedlichen Strategien. Nur so kann man die Qualität einer Strategie in Bezug auf Profitabilität feststellen.

## 5.2 Diskussion der Vor- und Nachteile

Nach einer ausführlichen Untersuchung beider Trading-Tools in den Kapiteln 3 und 4 werden nun die technisch-strategischen Vor- bzw. Nachteile beider Tools näher beleuchtet.

### 5.2.1 Vor-/Nachteile von Merchant of Venice

Merchant of Venice sticht durch mehrere Besonderheiten heraus:

- **Keine Kosten:** Merchant of Venice ist kostenlos. Somit ist es für Investment Anfänger oder Studenten, die (noch) nicht bereit sind eine größere Summe in Trading-Tools zu investieren, bestens geeignet.
- **Analysierbarkeit:** Aufgrund der GNU GPL Lizenzierung ist der Sourcecode und die Dokumentation von Merchant of Venice für alle zugänglich und somit einfach analysierbar.
- **Modifizierbarkeit:** Eine weitere Besonderheit ist die Modifizierbarkeit des Programms durch die zur Verfügungsstellung der API. Dadurch können fachkundige Anwender das Programm gezielt erweitern und eventuelle Fehlerquellen beheben.
- **Plattform-Unabhängigkeit:** Merchant of Venice wurde in Java implementiert. Das Tool kann auf unterschiedlichen Systemen mit installierter Java Laufzeit Umgebung verwendet werden.
- **Installierbarkeit:** Für Windows Benutzer wird eine ausführbare Datei angeboten. Mit geliefert wurde ebenfalls eine schrittweise Installationsanleitung. Für Unix Benutzer wird eine tar.gz Datei angeboten. Diese beinhaltet ebenfalls eine für Unix spezifische Installationsanleitung für die Java Laufzeit Umgebung und zur Ausführung notwendiger Pakete.
- **Erlernbarkeit und Bedienbarkeit:** Durch die einfache Gestaltung und übersichtlichen Funktionen des Tools, ist die Bedienung des Programms einfach erlernbar. Eine Dokumentation im HTML Format wird bei der Installation mitgeliefert und ist auch online verfügbar.

Neben diesen technisch-strategischen Vorteilen existieren auch eine Menge Nachteile:

- **Performanz:** Jede Programmiersprache hat ihre Vor- und Nachteile. Merchant of Venice wurde zur Gänze in Java programmiert. Die

plattform-unabhängige Architektur von Java wird durch die Inbetriebnahme der Java Virtual Maschine (JVM) realisiert. Die JVM verbraucht wie jeder anderen Prozess verfügbare Rechenleistung. Aus diesem Grund kann es bei Java Programmen zu Performanz Engpässen kommen [KCSL00] [Ach03]. Durch die Verbesserung der JVMs, dem Einsatz von Just-in-Time-Compilern und neuen leistungsstarken Prozessoren wurden die Performanz Defizite minimalisiert. In Rahmen der Untersuchungen konnten bei Merchant of Venice jedoch starke Performanz Engpässen festgestellt werden.

- **Stabilität:** Während der Untersuchung ist das Programm nach längerer Laufzeit mehrmals abgestürzt. Daten, die nicht gespeichert wurden, sind somit verloren. Das Programm bietet keine Wiederherstellungsfunktion nach einem Absturz.
- **Reife:** Bei der aktuellen Version des Programms handelt es sich um 0.71 beta, welche im Februar 2007 veröffentlicht wurde. Seitdem wurden keine Verbesserungen und Erweiterungen veröffentlicht. Laut der Projekt Homepage [LM09] sind zwar viele durchaus sinnvolle Erweiterungen angedacht. Es steht jedoch in Frage ob diese jemals umgesetzt werden.

### 5.2.2 Vor-/Nachteile von MultiCharts

MultiCharts ist ein professionelles Trading-Tool, das eine große Palette an Funktionen bietet. Jede Funktion ist sehr durchdacht implementiert und sticht durch verschiedene Besonderheiten heraus. Das Programm versucht unterschiedlichen Kundenwünschen und Anforderungen gerecht zu werden. Gleichzeitig ist das Programm sowohl für Einsteiger als auch für professionelle Trader geeignet. Der Einsteiger bekommt zahlreiche vorprogrammierte Signale und Strategien zur Verfügung gestellt, die ad-hoc einsetzbar sind. Der professionelle Trader kann nach Belieben in der Code Basis schnell und effizient seine Anforderungen umsetzen. Die Vorteile des Tools lassen sich so zusammenfassen:

- Große Anzahl an unterstützten Kursdatenquellen und Kursdatenformaten
- Hoch granulare Datenspeicherung
- Objekt-orientierte Benutzeroberfläche
- Vielzahl an unterstützten Indikatoren und Signalen

- Modifizierbarkeit durch zur Verfügungsstellung des PowerLanguage Editors
- Bereitstellung von Benutzer-Notifikationen in verschiedenen Kommunikationskanälen
- Innovative Portfolio-Backtesting-Funktionen
- Automatisierter Handel durch Anbindung an unterschiedliche Online-Broker

Die Nachteile des Tools entstehen durch den Tradeoff zwischen dem breiten Spektrum an Funktionalität, dem Einsatz von innovativen Technologien, der Benutzbarkeit und Effizienz. Der größte Nachteil in Bezug auf die objekt-orientierte Benutzeroberfläche und die Vielzahl an Funktionen ist die Erlernbarkeit und die Bedienbarkeit. Aufgrund der Innovativität muss ein neuer Benutzer vor dem Einsetzen die Bedeutungen der Buttons bzw. Fenster und die Interaktion mit dem Tool erlernen. Es ist eine Gewöhnungsphase notwendig, bevor die Oberfläche eine gewisse Intuitivität erlangt. Das Tool stellt dem Benutzer in dieser Hinsicht auch ausreichend Informationsmaterialien sowohl online als auch mitgeliefert zur Verfügung. MultiCharts ist auf ein Hauptprogramm und diverse Unterprogramme aufgeteilt. Während das Hauptprogramm ausreichend dokumentiert ist, fehlt bei den Unterprogrammen teilweise bzw. zur Gänze die Dokumentation. Die proprietäre Lizenz bringt auch Nachteile mit sich. Anders als bei der GNU Lizenz ist hier der Source Code des Tools nicht zugänglich. Somit leidet auch die Analysierbarkeit des Tools darunter. Hohe Kosten für die Lizenz sind ein weiterer negativer Punkt. Kosten müssen hier insofern erwähnt werden, da jeder Trader die Maximierung des Gewinns anstrebt. Somit muss jeder Trader selbst bestimmen, wie viel Mehrgewinn der Einsatz eines proprietären Trading-Tools erzielen kann und ob der Aufwand gerechtfertigt ist.

## 5.3 Vergleich der Ergebnisse aus der Performanz-Analyse

Es wurden in Merchant of Venice und in MultiCharts verschiedene auf GA-basierende Optimierungen durchgeführt. Die Ergebnisse werden in Folge besprochen und mit einander verglichen.

### 5.3.1 Kauf/Verkauf Regel Optimierung mittels GA

Merchant of Venice bietet die Funktion der GA-basierten Kauf/Verkauf Regel Optimierung. Bei der Kauf/Verkauf Regel Optimierung werden ausschließlich die Inputparameter der Regeln optimiert. Die Optimierung kann auf ein Portfolio oder auch auf eine einzige Aktie angewendet werden. Der Benutzer kann die Optimierung nur dann anwenden, wenn er über eine intakte und mathematisch korrekte Kauf/Verkauf Regel verfügt. Eine vergleichbare Funktion bietet auch MultiCharts. Dort wird sie „Portfolio-Optimierung“ genannt. Bei der Optimierung wird auch ausschließlich die Inputparameter der Regeln optimiert. Die Optimierung kann nur auf ein Portfolio angewendet werden. Die Größe des Portfolios ist beliebig gestaltbar. Sprich, der Benutzer kann genauso ein Portfolio aus einer einzigen Aktie anlegen und diese von MultiCharts optimieren lassen.

Die Performanz-Analysen wurden unter unterschiedlichen Marktbedingungen - im wachsenden und im turbulenten Markt - durchgeführt. Der wachsende Markt ist stabil und die Volatilität der Kurse relativ gering. Für die Untersuchung ausgewählte Aktien zeigten eine Volatilität zwischen 34.17 und 52.99 Prozent. Der turbulente Markt hingegen ist instabil und die Volatilität der Kurse sehr groß. Für die Untersuchung ausgesuchte Aktien zeigten eine Volatilität zwischen 64.83 und 167.41 Prozent. Weiters ist zu erwähnen, dass die Untersuchungen ohne die Berücksichtigung der Transaktionskosten durchgeführt wurden. Die Performanz-Analysen bei beiden Tools wurden mit den gleichen bzw. vergleichbaren Konfigurationen durchgeführt. Die ausgesuchten Aktien im wachsenden und im turbulenten Markt sind zu hundert Prozent ident. Zur besseren Übersicht werden die ausgesuchten Aktien hier nochmal tabellarisch aufgeführt (siehe Tabelle 5.1 und 5.2).

Die Kauf/Verkauf Regeln und deren Inputparameter, die in beiden Tools optimiert wurden, sind ebenfalls ident. Tabelle 5.3 zeigt die optimierte Kauf/Verkauf Regeln.

Die Anwendung der genetischen Algorithmen zur Optimierung von Kauf/Verkauf Regeln in Merchant of Venice brachte eine durchschnittliche Performanz von +30 Prozent im wachsenden und +6 Prozent im turbu-

Rang	Name	Branche	Volatilität
1	Fresenius Medical Care	Pharma u. Gesundheitswesen	34.17
2	Beiersdorf AG	Pharma u. Gesundheitswesen	35.50
3	Henkel AG	Haushalts Produkte	39.88
4	Linde AG	Maschinenbau	45.17
5	RWE AG	Energieversorgung	46.47
6	Deutsche Telekom AG	Telekommunikation	46.69
7	Muenchner Rueck	Versicherung	47.18
8	SAP AG	Software	49.15
9	Adidas AG	Textilien und Bekleidung	49.41
10	Deutsche Lufthansa AG	Fluggesellschaft	52.99

Tabelle 5.1: 10 ausgewählte Aktientitel aus dem DAX für den wachsenden Markt

Rang	Name	Branche	Volatilität
1	K+S Aktiengesellschaft	Rohstoffe	167.41
2	Volkswagen AG	Automobil Industrie	120.77
3	Commerzbank AG	Bankensektor	107.80
4	Deutsche Bank AG	Bankensektor	100.67
5	Salzgitter	Maschinenbau	77.82
6	Daimler AG	Automobil Industrie	72.44
7	Allianz SE	Versicherung	71.65
8	Thyssenkrupp AG	Metall- und Bergbau	71.00
9	Metro AG	Einzelhandel	66.17
10	Deutsche Post AG	Luftfracht und Kurierdienste	64.83

Tabelle 5.2: 10 ausgewählte Aktientitel aus dem DAX für den turbulenten Markt

Regel	Formel
Kauf	$rsi(LE\ length) < OverSold$
Verkauf	$rsi(SE\ length) > OverBought$
Parameter	Wert
LE length	7 - 14
OverSold	22 - 27
SE length	10 - 14
OverBought	42 - 62

Tabelle 5.3: In Merchant of Venice und in MultiCharts optimierte Kauf/-Verkauf Regeln

lenten Markt. In MultiCharts konnte im wachsenden Markt ein Gewinnzuwachs von +13.75 bis hin zu +209.78 Prozent erzielt werden. Tabelle 5.4 stellt die Ergebnisse der Untersuchungen im wachsenden Markt in einen direkten Vergleich. Beide Tools brachten in der Analyse positive Ergebnisse. Nun werden die Ergebnisse der Untersuchungen der Wachstumsrate des DAX gegenübergestellt. Abbildung 5.1 zeigt die Kursentwicklung des DAX im Zeitraum 2006 bis 2008. Man kann einen deutlichen Aufwärtstrend erkennen. In diesem Zeitraum wuchs der Index um knappe 50 Prozent. Die Ergebnisse von Merchant of Venice sind verglichen mit dem Wachstumsrate vom DAX nicht zufriedenstellend.

Performanz in % von 01/01/2006 bis 01/01/2008		
Merchant of Venice	MultiCharts	<b>DAX</b>
+30.32 bis +32.08	+13.75 bis +209.78	+50.0

Tabelle 5.4: Vergleich der Ergebnisse im wachsenden Markt

Im turbulenten Markt konnte ein Gewinnzuwachs von +21.22 bis hin zu +245.54 Prozent erzielt werden. Tabelle 5.5 stellt die Ergebnisse der Untersuchungen im turbulenten Markt im direkten Vergleich dar. Nun werden die Ergebnisse wieder mit der Kursentwicklung des DAX verglichen. Im Zeitraum 2008 bis 2009 verlor der DAX fast 50 Prozent. In der Abbildung 5.2 erkennt man eine rapide Abwärtstrend. Die Ergebnisse der Optimierungen von Merchant of Venice und MultiCharts sind verglichen mit dem Kursverlauf des DAX (von -50 Prozent) durchaus positiv. MultiCharts schneidet an dieser Stelle besser ab.

Performanz in % von 01/01/2008 bis 01/01/2009		
Merchant of Venice	MultiCharts	<b>DAX</b>
+5.49 bis +7.97	+21.22 bis +245.54	-50.0

Tabelle 5.5: Vergleich der Ergebnisse im turbulenten Markt

Wenn man an dieser Stelle die optimalsten Lösungen von Merchant of Venice in MultiCharts in gleichen Zeiträumen (01/01/2006 bis 01/01/2008 und 01/01/2008 bis 01/01/2009) einsetzt, ergäbe es im wachsenden Markt einen Gewinnzuwachs von +140.17 Prozent und im turbulenten Markt einen Verlust von -1.04 Prozent. Der Grund für den Unterschied der Performanz, obwohl mit gleichen Kursdaten und den gleichen Regeln getestet wurde, ist die Unterstützung von „Short Selling“ (Aktienleerverkauf) in MultiCharts. Beim „Short Selling“ wird mit ausgeliehenen Aktien gehandelt (vgl. [Pfe06]). Sprich, der Anleger borgt sich zum Beispiel von seinem Broker x Anteile



Abbildung 5.1: DAX im Zeitraum 2006 bis 2008



Abbildung 5.2: DAX im Zeitraum 2008 bis 2009



einer Aktie zum Preis  $y$  und verkauft diese  $x$  Anteile an der Börse. Vor dem vereinbarten Zeitpunkt muss der Anleger die geborgten Aktien wieder zurückkaufen und seinem Broker zurückverbuchen. Bei einem gesunkenen Aktienkurs, profitiert der Anleger. Umgekehrt bei einem gestiegenen Kurs muss er die Aktien zu einem höheren Preis zurückkaufen und macht somit Verluste. Die Unterstützung von Leerverkäufe in MultiCharts ist maßgeblich für die Performance Werte von bis zu +209 Prozent verantwortlich und ist auch mit enormen Risiken verbunden.

### 5.3.2 Kauf/Verkauf Regel Generierung mittels GP

Merchant of Venice bietet eine weitere Funktion an, die MultiCharts nicht bietet. Der Benutzer kann in Merchant of Venice Kauf/Verkauf Regeln mittels genetischer Algorithmen automatisiert generieren. Der Algorithmus stellt aus einer Liste von implementierten Indikatoren und Signalen verschiedene Kauf/Verkauf Regeln zusammen und wertet mögliche Gewinne aus. Es wurden ebenfalls Untersuchungen mit denselben Aktien und denselben Zeiträumen durchgeführt. Im wachsenden Markt konnte der Algorithmus ein Gewinnzuwachs von +96.97 Prozent erzielen. Im turbulenten Markt betrug der Gewinnzuwachs +152.20 Prozent. In der Tabelle 5.6 sieht man die Ergebnisse beider Auswertungen auf einen Blick. Die Performanz der Kauf/-Verkauf Regel Generierung ist wesentlich besser als die der Kauf/Verkauf Regel Optimierung. Die Ergebnisse einer Kauf/Verkauf Regel Optimierung hängen stark von der Qualität der zu optimierenden Regel ab. Wenn die gewählte Regeln zu einfach oder der Lösungsraum zu klein ausgerichtet ist, wird die Optimierung die Ergebnisse nicht maßgeblich verbessern können.

	Performanz in % 01/01/2006 bis 01/01/2008	Performanz in % 01/01/2008 bis 01/01/2009
Merchant of Venice	+96.97	+152.20
<b>DAX</b>	+50.0	-50.0

Tabelle 5.6: Ergebnisse der Kauf/Verkauf Regel Generierung im wachsenden und im turbulenten Markt

An dieser Stelle muss man hinterfragen, ob die Ergebnisse in einem Realfall wirklich erzielbar sind. Die Papertests haben gezeigt, dass beide optimal generierten Kauf/Verkauf Regeln im Zeitraum 01/01/2009 bis 30/04/2009 einen Gewinn von +1.46 Prozent und einen Verlust von -17.83 Prozent verzeichnet hätten. In beiden Fällen hätten sich die Investitionen nicht gelohnt.

Wenn man die Transaktionskosten mit einbezieht, hat man in beiden Fällen einen negativen Return-on-Initial-Capital.

### 5.3.3 Analyse der Ergebnissen zu den empirischen Versuchen

Zusätzlich zu den Performanz-Analysen wurden empirische Versuche durchgeführt. Diese sollten vorrangig die Funktionen der Tools veranschaulichen und desweiteren die Wirkung von unterschiedlichen GA-Parametern untersuchen.

#### 5.3.3.1 Versuch in Sektion 3.2.1.3

In der Sektion 3.2.1.3 wurden empirische Versuche zur GA basierten Kauf/-Verkauf Regel Optimierung durchgeführt. Die GA-Parameter wurden in der Tabelle 5.7 zusammengefasst. Setting 1 umfasst weniger Generationen, eine kleinere Population und keine Mutationswahrscheinlichkeit. Setting 2 wurde so gewählt, dass hier mehr Generationen erzeugt werden sollten, mit einer viel größeren Population und 55 Prozent Mutationswahrscheinlichkeit. Die Regeln wurden anschliessend mit beiden Settings im Zeitraum 01/01/2008 bis 01/01/2009 optimiert. Interessanterweise war die Lösung mit beiden Settings identisch.

GA-Parameter	Setting 1 Wert	Setting 2 Wert
Generations	10	50
Population	100	250
Breeding Population	10	30
Randomness	0	55

Tabelle 5.7: Vergleich der GA-Parameter-Settings

Mit den optimierten Regeln wurden noch weitere Versuche durchgeführt. Diesesmal wurden die optimierten Regeln auf unbekanntem Daten aus dem Zeitraum 01/01/2009 bis 30/04/2009 untersucht. Tabelle 5.8 fasst die Ergebnisse zusammen. In der 2. Spalte sieht man die Performanz der optimierten Regeln auf bekannte Daten. In der letzten Spalte sieht man die Performanz der Regeln auf unbekanntem Daten. Die optimierten Regeln würden in diesem Fall versagen und Verluste einbringen.

Es werden weitere Versuche mit Regeln durchgeführt, die im Zeitraum 01/04/2003 bis 01/01/2009 optimiert wurden. Sie wurden ebenfalls auf unbekanntem Daten aus dem Zeitraum 01/01/2009 bis 30/04/2009 untersucht.

Versuch	Performanz in % 01/01/2008 bis 01/01/2009	Performanz in % 01/01/2009 bis 30/04/2009
Paper Trade 1	+11.45	-0.66
Paper Trade 2	+10.81	-2.89
Paper Trade 3	+9.63	-2.89
Paper Trade 4	+7.75	-2.62
Paper Trade 5	+6.94	-2.62
<b>DAX</b>	+50.0	-5.0 bis -25.0

Tabelle 5.8: Ergebnisse der empirischen Versuche im Zeitraum 01/01/2008 bis 01/01/2009 und im Zeitraum 01/01/2009 bis 30/04/2009 mit unbekanntem Daten

Tabelle 5.9 fasst die Ergebnisse zusammen. Die Regeln scheinen in dem Fall besser optimiert zu sein und würden sogar Gewinne auf neuen, unbekanntem Daten einbringen. Die Ergebnisse sind insofern ertragreich, da der DAX im Zeitraum 01/01/2009 bis 30/04/2009 einen Verlust von -5.0 bis -25.0 Prozent verzeichnet.

Versuch	Performanz in % 01/04/2003 bis 01/01/2009	Performanz in % 01/01/2009 bis 30/04/2009
Paper Trade 6	+251.51	+7.71
Paper Trade 7	+245.73	+7.71
Paper Trade 8	+244.17	+7.87
Paper Trade 9	+243.90	+7.71
Paper Trade 10	+242.33	+7.87
<b>DAX</b>	+100.0 bis +250.0	-5.0 bis -25.0

Tabelle 5.9: Ergebnisse der empirischen Versuche zur GA-basierten Kauf/-Verkauf Regel Optimierung im Zeitraum 01/04/2003 bis 01/01/2009 und im Zeitraum 01/01/2009 bis 30/04/2009 mit unbekanntem Daten

### 5.3.3.2 Versuch in Sektion 3.2.2.3

Es wurden auch empirische Versuche durchgeführt, in denen GA generierte Kauf/Verkauf Regeln an unbekanntem Datensätzen getestet wurden. Diese Versuche sollten Hinweise geben, wie gut bzw. wie schlecht die generierten Regeln in einem realen Szenario abschneiden. Die Ergebnisse wurden

in der Tabelle 5.10 zusammengefasst und der Kursentwicklung des DAX gegenübergestellt.

Versuch	Performanz in % 01/04/2003 bis 01/01/2009	Performanz in % 01/01/2008 bis 01/01/2009	Performanz in % 01/01/2009 bis 30/04/2009
Regel #1	n.a	+206.68	n.a
Regel #2	+593.79	n.a	-0.58
Regel #3	n.a	+243.07	+7.25
<b>DAX</b>	+100.0 bis +250.0	+50.0	-5.0 bis -25.0

Tabelle 5.10: Ergebnisse der empirischen Versuche zur GP basierten Kauf/Verkauf Regel Generierung in den Zeiträumen 01/04/2003 bis 01/01/2009 und 01/01/2008 bis 01/01/2009 und im Zeitraum 01/01/2009 bis 30/04/2009 mit unbekanntem Daten

Die Performanz Werte sind hier außergewöhnlich hoch. Anders als bei MultiCharts unterstützt Merchant of Venice kein Short Selling. Es ist jedoch fraglich ob diese Werte nicht die Ergebnisse der Überoptimierung sind und in einem Realfall wirklich eingetreten wären.

## Kapitel 6

# Conclusio

Die Untersuchungen der auf GA-basierenden Trading-Tools und die Vergleiche der Tools und der Ergebnisse aus den Performanz-Analysen zeigten viele technisch-strategische Vorteile von Genetischen Algorithmen im Rahmen von Investmentgeschäften. Genetische Algorithmen zeigen Potentiale, besitzen aber gleichzeitig ihre technisch-mathematischen Grenzen. Zunächst werden die Untersuchungsergebnisse interpretiert. Als Abschluss der Arbeit werden die Grenzen der genetischen Algorithmen genannt.

## 6.1 Interpretation der Untersuchungsergebnisse

In Sektion 5.1 wurden die Funktionalitäten beider Tools mit einander verglichen. An dieser Stelle werden sie in tabellarischer Form (siehe Tabelle 6.1) zusammengefasst. Tabelle 6.2 fasst die Vor- und Nachteile beider Tools aus Sektion 5.2 zusammen.

Funktion	Merchant of Venice	MultiCharts
Kursdaten Import/Export	Unterstützung für - wenige Kursdatenformate - wenige Datenquellen keine Unterstützung für - Real Time Kursdaten (Ticks)	Unterstützung für - allen gängigen Formate - umfangreiche Datenquellen - Real Time Kursdaten
Grafische Oberfläche	Darstellung der Daten in Diagrammen und Tabellen	innovative GUI
Analyse Techniken	vorhanden	vorhanden über hundert Methoden
Erzeugen von Handelsstrategien	basierend auf GP	nicht vorhanden
Optimieren von Handelsstrategien	basierend auf GA	basierend auf GA und erschöpfende Suche
Testen von Handelsstrategien	Paper Trading Reporting in Listenform	vorhanden umfangreiches Reporting
Portfolio Management	Portfolio-Backtesting unterstützt Transaktionen	Portfolio-Backtesting
Systemanforderung	JAVA, ansonsten keine bekannt  Unix Derivate alle gängige Windows Versionen Speicherplatz 3.7MB	1 GHz, 256MB RAM, 1024x768 Auflösung, Windows 2000/XP SP2/XP x64, Windows Vista/Vista x64 Speicherplatz 200MB
Lizenz	GPL	kommerziell kostenpflichtig

Tabelle 6.1: Vergleich der Funktionalitäten von Merchant of Venice und MultiCharts

In Sektion 5.3 wurden die Ergebnisse aus allen durchgeführten Untersuchungen von beiden Tools tabellarisch zusammengefasst und mit einander verglichen. Merchant of Venice schneidet in vieler Hinsicht anders ab als MultiCharts. Die Hauptursache liegt in der Unterstützung von Short Selling in MultiCharts. Weitere Ursachen können in den unterschiedlichen Einstel-

	Merchant of Venice	MultiCharts
Vorteile	Analysierbarkeit Modifizierbarkeit Modifizierbarkeit Plattform Unabhängigkeit Installierbarkeit Erlernbarkeit Keine Kosten GPL Lizenz	Innovative und objekt-orientierte GUI Vielzahl an unterstützten Datenquellen und Formate Modifizierbarkeit Vielzahl an unterstützten Indikatoren und Signalen Installierbarkeit Benutzer-Notifikation Anbindung an Online-Broker umfangreiche Backtesting und Reporting Funktion
Nachteile	Performanz Stabilität Reife	Erlernbarkeit Bedienbarkeit Komplexität Hohe Kosten Fehlen von Dokumentation

Tabelle 6.2: Vor- und Nachteile von Merchant of Venice und MultiCharts

lungsmöglichkeit beider Tools liegen. Wichtige Variablen, die die Ergebnisse beeinflussen können:

- der Zeitpunkt zu dem der Kauf/Verkauf getätigt werden soll.
- die Limitierung des Transaktionsvolumen (maximale Transaktionshöhe)

Der Zeitpunkt, zu dem eine Transaktion durchgeführt wird, kann die Höhe des Gewinns stark verändern. Aktienkurse fluktuieren sehr stark innerhalb eines Tages. Diese Schwankungen werden im Tageskurs einer Aktie festgehalten. Die Tageskurse beinhalten unterschiedliche Informationen: Tageshoch, Tagestief, Eröffnungspreis und Schlusskurs. In Merchant of Venice kann der Benutzer den Zeitpunkt der Transaktion selbst bestimmen. Man kann zu einem Tageshoch bzw. Tagestief, einem Durchschnittskurs, zum Schlusskurs oder auch zum Eröffnungspreis an dem darauf folgenden Börsentag eine Transaktion durchführen lassen. Das ist ein möglicher Grund, weshalb Merchant of Venice andere Optimierungsergebnisse liefert als MultiCharts.

Eine Limitierung des Transaktionsvolumen soll mögliche Risiken verkleinern. Niedrige Transaktionsvolumen beschränken einerseits mögliche Ver-

luste, andererseits aber auch mögliche Gewinne. Ein kurzes Rechenbeispiel: Initialkapital liegt bei 100,000 Euro, Transaktionshöhe bei 10,000 Euro. Die Verkauf Regel entscheidet, dass die Aktie bei einem Preis von 30 Euro je Anteil verkauft wird. Der Gewinn, wenn die Transaktionshöhe auf 10,000 Euro limitiert ist, beträgt bei 20 Euro Kaufpreis 3,330 Euro. Bei einer Transaktionshöhe von 100,000 Euro würde der Netto Profit 33,333 Euro betragen. Das ist der Grund weshalb in MultiCharts die Ergebnisse um mehr als 200 Prozent divergieren (siehe Tabelle 5.4 und 5.5).

In MultiCharts gibt es eine weitere Konfigurationsvariable „Potential loss per contract“. Diese wird per Default auf 5 Prozent limitiert. Sprich, es werden nur Transaktionen durchgeführt wenn der Verlust dabei weniger als 5 Prozent beträgt. Jede Verkauf Regel kann auch in einer Verlustsituation einen Verkauf durchführen. Meistens hat es den Hintergrund, größere Verluste zu vermeiden, indem eine Position zur Gänze oder zu einem Teil vor einem möglichen Kursabsturz verkauft wird. Diese Variable hat auch Einflüsse auf die Optimierungsergebnisse in MultiCharts.

Weitere mögliche Gründe für die unterschiedlichen Ergebnisse von Merchant of Venice und MultiCharts sind einerseits unterschiedliche Implementierung der GA und andererseits unterschiedliche Konfigurationsmöglichkeit der GA spezifischen Parameter. In Merchant of Venice kann der Benutzer lediglich die Anzahl der Generationen, die Größe der Population, die Größe der Breeding Population und die Mutationswahrscheinlichkeit einstellen. In MultiCharts hat der Benutzer die Möglichkeit den Algorithmus spezifisch zu ändern. Man hat die Wahl zwischen zwei Subtypen, verschiedenen Ersetzungsregeln und zwei Selektionsalgorithmen<sup>33</sup>. Diese Details des implementierten Algorithmus haben große Auswirkungen auf die Optimalität der Ergebnisse.

---

<sup>33</sup>In den Grundlagen (siehe Sektion 2.1.2) ist beschrieben, welche verschiedene Selektionsalgorithmen innerhalb eines genetischen Algorithmus implementiert sein können.



## 6.2 Die Grenzen der genetischen Algorithmen

Viele Forscher sehen einen breiten Einsatzbereich von GA in der Lösung von Optimierungsproblemen bzw. Optimierungsaufgaben (siehe [JTM07], [KSSG96], [LS05], [Sch65], [Gol89], [LCWZ04], [Vac97], [FGLG06]). Es werden im Zuge dessen viele Vorteile der GA gegenüber traditionellen Lösungsansätzen genannt. Wirtschaftliche Vorteile der GA sind zum Beispiel die kürzere Laufzeit bzw. Einsparungen beim Rechenaufwand bei NP-Problemen wie etwa dem Traveling Salesman Problem (TSP) (vgl. [Wri91] und [CLP98]). Dies wird erreicht indem "schlechte" Lösungen bei der Berechnung nicht weiter verfolgt, sondern sofort gelöscht werden. Hier kommen enorme Zeit Einsparungen zusammen, weil nicht die gesamte Lösungsmenge ausgewertet werden muss. Der technische Vorteil der GA liegt im geringen Implementierungsaufwand, welcher auf das einfache und verständliche Konzept der Technologie zurückzuführen ist [GC00].

An dieser Stelle fragt man sich als Forscher und Entwickler: Sind genetische Algorithmen ein Allheilmittel? Haben Sie auch ihre Grenzen? Wo liegen ihre Grenzen? Darf man den Lösungen, die ein Algorithmus produziert, stets vertrauen?

Um diese Fragen beantworten zu können, muss zunächst der Terminus Optimierungsproblem ansich betrachtet werden. Bei einem Optimierungsproblem ist ein Lösungsraum  $\Omega$  und eine Ziel Funktion (auch Bewertungsfunktion)  $f : \Omega \rightarrow \mathbb{R}$  gegeben [MG00]. Ziel ist es, die beste Lösung  $x \in \Omega$  mit einem möglichst großen Wert  $f(x)$  zu finden.

Der größte Nachteil der genetischen Algorithmen nach Kopfer [Kop92] und Harbich [Har07] besteht darin, dass das Ergebnis nicht unbedingt ein globales Optimum sein muss. Eine gefundene Lösung kann ein globales Optimum, aber auch ein lokales Optimum sein. Im ungünstigsten Fall kann keine Lösung gefunden werden, weil der Algorithmus vorzeitig abgebrochen wurde. Ein weiterer wichtiger Faktor ist die Tatsache, dass sich die Optimalität nicht ohne weiteres bestimmen lässt (vgl. [Kop92]). Harbich erklärt die Wirkungsursache für das Phänomen auf folgender Art und Weise:

„Dies passiert, wenn die Chromosomen der Population sich einem lokalen Optimum der Ziel Funktion nähern und die genetischen Operatoren nicht ausreichend starke Änderungen hervorbringen, um die suboptimale Lösung zu überwinden. Auch funktionieren GAs effizient unter der Annahme, dass ähnliche Lösungen zu einem ähnlich gutem Ergebnis führen, was nicht unbedingt gegeben sein muss.“

*Einführung genetischer Algorithmen mit Anwendungsbeispiel [Har07]*

Dieses Phänomen soll aber nicht als ein Fehlverhalten von genetischen Algorithmen verstanden werden. Denn eine lokal optimierte Lösung ist besser als keine Lösung (siehe [CLP98]).

Ein weiterer Nachteil der GA, den die Untersuchungen mit Merchant of Venice und MultiCharts aufgezeigt hatten, ist die starke Abhängigkeit der Ergebnisse in Relation zu der gewählten Parameter-Konfiguration. Sowohl in Merchant of Venice als auch in MultiCharts war die Optimalität der Lösungen sehr stark abhängig von den GA-Parametern wie zum Beispiel der Größe der Population, dem Mutationsfaktor oder der Anzahl der Generationen. Je größer die Population und je mehr Generationen der Algorithmus durchlaufen soll, desto größer war der benötigte Zeitaufwand. Kleinere Population und zu wenige Generationen hingegen, haben stets schlechtere Resultate geliefert. Aus diesem Grund ist es sehr wichtig die „richtigen“ Parameter zu einem Problem zu finden. Und das ist wieder ein Problem ansich. Choy et Al. präsentiert in [CLP98] drei Ansätze das Problem der Parametersuche zu lösen und mitunter die Performanz von genetischen Algorithmen noch mehr zu verbessern. Im ersten Ansatz wird das Konzept der GA um einen Aspekt erweitert. Die Einbettung von adaptiven Mechanismen soll einem Algorithmus bewusst machen, welche Parameterwerte bessere bzw. optimalere Resultate liefern. Der zweite Ansatz involviert die Optimierung von statischen Parameterwerten<sup>34</sup>. Der letzte Ansatz beinhaltet die Einbettung von Fuzzy Logik um menschliches Eingreifen betreffend der Parameter-Konfiguration zu minimieren. Weder MultiCharts noch Merchant of Venice unterstützen diese Ansätze. Alle drei Ansätze sind im Bereich der Trading-Tools durchaus anwendbar und würden die Grenzen der GA erweitern.

Durch die Untersuchungen aus Sektion 3 und 4 konnten viele wertvolle Informationen über den Einsatz der GA in Trading-Tools gewonnen werden. Genetische Algorithmen könnten sowohl in einem stabilen, wachsenden Markt als auch in einem turbulenten Markt Gewinne erzielen. Jedoch darf man an dieser Stelle die Problematik der Überoptimierung nicht vergessen. Ferner muss man auch den Standpunkt betrachten, dass technische Analysen ausschließlich mit historischen Kursdaten arbeiten und alle relevanten wirtschaftlichen Daten bzw. Umstände vernachlässigen. Somit kann die Anwendbarkeit der technischen Analysen in Frage gestellt werden.

Die Autorin fand die Untersuchung von Merchant of Venice und MultiCharts durchaus interessant. Nach einer gründlichen Evaluationsphase würde sie beim professionellen Handeln auf ein Trading-Tool dieser Art zurückgreifen. Ihren Anforderungen entsprechend, zieht sie eindeutig Mul-

---

<sup>34</sup>Statische Parameter sind z.B. Die Mutationsrate und die Größe der Population

tiCharts von den beiden Tools vor. Gründe für MultiCharts sind: bessere Performanz, schnellere Rechenzeit, kein Absturz in der Evaluation, mehr Features, mehr Analysemethoden und besseres Datenmanagement.

# Literaturverzeichnis

- [Ach03] ACHER, G.: *JIFFY-ein FPGA-basierter Java Just-in-time-Compiler für eingebettete Anwendungen*, Technische Universität München, Universitätsbibliothek, Diss., 2003
- [AK94] ALLEN, F. ; KARJALAINEN, R.: Using genetic algorithms to find technical trading rules. (1994)
- [AK99] ALLEN, F. ; KARJALAINEN, R.: Using genetic algorithms to find technical trading rules. In: *Journal of Financial Economics* 51 (1999), Nr. 2, S. 245–271
- [AMI09] AMIBROKER.COM: *AmiBroker*. <http://www.amibroker.com/>. Version: 2009. – Zugriff am 15.01.2009
- [Ari95] ARIFOVIC, J.: Genetic algorithms and inflationary economies. In: *Journal of Monetary Economics* 36 (1995), Nr. 1, S. 219–243
- [Ari96] ARIFOVIC, J.: The Behavior of the Exchange Rate in the Genetic Algorithm and Experimental Economies. In: *Journal of Political Economy* 104 (1996), Nr. 3, S. 510
- [Arn06] ARNIM, M. von: *Erfolgreich mit Aktien!: Alles, was sie über die gängigste Anlageform wissen müssen*. Finanzbuch Verlag GmbH, 2006
- [Art91] ARTHUR, W.B.: Designing economic agents that act like human agents: A behavioral approach to bounded rationality. In: *American Economic Review* 81 (1991), Nr. 2, S. 353–359
- [BB02] BESSLER, W. ; BOOK, T.: Elektronischer Handel versus Präsenzhandel: Eine Untersuchung des Wettbewerbs von Terminbörsen am Beispiel des DM-Bund Future. In: *9th Symposium on Finance, Banking and Insurance, Universität Karlsruhe*, 2002

- [Bil02] BILL, R.W.: *Jython for Java programmers*. Sams, 2002
- [BK09] BAUERT, D.W.B.A.S. ; KOPP, C.: *Algorithmic Trading: Analyse der Transaktionskosten*. GRIN Verlag, 2009
- [BKR<sup>+</sup>00] BANZHAF, W. ; KOZA, JR ; RYAN, C. ; SPECTOR, L. ; JACOB, C.: Genetic programming. In: *IEEE Intelligent Systems and Their Applications* 15 (2000), Nr. 3, S. 74–84
- [BLL92] BROCK, W. ; LAKONISHOK, J. ; LEBARON, B.: Simple technical trading rules and the stochastic properties of stock returns. In: *Journal of Finance* (1992), S. 1731–1764
- [BS03] BECKER, L.A. ; SESHADRI, M.: GP-evolved Technical Trading Rules Can Outperform Buy and Hold. In: *3rd International Workshop on Computational Intelligence in Economics and Finance*. Sept, 2003
- [BT09] BROKER-TEST: [www.Broker-Test.de](http://www.broker-test.de). <http://www.broker-test.de/transaktionskosten-rechner/tools>.  
Version: 2009. – Zugriff am 20.12.2009
- [CLP98] CHOY, KL ; LAM, FW ; PETER, KH: Application of fuzzied genetic algorithms in optimizing parameters in a manufacturing system for resource allocation. (1998)
- [Col00] COLIN, A.: A genetic-programming-based approach to the generation of foreign-exchange trading models. In: *Commerce, Complexity, and Evolution: Topics in Economics, Finance, Marketing, and Management: Proceedings of the Twelfth International Symposium in Economic Theory and Econometrics* Cambridge University Press, 2000, S. 173
- [Dar02] DARWIN, C.: *Origin of Species by Means of Natural Selection, Or the Preservation of Favored Races in the Struggle for Life*. PF Collier, 1902
- [Dav94] DAVIS, L.: Genetic algorithms and financial applications. In: *Trading on the edge* (1994), S. 133–147
- [DHLN99] DUNIS, C. ; HARRIS, A. ; LEONG, S. ; NACASKUL, P.: Optimising intraday trading models with genetic algorithms. In: *NEURAL NETWORK WORLD* 9 (1999), S. 193–224

- [DM02] DRAKE, A.E. ; MARKS, R.E.: Genetic Algorithms in Economics and Finance: Forecasting Stock Market Prices and Foreign Exchange? A Review. In: *Genetic Algorithms and Genetic Programming in Computational Finance* (2002), S. 29
- [Ell96] ELLER, R.: *Handbuch Derivativer Instrumente*. Schäffer-Poeschel, 1996
- [ER06] ECK, C. ; RIECHERT, M.: *Professionelles Eurex Trading: Grundlagen, Strategien und Chancen mit Optionen und Futures*. Finanzbuch Verlag GmbH, 2006
- [FGLG06] FUENTE, David de l. ; GARRIDO, Alejandro ; LAVIADA, Jaime ; GÓMEZ, Alberto: Genetic algorithms to optimise the time to make stock market investment. In: *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*. New York, NY, USA : ACM, 2006. – ISBN 1–59593–186–4, S. 1857–1858
- [FRGMSR01] FERNÁNDEZ-RODRÍGUEZ, F. ; GONZÁLEZ-MARTEL, C. ; SOSVILLA-RIVERO, S.: Optimisation of technical rules by genetic algorithms: Evidence from the madrid stock market. In: *Documento de trabajo* (2001), S. 14
- [GC00] GEN, M. ; CHENG, R.: *Genetic algorithms and engineering optimization*. Wiley-interscience, 2000
- [Gei03] GEIRHOS, H.: *Strategische Wege zum Börsenerfolg: Der Weg zum erfolgreichen Anleger in allen Phasen der Börsenentwicklung!* BoD–Books on Demand, 2003
- [Gey08] *Hält die technische Analyse was sie verspricht?* 2008
- [GKK04] GERDES, I. ; KLAWONN, F. ; KRUSE, R.: *Evolutionäre Algorithmen: Genetische Algorithmen-Strategien und Optimierungsverfahren-Beispielanwendungen*. Vieweg+Teubner Verlag, 2004
- [Gol89] GOLDBERG, David E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 1989. – ISBN 0201157675
- [Göt01] GÖTTE, R.: *Aktien, Anleihen, Futures, Optionen*. Tectum Verlag DE, 2001

- [Gro09] GROUP, Ward S.: *Neuroshell*. <http://www.neuroshell.com/>. Version: 2009. – MD Patent 21,703
- [Gsc91] GSCHIEGL: *Börse von A bis Z*. Signum Verlag, 1991
- [Har03] HARRIS, L.: *Trading and exchanges: Market microstructure for practitioners*. Oxford University Press, USA, 2003
- [Har07] HARBICH, S.: Einführung genetischer Algorithmen mit Anwendungsbeispiel. (2007)
- [Hol75] HOLLAND, J.H.: *Adaption in Natural and Artificial Systems*. University of Michigan Press, 1975
- [Jam00] JAMROZ, M.: Neue Chancen gibt es wie Sand am Meer. In: *Börse Online* 33 (2000), S. 49–52
- [JTM07] JENABI, Masoud ; TORABI, S. A. ; MANSOURI, S. A.: A hybrid GA for a supply chain production planning problem. In: *GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation*. New York, NY, USA : ACM, 2007. – ISBN 978-1-59593-697-4, S. 2045–2052
- [Kab00] KABOUDAN, M. A.: Genetic Programming Prediction of Stock Prices. In: *Comput. Econ.* 16 (2000), Nr. 3, S. 207–236. <http://dx.doi.org/http://dx.doi.org/10.1023/A:1008768404046>. – DOI <http://dx.doi.org/10.1023/A:1008768404046>. – ISSN 0927-7099
- [KCM05] KWON, Yung-Keun ; CHOI, Sung-Soon ; MOON, Byung-Ro: Stock prediction based on financial correlation. In: *GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation*. New York, NY, USA : ACM, 2005. – ISBN 1-59593-010-8, S. 2061–2066
- [KCSL00] KAZI, Iffat H. ; CHEN, Howard H. ; STANLEY, Berdenia ; LILJA, David J.: Techniques for obtaining high performance in Java programs. In: *ACM Comput. Surv.* 32 (2000), Nr. 3, S. 213–240. <http://dx.doi.org/http://doi.acm.org/10.1145/367701.367714>. – DOI <http://doi.acm.org/10.1145/367701.367714>. – ISSN 0360-0300

- [KF95] KINGDON, J. ; FELDMAN, K.: Genetic algorithms and applications to finance. In: *Applied Mathematical Finance* 2 (1995), Nr. 2, S. 89–116
- [KF02] KORN, R. ; FRAUNHOFER, I. Berichte d.: Elementare Finanzmathematik. In: *Fraunhofer Institut für Techno-und Wirtschaftsmathematik ITWM Bericht* 39 (2002)
- [Knö09] KNÖPFEL, A.: *Investox-die Börsensoftware*. <http://www.investox.de/>. Version: 2009. – Zugriff am 15.01.2009
- [Kop92] KOPFER, H.: Konzepte genetischer Algorithmen und ihre Anwendung auf das Frachtoptimierungsproblem im gewerblichen Güterfernverkehr. In: *OR Spectrum* 14 (1992), Nr. 3, S. 137–147
- [KR92] KOZA, J.R. ; RICE, J.P.: *Genetic programming*. Springer, 1992
- [KSS06] KHALIFA, Y. ; SALEM, O. ; SHAHIN, A.: Cutting stock waste reduction using genetic algorithms. In: *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*. New York, NY, USA : ACM, 2006. – ISBN 1–59593–186–4, S. 1675–1680
- [KSSG96] KUMAR, A. ; SRIVASTAVA, A. ; SINGRU, A. ; GHOSH, R. K.: Robust and distributed genetic algorithm for ordering problems. In: *HPDC '96: Proceedings of the 5th IEEE International Symposium on High Performance Distributed Computing*. Washington, DC, USA : IEEE Computer Society, 1996. – ISBN 0–8186–7582–9, S. 253
- [LCWZ04] LIN, L. ; CAO, L. ; WANG, J. ; ZHANG, C.: The applications of genetic algorithms in stock market data mining optimization. In: *Proceedings of Fifth International Conference on Data Mining, Text Mining and their Business Applications*, 2004, S. 273–280
- [Lep07] LEPPARD, Andrew: *Merchant of Venice - Manual*. (2007)
- [Lip67] LIPFERT, H.: *Internationaler Devisen-und Geldhandel*. Knapp, 1967



- [LL95] LIANG, Simon J. T. ; LEWIS, John M.: A sparse matrix representation for production scheduling using genetic algorithms. In: *SAC '95: Proceedings of the 1995 ACM symposium on Applied computing*. New York, NY, USA : ACM, 1995. – ISBN 0-89791-658-1, S. 313–317
- [LM09] LEPPARD, A. ; MAKOVEC, D.: *Merchant of Venice*. <http://mov.sourceforge.net/>. Version: 2009. – Zugriff am 15.01.2009
- [LS05] LIM, Chihoon ; SIM, Eoksu: Production planning in manufacturing/remanufacturing environment using genetic algorithm. In: *GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation*. New York, NY, USA : ACM, 2005. – ISBN 1-59593-010-8, S. 2217–2218
- [LT96] LORASCHI, A. ; TETTAMANZI, A.: An evolutionary algorithm for portfolio selection within a downside risk framework. In: *Forecasting Financial Markets, Series in Financial Economics and Quantitative Analysis* (1996), S. 275–285
- [LTT+95] LORASCHI, A. ; TETTAMANZI, A. ; TOMASSINI, M. ; SVIZZERO, C. ; SCIENTIFICO, C. ; VERDA, P.: Distributed genetic algorithms with an application to portfolio selection problems. In: *Artificial neural nets and genetic* (1995), S. 384–387
- [Mah08] MAHRT, M.: *Open Office. org Base 3.0*. O'Reilly Germany, 2008
- [Mar07] MARRONE, P.: Java object-oriented neural engine (JOONE). In: *Accessed March 13* (2007), S. 2007
- [May08] MAY, H.: *Handbuch zur ökonomischen Bildung*. Oldenbourg Wissenschaftsverlag, 2008
- [MG00] MARTI, K. ; GRÖGER, D.: *Einführung in die lineare und nichtlineare Optimierung*. Physica-Verlag Heidelberg, 2000
- [Mic96] MICHALEWICZ, Z.: *Genetic Algorithms+ Data Structures = Evolution Programs*. Springer, 1996
- [MM96] MAHFOUD, S. ; MANI, G.: Financial forecasting using genetic algorithms. In: *Applied Artificial Intelligence* 10 (1996), Nr. 6, S. 543–565

- [MN98] MÜLLER, T. ; NIETZER, H.: *Das große Buch der technischen Indikatoren*. TM-Börsenverl., 1998
- [Mül07] MÜLLER, M.: *Die Ansätze der modernen Aktienanalyse im Vergleich*. GRIN Verlag OHG, 2007
- [MyS06] MYSQL, AB: *Mysql connector/j*. 2006
- [Neu09] NEURODIMENSION: *TradingSolutions*. [www.tradingsolutions.com/](http://www.tradingsolutions.com/). Version: 2009. – Zugriff am 15.01.2009
- [NWD97] NEELY, C. ; WELLER, P. ; DITTMAR, R.: Is technical analysis in the foreign exchange market profitable? A genetic programming approach. In: *Journal of Financial and Quantitative Analysis* (1997), S. 405–426
- [Pac90] PACKARD, N.H.: A genetic learning algorithm for the analysis of complex data. In: *Complex Systems* 4 (1990), Nr. 5, S. 543–572
- [PB99] PEREIRA, R. ; BUSINESS, La Trobe University. S.: *Forecasting ability but no profitability: An empirical evaluation of genetic algorithm-optimised technical trading rules*. La Trobe University, School of Business, 1999
- [PB00] PEREIRA, R. ; BUSINESS, La Trobe University S.: *Genetic Algorithm Optimisation for Finance and Investment*. La Trobe University, School of Business, 2000
- [Per96] PEREIRA, R.: Selecting parameters for technical trading rules using genetic algorithms. In: *Journal of Applied Finance and Investment* 1 (1996), Nr. 3, S. 27–34
- [Pez90] PEZOLD, C.: *Dynamic Data Exchange (DDE)–Chapter 17, Programming Windows*. 1990
- [PF07] PRESSMAR, D.B. ; FRIEDRICH, S.: Optimierung von Geschäftsprozessen mit Evolutionären Algorithmen. Version: 2007. <http://www.springerlink.com/content/h634137587053771/>. In: *Architekturen und Prozesse*. Springer, Heidelberg, 2007
- [Pfe06] PFEIFER, D.: *Diskrete Stochastische Finanzmathematik*. 2006

- [PH03] PRUITT, G. ; HILL, J.R.: *Building winning trading systems with TradeStation*. Wiley, 2003
- [Pis97] PISCHULTI, H.: *Direktbankgeschäft*. Knapp, 1997
- [PR02] PEDRONI, S. ; RAPPIN, N.: *Jython essentials*. O'Reilly Media, Inc., 2002
- [PSV04] POTVIN, J.Y. ; SORIANO, P. ; VALLEE, M.: Generating trading rules on the stock markets with genetic programming. In: *Computers and Operations Research* 31 (2004), Nr. 7, S. 1033–1048
- [Rec73] RECHENBERG, I.: *Evolutionsstrategie*. Frommann-Holzboog, 1973
- [Rei96] REICHMANN, D.: Vom Ende der Welle-Umkehrformationen. In: *Börse Online* 18 (1996)
- [Res09] RESEARCH, Alyuda: *Tradecision*. <http://www.tradecision.com/>. Version: 2009. – Zugriff am 15.01.2009
- [Rim03] RIME, D.: *New electronic trading systems in foreign exchange markets*. Norges bank, 2003
- [RSC07] RIMCHAROEN, Sunisa ; SUTIVONG, Daricha ; CHONGSTITVATANA, Prabhas: A synthesis of optimal stopping time in compact genetic algorithm based on real options approach. In: *GECCO '07: Proceedings of the 9th annual conference on Genetic and evolutionary computation*. New York, NY, USA : ACM, 2007. – ISBN 978-1-59593-697-4, S. 630–630
- [S<sup>+</sup>05] STALLMAN, R. u. a.: The GNU project. In: *at http://www.indymedia.org.uk/en/2004/05/292609.html*, accessed 29 (2005)
- [SAB85] SHARPE, W.F. ; ALEXANDER, G.J. ; BAILEY, J.V.: *Investments*. Prentice-Hall Englewood Cliffs, NJ, 1985
- [SB00] STEINER, M. ; BRUNS, C.: Aktien-Analysemethoden versus Effizienzmarkttheorie. In: *Schaffer-Poeschel Verlag, Stuttgart* 7 (2000)
- [Sch65] SCHWEFEL, H.P.: Kybernetische Evolution als Strategie der experimentellen Forschung in der Stromungstechnik. In: *Master's thesis, Technical University of Berlin* (1965)

- [Sch02] SCHREDELSEKER, K.: *Grundlagen der Finanzwirtschaft: Ein informationsökonomischer Zugang*. Oldenbourg Wissenschaftsverlag, 2002
- [Sch08] SCHLITTER, N.: » *Analyse und Prognose ökonomischer Zeitreihen mittels evolutionär optimierter Backpropagation-Netze*«, Diss., 2008
- [SDJ98] SEXTON, R.S. ; DORSEY, R.E. ; JOHNSON, J.D.: Toward global optimization of neural networks: A comparison of the genetic algorithm and backpropagation. In: *Decision Support Systems* 22 (1998), Nr. 2, S. 171–185
- [Sha92] SHAPCOTT, J.: Index tracking: genetic algorithms for investment portfolio selection. In: *Edinburgh Parallel Computing Centre* (1992)
- [Spa02] SPANN, M.: *Virtuelle Börsen als Instrument zur Marktforschung*. Deutscher Universitätsverlag, 2002
- [ST04] SIMPSON, B. ; TOUSSI, F.: *Hsqldb User Guide*. 2004
- [Sup09] SUPPORT, TS: *MultiCharts*. <http://www.tssupport.com/multicharts/>. Version: 2009. – Zugriff am 15.04.2009
- [Szp97] SZPIRO, G.G.: The Emergence of Risk Aversion. In: *Complexity* 2(4) (1997), S. 31–39
- [Szp02] SZPIRO, G.G.: Tinkering with Genetic Algorithms: Forecasting and Data Mining in Finance and Economics. In: *STUDIES IN FUZZINESS AND SOFT COMPUTING* 100 (2002), S. 273–286
- [Tec09] TECHNOLOGIES, Wave59: *Wave59*. <http://www.wave59.de/>. Version: 2009. – Zugriff am 15.01.2009
- [UN01] URBATSCH, R.C. ; NAGLER, F.: *Technische Wertpapieranalyse–Grundlagen, Technische Wertpapieranalyse, Stop-Loss-Orders-Teil 2*. 2001
- [Vac97] VACCA, L.: Managing options risk with genetic algorithms. In: *Computational Intelligence for Financial Engineering (CIFEr), 1997., Proceedings of the IEEE/IAFE 1997*, 1997, S. 29–35

- [WCL04] WONG, E.Y.C. ; CHAN, A.T.S. ; LEONG, H.V.: Efficient management of XML contents over wireless environment by Xstream. In: *Proceedings of the 2004 ACM symposium on Applied computing* ACM, 2004, S. 1122–1127
- [Wei01] WEINBERGER, M.: Österreichische Online Broker im internationalen Vergleich. (2001)
- [WJ01] WICKHAM-JONES, T.: web Mathematica: A User Guide. In: *Wolfram Research, Inc* (2001)
- [Wri91] WRIGHT, A.H.: Genetic algorithms for real parameter optimization. In: *Foundations of genetic algorithms* 1 (1991), S. 205–218
- [XSt09] XSTREAM, C.: *Architecture Overview*. <http://xstream.codehaus.org/architecture.html>. Version: 2009. – Zugriff am 15.04.2009