

**Dezentrale Strategieelemente
für Evolutionsstrategien
und Anwendung zum Reglerentwurf**

von der
Fakultät für Elektrotechnik und Informationstechnik
der Universität Dortmund
genehmigte Dissertation
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften

von

Barbara Ottersbach

Dortmund, August 2004

Tag der mündlichen Prüfung: 27. Januar 2005
Hauptreferent: Prof. em. Dr. rer. nat. H. Kiendl
Korreferent: Prof. Dr.-Ing. B. Lohmann

Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftliche Mitarbeiterin am Lehrstuhl für Elektrische Steuerung und Regelung der Universität Dortmund. Die Arbeit wurde angeregt und betreut von Herrn Prof. em. Dr. rer. nat. Harro Kiendl. Auf ihn geht die Grundidee dieser Arbeit zurück und ihm danke ich herzlich für die stete Förderung meiner wissenschaftlichen Arbeit. Herrn Prof. Dr.–Ing. Boris Lohmann danke ich für die Übernahme des Korreferats.

Der Deutschen Forschungsgemeinschaft danke ich für die finanzielle Unterstützung im Rahmen des Graduiertenkollegs *Modellierung und modellbasierte Entwicklung komplexer technischer Systeme* der Universität Dortmund zu Beginn meiner wissenschaftlichen Tätigkeit.

Ferner bedanke ich mich bei allen Mitarbeiterinnen und Mitarbeitern des Lehrstuhls für die vielfältige Unterstützung und die gute Zusammenarbeit. Besonderer Dank gilt dabei den Herren Daniel Schauten und Martin Sternke. Wertvolle Hinweise inhaltlicher und formaler Art für die endgültige Fassung des Manuskripts verdanke ich außerdem den Herren Dr.–Ing. Renato Krohling und Stefan Meier sowie Frau Gertrud Kasimir. Insbesondere bedanke ich mich auch bei Herrn Dr. rer. nat. Frank Hoffmann als Lehrstuhlvertreter für die Möglichkeit, meine Arbeit wie gewünscht abzuschließen.

Für die gute Zusammenarbeit im Rahmen einer langjährigen Industriekoope-
ration bedanke ich mich bei der Bosch Rexroth AG, ehemals Robert Bosch GmbH. Mein besonderer Dank gilt hier Herrn Wilhelm Blumendeller für wertvolle Hinweise und Diskussionen.

Für die Untersuchung einzelner Fragestellungen danke ich schließlich den Studenten, deren Studien- und Diplomarbeiten ich betreuen durfte.

Dortmund, Juli 2005

Barbara Ottersbach

”Alles durch Mutation und Selektion entstanden?
Da könnte man doch auch einem Steinzeitmenschen einen Haufen Sand geben
und ihm sagen, er soll solange herumprobieren,
bis daraus ein Pentium-Prozessor geworden ist?
Aber siehe da: Schon nach 20000 Jahren hatte er es geschafft!”

Peter Hohl, dt. Journalist und Verleger
(aus: ”Ein Mittel gegen Einsamkeit...”, 1999)

Inhaltsverzeichnis

1	Einleitung	1
2	Einordnung und Zielsetzung	6
2.1	Aufgabenstellung	8
2.2	Optimierung mit Evolutionären Algorithmen	13
2.3	Ziele dieser Arbeit	18
3	Klassifizierung von Optimierungsproblemen	21
3.1	Skalare Optimierungsprobleme	23
3.2	Multikriterielle Optimierungsprobleme	23
3.3	Rauschen und Drift bei Optimierungsproblemen	29
4	Schwachstellen-Analyse von Optimierungsverfahren	31
4.1	Evolutionstrategien (ES)	32
4.1.1	Generierung der Startpopulation	33
4.1.2	Rekombination	36
4.1.3	Mutation	39
4.1.4	Selektion	45
4.1.5	Abbruchbedingung	53
4.2	Hooke-Jeeves-Verfahren	54
5	Auswahl analytischer Testfunktionen	57
6	Neue Strategieelemente: Konzeption, Wirkungsweise	62
6.1	ES mit wechselnder Breiten- und Tiefensuche (ES_{WBT})	62
6.1.1	ES_{WBT} mit variabler Laufzeit: ES_{WBT}^{Tvar}	64

6.1.2	ES _{WBT} mit konstanter Laufzeit: ES _{WBT} ^{T_{const}}	66
6.1.3	ES _{WBT} bei multikriteriellen Optimierungsproblemen . . .	70
6.1.4	Ergebnisse mit ES _{WBT} bei Verwendung einer analytischen Testfunktion als Gütemaß	72
6.2	ES mit gleichzeitiger Breiten- und Tiefensuche (ES _{GBT})	73
6.2.1	ES _{GBT} mit symmetrischen Wahrscheinlichkeitsdichtefunktionen: ES _{GBT} ^{symm}	76
6.2.2	ES _{GBT} mit unsymmetrischen Wahrscheinlichkeitsdichtefunktionen: ES _{GBT} ^{unsymm}	79
6.2.3	ES _{GBT} bei multikriteriellen Optimierungsproblemen . . .	81
6.2.4	Ergebnisse mit ES _{GBT} bei Verwendung einer analytischen Testfunktion als Gütemaß	83
6.3	Kombination ES mit Hooke-Jeeves-Verfahren	86
6.3.1	Ergebnisse mit ES _{hybrid} bei Verwendung einer analytischen Testfunktion als Gütemaß	93
6.4	Hooke-Jeeves-Verfahren mit Neustart (HoJe _{Neustart})	96
6.5	Ergebnisse bei höherdimensionalen Testfunktionen	99
7	Industrielles Hydraulikventil	101
7.1	Hardware	101
7.2	Software	104
7.3	Reglerstruktur	104
7.4	Bewertung der Regelungsgüte	105
7.5	Optimierungsergebnisse	121
8	Zusammenfassung und Ausblick	128
A	Symbole, Definitionsgleichungen, Abkürzungen	133
B	Fuzzy-Gütemaß	137
	Literaturverzeichnis	140

Kapitel 1

Einleitung

In wissenschaftlichen und technischen Bereichen lassen sich viele Aufgabenstellungen auf Optimierungsprobleme zurückführen. Dies können Modellierungsaufgaben sowie allgemein Aufgabenstellungen sein, die eine möglichst günstige Einstellung bestimmter Strukturen oder Parameter – wie beispielsweise die Einstellung der freien Parameter eines Reglers – betreffen. Hierbei ist es möglich, dass bereits eine Struktur für die Optimierungsaufgabe bekannt ist und nur noch freie Parameter innerhalb dieser Struktur möglichst günstig eingestellt werden sollen. Es kann aber auch die Aufgabe bestehen, die Struktur selbst ganz oder teilweise durch einen Optimierungsprozess in möglichst günstiger Weise zu ermitteln.

Es existiert eine Vielzahl unterschiedlicher Optimierungsverfahren, wobei die Auswahl eines für das zu lösende Optimierungsproblem geeigneten Verfahrens häufig eine nicht zu unterschätzende Schwierigkeit darstellt. Eine mögliche Einteilung der existierenden Optimierungsverfahren besteht in der Unterscheidung, ob sie lokal arbeiten, d. h. ausgehend von einem Startpunkt das nächstgelegene lokale Optimum ermitteln, oder ob sie global arbeiten, d. h. darauf abzielen, den gesamten Suchraum möglichst breit auszuschöpfen und darin eine möglichst gute Lösung zu finden.

Bei komplexen Optimierungsproblemen ist häufig der Einsatz von Evolutionären Algorithmen (EA) zum Auffinden einer möglichst optimalen Lösung des vorliegenden Optimierungsproblems sinnvoll, da Evolutionäre Algorithmen populationsbasiert arbeiten und somit mehrere mögliche Lösungen den Optimierungsprozess gleichzeitig beeinflussen. Ein Anhaltspunkt für eine hohe Komplexität des Optimierungsproblems besteht in einer stark *zerklüfteten Gütelandschaft*. Dies ist die virtuelle Landschaft der Gütewerte, die über den einzustellenden Parametervektoren aufgetragen sind. Des Weiteren kann eine *hohe Anzahl* einzustellender Parameter zu einem komplexen Optimierungsproblem führen.

Im Bereich der Technik treten zunehmend Optimierungsaufgaben auf, die dadurch gekennzeichnet sind, dass nur wenig über die Topologie der Güte-landschaft bekannt ist. Eine Ursache hierfür ist z. B. die zunehmende Vernetzung technischer Systeme, wodurch das Verhalten eines Teilsystems vom Verhalten mehrerer anderer Teilsysteme abhängig ist. Eine weitere Ursache liegt in gestiegenen Genauigkeitsanforderungen, beispielsweise an die Positioniergenauigkeit der Lageregelung eines Hydraulikventils. Diese Anforderungen führen dazu, dass eine Vielzahl von Effekten, darunter auch undurchsichtige wie die Reibung, für die Optimierung zu berücksichtigen sind.

Für die Lösung derartiger Optimierungsaufgaben haben sich Evolutionäre Algorithmen, die die biologischen Mechanismen der Fortpflanzung, Mutation und Selektion nachahmen, oftmals bewährt. Ihre Nützlichkeit gerade für die Lösung technischer Probleme liegt u. a. darin begründet, dass diese Algorithmen bereits von ihrer Konzeption her den meist gar nicht einlösbaren Anspruch aufgeben, die bestmögliche Lösung überhaupt, d. h. das absolute Optimum, zu finden. Stattdessen streben sie das realistischere Ziel an, eine Lösung zu finden, die besser als bisher bekannte Lösungen ist. In diesem Zusammenhang ist anzumerken, dass die Evolutionären Algorithmen zusammen mit Fuzzy-Logik (FL) und Neuronalen Netzen (NN) unter dem Begriff CI-Methoden (CI: Computational Intelligence) zusammengefasst werden. Die Ansätze dieser Methoden sind insofern verwandt, als auch die Fuzzy-Logik und die Technik der Neuronalen Netze natürlich gewachsene Problemlösungsstrategien, die in der menschlichen Sprache bzw. in den biologischen neuronalen Netzen angetroffen werden, nachahmen.

Die vorliegende Arbeit geht von dem Anwendungsproblem der Optimierung der Reglerparameter für ein Hydraulikventil aus. Es zeigt sich, dass die Anwendung herkömmlicher Evolutionsstrategien – einer Form der Evolutionären Algorithmen – auf dieses Problem zwar möglich ist, die damit erzielten Ergebnisse jedoch nicht allen Anwenderwünschen entsprechen.

Bei dem vorliegenden Anwendungsproblem handelt es sich um die Optimierung eines Reglers mit 24 einstellbaren Parametern. Zwar zielen Evolutionäre Algorithmen – wie erwähnt – bereits von der Konzeption her nicht darauf ab, in diesem riesigen Suchraum, der von den 24 Parametern aufgespannt wird, das absolute Optimum exakt zu finden. Aber dennoch besteht aus Anwendersicht der Wunsch, nicht nur eine gute Lösung zu finden, sondern zugleich auch zumindest eine Einschätzung darüber zu gewinnen, ob sich in dem Suchraum u. U. noch bessere Lösungen finden lassen. Abhängig davon kann der Anwender dann nämlich entscheiden, ob eine weitere Verbesserung der Regelgüte durch eine weitere Optimierung der Parameter bei unveränderter Reglerstruktur möglich

erscheint oder ob aufwendige Änderungen der Reglerstruktur angeraten sind. Es zeigt sich, dass Standard-Evolutionsstrategien potenzielle Lösungen produzieren, die vergleichsweise schnell gegen eine Lage in dem Suchraum konvergieren. Stattdessen ist eine bessere Absicherung des letztendlich erhaltenen Ergebnisses dadurch erwünscht, dass der evolutionäre Prozess den Suchraum breiter ausschöpft. In dieser Arbeit werden Strategien entwickelt, die für eine verbesserte Breitensuche sorgen.

Ein erstes Konzept besteht darin, dass in der Selektion bevorzugt räumlich benachbarte Parametervektoren miteinander hinsichtlich ihrer Güte verglichen und in diesem Vergleich schlechter bewertete Lösungsvorschläge verworfen werden. Somit verbleiben im Optimierungsprozess mit großer Wahrscheinlichkeit weiter voneinander entfernte Parametervektoren.

In einem zweiten Konzept wird bei relativ schlecht bewerteten Parametervektoren, die vor allem zu Beginn des Optimierungsprozesses auftreten, zum einen die Fortpflanzung dahingehend vorgenommen, dass bevorzugt solche Parametervektoren herangezogen werden, die räumlich relativ weit voneinander entfernt sind. Zum anderen werden dann in der Selektion bevorzugt Parametervektoren, die räumlich benachbart sind, hinsichtlich ihrer Güte miteinander verglichen. Beide Mechanismen begünstigen eine, verglichen mit einer Standard-Evolutionsstrategie, breite Ausschöpfung des Suchraums.

Eine weitere generelle Schwachstelle von Evolutionsstrategien besteht darin, dass sie eine sehr große Anzahl an Gütebewertungen erfordern. Dies ist insbesondere dann zeitaufwendig, wenn die Gütebewertung – wie bei Hardware-in-the-loop-Aufbauten – Experimente an realen Prozessen erfordert. Zur Milderung dieser Schwachstelle werden weitere Strategieelemente entwickelt.

Sie bestehen erstens darin, dass in der näheren Umgebung bereits gefundener, vergleichsweise guter Lösungsvorschläge keine Breitensuche, sondern eine Tiefensuche durchgeführt wird: In Abhängigkeit vom bereits erreichten Optimierungsergebnis erfolgt eine Umschaltung von Breiten- auf Tiefensuche. So wird ein günstiger Kompromiss zwischen Breiten- und Tiefensuche während des Optimierungsprozesses und damit ein Zeitersparnis erreicht.

Zweitens wird zur Zeitersparnis ein hybrides Lösungskonzept, bestehend aus einer Kombination der Evolutionsstrategie mit einem klassischen lokalen Optimierungsverfahren, entwickelt.

Das vorliegende Anwendungsproblem ist – wie die meisten technischen Optimierungsaufgaben – eine *multikriterielle Optimierungsaufgabe*, bei der mehrere, teilweise gegenläufige Gütegesichtspunkte während des Optimierungsprozesses zu berücksichtigen sind. Vielfach ist es zielführend, die relevanten Gütegesichts-

punkte zu einem skalaren Gütemaß zusammenzufassen und dieses dann zu optimieren. So wird auch hier verfahren, um unterschiedliche Kenngrößen, wie beispielsweise die Anregelzeit und die Überschwingweite, die eine Sprungantwort charakterisieren, zu bewerten. Bei den häufig angetroffenen gegenläufigen Gütegesichtspunkten reicht aber diese Vorgehensweise oft nicht aus, weil man vorab nicht weiß, welche Verbesserung bezüglich eines Gütegesichtspunktes durch Hinnahme welcher Verschlechterungen bezüglich anderer Gütegesichtspunkte erzielbar ist. Deshalb ist aus Anwendersicht eine Einstellbarkeit der Evolutionsstrategie derart zu fordern, dass sie optional auch vektoriell arbeitet, d. h. eine simultane Optimierung hinsichtlich mehrerer Gütegesichtspunkte vornimmt. Dementsprechend werden die obigen Strategieelemente – mit Ausnahme der hybriden Strategie – so ausgelegt, dass sie auch im multikriteriellen Fall einsetzbar sind.

Das bereits genannte Anwendungsproblem aus der industriellen Praxis ist insofern besonders realitätsnah, als das Ventil in einem Hardware-in-the-loop-Laborstand für den Optimierungsprozess verwendet wird. Im Gegensatz zum Einsatz eines Simulationsmodells entfällt so das in anderen Anwendungen häufig auftretende Problem, dass zwar anhand eines Simulationsmodells gute Ergebnisse erzielt werden, diese aber aufgrund von Modellierungsfehlern oder Modellierungsungenauigkeiten nicht unmittelbar auf das modellierte reale System übertragbar sind.

Die Nützlichkeit der in dieser Arbeit entwickelten Strategieelemente wird nicht nur simulatorisch, sondern anhand dieses Anwendungsproblems nachgewiesen.

Die Arbeit ist folgendermaßen gegliedert:

In Kapitel 2 wird die dieser Arbeit zugrunde liegende Aufgabenstellung vorgestellt und es werden die Einsatzmöglichkeiten sowie die Vor- und Nachteile verschiedener Optimierungsverfahren zusammengefasst. Insbesondere wird ein Überblick über Evolutionsstrategien gegeben. Es erfolgt eine Einordnung dieser Arbeit in die bekannte Fachliteratur.

Kapitel 3 dient der Einordnung verschiedenartiger Optimierungsprobleme. Hierbei wird eine Unterscheidung zwischen skalaren und multikriteriellen Optimierungsproblemen vorgenommen. Auch wird der Einfluss von Rauschen und Drift bei der Gütebewertung auf den Ablauf der Optimierung dargestellt und es werden sinnvolle Maßnahmen zur Berücksichtigung dieser Effekte diskutiert.

In Kapitel 4 wird ein Überblick über bekannte Optimierungsverfahren gegeben. Insbesondere Evolutionsstrategien und das für die hybride Strategie genutzte Hooke-Jeeves-Verfahren werden detailliert dargestellt und hinsichtlich

ihrer Schwachstellen analysiert. Bei den Evolutionsstrategien werden Standard-Vorgehensweisen und verschiedene, aus der Literatur bekannte, spezielle Vorgehensweisen untersucht, um die im Rahmen dieser Arbeit entwickelten neuen Strategieelemente gegenüber bereits bekannten Ansätzen abzugrenzen und die erzielten Vorteile zu verdeutlichen.

In Kapitel 5 werden aus der Literatur bekannte analytische Testfunktionen ausgewählt, die anschließend in Kapitel 6 zur qualitativen Beurteilung der neuen Strategieelemente verwendet werden.

Das Kapitel 6 stellt den methodischen Kern der vorliegenden Arbeit dar. Die neuen Strategieelemente werden umfassend vorgestellt. Das Ziel besteht zum einen in einem besseren Kompromiss zwischen Breiten- und Tiefensuche. Durch das neue hybride Optimierungsverfahren wird zum anderen das Ziel verfolgt, eine automatische Einstellung des Optimierungsprozesses entweder auf eine ungerichtete Suche oder auf eine bevorzugte Suchrichtung im Suchraum hervorzurufen. Somit passt sich der Optimierungsprozess bei Einsatz des hybriden Optimierungsverfahrens an die Beschaffenheit des Optimierungsproblems an, ohne dass der Anwender vorab entscheiden muss, ob eine ungerichtete oder gerichtete Suche durchgeführt werden soll. Es wird gezeigt, dass durch diese automatische Anpassung ein wesentliches Zeitersparnis bei der Optimierung möglich ist. Es werden somit die Vorteile eines global suchenden mit denen eines lokal arbeitenden Verfahrens kombiniert. Für die neuen Strategieelemente werden vergleichende Ergebnisse präsentiert und diskutiert.

Anschließend werden in Kapitel 7 die vorgestellten neuen Strategieelemente zur Lösung des genannten praxisbezogenen Optimierungsproblems, der Regleroptimierung für ein industrielles Hydraulikventil, eingesetzt und anhand der erzielten Ergebnisse miteinander verglichen. Die Konzeption der Gütebewertung für diese Anwendung wird ausführlich beschrieben. Das Gütemaß basiert auf der Fuzzy-Logik, deren Grundlagen hier ebenfalls skizziert werden.

In Kapitel 8 werden die wesentlichen Aspekte der Arbeit zusammengefasst. Es werden weitere Arbeiten angesprochen, die mit dieser Arbeit eng verknüpft sind. Abschließend wird ein Ausblick auf mögliche zukünftige Arbeiten gegeben.

Kapitel 2

Einordnung und Zielsetzung

In diesem Kapitel wird ein Überblick über die Einsatzmöglichkeiten von Optimierungsverfahren, insbesondere von Evolutionären Algorithmen, gegeben. Es existiert eine Vielzahl von solchen Verfahren, so dass es für einen Anwender oft schwierig ist, ein für seine Problemstellung geeignetes Verfahren auszuwählen. Hinzu kommen oft noch die Schwierigkeiten, dass das vorliegende Problem zunächst nicht in einer derart aufbereiteten Art vorliegt, dass unmittelbar ein Optimierungsverfahren einsetzbar ist und dass sowohl für diese Aufbereitung als auch für die Auswahl eines geeigneten Verfahrens nur eine geringe Zeitdauer zur Verfügung steht.

Optimierungsverfahren sind überall dort einsetzbar, wo bestimmte Parameter eingestellt werden müssen. Dies kann beispielsweise die Approximation einer unbekanntenen Funktion sein, wobei ein Funktionsansatz aufgestellt wird, in dem noch freie, einstellbare Parameter enthalten sind, die mit Hilfe eines Optimierungsverfahrens bestimmt werden können. Auch das Auffinden eines solchen Funktionsansatzes kann durch ein Optimierungsverfahren geleistet werden.

Beispiele für technische Optimierungsprobleme sind die Optimierung des Designs von Flugzeug-Tragflächen [BNSW00], die Optimierung der Parametereinstellung eines Trinkwasser-Versorgungssystems [AL95] oder die Optimierung von elektrischen Energieübertragungsnetzen [SV01].

Eine weitere Einsatzmöglichkeit besteht beispielsweise in der möglichst günstigen Regelung einer technischen Regelstrecke. Hierzu kann eine Reglerstruktur mit darin enthaltenen noch frei einstellbaren Parametern festgelegt werden, beispielsweise kann ein PID-Regler ausgewählt werden. Die dann noch einzustellenden Parameter werden in der Praxis oft interaktiv von Hand eingestellt, was bei einer komplizierten Reglerstruktur sehr viel Zeit und Erfahrung kostet, da die Gütebewertung häufig aufgrund gegenläufiger Teilziele schwierig

ist. Sehr viel komfortabler können die Reglerparameter unter Verwendung eines Optimierungsverfahrens automatisiert eingestellt werden. Auch die Auswahl der geeigneten Reglerstruktur muss nicht notwendigerweise vom Anwender vorgenommen werden, sondern zur Lösung dieser Aufgabe ist ebenfalls der Einsatz eines Optimierungsverfahrens denkbar.

Ein weiteres umfangreiches Anwendungsfeld für Optimierungsverfahren ist die Modellierung. Überall dort, wo ein Modell für einen technischen Prozess oder viel allgemeiner für das Eingangs–Ausgangsverhalten eines Systems bestimmt werden soll, kommen Optimierungsverfahren zum Einsatz. Liegt bereits Vorwissen über die Struktur des zu bestimmenden Modells vor, so ist es im Allgemeinen sinnvoll, dieses Vorwissen bereits in den Modellansatz einfließen zu lassen und nur noch gewisse Freiheitsgrade durch ein Optimierungsverfahren festzulegen. Ist dieses Vorwissen nicht oder nur unzureichend vorhanden, so kann auch der Modellansatz mit Hilfe eines Optimierungsverfahrens aufgestellt werden. Beispiele für die Modellierung technischer Systeme sind die Modellierung eines Hydraulikventils oder als Teilaufgabe die Modellierung der Strömungskräfte, die in einem Hydraulikventil wirken.

Auch bekannte Verfahren zur datenbasierten Modellierung, wie etwa das Fuzzy-ROSA-Verfahren [Kro99, Sla01, Kra02], nutzen Optimierungsverfahren, um das Eingangs–Ausgangsverhalten eines Systems anhand einer vorliegenden Datenbasis von Eingangs- und Ausgangsdaten bestmöglich nachzubilden.

Für Anwender ist es häufig nicht entscheidend, für ein Optimierungsproblem die bestmögliche Lösung zu finden. Häufig genügt auch eine suboptimale Lösung, die alle gestellten Anforderungen erfüllt. Es ist hierbei entscheidend, diese suboptimale Lösung mit akzeptablem Aufwand zu finden oder eine Verbesserung gegenüber bereits bekannten Lösungen zu erreichen.

Hierfür ist es zunächst wesentlich, das Problem in einer Weise aufzubereiten, die es für ein nachgeschaltetes Optimierungsverfahren lösbar macht. Der Anwender muss also zunächst analysieren, von welcher Art das Problem ist und welche Freiheitsgrade durch das Optimierungsverfahren eingestellt werden sollen. Hierbei muss er festlegen, ob die Struktur des Optimierungsproblems bereits feststeht und in dieser Struktur noch enthaltene freie Parameter eingestellt werden sollen oder ob das Auffinden der Struktur selbst Teil der Optimierungsaufgabe ist. Im zweiten Fall spricht man von *Strukturoptimierung*.

Sämtliche Optimierungsverfahren benötigen für das Einstellen günstiger Parameter mindestens *ein* Gütemaß, das eine Aussage darüber macht, wie gut oder schlecht ein eingestellter Satz von Parameterwerten das Optimierungsproblem löst. Auch die Auswahl oder sogar die Entwicklung eines geeigneten Gütema-

ßes ist aufwendig. Oft ist es möglich, bekannte Fehlermaße für die Gütebewertung zu verwenden. Je nach Anforderungsprofil kann es aber auch notwendig sein, Expertenwissen über das Verhalten eines technischen Systems datenbasiert oder wissensbasiert nachzubilden. Wurde die Einstellung von Reglerparametern bisher von einem Prozessexperten von Hand vorgenommen, so besitzt dieser Prozessexperte spezifisches Wissen über die gewünschte Parametereinstellung. Beispielsweise wird er durch visuelle Betrachtung der Auswirkung bestimmter Einstellungen der Parameter auf gewisse Ausgangsgrößen des Prozesses und gedanklichen Vergleich mit dem gewünschten Prozessverhalten die Parameter so lange verstellen, bis das reale Verhalten mit der Wunschvorstellung möglichst gut übereinstimmt. In einigen Anwendungsfällen ist jedoch keine objektive Bewertung einer bestimmten Parametereinstellung durch einen Zahlenwert möglich. Wie bei o. g. Beispiel der Reglereinstellung durch einen Prozessexperten ist es dennoch möglich, einen Optimierungsprozess durchzuführen, wenn bewertet werden kann, ob eine bestimmte Parametereinstellung qualitativ besser oder schlechter als eine andere Einstellung ist. In [Her98] wird beispielsweise die Optimierung von Kaffeemischungen anhand des qualitativen Testurteils von professionellen Testern einer Kaffeerösterei vorgestellt.

Die virtuelle Landschaft, die aufgespannt wird, wenn man über dem Raum der Parametervektoren die jeweils zugehörigen Gütewerte aufträgt, wird häufig Gütelandschaft genannt. Von der Beschaffenheit dieser Gütelandschaft ist es auch abhängig, welches Optimierungsverfahren eingesetzt werden sollte, damit der Optimierungsprozess möglichst günstig abläuft.

Ein gesuchtes Optimum zur Lösung des Optimierungsproblems kann entweder ein Maximum oder ein Minimum in der Gütelandschaft sein. Die Gütelandschaft kann stetig sein oder auch Unstetigkeitsstellen aufweisen, sie kann freundlich (d. h. differenzierbar) oder stark zerklüftet (d. h. nicht differenzierbar) sein und sie kann für einen jeweils begrenzten Wertebereich der einzustellenden Parameter ein globales Optimum oder mehrere – zum Teil lokale – Optima aufweisen.

Ist die Beschaffenheit der Gütelandschaft vorab bekannt, so kann das Optimierungsverfahren in Abhängigkeit davon ausgewählt werden. Oft liegen aber die dafür notwendigen Kenntnisse nicht vor.

2.1 Aufgabenstellung

Optimierungsprobleme werden meist durch die Suche nach minimalen oder maximalen Funktionswerten einer Funktion Q , dem sogenannten Gütemaß, defi-

niert. Enthält das Gütemaß mehr als ein Gütekriterium, so spricht man von einem vektoriellen Gütemaß \mathbf{Q} . Ein Vektor \mathbf{x} enthält einen Satz von Parameterwerten, die eine potenzielle Lösung des Optimierungsproblems darstellen: $\mathbf{x} = (x_1, \dots, x_n)$. Wird für diesen Vektor \mathbf{x} die Funktion $Q(\mathbf{x})$ bzw. $\mathbf{Q}(\mathbf{x})$ ausgewertet, so wird dem Vektor \mathbf{x} ein Gütewert $Q(\mathbf{x})$ oder es werden mehrere Gütewerte $\mathbf{Q}(\mathbf{x}) = (Q_1(\mathbf{x}), \dots, Q_k(\mathbf{x}))$ zugewiesen.

Im Zusammenhang mit Evolutionären Algorithmen wird ein Vektor \mathbf{x} auch als Individuum bezeichnet. Eine Menge von Individuen wird dann Population genannt. Ausgehend von einer Startpopulation werden schrittweise durch die Programmschritte Rekombination, Mutation und Selektion neue Mengen von Individuen erzeugt. Eine durch Rekombination oder Mutation erzeugte Menge wird Nachkommenpopulation, eine durch Selektion der jeweils besten ('fittesten') Individuen erzeugte Menge wird Elternpopulation genannt. Dabei wird ein skalares oder vektorielles Gütemaß Q bzw. \mathbf{Q} im Rahmen der Evolutionären Algorithmen auch als Fitnessfunktion bezeichnet.

Die Parameter, die zur Lösung des Optimierungsproblems eingestellt werden sollen, sind häufig gewissen Grenzen unterworfen, d. h. jede Komponente des Parameterraums wird beispielsweise durch einen minimal und einen maximal zulässigen Wert des Parameters begrenzt (\mathbf{x}_{min} bzw. \mathbf{x}_{max}). Auf diese Art wird für die einzustellenden Parameter ein mehrdimensionaler Hyperquader aufgespannt, innerhalb dessen die zulässigen Vektoren \mathbf{x} liegen. Diese Grenzen \mathbf{x}_{min} und \mathbf{x}_{max} können daraus resultieren, dass das verwendete Gütemaß nur für einen bestimmten Bereich der einzustellenden Parameterwerte definiert ist. Eine weitere Ursache für diese Bereichsgrenzen kann darin bestehen, dass Parameterwerte außerhalb des zulässigen Hyperquaders zu unerwünschten oder gefährlichen Systemzuständen des technischen Systems führen können, auf das sie wirken. Die Bereichsgrenzen für die einzustellenden Parameter können auch aus Vorwissen resultieren, wenn für die Parameter bereits aus dem Anwendungszusammenhang heraus sinnvolle Grenzen festgelegt werden können. Die Begrenzung des Parameterraums durch \mathbf{x}_{min} und \mathbf{x}_{max} wird Begrenzung durch *harte Nebenbedingungen* genannt.

Es existieren auch Optimierungsprobleme, bei denen harte Nebenbedingungen in Form von Ungleichungen $u_i(\mathbf{x}) \leq 0$ mit $i = 1, \dots, l$ vorliegen. Dann wird durch die gegebenen Nebenbedingungen nicht ein n -dimensionaler Hyperquader, sondern ein n -dimensionaler Teilraum des Parameterraums festgelegt. Die oben beschriebene Begrenzung des Parameterraums in Form eines Hyperquaders ist in dieser Form der Nebenbedingungen als Sonderfall enthalten.

Ferner gibt es Optimierungsprobleme, die ohne eine Begrenzung des Parame-

terraums durch harte Nebenbedingungen definiert werden. Die Motivation für den Verzicht auf Bereichsgrenzen kann darin begründet sein, dass diese Grenzen nicht sinnvoll festgelegt werden können.

Je nach Optimierungsproblem kann es auch sinnvoll sein, im begrenzten oder unbegrenzten Parameterraum einen Teilraum durch zusätzliche *weiche Nebenbedingungen* festzulegen, die aber – im Gegensatz zu harten Nebenbedingungen – während des Optimierungsprozesses verletzt werden dürfen und somit nur die Aufgabe haben, den Optimierungsprozess in Richtung der Parameterwerte innerhalb dieses Teilraums zu lenken.

Ziel eines Optimierungsprozesses ist es also, eine möglichst gute Lösung des Optimierungsproblems zu finden. Diese Lösung besteht bei einem skalaren Gütemaß häufig in einem oder mehreren Vektoren \mathbf{x} , die zu einem minimalen oder maximalen Gütewert führen. Bei einem vektoriellen Gütemaß existiert häufig eine Menge \mathcal{X} von Vektoren \mathbf{x} , die jeweils eine unterschiedliche Kompromissbildung zwischen den einzelnen Gütekriterien vornehmen und jeweils im Sinne einer bestimmten Kompromissbildung ein Optimum darstellen. Das Ziel des Optimierungsprozesses ist dann entweder das Auffinden der gesamten Kompromissmenge, der sogenannten Pareto-Menge, oder das Auffinden eines Punktes innerhalb der Pareto-Menge. Kann eine existierende Lösung nicht exakt aufgefunden werden, so besteht das Ziel in der möglichst genauen Annäherung an diese existierende Lösung.

Bei einem skalaren Gütemaß Q besteht ein weiteres Ziel darin, mit möglichst wenigen Bewertungen mittels des Gütemaßes einen oder mehrere Vektoren \mathbf{x} zu finden, die das Optimierungsproblem möglichst gut lösen, d. h. die der wirklichen Lösung, einem globalen Maximum oder Minimum oder mehreren globalen Maxima oder Minima in der Gütelandschaft, möglichst nahe kommen. Bei Vorliegen eines vektoriellen Gütemaßes \mathbf{Q} möchte man die Pareto-Menge mit möglichst wenigen Gütebewertungen möglichst gut approximieren. Diese Gütebewertungen erfordern bei der Optimierung in vielen Fällen den größten Zeitaufwand, da beispielsweise die Auswirkungen einer bestimmten Parametereinstellung – also eines bestimmten Vektors \mathbf{x} – auf ein reales technisches System zunächst ‘abgewartet’ und anschließend ausgewertet werden müssen.

Im Folgenden wird eine Übersicht über die grundsätzlichen Eigenschaften bisher bekannter Optimierungsverfahren gegeben. Dabei liegt ein Schwerpunkt in der Analyse und Verbesserung von Evolutionsstrategien. Daher wird auf Evolutionsstrategien besonders detailliert eingegangen.

Für einen Anwender ist es oftmals schwierig, ein geeignetes Optimierungsverfahren auszuwählen. Bei einer freundlichen Gütelandschaft, die nur ein globales Optimum aufweist, ist der Einsatz eines Gradientenverfahrens sinnvoll. Denn bei einer derartigen Gütelandschaft werden bei Einsatz eines solchen Verfahrens zum Erreichen des Ziels nicht unnötig viele Gütebewertungen durchgeführt. Im Gegensatz dazu würde ein Evolutionärer Algorithmus, da er populationsbasiert arbeitet, im Allgemeinen vergleichsweise mehr Gütebewertungen erfordern, ohne ein besseres Ergebnis zu erzielen.

Ist die Gütelandschaft hingegen stark zerklüftet, weist sie also viele nicht differenzierbare Stellen oder auch Unstetigkeitsstellen auf, so ist ein Evolutionärer Algorithmus besser zur Lösung des Optimierungsproblems geeignet als ein Verfahren, das nach einer deterministischen Vorgehensweise arbeitet, wie etwa ein Gradientenverfahren oder das Hooke-Jeeves-Verfahren [HJ61]. Auch bei Vorliegen mehrerer Optima ist der Einsatz eines Evolutionären Algorithmus i. Allg. vorteilhaft, d. h. der damit verbundene größere Aufwand durch mehr Gütebewertungen ist gerechtfertigt. Ein Gradientenverfahren liefert nämlich bei einer Gütelandschaft mit mehreren Optima bei Wahl eines ungünstigen Startpunktes für den Optimierungsprozess oft nur in ein nahe gelegenes lokales Optimum.

Bei praktischen Optimierungsproblemen kann allerdings auch der Fall auftreten, dass Teile der Gütelandschaft stark zerklüftet und wiederum andere Teile freundlich sind. Dann ist die Auswahl eines geeigneten Verfahrens stark erschwert. Ebenso schwierig ist diese Auswahl, wenn die Gütelandschaft weitgehend unbekannt ist, was in praktischen Anwendungen ebenfalls auftreten kann. Um in diesem Fall ein geeignetes Optimierungsverfahren auszuwählen, ist zunächst eine umfangreiche Analyse der Gütelandschaft erforderlich. Dies ist jedoch aufwendig und eine eindeutige Entscheidung zugunsten eines bestimmten Optimierungsverfahrens ist keineswegs garantiert. Bei einem hochdimensionalen Optimierungsproblem, d. h. bei einer hohen Anzahl einzustellender Parameter und damit einer hohen Dimension des Vektors \mathbf{x} , ist eine solche Analyse eventuell praktisch unmöglich.

Bisher bekannte Varianten der Evolutionsstrategien arbeiten in der Regel so, dass der Optimierungsprozess schnell gegen ein einziges – lokales oder globales – Optimum in der Gütelandschaft konvergiert. Der Raum der zulässigen Vektoren \mathbf{x} – im Folgenden auch Suchraum S genannt – wird damit oft nicht ausreichend breit durchsucht. Insbesondere bei hochdimensionalen Optimierungsproblemen besteht die Gefahr, ein globales Optimum in der Gütelandschaft nicht zu finden und stattdessen nur ein lokales Optimum zu ermitteln.

Bei Vorliegen eines vektoriellen Gütemaßes \mathbf{Q} treten bei den bisher bekannten Evolutionsstrategien im Wesentlichen drei Schwierigkeiten auf:

1. Oft werden Optimierungsprobleme mit vektorielltem Gütemaß dadurch angegangen, dass aus den verschiedenen Gütekriterien beispielsweise durch eine gewichtete Summe ein skalares Gütemaß gebildet wird. Diese Vorgehensweise ist ungünstig, wenn man an der gesamten Pareto-Menge interessiert ist, denn dadurch, dass man schon im Voraus Gewichtungsfaktoren festlegt, wird das Optimierungsproblem auf die Suche nach einzelnen Punkten innerhalb der Pareto-Menge reduziert. Auch eine Variation der Gewichtungsfaktoren während des Optimierungsprozesses führt zu gewissen Nachteilen (s. Abschnitt 3.2). In [Kah91] wird eine Übersicht über bekannte Verfahren zur Lösung von Optimierungsproblemen mit vektorielltem Gütemaß durch die Lösung skalarer Ersatzprobleme gegeben. Laut J. KAHLERT existiert jedoch kein solches Verfahren, das einerseits nur Lösungen liefert, die zur Pareto-Menge gehören, andererseits aber durch geeignete Variation der Gewichtungsfaktoren jeden Punkt der Pareto-Menge erreichen kann.
2. Möchte man mehrere Gütekriterien simultan optimieren, so tritt das Problem sogenannter *nicht vergleichbarer* Individuen auf. Bezogen auf zwei Individuen bedeutet dies, dass eines der beiden Individuen hinsichtlich eines der Gütekriterien und das andere hinsichtlich eines anderen Gütekriteriums günstiger ist. Dann kann man also nicht entscheiden, welches der beiden Individuen ‘insgesamt besser’ ist und somit für den Optimierungsprozess weiterhin verwendet werden sollte. Zwar sind zur Problemlösung die beiden nachfolgenden Vorgehensweisen (a) und (b) einsetzbar, aber sie sind nicht frei von Nachteilen:
 - (a) Verwendet man alle paarweise nicht vergleichbaren Individuen weiterhin für den Optimierungsprozess, so kann dies zu einer übermäßigen Vergrößerung der Menge der Individuen und somit zu einem erheblich höheren Rechenaufwand innerhalb einer Generation (s. Abschnitt 2.2) führen. Als Hinweis sei hier genannt, dass die Pareto-Menge oft unendlich viele Vektoren \mathbf{x} beinhaltet und somit keine obere Grenze für die Anzahl der paarweise nicht vergleichbaren Individuen besteht.
 - (b) Entscheidet man sich dafür, während des Optimierungsprozesses mit einer konstanten Populationsgröße zu arbeiten, um den Rechenaufwand innerhalb einer Generation zu begrenzen, so kommt man in vielen Fällen zu der Situation, dass ein Individuum für den weiteren Optimierungsprozess im Interesse einer Konstanthaltung der Populations-

größe nicht mehr verwendet wird, obwohl es mit allen anderen noch verbliebenen Individuen paarweise nicht vergleichbar ist.

3. Auch bei Optimierungsproblemen mit mehreren Gütekriterien besteht das Problem, dass der Suchraum in vielen Fällen nicht breit genug durchsucht wird. Hierdurch wird es erschwert, die Pareto-Menge gut zu approximieren.

2.2 Optimierung mit Evolutionären Algorithmen

Im Laufe der natürlichen Evolution haben sich die unterschiedlichsten Lebensformen entwickelt und entwickeln sich fortlaufend weiter. Das Ziel des natürlichen Evolutionsprozesses besteht für jede Lebensform darin, ihr Überleben durch die Erschließung neuer Lebensräume und durch ständige Anpassung an sich ändernde Umweltbedingungen zu sichern.

C. DARWIN legte mit seinem 1859 erschienenen Buch *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life* den Grundstein der Evolutionstheorie [Mic04].

In der natürlichen Evolution treten die Vorgänge Fortpflanzung, Mutation und Selektion auf. Diese Vorgänge können durch Algorithmen in vereinfachter bzw. abgewandelter Form nachgeahmt und somit zur Lösung von Optimierungsproblemen insbesondere im technischen Bereich eingesetzt werden. Hierbei ist es nicht notwendig, die natürlichen Evolutionsschritte möglichst genau nachzubilden, sondern die natürlichen Vorgänge in geeigneter Weise auf *wesentliche* Ablaufschritte zu reduzieren.

Algorithmen, die die natürliche Evolution nachbilden, stellen leistungsfähige Optimierungsverfahren dar. Unter dem Oberbegriff *Evolutionäre Algorithmen* werden *Genetische Algorithmen*, *Evolutionary Programming* und *Evolutionstrategien* zusammengefasst. Genetische Algorithmen und Evolutionstrategien wurden unabhängig voneinander entwickelt. J. H. HOLLAND legte in den 1960er Jahren den Grundstein der Genetischen Algorithmen [Hol75, Gol89]. Nahezu zeitgleich entwickelte I. RECHENBERG ein Modell der natürlichen Evolution und damit die ersten Ansätze der Evolutionstrategien [Rec73]. Auch H.-P. SCHWEFEL trägt seit den 1970er Jahren wesentlich zur Weiterentwicklung der Evolutionstrategien bei [Sch77, Sch95b]. Ein Überblick über Genetische Algorithmen und Evolutionstrategien wird in [SHF94, Spr95, BS96, BHS97] gegeben. Bezüglich Evolutionary Programming sei auf [FOW66] verwiesen.

In der Natur ist der Aufbau jedes Lebewesens durch seinen genetischen

Code (Genotyp) bestimmt. Die Ausprägung der Eigenschaften, die durch diese Codierung festgelegt werden, wird Phänotyp genannt. Die Begriffe Genotyp und Phänotyp werden auch im Zusammenhang mit Evolutionären Algorithmen verwendet.

Zur Lösung von Optimierungsproblemen werden sowohl bei Genetischen Algorithmen als auch bei Evolutionsstrategien sogenannte Individuen verwendet. Jedes Individuum \mathbf{x} besteht aus einem Satz von skalaren Werten (x_1, \dots, x_n) . Jedes Individuum kann anschaulich als künstliches Lebewesen in einem n -dimensionalen Raum betrachtet werden. Eine Menge von Individuen wird als Population bezeichnet. Auf eine Population wirken die Programmschritte Rekombination, Mutation und Selektion – benannt nach den entsprechenden Vorgängen der natürlichen Evolution. Durch die Programmschritte Rekombination oder Mutation generierte Individuen werden auch Nachkommen oder Nachkommen-Individuen genannt, durch den Programmschritt Selektion für den nächsten Programmschritt Rekombination ausgewählte Individuen werden auch Eltern oder Eltern-Individuen genannt. Eine Folge der Programmschritte Rekombination, Mutation und Selektion wird als Generation bezeichnet.

Bei der Selektion werden Individuen als Eltern der nächsten Generation ausgewählt, die mit großer Wahrscheinlichkeit für den weiteren Optimierungsprozess von Nutzen sind. Die Grundlage dieser Auswahl bildet eine Bewertung der Individuen anhand eines skalaren oder vektoriellen Gütemaßes Q bzw. \mathbf{Q} . Es werden also beispielsweise diejenigen Individuen als Eltern der nächsten Generation ausgewählt, die im Sinne des Gütemaßes besonders günstig bewertet wurden.

Der wesentliche Unterschied zwischen Genetischen Algorithmen und Evolutionsstrategien besteht in der Codierung der Individuen. Während bei Genetischen Algorithmen üblicherweise eine Binär-Codierung der Individuen verwendet wird, arbeiten Evolutionsstrategien in der Regel auf der Basis reellwertig codierter Individuen. Bei Evolutionsstrategien ist also n die Anzahl der Parameter, die durch den Optimierungsprozess möglichst günstig im Sinne eines skalaren oder vektoriellen Gütemaßes Q bzw. \mathbf{Q} eingestellt werden sollen.

In [KS98] sind die Unterschiede zwischen Evolutionsstrategien und Genetischen Algorithmen anschaulich dargestellt. Hier wird auch auf die Unterscheidung zwischen Genotyp und Phänotyp genauer eingegangen.

Durch diese Unterschiede kommen auch den Programmschritten Rekombination und Mutation jeweils unterschiedliche Bedeutungen zu. Bei Genetischen Algorithmen werden Variationen hauptsächlich durch das sogenannte *Crossover* erreicht, das Auftrennen meist zweier Bitstrings und erneutes Zusammenfügen

der Teilstrings, die zuvor zu unterschiedlichen Individuen gehörten. Die Mutation, die zufällige Variation einzelner Bits, hat hier nur eine geringe Bedeutung. In Evolutionsstrategien kommt der Mutation eine größere Rolle zu.

Auch die Selektion unterscheidet sich üblicherweise bei beiden Verfahren. Während bei Genetischen Algorithmen meist stochastische Selektionsmechanismen eingesetzt werden, basieren Evolutionsstrategien meist auf deterministischen Selektionsmechanismen (s. Abschnitt 4.1.4).

Ein praktisches Anwendungsbeispiel für den Einsatz von Evolutionsstrategien ist die Optimierung von Kaffeemischungen [Her98]. Praktische Beispiele für die Verwendung Genetischer Algorithmen sind beispielsweise in [BHM⁺98, BL04] zu finden. In [BHM⁺98] wird die Optimierung von Molekülen durch einen an die Problemstellung angepassten und somit gegenüber einem Standardalgorithmus veränderten Genetischen Algorithmus vorgestellt. In [BL04] wird die Anwendung von Genetischen Algorithmen zur Reduktion der Modellordnung bei gleichzeitig möglichst guter Modellnachbildung für ein technisches System vorgestellt.

Im Folgenden wird hauptsächlich auf die Evolutionsstrategie näher eingegangen, weil sie als leistungsfähiges Optimierungsverfahren insbesondere zur Lösung multikriterieller Optimierungsprobleme (s. Kapitel 3.2) den Schwerpunkt dieser Arbeit bildet.

Oben wurde bereits das Problem erläutert, dass die Gütelandschaft in vielen praktischen Anwendungsfällen weitgehend unbekannt ist oder in bestimmten Bereichen eine andere Beschaffenheit aufweist als in anderen Bereichen. Somit werden durch den Einsatz eines Evolutionären Algorithmus unter Umständen zu viele Bewertungen mittels des Gütemaßes aufgewendet, obwohl beispielsweise der Einsatz eines Gradientenverfahrens für einige Bereiche der Gütelandschaft viel lohnender wäre.

Dieses Problem kann dadurch angegangen werden, dass die zufallsgesteuerten Suchschritte der Individuen innerhalb des Evolutionären Algorithmus in Größe und Richtung an die Beschaffenheit der Gütelandschaft angepasst werden. Im Rahmen der Evolutionsstrategien werden jedem Individuum \mathbf{x} als zusätzliche Strategieparameter sogenannte Schrittweiten zugewiesen, die im Schrittweitenvektor \mathbf{s} zusammengefasst werden. Häufig hat \mathbf{s} ebenso viele Komponenten wie \mathbf{x} . Jede Schrittweite stellt die Standardabweichung einer normalverteilten Wahrscheinlichkeitsdichte-Funktion dar. Die Wahrscheinlichkeitsdichte für einen Suchschritt eines Individuums wird für jede Komponente des Individuums nach dieser Wahrscheinlichkeitsdichte-Funktion bestimmt. Ein klei-

ner Suchschritt ist hierbei wahrscheinlicher als ein größerer Suchschritt. Die Wahrscheinlichkeitsdichte-Funktionen für die einzelnen Komponenten der Individuen werden zusammen bei jedem zufälligen Suchschritt berücksichtigt. Im Suchraum liegt somit jedes Individuum im Zentrum eines Hyperellipsoids, auf dem jeder Suchschritt gleich wahrscheinlich ist. Die Anpassung der einzelnen Schrittweiten und damit der Länge der Halbachsen der Hyperellipsoide während des Optimierungsprozesses kann auf unterschiedliche Weise erfolgen (s. Abschnitt 4.1).

Die Ausrichtung der Hyperellipsoide kann so erfolgen, dass die Halbachsen parallel zu den Koordinatenachsen der einzustellenden Parameter liegen. Dann kann nur eine Anpassung der zufälligen Suchschritte an die Gütelandschaft über die Länge der Halbachsen erfolgen.

Eine bessere Anpassung an die Beschaffenheit der Gütelandschaft kann durch die Einführung zusätzlicher Rotationswinkel der Hyperellipsoide erreicht werden [Rud92]. Die Halbachsen der Hyperellipsoide stimmen dann nicht mehr notwendigerweise mit den Koordinatenrichtungen überein. Nachteilig an dieser Vorgehensweise ist aber, dass sich zwar eine bevorzugte Richtung der zufälligen Suchschritte einstellen kann, die entgegengesetzte Richtung aber genauso wahrscheinlich ist.

Das bereits genannte Problem, dass der Suchraum nicht ausreichend breit durchsucht wird und somit ein eventuell vorhandenes globales Optimum nicht gefunden wird, wird von KAHLERT in [Kah91] und [Kah95] behandelt. Sogenannte Nischen-Individuen erhalten für den Programmschritt der Selektion eine Sonderstellung. Üblicherweise werden im Programmschritt Selektion die Individuen als Eltern der nächsten Generation ausgewählt, die besonders günstige Gütewerte aufweisen. Aus der Natur sind Lebewesen bekannt, die sich in einer *ökologischen Nische* angesiedelt haben. Dies kann beispielsweise die Bevorzugung einer ungewöhnlichen Nahrungsquelle oder eines ungewöhnlichen Lebensraumes sein. In [Kah91, Kah95] wird die algorithmische Nachahmung dieser natürlichen Strategie vorgeschlagen. Es wird eine bestimmte Anzahl von Nischen-Individuen festgelegt. Diese Nischen-Individuen werden für die nachfolgende Population ausgewählt, obwohl ihre Gütewerte nicht zu den günstigsten gehören, wenn sie einen relativ großen räumlichen Abstand zum derzeit besten Individuum aufweisen. Somit können mehrere lokale Optima gleichzeitig von Individuen besetzt werden. Die Gefahr, frühzeitig den Optimierungsprozess durch das Auffinden eines nur lokalen Optimums zu beenden, wird verringert. Nachteilig an der vorgestellten Vorgehensweise ist, dass die Anzahl der verwendeten Nischen-Individuen vom Anwender vorgegeben werden

muss, was bei einer unbekanntem Gütelandschaft nicht immer sinnvoll möglich ist. Außerdem können sich mehrere Nischen-Individuen in demselben lokalen Optimum ansammeln, was ebenfalls ungünstig ist und nur bedingt dafür sorgen kann, dass eine größere Anzahl lokaler Optima von Nischen-Individuen besetzt wird.

Es existieren auch hybride Verfahren, die sich beispielsweise aus einem klassischen analytischen oder numerischen Optimierungsverfahren und einem Evolutionären Algorithmus zusammensetzen. In [AL95] wird die Optimierung der Parametereinstellung eines Trinkwasser-Versorgungssystems vorgestellt. Da das Optimierungsproblem teilweise kontinuierliche und teilweise diskretwertige zu optimierende Parameter enthält, wurde hier ein hybrider Optimierungsalgorithmus eingesetzt.

Für die Lösung multikriterieller Optimierungsprobleme, bei denen mehrere Gütekriterien zu berücksichtigen sind (s. Abschnitt 3.2), gibt es zahlreiche Ansätze.

In [Kah91, Kah95] wird ein Verfahren vorgestellt, das sich zur Lösung multikriterieller Optimierungsprobleme eignet. Das Verfahren zeichnet sich durch eine *lokale* und eine *globale Selektion* aus.

Die lokale Selektion ist den Programmschritten Mutation und Rekombination – hier wurde die sonst übliche Reihenfolge der Programmschritte vertauscht – inhärent. Bei der Mutation wird über einen Mutations-Letalfaktor eingestellt, in welchem Maße der Gütewert des mutierten Individuums schlechter als der Gütewert des Individuums sein darf, aus dem es durch Mutation erzeugt wurde, um dieses in der Population zu ersetzen. Bei der Rekombination wird über einen Auswahlfaktor gesteuert, in welchem Bereich die Gütewerte der Individuen liegen müssen, die sich fortpflanzen dürfen. Über diesen Auswahlfaktor wird zum einen eine bestimmte Mindestgüte der zur Rekombination zugelassenen Individuen gewährleistet, zum anderen darf die Differenz der Gütewerte der herangezogenen Individuen in einem Rekombinationsschritt einen bestimmten Wert nicht überschreiten, d. h. eine bestimmte Ähnlichkeit der Gütewerte der verwendeten Individuen eines Rekombinationsschritts wird erzwungen.

Bei der globalen Selektion werden alle mutierten oder nicht mutierten Individuen nach dem Programmschritt Mutation sowie alle bei der Rekombination erzeugten Nachkommen berücksichtigt. Hierbei kann der Fall auftreten, dass von zwei nicht vergleichbaren Individuen eines nicht für den weiteren Optimierungsprozess verwendet wird.

In [SV01] wird die Optimierung von elektrischen Energieübertragungsnetzen behandelt. Es wird zwischen *Aggregationsmethoden* und *Pareto-basierten Methoden* unterscheiden.

Bei Aggregationsmethoden werden die einzelnen Gütekriterien als Funktionen formuliert, es wird eine Gewichtung der einzelnen Gütekriterien anhand von Gewichtungsfaktoren vorgenommen und die gewichteten Gütekriterien werden zu einem skalaren Gütemaß Q_{Ersatz} zusammengesetzt. Das multikriterielle Optimierungsproblem wird also, wie bereits in [Kah91] dargestellt, in ein skalares Optimierungsproblem (s. Abschnitt 3.1) überführt.

Bei den Pareto-basierten Methoden besteht das Ziel, die gesamte Pareto-Menge als Lösungsmenge zu erhalten (s. Abschnitt 3.2). In [SV01] werden vier Pareto-basierte Evolutionsstrategien vorgestellt, die in Abschnitt 4.1.4 detailliert dargestellt werden.

Als weitere wichtige Arbeiten bzw. Übersichtsbeiträge, bezogen auf Evolutionäre Algorithmen zur Lösung multikriterieller Optimierungsprobleme, sind [Deb01, ZT99, ZDT00, CDFK03, CCVVL02, CC99] zu nennen.

2.3 Ziele dieser Arbeit

Die Erarbeitung neuer Strategien zur Lösung von Optimierungsproblemen ist häufig aufgrund komplexer Anwendungen erforderlich, bei denen sich die Grenzen der bisher bekannten Optimierungsverfahren zeigen. Das dieser Arbeit hauptsächlich zugrunde liegende Optimierungsproblem besteht in der möglichst günstigen Einstellung der Parameter des Reglers eines industriellen Hydraulikventils der Firma Bosch Rexroth AG [BLSK00, SNK⁺00, NKBS01]. Dieser Regler weist je nach Variante ca. 20 einstellbare Parameter auf. In der vorliegenden Arbeit werden – initiiert durch die Schwierigkeiten, die bei der Lösung dieses komplexen Optimierungsproblems mit herkömmlichen Evolutionsstrategien aufgetreten sind – neue, vorteilhafte Strategieelemente für Evolutionsstrategien entwickelt. Hierzu wird zunächst die notwendige Vorarbeit demonstriert, die zur Lösung eines derart komplexen Optimierungsproblems häufig erforderlich ist. Von dieser – in vielen Anwendungsfällen sehr umfangreichen – Vorarbeit ist es auch abhängig, ob das Optimierungsproblem mit Hilfe eines anschließend eingesetzten Optimierungsverfahrens gelöst werden kann.

Die Festlegung der zu optimierenden Parameter sowie die Entwicklung eines Gütemaßes, das die an die Lösung des Optimierungsproblems gestellten Anforderungen widerspiegelt, sind zentrale Punkte der durchzuführenden Vorarbeiten. Bei der Entwicklung eines geeigneten Gütemaßes sind häufig viele, teilweise widersprüchliche Güteaspekte zu berücksichtigen.

Zu diesem Zweck wurde für das hier vorgestellte Anwendungsproblem u. a. ein Fuzzy-Gütemaß entwickelt, das sowohl Empfehlungen in Form von positiven Regeln als auch Warnungen bzw. Verbote in Form von negativen Regeln berücksichtigt [Kie97, Kie99].

Auch die Monotonie bezüglich des Eingangs–Ausgangsverhaltens des Gütemaßes muss gewährleistet sein, damit der Optimierungsprozess erfolgreich durchgeführt werden kann. Setzt sich das Gütemaß aus der Berücksichtigung mehrerer einzelner Güteaspekte zusammen und einer dieser Güteaspekte verschlechtert sich, während die restlichen Güteaspekte gleich bleiben, so sollte auch das Gütemaß einen schlechteren Gütewert liefern. Hiermit ist nicht notwendigerweise eine Monotonie der Gütelandschaft verbunden, denn durch die Gütelandschaft werden die Gütewerte nicht in Abhängigkeit einzelner Güteaspekte, sondern in Abhängigkeit der einzustellenden Parameter betrachtet.

Bei einem realen Anwendungsproblem spielen auch Störfaktoren, wie beispielsweise Messrauschen und variierende Systemeigenschaften, eine entscheidende Rolle. Es muss dafür gesorgt werden, dass diese Störeffekte möglichst gering bleiben. Außerdem muss bei der Auswahl eines geeigneten Optimierungsverfahrens darauf geachtet werden, dass geringe Störungen den Optimierungsverlauf nicht wesentlich beeinflussen.

Im Rahmen des genannten regelungstechnischen Anwendungsproblems wurden die bereits angesprochenen Anwendungsgrenzen bzw. Mängel der bisher bekannten Verfahren aufgedeckt. Diese Arbeit liefert einen Beitrag zur Beseitigung bzw. Milderung dieser Mängel.

Wie bereits erläutert, ist es bei unbekanntem Gütelandschaften schwierig, überhaupt ein geeignetes Optimierungsverfahren auszuwählen. Wie in [WM97] beschrieben, existiert kein Optimierungsverfahren, das für alle möglichen Optimierungsprobleme am besten geeignet ist (*no free lunch theorem*). In dieser Arbeit wird eine hybride Strategie, bestehend aus Evolutionsstrategie und Hooke-Jeeves-Verfahren (s. Abschnitt 6.3), zur Milderung dieses Entscheidungsproblems vorgestellt. Sie ermöglicht es, dass der Anwender nicht im Voraus entscheiden muss, ob der Optimierungsprozess mit Hilfe eines evolutionären Algorithmus oder unter Verwendung eines deterministischen Verfahrens

durchgeführt werden soll. Vielmehr findet eine Adaption des Optimierungsprozesses an die Beschaffenheit der Gütelandschaft statt. Die vorgenommene Einbettung des Hooke-Jeeves-Verfahrens gewährleistet außerdem, dass der Aufwand, der für die Evolutionsstrategie bzw. für das Hooke-Jeeves-Verfahren aufgewendet wird, durch den Erfolg des jeweiligen Verfahrens gesteuert wird.

Die mangelnde Breitensuche bisher bekannter Evolutionsstrategien wird durch die in den Abschnitten 6.1 und 6.2 vorgestellten Strategieelemente deutlich verbessert. Der Optimierungsprozess durchmustert bei Verwendung dieser Strategieelemente den Suchraum in der Regel vergleichsweise breiter. Somit wird die Wahrscheinlichkeit erhöht, ein globales Optimum auch tatsächlich zu finden. Durch weitere in den Abschnitten 6.1 und 6.2 vorgestellte Strategieelemente wird eine verbesserte Tiefensuche erreicht, d. h. unter bestimmten Bedingungen findet eine stärkere Konzentration auf ein ermitteltes Optimum statt. Das Zusammenspiel von Breiten- und Tiefensuche wurde gegenüber bekannten Evolutionsstrategien verbessert. Bei gleicher Anzahl an Bewertungen mittels des Gütemaßes wird somit in der Regel ein besserer Kompromiss zwischen Breiten- und Tiefensuche als bei bisher bekannten Verfahren erreicht.

Auch bei multikriteriellen Optimierungsproblemen wird die unzureichende Breitensuche bisher bekannter Verfahren durch die in den Abschnitten 6.1 und 6.2 beschriebenen Maßnahmen gemildert. Die Maßnahmen, die für den Fall eines skalaren Gütemaßes getroffen werden, sind in entsprechend abgewandelter Form auch bei multikriteriellen Optimierungsproblemen einsetzbar und führen auch bei Vorliegen eines vektoriellen Gütemaßes zu einem besseren Kompromiss zwischen Breiten- und Tiefensuche. Durch die verbesserte Breitensuche und die ausgewählte Verfahrensweise bei Vorliegen nicht vergleichbarer Individuen in einem Aussonderungsschritt innerhalb des Programmschritts Selektion (s. Abschnitt 4.1.4) kann die Pareto-Menge besser approximiert werden.

Kapitel 3

Klassifizierung von Optimierungsproblemen

Optimierungsprobleme lassen sich durch Vektoren einstellbarer Parameter $\mathbf{x} = (x_1, \dots, x_n)$ charakterisieren, für die jeweils ein oder mehrere Funktionswerte $Q(\mathbf{x})$ bzw. $\mathbf{Q}(\mathbf{x}) = (Q_1(\mathbf{x}), \dots, Q_k(\mathbf{x}))$ experimentell, simulatorisch oder analytisch ermittelt werden können. Die Funktion Q bzw. \mathbf{Q} wird Gütemaß genannt, die Auswertung der Funktion für einen bestimmten Vektor \mathbf{x} liefert einen oder mehrere Gütewerte. Im Rahmen der Evolutionären Algorithmen wird die Funktion Q bzw. \mathbf{Q} auch Fitnessfunktion genannt.

Für die einstellbaren Parameter x_1, \dots, x_n wird ein zulässiger Wertebereich, der sogenannte Suchraum S , festgelegt. Hierbei wird unterschieden zwischen einem unbegrenzten Suchraum

$$S = \{\mathbf{x} \in \mathbb{R}^n\} \quad (3.1)$$

und einem begrenzten Suchraum

$$S = \{\mathbf{x} \in \mathbb{R}^n \mid \forall i = 1, \dots, l : u_i(\mathbf{x}) \leq 0\}. \quad (3.2)$$

Einen Sonderfall des durch Gleichung (3.2) gegebenen Wertebereichs stellt die Begrenzung durch einen n -dimensionalen Hyperquader dar:

$$S = \{\mathbf{x} \in \mathbb{R}^n \mid \forall i = 1, \dots, n : x_{i,min} \leq x_i \leq x_{i,max}\} \quad (3.3)$$

mit $\mathbf{x}_{min} = (x_{1,min}, \dots, x_{n,min})$ und $\mathbf{x}_{max} = (x_{1,max}, \dots, x_{n,max})$.

Des Weiteren wird unterschieden zwischen einer Begrenzung durch *harte Nebenbedingungen* und durch *weiche Nebenbedingungen*.

Eine Begrenzung durch harte Nebenbedingungen darf im Laufe des Optimierungsprozesses nicht verletzt werden, d. h. während des Optimierungsprozesses können nur Vektoren \mathbf{x} mit Hilfe des Gütemaßes bewertet werden, die innerhalb des angegebenen Suchraums S liegen.

Eine Begrenzung durch weiche Nebenbedingungen darf während des Optimierungsprozesses verletzt werden, d. h. auch Vektoren \mathbf{x} , die außerhalb des Suchraums S liegen, können mittels des Gütemaßes bewertet werden. Die Gütewerte dieser Vektoren werden z. B. durch einen Strafterm abgewertet, der berücksichtigt, wie weit die Vektoren außerhalb von S liegen. Der Vorteil dieser Vorgehensweise besteht darin, dass am Rand des zulässigen Gebiets S liegende Optima leichter gefunden werden können.

Die Aufgabe besteht darin, für ein bestimmtes Gütemaß Q bzw. \mathbf{Q} die Werte der einzustellenden Parameter $\mathbf{x} = (x_1, \dots, x_n)$ zu finden, die zu minimalen oder maximalen Gütewerten führen. Aus Gründen der Übersichtlichkeit werden im Folgenden nur Minimierungsprobleme betrachtet. Jede Suche nach einem Maximum kann in ein Minimierungsproblem überführt werden.

Es existieren *deterministische* und *stochastische Optimierungsprobleme*. Bei den deterministischen Optimierungsproblemen kann nochmals unterschieden werden zwischen den *statischen Optimierungsproblemen* und den *dynamischen Optimierungsproblemen* [Pap91].

In dieser Arbeit werden statische Optimierungsprobleme betrachtet. Das bedeutet, dass Elemente des Euklidischen Raumes \mathbb{R}^n oder eines Unterraumes $S \subset \mathbb{R}^n$ gesucht werden, die zu minimalen Funktionswerten des Gütemaßes führen.

Bei dynamischen Optimierungsproblemen werden hingegen Funktionen $\mathbf{x}(t)$ einer unabhängigen Variable t , also Elemente des Hilbert-Raumes, gesucht.

Bei stochastischen Optimierungsproblemen werden für unbekannte Ereignisse Zufallsvariablen angesetzt, deren genauer Wert zwar unbekannt ist, von denen aber eine Wahrscheinlichkeitsdichte-Funktion bekannt ist. Das Ziel besteht dann darin, unter Berücksichtigung aller bekannten Gegebenheiten und unter Berücksichtigung der Wahrscheinlichkeitsdichte-Funktion der Zufallsvariablen die einzustellenden Parameter so festzulegen, dass der Erwartungswert des Gütemaßes einen minimalen Wert annimmt.

3.1 Skalare Optimierungsprobleme

Es liegt ein *skalares* Optimierungsproblem vor, wenn ein skalares Gütemaß verwendet wird, also jeder Vektor \mathbf{x} nur einen Gütewert $Q(\mathbf{x})$ zugewiesen bekommt.

Das Ziel besteht darin, Vektoren $\mathbf{x} \in S$ zu ermitteln, so dass gilt ¹:

$$Q(\mathbf{x}) \xrightarrow{!} \text{Min.} \quad (3.4)$$

Ein Punkt \mathbf{x}^* im Suchraum S kennzeichnet ein *globales Optimum*, wenn gilt:

$$\forall \mathbf{x} \in S : Q(\mathbf{x}^*) \leq Q(\mathbf{x}). \quad (3.5)$$

Wird Gleichung (3.5) nur für alle Vektoren \mathbf{x} in einer beliebig kleinen ε -Umgebung von \mathbf{x}^* erfüllt, so liegt ein *lokales Optimum* vor.

Liegt das Gütemaß Q in Form einer *unimodalen Funktion* vor, so existiert ein einziges globales Optimum innerhalb des Suchraums S . ²

Existieren zusätzlich zum globalen Optimum weitere relative Extrema oder Wendepunkte mit Steigung Null, so liegt eine *nicht-unimodale Funktion* als Gütemaß vor.

3.2 Multikriterielle Optimierungsprobleme

Bei einem *multikriteriellen* Optimierungsproblem kommt ein vektorielles Gütemaß zum Einsatz. Jeder Vektor \mathbf{x} wird dann unter Berücksichtigung mehrerer Gütekriterien $\mathbf{Q}(\mathbf{x}) = (Q_1(\mathbf{x}), \dots, Q_k(\mathbf{x}))$ bewertet.

Ein Sonderfall liegt dann vor, wenn durch geeignete Gewichtungsfaktoren g_1, \dots, g_k und anschließende Summenbildung über die einzelnen Gütekriterien eine skalare Ersatzfunktion gebildet werden kann:

$$Q_{\text{Ersatz}}(\mathbf{x}) = g_1 \cdot Q_1(\mathbf{x}) + \dots + g_k \cdot Q_k(\mathbf{x}). \quad (3.6)$$

Zur Lösung des Optimierungsproblems sind dann Verfahren zur Lösung skalarer Optimierungsprobleme einsetzbar. Bei dieser Vorgehensweise muss im Voraus

¹Es werden Minimierungsprobleme betrachtet. Jedes Maximierungsproblem kann in ein Minimierungsproblem überführt werden.

²Es wird vorausgesetzt, dass überhaupt ein Optimum existiert. Bei einem unbegrenzten Suchraum (Gleichung (3.1)) kann dies nicht garantiert werden. Bei einem begrenzten Suchraum nach Gleichung (3.3) kann die Existenz eines globalen Optimums bei einer stetigen Funktion Q nach dem Satz von Weierstraß garantiert werden [GZZ91].

eine geeignete Gewichtung der einzelnen Gütekriterien bekannt sein. Diese Gewichtung ist aber in vielen praktischen Anwendungsfällen schwierig.

Aus diesem Grund kann das Ziel bei der Lösung multikriterieller Optimierungsprobleme so formuliert werden, dass die gesamte Menge \mathcal{X} aller optimalen Lösungen im Sinne jeweils unterschiedlicher Kompromissbildungen möglichst genau durch den Optimierungsprozess approximiert werden soll.

Formal lässt sich dieses Optimierungsziel folgendermaßen darstellen:

$$\mathbf{Q}(\mathbf{x}) \xrightarrow{!} \text{Min.} \quad (3.7)$$

Die Lösungsmenge \mathcal{X} eines multikriteriellen Optimierungsproblems, die sogenannte *Pareto-Menge*, wird wie folgt definiert:

$$\mathbf{x}^* \in \mathcal{X} \Leftrightarrow \{\mathbf{x} \in S, \mathbf{x} \neq \mathbf{x}^* \mid \forall i = 1, \dots, k : Q_i(\mathbf{x}) \leq Q_i(\mathbf{x}^*)\} = \{\}. \quad (3.8)$$

Für einen Vektor \mathbf{x}^* , der zur Pareto-Menge gehört, gilt, dass bei Variation des Vektors im Suchraum eine Verbesserung bezüglich eines Gütekriteriums nur bei einer Verschlechterung hinsichtlich eines anderen Gütekriteriums erreichbar ist.

Abbildung 3.1 stellt die Gütwerte der zu zwei beispielhaften Pareto-Mengen gehörenden Parametervektoren \mathbf{x}^* bei Verwendung eines Gütemaßes, das zwei Gütekriterien Q_1 und Q_2 enthält, im Raum der Gütwerte – im Folgenden Güteraum genannt – dar. Vereinfachend wird im Folgenden auch die Abbildung $\mathbf{Q}(\mathcal{X})$ als Pareto-Menge bezeichnet. Abbildung 3.1 (a) stellt eine *konvexe* Pareto-Menge dar, in Abbildung 3.1 (b) ist eine *konkave* Pareto-Menge dargestellt.

Für $\min(Q_i(\mathbf{x}^*))$ mit $\mathbf{x}^* \in \mathcal{X}$ wird vereinfachend der Ausdruck $Q_{i,\min}$ verwendet. $Q_{i,\min}$ stimmt mit $\min(Q_i(\mathbf{x}) \mid \mathbf{x} \in S)$ überein.

Für $\max(Q_i(\mathbf{x}^*))$ mit $\mathbf{x}^* \in \mathcal{X}$ wird entsprechend der Ausdruck $Q_{i,\max}$ verwendet. $Q_{i,\max}$ stimmt hingegen *nicht* mit $\max(Q_i(\mathbf{x}) \mid \mathbf{x} \in S)$ überein, da sich $Q_{i,\max}$ *nur* auf die Gütwerte von Vektoren bezieht, die Teil der Pareto-Menge sind.

Abbildung 3.2 dient der Veranschaulichung der Begriffe ‘*dominierende Vektoren*’, ‘*dominierte Vektoren*’ sowie ‘*nicht vergleichbare Vektoren*’. Ein Vektor \mathbf{x} dominiert einen anderen Vektor \mathbf{x}_0 , wenn \mathbf{x} in allen Gütekriterien bessere oder

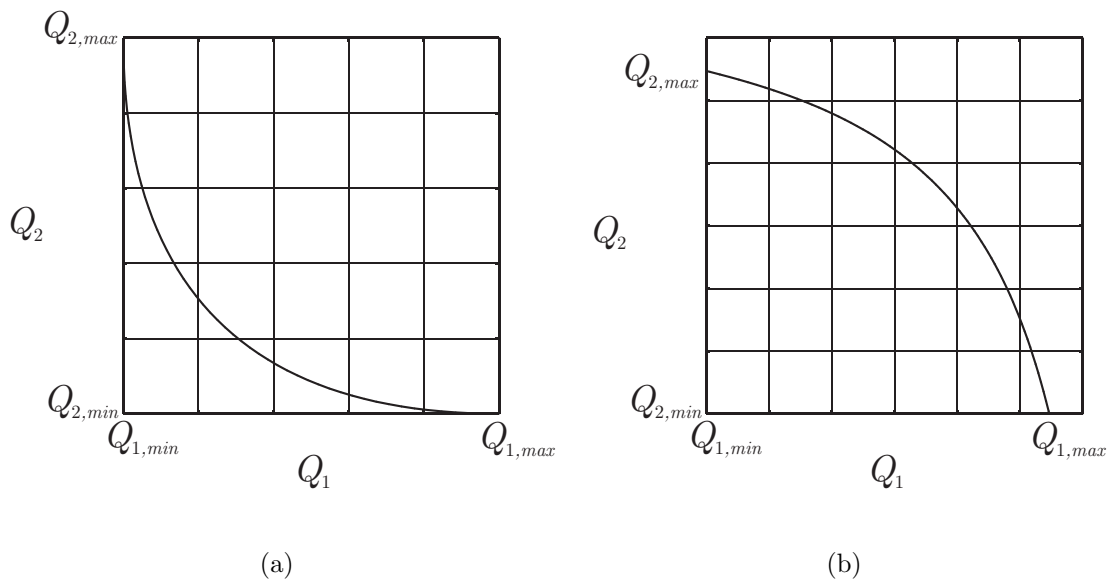


Abbildung 3.1: Konvexe Pareto-Menge (a) bzw. konkave Pareto-Menge (b) bei Verwendung zweier Gütekriterien Q_1 und Q_2 in der Q_1 - Q_2 -Ebene.

zumindest gleich gute Gütwerte aufweist wie \mathbf{x}_0 . \mathbf{x} wird von \mathbf{x}_0 dominiert, wenn \mathbf{x} in allen Gütekriterien schlechtere oder bestenfalls gleich gute Gütwerte aufweist wie \mathbf{x}_0 .³ Hat \mathbf{x} bezogen auf mindestens ein Gütekriterium einen besseren und bezogen auf mindestens ein Gütekriterium einen schlechteren Gütwert als \mathbf{x}_0 , so sind beide Vektoren nicht vergleichbar.

Von entscheidender Bedeutung für den Optimierungsprozess ist die Antwort auf die Frage, wie bei nicht vergleichbaren Vektoren entschieden werden kann, welcher der beiden Vektoren ‘besser’ im Sinne des Gütemaßes ist, d. h. welcher für den weiteren Optimierungsprozess berücksichtigt werden sollte. Ohne zusätzliche Informationen über das vorliegende Optimierungsproblem, z. B. in Form einer Gewichtung der einzelnen Gütekriterien, kann diese Frage nicht beantwortet werden.

Wird eine skalare Ersatzfunktion nach Gleichung (3.6) mit festen Gewichtungsfaktoren g_1, \dots, g_k verwendet, so wird ein Punkt der Pareto-Menge im Güteraum durch den Optimierungsprozess angenähert. Dieser Punkt $Q_{Ersatz,min}$ hängt von der Wahl der Gewichtungsfaktoren ab (s. Abbildung 3.3).

Durch Variation der Gewichtungsfaktoren während des Optimierungsprozesses in folgender Weise

³Der Sonderfall $\mathbf{Q}(\mathbf{x}) = \mathbf{Q}(\mathbf{x}_0)$ mit $\mathbf{x} \neq \mathbf{x}_0$ muss problemabhängig entschieden werden.

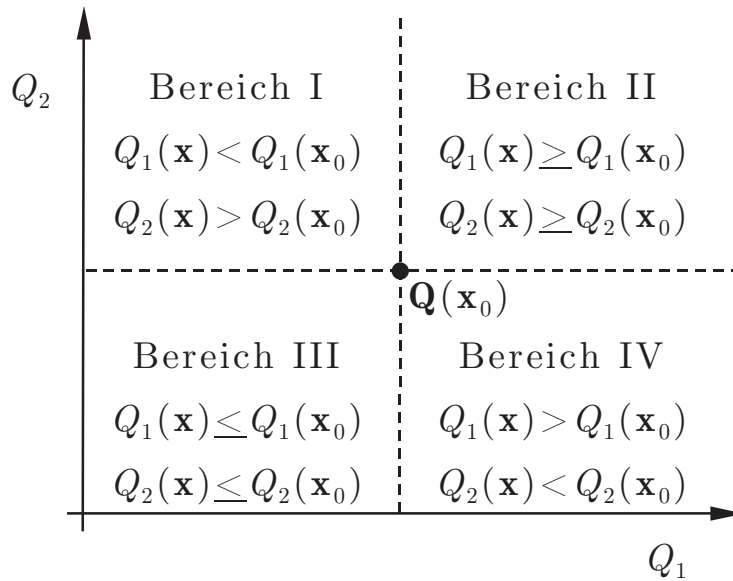


Abbildung 3.2: Darstellung der Bereiche, in denen die Güterwerte von Vektoren $\mathbf{x} \neq \mathbf{x}_0$ liegen, die \mathbf{x}_0 dominieren (Bereich III), die von \mathbf{x}_0 dominiert werden (Bereich II) und die mit \mathbf{x}_0 nicht vergleichbar sind (Bereiche I und IV) bei Verwendung zweier Gütekriterien Q_1 und Q_2 (s. Fußnote 3 auf Seite 25).

$$\forall i = 1, \dots, k : 0 \leq g_i \leq 1 \text{ mit } \sum_{i=1}^k g_i = 1 \quad (3.9)$$

können, wie durch Abbildung 3.3 veranschaulicht wird, bei einer konvexen Pareto-Menge prinzipiell alle Punkte dieser Menge im Güteraum approximiert werden. In der praktischen Anwendung sind dieser Approximation Grenzen gesetzt, da eine Auswertung des Gütemaßes für unendlich viele Vektoren \mathbf{x} bzw. Gewichtungsfaktoren g_1, \dots, g_k nicht möglich ist.

Bei einer konkaven Pareto-Menge hat die Verwendung einer skalaren Ersatzfunktion nach Gleichung (3.6) nicht den Effekt, dass alle Punkte der Pareto-Menge im Güteraum approximierbar sind. Durch den Optimierungsprozess sind dann die Punkte

$$(Q_{1,max}, Q_{2,min}, Q_{3,min}, \dots, Q_{k,min}),$$

$$(Q_{1,min}, Q_{2,max}, Q_{3,min}, \dots, Q_{k,min}),$$

⋮

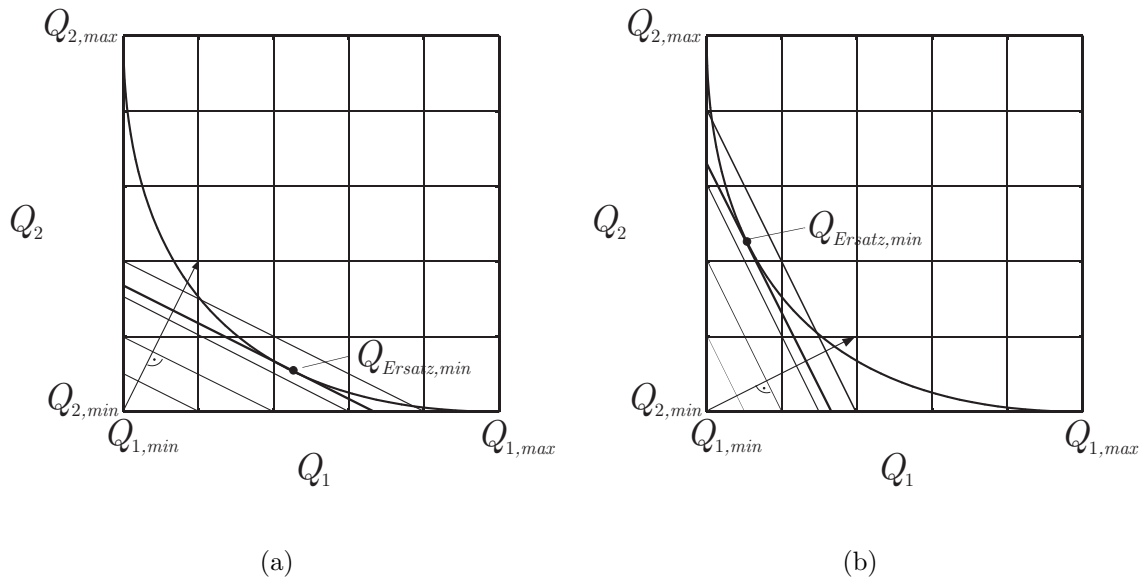


Abbildung 3.3: Optimum $Q_{Ersatz,min}$ bei Verwendung einer skalaren Ersatzfunktion nach Gleichung (3.6) mit $(g_1, g_2) = (0.33, 0.67)$ (a) bzw. $(g_1, g_2) = (0.67, 0.33)$ (b) bei Verwendung zweier Gütekriterien Q_1 und Q_2 und einer konvexen Pareto-Menge.

$$(Q_{1,min}, Q_{2,min}, \dots, Q_{k-1,min}, Q_{k,max})$$

erreichbar.

Ein allgemeinerer Ansatz für die skalare Ersatzfunktion kann folgendermaßen aufgestellt werden:

$$Q_{Ersatz}(\mathbf{x}) = \sum_{i=1}^k g_i \cdot (Q_i(\mathbf{x}))^{e_i} \quad \text{mit} \quad e_i \in \mathbb{R}^+ \quad \text{und} \quad \sum_{i=1}^k g_i = 1. \quad (3.10)$$

Für gleiche Werte $e_1 = e_2 = 2$ werden die Höhenlinien für die Minimierung der Ersatzfunktion, die in Abbildung 3.3 als parallele Geraden eingezeichnet sind, bei Wahl gleicher Gewichtungsfaktoren $g_1 = g_2 = 1/k = 0.5$ zu Kreisen, bei ungleichen Gewichtungsfaktoren ergeben sich elliptische Höhenlinien. In höherdimensionalen Güteräumen ergeben sich entsprechend Hyperellipsoide.

Durch die Wahl der Exponenten e_i wird die Krümmung der Höhenlinien der Ersatzfunktion beeinflusst, durch die Gewichtungsfaktoren g_i werden diese Höhenlinien, bezogen auf die Skalierung der Q_i -Achsen, gestaucht oder gestreckt. In Abbildung 3.4 sind Höhenlinien mit unterschiedlichen Werten $e_1 = e_2$ bei gleichen Gewichtungsfaktoren $g_1 = g_2 = 1/k = 0.5$ dargestellt. Hierdurch werden die Unterschiede im Krümmungsverhalten der Höhenlinien verdeutlicht.

Unter Verwendung des allgemeineren Ansatzes nach Gleichung (3.10) ist es also durchaus auch möglich, konkave Pareto-Mengen zu approximieren, wenn Werte

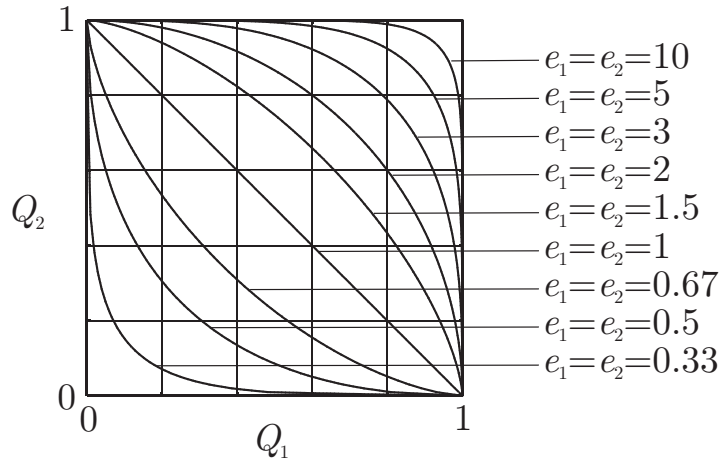


Abbildung 3.4: Höhenlinien der Ersatzfunktion nach Gleichung (3.10) mit Funktionswert $Q_{Ersatz} = 0.5$ mit $g_1 = g_2 = 1/k = 0.5$ und $e_1 = e_2$.

$e_i > 1$ gewählt werden. Alle Punkte einer konkaven Pareto-Menge im Güteraum sind durch Variation der Gewichtungsfaktoren g_i approximierbar, wenn die Krümmung der konkaven Pareto-Menge weniger stark ist als die Krümmung der Höhenlinien der Ersatzfunktion Q_{Ersatz} mit $e_i > 1$.

Hier zeigt sich der Nachteil dieses Ansatzes. Auf den ersten Blick scheint es für die Lösung eines wie auch immer gearteten Optimierungsproblems sinnvoll zu sein, die Werte der Exponenten e_i sehr groß zu wählen, bei $e_i \rightarrow \infty$ wäre eine Approximation jedes Punktes der Pareto-Menge bei jeder möglichen Krümmung der Pareto-Menge möglich. Diese Vorgehensweise würde aber zu stark erschwerten Bedingungen während des Optimierungsprozesses führen. Da die Höhenlinien der Ersatzfunktion dann über weite Bereiche parallel zu den Koordinatenachsen im Güteraum verlaufen würden, könnte es vermehrt vorkommen, dass Parametervektoren \mathbf{x} bei einem identischen und ansonsten stark unterschiedlichen Gütewerten Q_i nahezu den gleichen Wert $Q_{Ersatz}(\mathbf{x})$ erhalten. Diese Gegebenheiten machen es sehr unwahrscheinlich, den prinzipiell gegebenen Vorteil – jeden Punkt der Pareto-Menge approximieren zu können – während des Optimierungsprozesses nutzen zu können.

Werden für die Exponenten e_i kleinere Werte verwendet, so ergibt sich das Problem, dass die Krümmung der Pareto-Menge vorab bekannt sein müsste, damit die Exponenten sinnvoll festgelegt werden können.

3.3 Verrauschte Optimierungsprobleme bzw. Optimierungsprobleme mit Drift

Bei praktischen Optimierungsproblemen wird mittels des Gütemaßes häufig die Beeinflussung eines realen Systems durch die einzustellenden Parameter x_1, \dots, x_n bewertet. Als Beispiel sei hier die Optimierung der Regelung eines industriellen Hydraulikventils genannt (s. Kapitel 7). Bei einem realen System tritt oft Rauschen auf, d. h. auch bei identischer Parametereinstellung ist das Systemverhalten unterschiedlich. Somit erhält ein eingestellter Vektor \mathbf{x} bei mehreren Bewertungen mittels des Gütemaßes in der Regel leicht unterschiedliche Gütwerte. Das Rauschen des Systems darf allerdings nicht zu stark sein, anderenfalls wäre eine möglichst günstige Einstellung der Parameter x_1, \dots, x_n kaum möglich. Wird ein Optimierungsprozess durch Rauschen beeinflusst, so treten in der Regel lokale Maxima und Minima auch an Stellen im Suchraum auf, wo ohne Rauschen keine lokalen Extrema vorhanden wären. Dies kann den Optimierungsprozess erschweren und in ungünstigen Fällen dazu führen, dass eine Konvergenz gegen ein allein durch Rauschen entstandenes lokales Optimum eintritt.

Bei Vorliegen von Rauschen können mehrere Zielsetzungen, auch in Kombination miteinander, verfolgt werden. Durch geeignete Maßnahmen kann der Einfluss von Rauschen reduziert werden. Eine solche Maßnahme kann beispielsweise darin bestehen, die Bewertung eines Vektors \mathbf{x} mehrfach durchzuführen und die erhaltenen Gütwerte durch Mittelwertbildung miteinander zu verrechnen. Hierbei muss im Vorfeld analysiert werden, wie groß der Aufwand für die Bewertung eines Parametersatzes sein darf und wie groß der entsprechende Nutzen der Gütemittelung ist. Auch eine Glättung der zur Gütebewertung herangezogenen Signale kann die Auswirkung von Rauschen, insbesondere auch Messrauschen, auf den Optimierungsprozess deutlich reduzieren. Hierbei führt eine stärkere Glättung zu einer größeren Reduzierung des Rauschens, aber ebenfalls zu einer zunehmenden Verfälschung der betrachteten Signale. Auch bei der Auswahl eines Optimierungsverfahrens sollte das Vorliegen von Rauschen berücksichtigt werden. Gegebenenfalls ist es sinnvoll, ein Optimierungsverfahren durch geeignete Modifikationen an ein verrauschtes Optimierungsproblem anzupassen. Bei Auftreten von Rauschen haben sich heuristische Optimierungsverfahren (s. Kapitel 4) vielfach bewährt. Ein weiteres Ziel bei Vorliegen von Rauschen kann darin bestehen, gerade solche Parametersätze \mathbf{x} zu ermitteln, die sich durch besonders kleine Variationen der Systemeigenschaften auszeichnen.

Neben Rauschen kann auch eine Drift des Systemverhaltens auftreten. Beispielsweise durch Erwärmung oder Alterung können sich die Eigenschaften eines rea-

len Systems langsam in eine bestimmte Richtung verändern. Dieses Verhalten sollte ebenfalls bei der Auswahl des Optimierungsverfahrens berücksichtigt werden. Hierbei kann beispielsweise eine Neubewertung bestimmter Vektoren \mathbf{x} im Laufe des Optimierungsprozesses vorgenommen werden. Auch hier kann das Ziel darin bestehen, Vektoren \mathbf{x} für die einzustellenden Parameter zu finden, die für weite Bereiche der Drift zu einem günstigen Systemverhalten führen. Handelt es sich bei der Drift um einen relativ langsamen Prozess, wie er beispielsweise durch Alterung bestimmter Systemkomponenten hervorgerufen werden kann, so kann auch eine Optimierung ohne Berücksichtigung der Drift mit nachträglicher Anpassung der Parameter in bestimmten zeitlichen Abständen sinnvoll sein.

Die Analyse des vorliegenden Optimierungsproblems hinsichtlich Rauschen und Drift sowie die Festlegung der geeigneten Maßnahmen ist Teil der notwendigen und aufwendigen Vorarbeiten zur Lösung eines realen Optimierungsproblems.

Kapitel 4

Schwachstellen-Analyse bekannter Optimierungsverfahren

Optimierungsverfahren können unterteilt werden in die Gruppe der *deterministischen Verfahren* und die Gruppe der *stochastischen Verfahren*. Hierbei ist allerdings zu beachten, dass diese Einteilung nicht immer einheitlich vorgenommen wird.

Während in [Pap91] unter deterministischen Verfahren solche Verfahren verstanden werden, die zur Lösung deterministischer Optimierungsprobleme verwendet werden (s. Kapitel 3), werden in [Kah91] unter deterministischen Verfahren Verfahren verstanden, die nach einer deterministischen Vorgehensweise arbeiten. In [Kah91] wird bei den deterministischen Verfahren zwischen *Gradientenverfahren*, *Newton-Verfahren* und *direkten Verfahren*, die auch *Suchverfahren* genannt werden, unterschieden.

Auch bezogen auf die Gruppe der stochastischen Verfahren besteht keine Einheitlichkeit. In [Pap91] werden unter stochastischen Verfahren solche Verfahren verstanden, die zur Lösung stochastischer Optimierungsprobleme eingesetzt werden (s. Kapitel 3). Im Gegensatz hierzu werden in [Kah91] unter dem Begriff ‘stochastische Verfahren’ Verfahren zusammengefasst, deren Ablauf zufallsabhängige Elemente enthält. Es wird hier die Untergruppe *reine Zufallsverfahren* (*Monte-Carlo-Verfahren*), die Untergruppe der *Evolutionären Algorithmen* (*Genetische Algorithmen* und *Evolutionstrategien*) und die Untergruppe der Zwischenstufen, der sogenannten *Lern- und Vergessverfahren*, genannt.

In [Sch95a] wird eine Unterteilung in *klassische analytische und numerische Optimierungsverfahren*, in *heuristische Such- und Optimierungsverfahren* sowie in *hybride Such- und Optimierungsverfahren* vorgenommen. Unter klassischen Verfahren versteht man Verfahren, die auf einer Gradientenberechnung basieren. Heuristische Such- und Optimierungsverfahren werden auch *gezielte Zufallssuchverfahren* genannt. Hierunter werden Verfahren verstanden, die

sich an biologischen oder physikalischen Vorgängen orientieren, beispielsweise Evolutionäre Algorithmen oder *Simuliertes Kühlen* (*Simulated Annealing*¹ [MRR⁺53, KGV83, Fle95, KRO⁺03]). Hybride Verfahren stellen hier eine Verbindung zwischen klassischen und heuristischen Verfahren dar.

Des Weiteren kann zwischen *lokalen* und *globalen Verfahren* unterschieden werden. Lokale Verfahren führen eine lokale Optimierung in einem Teilraum des Suchraums S durch, globale Verfahren haben das Ziel und somit eine größere Wahrscheinlichkeit, das globale Optimum eines gegebenen Optimierungsproblems zu finden.

In [BBTZ04] wird ein allgemeines Konzept zur Kombination eines Evolutionären Algorithmus als globales Verfahren mit einem lokalen Verfahren vorgestellt. Der für das lokale Verfahren aufgebrauchte Aufwand wird hierbei über einen Parameter gesteuert. Dieses allgemeine Konzept ist unabhängig von der Wahl des Evolutionären Algorithmus und des lokalen Optimierungsverfahrens. Der genannte Parameter wird im Laufe des Optimierungsprozesses so verändert, dass zunehmend mehr Aufwand für das lokale Verfahren aufgewendet wird. Dieses Konzept wird als *Simulated Heating* bezeichnet.

4.1 Evolutionsstrategien (ES)

Die wesentliche Eigenschaft von Evolutionären Algorithmen besteht darin, dass diese populationsbasiert arbeiten, d. h. bei Evolutionsstrategien werden die einzelnen potenziellen Lösungen \mathbf{x} des Optimierungsproblems im Laufe des Optimierungsprozesses nicht unabhängig voneinander betrachtet und verändert. Vielmehr ist es so, dass häufig mehrere Vektoren \mathbf{x} simultan zur Erzeugung neuer potenzieller Lösungen herangezogen werden. Dies macht den Einsatz von Evolutionsstrategien bzw. allgemein den Einsatz Evolutionärer Algorithmen insbesondere zur Lösung von realen, durch Rauschen beeinflussten, Optimierungsproblemen (s. Abschnitt 3.3) besonders attraktiv.

Der übliche Ablauf einer Evolutionsstrategie ist in Abbildung 4.1 schematisch dargestellt. In den nachfolgenden Abschnitten werden die einzelnen Programmschritte detailliert erläutert. Nach der Initialisierung einer Startpopulation (Abschnitt 4.1.1) werden die Programmschritte Rekombination (Abschnitt 4.1.2),

¹Simulated Annealing ist ein Optimierungsverfahren, das zur Lösung kombinatorischer Optimierungsprobleme gut geeignet ist. Das Verfahren wurde aus Gesetzmäßigkeiten der Thermodynamik entwickelt [MRR⁺53, KGV83]. Beginnend mit einer anfänglich nahezu reinen Zufallssuche, bei der auch Verschlechterungen in Kauf genommen werden, wird durch Reduktion eines zur Temperatur analogen Parameters im Laufe des Optimierungsprozesses zunehmend dahingehend umgeschaltet, dass nur noch Verbesserungen der bereits gefundenen Lösung zugelassen werden [Fle95, KRO⁺03].

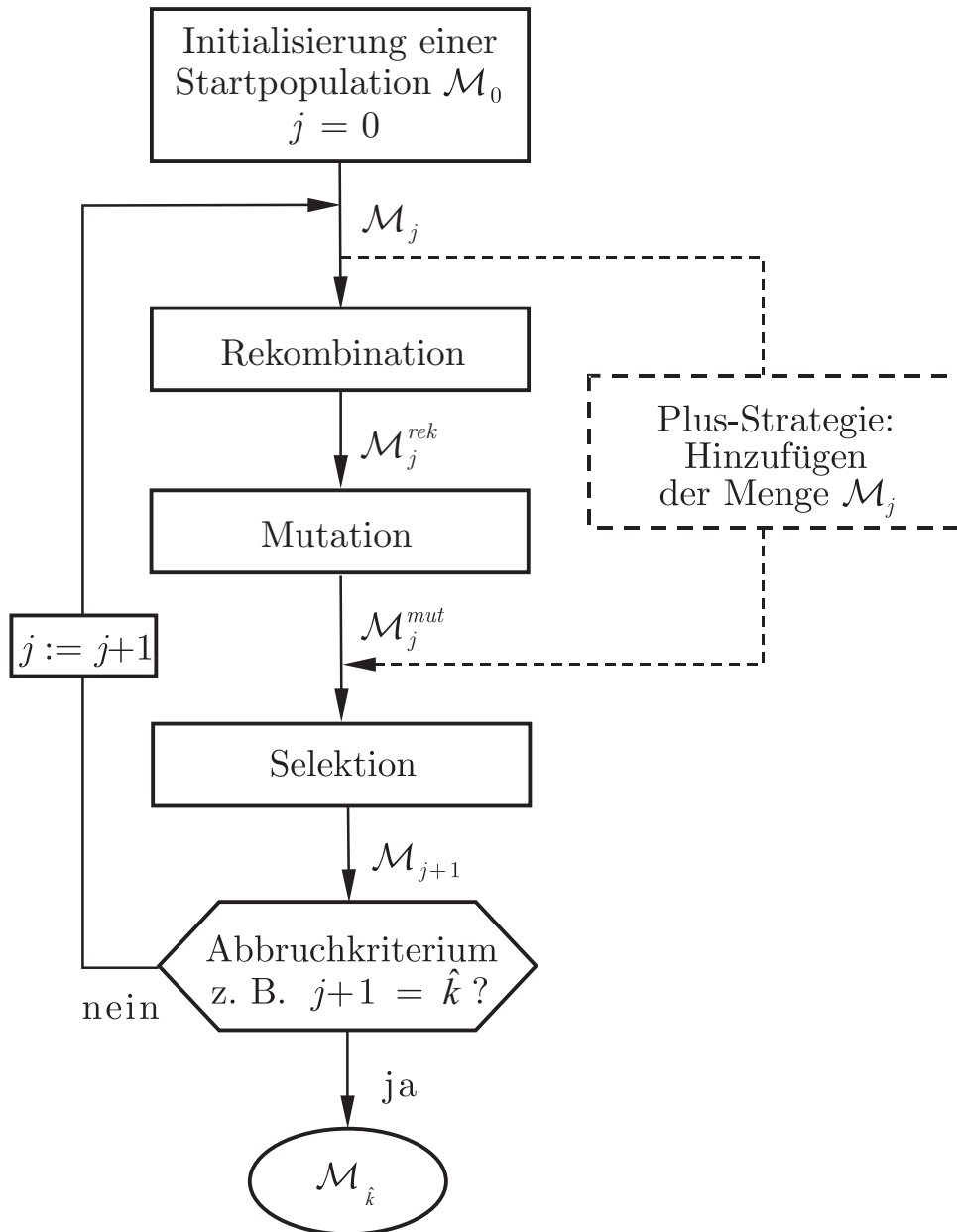


Abbildung 4.1: Üblicher Ablauf einer Evolutionsstrategie.

Mutation (Abschnitt 4.1.3) und Selektion (Abschnitt 4.1.4) so lange durchlaufen, bis eine Abbruchbedingung (Abschnitt 4.1.5) erfüllt ist. Ein Zyklus aus Rekombination, Mutation und Selektion wird Generation genannt. Jeder dieser Programmschritte führt zu einer neuen Menge von Individuen, einer sogenannten Population.

4.1.1 Generierung der Startpopulation

Der Ablauf der Evolutionsstrategie beginnt zunächst mit der Generierung einer Menge von Vektoren \mathbf{x} , der sogenannten Anfangs- oder Startpopulation \mathcal{M}_0 .

Sind durch den Optimierungsprozess n Parameter möglichst optimal einzustellen, so haben die einzelnen Vektoren \mathbf{x} jeweils n Komponenten $\mathbf{x} = (x_1, \dots, x_n)$. Jeder Vektor \mathbf{x} wird auch als Individuum bezeichnet.

Die Startpopulation kann zufallsgesteuert im Suchraum verteilt sein. In den meisten Fällen wird bei einem Hyperquader als Suchraum (s. Gleichung (3.3)) eine in diesem Hyperquader gleichverteilte Startpopulation festgelegt. Liegt keine Begrenzung des Parameterraums vor, so kann eine normalverteilte Startpopulation um einen ausgewählten Punkt im Suchraum erzeugt werden. In diesem Fall muss ein geeignetes Maß für die Streuung der Vektoren \mathbf{x} im Suchraum ermittelt und als Standardabweichung der Normalverteilung eingebracht werden. Liegt bereits Vorwissen über eine relativ gute Lösung des Optimierungsproblems vor, so kann eine normalverteilte Startpopulation um diese bereits bekannte Lösung ebenfalls sinnvoll sein. Andererseits besteht hierbei auch die Gefahr, dass der Optimierungsprozess gegen diese bereits bekannte Lösung konvergiert oder diese nur noch wenig verbessern kann, obwohl an anderen Stellen im Suchraum bessere Lösungen zu finden sind. In diesem Fall wäre der Suchraum nicht breit genug durchsucht worden.

Die Startpopulation kann auch deterministisch festgelegt werden. Sind bereits mehrere relativ gute Lösungen des Optimierungsproblems bekannt, so kann die Menge dieser Vektoren als Startpopulation verwendet werden. Allerdings besteht hierbei ebenfalls die oben genannte Gefahr der frühzeitigen Konvergenz gegen eine nur suboptimale Lösung. Auch die Festlegung der Startpopulation auf einem festen Raster im Suchraum ist denkbar. Hierbei sind aber um so mehr Vektoren \mathbf{x} zu berücksichtigen, je feiner das Raster festgelegt wird oder je mehr Parameter durch den Optimierungsprozess einzustellen sind. Ein sinnvolles Raster führt hierbei durch die Bestimmung der Gütewerte der einzelnen Vektoren \mathbf{x} häufig zu größerem Aufwand, als für den gesamten Optimierungsprozess vorgesehen ist.

In dieser Arbeit wird die Startpopulation teils gleichverteilt im Suchraum erzeugt, teilweise werden aber auch alle Individuen der Startpopulation auf einem einzigen Punkt im Suchraum platziert, um beim Vergleich neuer Strategieelemente für den Optimierungsprozess erschwerte Bedingungen zu schaffen.

Für den Programmschritt der Mutation, der an späterer Stelle detailliert erläutert wird (s. Abschnitt 4.1.3), ist die sogenannte Schrittweite der Individuen von entscheidender Bedeutung.

Die Start-Schrittweiten der Individuen werden bei der Generierung der Startpopulation ebenfalls erzeugt. Hierbei kann eine globale Schrittweite für alle In-

dividuen der Startpopulation \mathcal{M}_0 verwendet werden. Diese globale Schrittweite kann sowohl skalar als auch vektoriell sein.

Üblicher ist es jedoch, jedem Individuum eine individuelle Schrittweite zuzuordnen. Auch hier kann zwischen einer skalaren Schrittweite s und einer vektoriellen Schrittweite \mathbf{s} unterschieden werden. Wird eine vektorielle Schrittweite verwendet, so ist es in vielen Anwendungsfällen günstig, für jede Komponente eines Individuums eine eigene Schrittweitenkomponente vorzusehen: $\mathbf{s} = (s_1, \dots, s_n)$. Der Vorteil besteht darin, dass für jedes Individuum je nach Position im Parameterraum eine individuelle Anpassung der Suchschritte in jeder Dimension des Parameterraums vorgenommen werden kann. Es können aber auch weniger als n Schrittweitenkomponenten pro Individuum verwendet werden.

In der Regel werden die Start-Schrittweiten ebenfalls zufallsabhängig, meist nach einer Normalverteilung, generiert. Hierbei muss wiederum eine sinnvolle Standardabweichung der Schrittweiten berücksichtigt werden.

Bei der in dieser Arbeit verwendeten Evolutionsstrategie wird für jedes Individuum ein Schrittweitenvektor, bestehend aus n Komponenten, verwendet, wobei jede Komponente des Start-Schrittweitenvektors eines Individuums folgendermaßen erzeugt wird:

$$s_i = |\sigma_{NORM\ i} \cdot N_i(0, 1)| \quad \text{mit } i = 1, \dots, n. \quad (4.1)$$

Hierbei ist $N(0, 1)$ eine normalverteilte, skalare Zufallsvariable mit Erwartungswert Null und Standardabweichung Eins. $N_i(0, 1)$ ist der Ausdruck für eine solche Zufallsvariable, deren Wert für jeden möglichen Wert von i neu erzeugt wird. Allgemein ist $N(a, b)$ eine Zufallsvariable, die eine normalverteilte Wahrscheinlichkeitsdichte P annimmt:

$$P(s) = \frac{1}{\sqrt{2 \cdot \pi \cdot b}} \cdot e^{-\frac{1}{2} \frac{(s-a)^2}{b^2}}. \quad (4.2)$$

$\sigma_{NORM\ i}$ ist eine auf die Breite des Suchraums in der Dimension i normierte Standardabweichung:

$$\sigma_{NORM\ i} = 0.05 \cdot (x_{i,max} - x_{i,min}). \quad (4.3)$$

Die Schrittweitenvektoren werden in dieser Arbeit komponentenweise durch $\mathbf{s}_{min} = 0.001 \cdot (\mathbf{x}_{max} - \mathbf{x}_{min})$ und $\mathbf{s}_{max} = 0.1 \cdot (\mathbf{x}_{max} - \mathbf{x}_{min})$ begrenzt. Kleinere oder größere Suchschritte sind dennoch möglich, da die Schrittweitenvektoren der einzelnen Individuen die Standardabweichungen der im Programmschritt Mutation auftretenden normalverteilten Suchschritte festlegen (s. Abschnitt 4.1.3).

Bei einigen Evolutionsstrategien wird die sogenannte *Korrelierte Mutation* [Rud92] eingesetzt (s. Abschnitt 4.1.3). Dann wird zusätzlich zu dem Schrittweitenvektor \mathbf{s} ein Vektor von Drehwinkeln \mathbf{a} für die Mutation der Individuen verwendet. Die Start-Drehwinkel werden dann ebenfalls zusammen mit der Startpopulation generiert.

Die Schrittweiten und Drehwinkel, die einem Individuum zugeordnet werden, werden zusammenfassend als *Strategieparameter* des Individuums bezeichnet.

Bei der im Rahmen dieser Arbeit verwendeten Evolutionsstrategie wird keine Korrelierte Mutation eingesetzt.

Ob es sinnvoll ist, eine globale Schrittweite für alle Individuen oder individuelle Schrittweiten zu verwenden, ob skalare oder vektorielle Schrittweiten vorgesehen werden sollten, ist abhängig vom jeweiligen Anwendungsfall. Auch der Einsatz der Korrelierten Mutation kann je nach Optimierungsproblem mehr oder weniger erfolgsversprechend sein.

4.1.2 Rekombination

Bei der Rekombination – die häufig auch Fortpflanzung genannt wird – der Generation j werden neue potenzielle Lösungen des Optimierungsproblems üblicherweise aus mehreren bereits bestehenden Vektoren \mathbf{x} der Menge \mathcal{M}_j erzeugt. Die Menge \mathcal{M}_j wird auch Elternpopulation der Generation j genannt, die einzelnen Vektoren \mathbf{x} dieser Menge werden auch als Eltern-Individuen bezeichnet. Die Erzeugung neuer Individuen geschieht in der Regel zufallsabhängig.

Auch die Strategieparameter jedes neu erzeugten Individuums werden bei der Rekombination in Abhängigkeit von den Strategieparametern der Eltern-Individuen, die zur Erzeugung dieses Individuums verwendet wurden, ebenfalls zufallsabhängig neu generiert.

Im Wesentlichen wird unterschieden zwischen der *Diskreten Rekombination* und der *Intermediären Rekombination*. Es sind aber auch Zwischenformen möglich.

Im Folgenden wird sowohl die diskrete als auch die intermediäre Rekombination anhand der beteiligten Individuen \mathbf{x} dargestellt. Für die Strategieparameter findet die Rekombination in analoger Weise statt.

Hierbei sei $G_{[0..1]}$ eine gleichverteilte eindimensionale Zufallsvariable im Bereich $[0 \dots 1]$. G_i $_{[0..1]}$ bedeutet, dass der Wert dieser Zufallsvariablen für jeden möglichen

chen Wert des Zählers i neu erzeugt wird. Die Zufallsvariable $G_{[0...1]}$ nimmt also folgende Wahrscheinlichkeitsdichte an:

$$P(z) = \begin{cases} 1 & \text{wenn } 0 \leq z \leq 1, \\ 0 & \text{sonst.} \end{cases} \quad (4.4)$$

Werden beispielsweise in jedem Rekombinationsschritt zwei Eltern-Individuen

$$\mathbf{x}_{\text{Elter 1}} = (x_{1,\text{Elter 1}}, \dots, x_{n,\text{Elter 1}})$$

$$\mathbf{x}_{\text{Elter 2}} = (x_{1,\text{Elter 2}}, \dots, x_{n,\text{Elter 2}})$$

zur Erzeugung neuer Individuen herangezogen, so erfolgt die diskrete Rekombination häufig folgendermaßen:

$$\begin{aligned} z_{i,\text{Nachkomme}} &= G_{i[0...1]} \quad \text{für } i = 1, \dots, n \\ x_{i,\text{Nachkomme}} &= \begin{cases} x_{i,\text{Elter 1}} & \text{wenn } z_{i,\text{Nachkomme}} < 0.5, \\ x_{i,\text{Elter 2}} & \text{sonst} \end{cases} \\ \mathbf{x}_{\text{Nachkomme}} &= (x_{1,\text{Nachkomme}}, \dots, x_{n,\text{Nachkomme}}). \end{aligned} \quad (4.5)$$

Somit setzt sich jedes Nachkommen-Individuum zufallsabhängig aus den einzelnen Komponenten der Eltern-Individuen zusammen. Der Sonderfall, dass ein Nachkommen-Individuum mit einem seiner Eltern-Individuen identisch ist, ist in Gleichung (4.5) enthalten, kann aber durch zusätzliche Abfragen verhindert werden.

Die intermediäre Rekombination kann wie folgt durchgeführt werden:

$$\begin{aligned} z_{\text{Nachkomme}} &= G_{[0...1]} \quad \forall i \\ \mathbf{x}_{\text{Nachkomme}} &= \mathbf{x}_{\text{Elter 1}} + z_{\text{Nachkomme}} \cdot (\mathbf{x}_{\text{Elter 2}} - \mathbf{x}_{\text{Elter 1}}). \end{aligned} \quad (4.6)$$

Hierbei liegt jedes Nachkommen-Individuum zufallsabhängig auf der Verbindungslinie der Eltern-Individuen im Suchraum. Auch bei dieser Rekombinationsmethode kann (wenn auch nur mit verschwindend geringer Wahrscheinlichkeit) das Nachkommen-Individuum mit einem seiner Eltern-Individuen übereinstimmen.

Bei identischen Eltern-Individuen findet durch den Rekombinationsschritt eine Reduplikation (Kopie) des Eltern-Individuums statt.

Es existiert im Suchraum S ein n -dimensionaler Hyperquader H_j , der alle Individuen der Menge \mathcal{M}_j einschließt und dabei das kleinstmögliche Volumen aufweist.

Mit

$$\hat{x}_{i,min} = \{min(x_i) \forall \mathbf{x} \in \mathcal{M}_j\} \quad \text{für } i = 1, \dots, n \quad (4.7)$$

und

$$\hat{x}_{i,max} = \{max(x_i) \forall \mathbf{x} \in \mathcal{M}_j\} \quad \text{für } i = 1, \dots, n \quad (4.8)$$

wird H_j wie folgt definiert:

$$H_j = \{\mathbf{x} \mid \forall i = 1, \dots, n : \hat{x}_{i,min} \leq x_i \leq \hat{x}_{i,max}\}. \quad (4.9)$$

Sämtliche durch die üblichen Rekombinationsmethoden erzeugten Individuen der Menge \mathcal{M}_j^{rek} liegen innerhalb des Gebiets H_j .

Es werden so oft neue Eltern-Individuen bestimmt und aus diesen Eltern-Individuen ein oder mehrere Nachkommen-Individuen erzeugt, bis eine bestimmte Anzahl an Nachkommen vorliegt. Diese λ Nachkommen bilden die Nachkommen-Population \mathcal{M}_j^{rek} . Da üblicherweise im Anschluss an den Programmschritt Rekombination der Programmschritt Mutation stattfindet, können identische Individuen in \mathcal{M}_j^{rek} zu unterschiedlichen Individuen in \mathcal{M}_j^{mut} werden.

Die Auswahl der mindestens zwei Eltern-Individuen geschieht üblicherweise ebenfalls zufallsgesteuert. Hierbei wird häufig mit gleicher Wahrscheinlichkeit für alle Eltern-Individuen gearbeitet, wie auch in der im Rahmen dieser Arbeit zu Vergleichszwecken herangezogenen Standard-Evolutionsstrategie (ES_{std} , s. Kapitel 6). Andere Ansätze verwenden nicht alle Eltern-Individuen mit gleicher Wahrscheinlichkeit für die Rekombination. In [Kah91] wird beispielsweise ein Ansatz beschrieben, bei dem die zur Rekombination zugelassenen Individuen eine einstellbare Mindestgüte aufweisen müssen². Die zur Rekombination herangezogenen Individuen werden aus allen zugelassenen Individuen mit gleicher Wahrscheinlichkeit zufallsabhängig bestimmt.

Für die Rekombination der zu optimierenden Parameter des Vektors \mathbf{x} wird nicht immer dieselbe Rekombinationsmethode wie für die Rekombination der zugehörigen Strategieparameter ausgewählt. Laut [Spr95] werden für die Rekombination der Individuen häufig eine diskrete Rekombination und für die Schrittweiten eine intermediäre Rekombination gewählt.

²Im Rahmen der Arbeiten [Kah91] und [Kah95] ist die Reihenfolge der Programmschritte einer Generation: Mutation, Rekombination, Selektion.

In der im Rahmen dieser Arbeit eingesetzten Evolutionsstrategie findet eine Kombination aus diskreter und intermediärer Rekombination statt. Vor jedem Programmschritt Rekombination wird zufallsabhängig festgelegt, ob für diese Generation diskrete oder intermediäre Rekombination verwendet wird (Wahrscheinlichkeit 0.9 für diskrete Rekombination, Wahrscheinlichkeit 0.1 für intermediäre Rekombination). Es werden in jedem einzelnen Rekombinationsschritt jeweils zwei Eltern-Individuen verwendet, wobei alle Eltern-Individuen zugelassen und identische Eltern-Individuen nicht zugelassen werden. Jedes Eltern-Individuum kann hierbei in mehreren Rekombinationsschritten verwendet werden. Aus diesen beiden Eltern-Individuen werden gemäß den obigen Vorschriften zwei – normalerweise unterschiedliche – Nachkommen-Individuen gebildet. Die Rekombination findet für die Strategieparameter in analoger Weise wie für die Individuen \mathbf{x} statt. Bei der Rekombination der Schrittweiten werden in jedem Rekombinationsschritt dieselben Werte $z_{i,Nachkomme}$ und $z_{Nachkomme}$ verwendet wie für die Rekombination der Individuen \mathbf{x} (vgl. Gleichungen (4.5) und (4.6)).

Insbesondere für die *Auswahl* der verwendeten Eltern-Individuen wurden im Rahmen dieser Arbeit mehrere neue Konzepte entwickelt (s. Abschnitte 6.1 und 6.2) und anhand beispielhafter Anwendungen untersucht.

4.1.3 Mutation

Im Anschluss an die Rekombination wird üblicherweise die Menge \mathcal{M}_j^{rek} dem Programmschritt Mutation unterworfen und dadurch die Menge \mathcal{M}_j^{mut} erzeugt. Hierbei können alle oder nur einige der Vektoren \mathbf{x} aus \mathcal{M}_j^{rek} durch die Mutation verändert werden.

Im Programmschritt Mutation werden an den bereits vorhandenen Individuen der Menge \mathcal{M}_j^{rek} zufällige Änderungen vorgenommen. Hierbei sind kleine Änderungen wahrscheinlicher als große. Bei Evolutionsstrategien wird die Mutation durch Addition eines n -dimensionalen Zufallsvektors zu einem bereits bestehenden Individuum \mathbf{x} realisiert. Üblicherweise wird hierbei ein normalverteilter Zufallsvektor mit Erwartungswert Null und vektorieller Standardabweichung verwendet. Es kann aber auch eine skalare Standardabweichung eingesetzt werden.

Die Mutation stellt eine wesentliche Komponente der Evolutionsstrategie dar, denn nur durch die Mutation ist es möglich, in bestimmte Bereiche des Suchraums S zu gelangen. Dies können beispielsweise Bereiche sein, die am Rand des Suchraums liegen. Bei den gängigen Rekombinationsmethoden treten nur

Nachkommen-Individuen auf, deren Parameterwerte zwischen denen der Eltern-Individuen liegen (intermediäre Rekombination) oder mit diesen identisch sind (diskrete Rekombination). Alle in der Generation j erzeugten Nachkommen liegen also in dem Gebiet H_j (s. Gleichung (4.9)). Die Mutation dient dazu, auch Individuen außerhalb dieses Gebiets zu erzeugen.

Von entscheidender Bedeutung für die erfolgreiche Wirkungsweise des Programmschritts Mutation ist eine geeignete Schrittweite für die zufallsgesteuerten Mutationsschritte. Zu große Schrittweiten führen dazu, dass die Mutation zu einer reinen Zufallskomponente der Evolutionsstrategie wird, zu kleine Schrittweiten bringen kaum Änderung und führen somit zu einer sehr geringen Konvergenzgeschwindigkeit gegen ein – möglicherweise globales – Optimum. Die Festlegung der Startschrittweiten erfolgt bereits bei Generierung der Startpopulation (s. Abschnitt 4.1.1). Im Programmschritt Rekombination werden neben den Nachkommen-Individuen der Menge \mathcal{M}_j^{rek} auch deren Schrittweiten aus den Schrittweiten der Eltern-Individuen generiert (s. Abschnitt 4.1.2).

Für die Veränderung der Schrittweiten sowie der Individuen der Menge \mathcal{M}_j^{rek} im Programmschritt Mutation existieren verschiedene Strategien.

Auf RECHENBERG ist die 1/5-Erfolgsregel zurückzuführen. Für zwei einfache skalare Gütemaße, das Korridormodell und das Kugelmodell, hat RECHENBERG theoretisch eine Regel zur adaptiven Steuerung der Schrittweiten hergeleitet [Rec73]. Er betrachtet hierbei eine zweigliedrige Evolutionsstrategie³.

Der Mutationsschritt erfolgt durch Addition eines n -dimensionalen Zufallsvektors, wobei eine skalare Schrittweite s als Standardabweichung der Normalverteilungen verwendet wird.

Mit

$$\mathbf{z}_{Mutation} = (N_1(0, s), \dots, N_n(0, s)) \quad (4.10)$$

wird der Mutationsschritt folgendermaßen vorgenommen:

$$\mathbf{x}_{Mutation} = \mathbf{x}_{Elter} + \mathbf{z}_{Mutation}. \quad (4.11)$$

In jeder Dimension ist $\mathbf{z}_{Mutation}$ normalverteilt (Gleichung (4.10)). Durch die Unabhängigkeit der einzelnen Komponenten von $\mathbf{z}_{Mutation}$ ergibt sich die Wahrscheinlichkeitsdichte für $\mathbf{x}_{Mutation}$ als n -dimensionale Normalverteilung:

³D. h.: Es existiert nur ein Individuum. Dieses Individuum wird abweichend von Abschnitt 4.1.2 als Eltern-Individuum bezeichnet. Das Individuum wird mutiert und der Gütewert des mutierten Individuums wird mit dem des Eltern-Individuums verglichen. Das bessere der beiden Individuen wird neues Eltern-Individuum. Es findet somit keine Rekombination statt.

$$P(\mathbf{x}_{Mutation}) = \left(\frac{1}{\sqrt{2 \cdot \pi \cdot s}} \right)^n \cdot e^{-\frac{1}{2} \sum_{i=1}^n \frac{(x_{i,Mutation} - x_{i,Elter})^2}{s^2}}. \quad (4.12)$$

Somit ergibt sich bei skalarer Schrittweite s die gleiche Wahrscheinlichkeitsdichte für einen Mutationsschritt auf konzentrischen Hyperkugeln um \mathbf{x}_{Elter} .

An dieser Stelle wird auf den Programmschritt Selektion vorgegriffen (s. Abschnitt 4.1.4), der bei RECHENBERG wie folgt durchgeführt wird:

$$\mathcal{M}_{j+1} = \begin{cases} \mathbf{x}_{Mutation} & \text{wenn } \mathbf{x}_{Mutation} \text{ günstigeren Gütewert aufweist,} \\ \mathbf{x}_{Elter} & \text{wenn } \mathbf{x}_{Elter} \text{ günstigeren Gütewert aufweist.} \end{cases} \quad (4.13)$$

Die genannte 1/5-Erfolgsregel lautet: "Während des Optimierungsprozesses sollte 1/5 der Mutationen erfolgreich sein, also zu einer Verbesserung des Gütewertes führen. Sind weniger als 1/5 der Mutationen erfolgreich, so ist die Schrittweite zu verkleinern, sind mehr als 1/5 der Mutationen erfolgreich, so ist die Schrittweite zu vergrößern."

Von SCHWEFEL wurde die 1/5-Erfolgsregel konkretisiert, d. h. es wurde eine algorithmische Umsetzung erarbeitet [Sch77].

Eine Erweiterung der Vorgehensweise von RECHENBERG stellen Mutationsschritte unter Verwendung eines n -dimensionalen Zufallsvektors mit *vektorieller* Schrittweite \mathbf{s} dar.

In diesem Zusammenhang geht auf SCHWEFEL das Verfahren der *Selbstadaptation der Schrittweiten* zurück [Sch77]. Hierbei wird bei jedem Mutationsschritt zunächst die Schrittweite des betrachteten Nachkommen-Individuums aus \mathcal{M}_j^{rek} mutiert, d. h. einer zufälligen Änderung unterworfen, und anschließend unter Verwendung dieser mutierten Schrittweite der eigentliche Mutationsschritt des betrachteten Individuums durchgeführt.

Zunächst wird die neue Schrittweite des zu mutierenden Nachkommen-Individuums erzeugt:

$$s_{i,Mutation} = s_{i,Nachkomme} \cdot e^{N(0,\tau_1) + N_i(0,\tau_2)} \quad \text{für } i = 1, \dots, n \quad (4.14)$$

mit

$$\tau_1 = \frac{c}{\sqrt{2 \cdot n}}, \quad \tau_2 = \frac{c}{\sqrt{2 \cdot \sqrt{n}}} \quad \text{und } c = 1. \quad (4.15)$$

Die Standardabweichungen der Normalverteilungen in Gleichung (4.14) τ_1 und τ_2 Gleichung (4.15) können dazu genutzt werden, den Optimierungsprozess an spezielle Gütefunktionen anzupassen. τ_1 wirkt global für alle Komponenten eines Individuums, τ_2 wirkt individuell für jede einzelne Komponente.

Das Nachkommen-Individuum wird anschließend nach folgender Vorschrift in einem Mutationsschritt verändert:

$$z_{i,Mutation} = N_i(0, s_{i,Mutation}) \quad \text{für } i = 1, \dots, n \quad (4.16)$$

$$\mathbf{x}_{Mutation} = \mathbf{x}_{Nachkomme} + \mathbf{z}_{Mutation}. \quad (4.17)$$

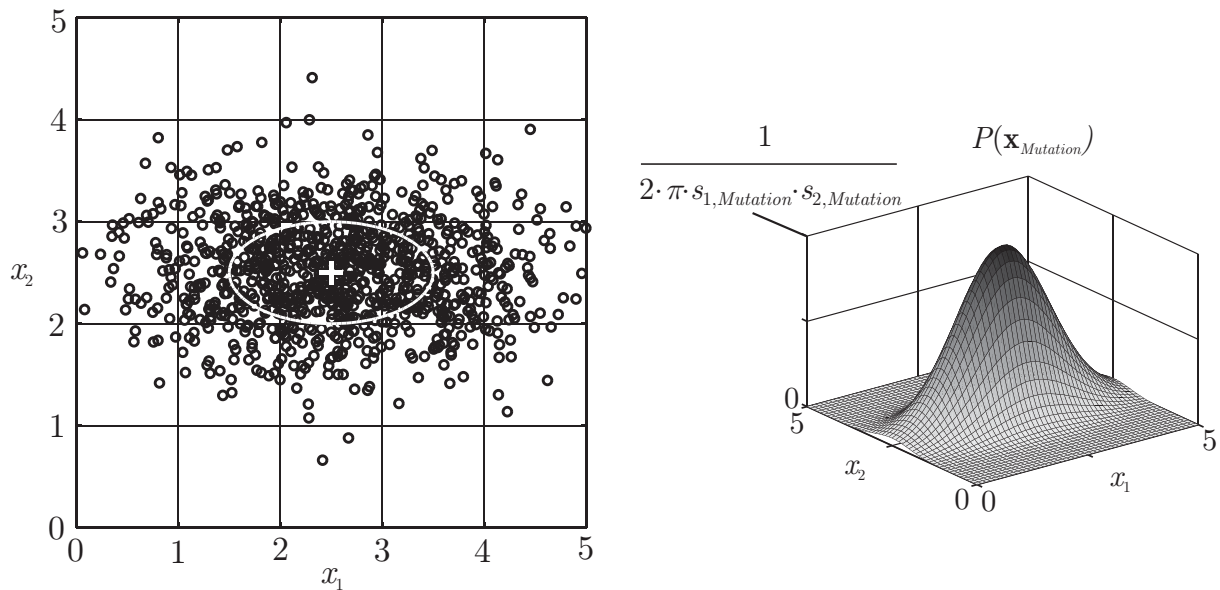
Nähere Ausführungen bezüglich obiger Vorgehensweise sind auch in [Bäc94, BS96] zu finden. Zusammenfassend sei zu der dargestellten Vorgehensweise genannt, dass durch die Multiplikation mit der natürlichen Exponentialfunktion in Gleichung (4.14) bei positiven Werten der Schrittweite $\mathbf{s}_{Nachkomme}$ auch die generierte Schrittweite $\mathbf{s}_{Mutation}$ nur positive Werte enthält. Da beide Normalverteilungen im Exponenten den Erwartungswert Null haben, ist der Erwartungswert des Multiplikationsfaktors, mit dem die Schrittweitenkomponente $s_{i,Nachkomme}$ multipliziert wird, Eins. So ist die Wahrscheinlichkeit für das Auftreten eines bestimmten Multiplikationsfaktors und seines reziproken Wertes gleich groß, d. h. es kommt ebenso häufig zu einer Vergrößerung wie zu einer Verkleinerung der Schrittweitenkomponente $s_{i,Mutation}$ gegenüber $s_{i,Nachkomme}$. Des Weiteren treten häufiger kleine Änderungen von $\mathbf{x}_{Mutation}$ gegenüber $\mathbf{x}_{Nachkomme}$ auf als große Änderungen.

Im Gegensatz zu Gleichung (4.12) ergibt sich hier gleiche Wahrscheinlichkeitsdichte für $\mathbf{x}_{Mutation}$ auf konzentrischen Hyperellipsoiden um $\mathbf{x}_{Nachkomme}$, da hier eine n -dimensionale Normalverteilung mit vektorieller Standardabweichung (durch die vektorielle Schrittweite $\mathbf{s}_{Mutation}$) vorliegt:

$$P(\mathbf{x}_{Mutation}) = \left(\prod_{i=1}^n \frac{1}{\sqrt{2 \cdot \pi \cdot s_{i,Mutation}}} \right) \cdot e^{-\frac{1}{2} \sum_{i=1}^n \frac{(x_{i,Mutation} - x_{i,Nachkomme})^2}{s_{i,Mutation}^2}}. \quad (4.18)$$

Abbildung 4.2 verdeutlicht Mutationsschritte durch Addition eines n -dimensionalen Zufallsvektors mit vektorieller Schrittweite $\mathbf{s}_{Mutation}$.

Die Strategie, die zu der von SCHWEFEL vorgestellten Vorgehensweise der Selbstadaption der Schrittweiten führte, ist, dass die Schrittweiten sich ohne äußere Beeinflussung aus dem Optimierungsprozess heraus auf günstige Werte



(a) Zufallsabhängige Mutationsschritte (schwarz) und Ellipse gleicher Wahrscheinlichkeitsdichte (weiß) für einen Mutationsschritt

(b) Wahrscheinlichkeitsdichte-Funktion für einen Mutationsschritt

Abbildung 4.2: Mutationsschritte mit vektorieller Schrittweite $\mathbf{s}_{Mutation} = (1.0, 0.5)$ ausgehend von $\mathbf{x}_{Nachkomme} = (2.5, 2.5)$.

einstellen sollen. SCHWEFEL weist darauf hin, dass sich die Schrittweiten, d. h. Streuungen, im Rahmen der natürlichen Evolution ebenfalls den Gegebenheiten angepasst haben und sich noch anpassen. Bei Einstellung von günstigen Schrittweiten werden auch die unter Verwendung dieser Schrittweiten erzeugten Mutationsschritte im Mittel zu Individuen mit günstigeren Gütewerten führen. Bei ungünstigen Schrittweiten werden auch die Mutationsschritte zu Individuen mit durchschnittlich schlechteren Gütewerten führen. Diese letzteren Individuen mit schlecht an die Gütelandschaft angepassten Schrittweiten werden vermehrt durch die anschließende Selektion (s. Abschnitt 4.1.4) für den weiteren Optimierungsprozess nicht berücksichtigt. Somit werden aber auch die zu diesen Individuen gehörenden Schrittweiten nicht mehr für den weiteren Optimierungsprozess verwendet. Die Individuen und ihre schlecht angepassten Schrittweiten ‘sterben aus’.

Die Halbachsen der durch Gleichung (4.18) gegebenen Hyperellipsoide gleicher Wahrscheinlichkeitsdichte sind in Richtung der x_i -Koordinatenachsen ausgerichtet. Durch die Vektoren $\mathbf{s}_{Mutation}$ kann die Länge der Halbachsen beeinflusst werden, nicht aber ihre Lage im Suchraum. Hier kann die Verwendung zusätz-

licher Rotationswinkel, die zu einer Kovarianzmatrix führen, Abhilfe schaffen, um die Hyperellipsoide gleicher Mutations-Wahrscheinlichkeitsdichte besser an die vorliegende Gütelandschaft anpassen zu können. Durch die Kovarianzmatrix werden die Hyperellipsoide im Suchraum gedreht. Dieses Verfahren wird *Korrelierte Mutation* genannt [Rud92].

Bei Verwendung der Korrelierten Mutation werden nicht nur die Schrittweiten s der Individuen, sondern auch die Drehwinkel \mathbf{a} im Programmschritt Mutation verändert, bevor mit den veränderten Strategieparametern neue Individuen generiert werden.

Bei der von SCHWEFEL entwickelten Vorgehensweise der Selbstadaption der Schrittweiten können folgende Nachteile auftreten:

- Auch kleine Schrittweiten können durch die Zufallsabhängigkeit große Mutationsschritte hervorrufen und umgekehrt. Die Selektion (s. Abschnitt 4.1.4) – d. h. Auswahl – eines Individuums, das durch den Programmschritt Mutation nur geringfügig verändert wurde, kann beispielsweise zu der ‘Vererbung’ einer zu großen Schrittweite führen. Dies kann vor allem bei kleinen Populationen zu Problemen führen.
- Die Variation der Schrittweiten zwischen konkurrierenden Individuen einer Generation entspricht im Mittel der Schrittweiten-Variation von einer Generation zur nächsten. Hier liegt ein Konflikt vor: Einerseits sollten die Individuen einer Generation unterschiedliche Schrittweiten haben, damit die Selektion einer günstigen Schrittweite erfolgen kann, andererseits dürfen die Änderungen der Schrittweiten in der Generationenfolge nicht zu groß sein, um zufallsbedingte Schwankungen zu vermeiden.

Um diese Nachteile zu beseitigen, wurde das Prinzip der *Derandomisierten Schrittweitenkontrolle* entwickelt [OGH94a]. Nähere Ausführungen zu dieser Schrittweitenanpassung sind auch in [OGH94b] zu finden, in [BNSW00] wird die Anwendung dieses Verfahrens zur Optimierung von Tragflächen vorgestellt.

Die Menge \mathcal{M}_j^{mut} wird entweder aus allen durch Mutationsschritte generierten Individuen $\mathbf{x}_{Mutation}$ oder aus allen Individuen $\mathbf{x}_{Mutation}$, die eine bestimmte Anforderung bezüglich des verwendeten Gütemaßes erfüllen, oder zu einem Teil aus Individuen $\mathbf{x}_{Mutation}$, die eine bestimmte Mindestgüte aufweisen, und zu einem Teil aus Individuen der Menge \mathcal{M}_j^{rek} gebildet.

Von KAHLERT wird beispielsweise ein sogenannter *Mutations-Letalfaktor* eingeführt, über den eingestellt wird, in welchem Maße sich das Individuum

$\mathbf{x}_{Mutation}$ gegenüber dem Individuum, aus dem es durch einen Mutationsschritt hervorgegangen ist, verschlechtern darf [Kah91, Kah95] (s. Fußnote 2 auf Seite 38).

Alle in dieser Arbeit eingesetzten Evolutionsstrategien verwenden eine Variante der Mutation, bei der alle λ Individuen der Menge \mathcal{M}_j^{rek} im Programmschritt Mutation einmalig modifiziert werden. Es kommt die Selbstadaption der Schrittweiten nach SCHWEFEL zum Einsatz. Die Menge \mathcal{M}_j^{mut} setzt sich zum einen aus allen Individuen $\mathbf{x}_{Mutation}$ zusammen, die, bezogen auf mindestens ein Gütekriterium, einen günstigeren Gütewert aufweisen als die Individuen $\mathbf{x}_{Nachkomme}$, aus denen sie generiert wurden. Um stets λ Individuen in der Menge \mathcal{M}_j^{mut} zu gewährleisten, werden zum anderen die Individuen $\mathbf{x}_{Nachkomme}$ zu der Menge \mathcal{M}_j^{mut} hinzugefügt, die, bezogen auf alle Gütekriterien, identische oder günstigere Gütewerte aufweisen als die Individuen $\mathbf{x}_{Mutation}$, die aus ihnen durch Mutationsschritte erzeugt wurden.

Die Menge \mathcal{M}_j^{mut} setzt sich also zunächst aus den gleichen Individuen zusammen wie die Menge \mathcal{M}_j^{rek} , wobei diejenigen Individuen $\mathbf{x}_{Nachkomme}$ durch die aus ihnen jeweils entstandenen Individuen $\mathbf{x}_{Mutation}$ ersetzt werden, die durch den Programmschritt Mutation, bezogen auf mindestens ein Gütekriterium, verbessert werden konnten. Diese Vorgehensweise resultiert aus der Überlegung, dass nur solche Individuen $\mathbf{x}_{Mutation}$ und die zugehörigen modifizierten Schrittweiten für den weiteren Optimierungsprozess verwendet werden sollen, die zu einer Verbesserung in mindestens einem Gütekriterium geführt haben.

4.1.4 Selektion

Das Ziel der Selektion besteht darin, aus einer größeren Menge von Individuen \mathbf{x} diejenigen auszuwählen, die für den weiteren Optimierungsprozess nützlich sein könnten und somit beibehalten werden sollen bzw. diejenigen Individuen auszusondern und zu eliminieren, die für den weiteren Optimierungsprozess voraussichtlich nicht nützlich sein werden. Die Prognose darüber, ob ein Vektor \mathbf{x} für den weiteren Optimierungsprozess verwendet werden soll, richtet sich danach, wie der Vektor mittels des Gütemaßes bewertet wird.

Es wird unterschieden zwischen einer *Plus-Strategie* und einer *Komma-Strategie*.

Bei einer Plus-Strategie ($(\mu + \lambda)$ -Strategie) wird in der Standard-Variante die Vereinigungsmenge der μ Eltern-Individuen der Menge \mathcal{M}_j und der λ Nachkommen-Individuen der Menge \mathcal{M}_j^{mut} gebildet. Aus dieser Vereinigungsmenge wird durch den Programmschritt Selektion die Menge \mathcal{M}_{j+1} gebildet.

Durch den Programmschritt Selektion werden also die für den weiteren Optimierungsprozess nützlichsten Individuen aus der Gruppe der Eltern-Individuen und der Nachkommen-Individuen der Menge \mathcal{M}_j^{mut} ausgewählt. Diese bilden die Menge der Eltern-Individuen der Generation $j + 1$. Bei dieser Vorgehensweise ist es möglich, dass ein Individuum über viele Generationen hinweg Teil des Optimierungsprozesses bleibt.

Wird hingegen eine Komma-Strategie ((μ, λ) -Strategie) eingesetzt, so wird der Programmschritt Selektion nur auf die λ Individuen der Menge \mathcal{M}_j^{mut} angewendet und aus dieser Menge wird durch den Programmschritt Selektion die Menge \mathcal{M}_{j+1} gebildet. Somit werden die Eltern-Individuen der Generation j nicht mehr im Programmschritt Selektion berücksichtigt. Es werden die nützlichsten Individuen aus der Menge der Nachkommen-Individuen aus \mathcal{M}_j^{mut} zu Eltern-Individuen für die Generation $j + 1$.

Es sind auch Mischformen aus beiden Strategien möglich. Beispielsweise kann die ‘Lebensdauer’ der Individuen bei Verwendung der Plus-Strategie auf eine bestimmte Anzahl an Generationen begrenzt werden. Hierdurch wird eine frühzeitige Konvergenz des Optimierungsprozesses gegen die Bereiche im Suchraum verhindert, die bereits von relativ gut bewerteten Individuen besetzt sind. Es wird also ermöglicht, dass bessere Bereiche innerhalb des Suchraums untersucht und von Individuen besetzt werden.

Weitere Einzelheiten zur formalen Darstellung der verschiedenen Plus- und Komma-Strategien sind in [SHF94] zu finden. Zur Festlegung der Strategieparameter μ und λ sei beispielsweise auf [Sch87] verwiesen. Für ein bestimmtes Gütemaß wird gezeigt, dass $\mu = 15$ und $\lambda = 100$ zu einem günstigen Verlauf des Optimierungsprozesses führen. SCHWEFEL verdeutlicht außerdem die Nachteile, die durch Anwendung einer $(\mu + \lambda)$ -Strategie entstehen können. Es kann vorkommen, dass Individuen mit ungünstigen Schrittweiten durch günstige Güterwerte nicht aus dem Optimierungsprozess ausgesondert werden. Dies ist für die Anpassung der Schrittweiten an die Gütelandschaft hinderlich. Durch Verwendung einer (μ, λ) -Strategie haben Individuen mit günstigen Güterwerten, aber schlecht angepassten Schrittweiten, eine weniger starke Wirkung auf den Optimierungsprozess. Beispielsweise durch Mutationsschritte, die zu schlechter bewerteten Individuen führen, werden die mutierten Individuen und damit die schlecht angepassten Schrittweiten aus dem Optimierungsprozess eliminiert. Der Vorteil der Komma-Strategie besteht somit darin, dass eine Anpassung der Schrittweiten erleichtert wird, nachteilig kann hingegen sein, dass bereits gefundene, als gut bewertete Vektoren \mathbf{x} wieder ‘vergessen’ werden, wenn sie mit einer ungünstigen Schrittweite verbunden sind.

Bei den im Rahmen dieser Arbeit eingesetzten Evolutionsstrategien wurde eine $(\mu + \lambda)$ -Strategie verwendet, damit bereits gefundene, als gut bewertete Vektoren \mathbf{x} den weiteren Optimierungsprozess positiv beeinflussen. Durch die neu entwickelten Strategieelemente zur verbesserten Breitensuche werden die Nachteile, die eine Plus-Strategie aufweist, relativiert.

Bei den existierenden Selektionsmechanismen wird unterschieden, ob ein skalares oder ein multikriterielles Optimierungsproblem vorliegt (s. Kapitel 3).

Bei skalaren Optimierungsproblemen besteht die gängigste Strategie in der *Selektion mit Ranking*. Hierbei werden alle Individuen der Menge \mathcal{M}_j^{mut} (Komma-Strategie) bzw. der Vereinigungsmenge $\mathcal{M}_j^{mut} \cup \mathcal{M}_j$ (Plus-Strategie) entsprechend ihrer Gütewerte sortiert. Die μ Individuen, die die günstigsten Gütewerte aufweisen, werden in der Menge \mathcal{M}_{j+1} zusammengefasst. Die Strategieparameter dieser μ besten Individuen werden unverändert übernommen. Diese Menge der Eltern-Individuen der Generation $j + 1$ ergibt sich also durch Auswahl der besten bzw. Elimination der schlechtesten Individuen.

Eine spezielle Variante der Selektion mit Ranking wird von KAHLERT vorgestellt [Kah91, Kah95]. Diese Variante wird *Selektion mit Einnischung* genannt. Das Ziel besteht darin, eine frühzeitige Konvergenz des Optimierungsprozesses gegen ein lokales Optimum zu verhindern und stattdessen durch die Besetzung mehrerer Optima mit Individuen erhöhte Chancen zu schaffen, ein globales Optimum zu ermitteln. Jedem Individuum \mathbf{x} der Menge, die dem Programmschritt Selektion zugeführt wird (s. Fußnote 2 auf Seite 38), wird ein sogenannter Einnischungs-Kennwert zugeordnet:

$$Q_{Nische}(\mathbf{x}) = \frac{Q(\mathbf{x}) - Q_{opt}}{\|\mathbf{x} - \mathbf{x}_{opt}\|_2^\beta} \quad \text{mit } \beta > 0, \mathbf{x} \neq \mathbf{x}_{opt}. \quad (4.19)$$

Hierbei ist \mathbf{x}_{opt} das Individuum aus der Menge, auf die der Programmschritt Selektion wirkt, das den besten Gütewert aufweist. Der zugehörige Gütewert ist Q_{opt} . Die Gütewerte der Nischen-Individuen sollen möglichst günstig sein, also einen möglichst geringen Zahlenwert haben. Aus diesem Grund befindet sich im Zähler die Differenz zum derzeit optimalen Gütewert. Gleichzeitig soll der räumliche Abstand der Nischen-Individuen zum derzeit besten Individuum möglichst groß sein. Somit ist der Euklidische Abstand zum derzeit besten Individuum im Nenner sinnvoll. Hier sei aber angemerkt, dass stark unterschiedliche Größenordnungen der zu optimierenden Parameter es erforderlich machen, diesen Euklidischen Abstand innerhalb eines auf die Bereichsbreiten des Suchraums $\mathbf{x}_{max} - \mathbf{x}_{min}$ normierten Raumes zu verwenden. Durch den Exponenten

β kann dieser Euklidische Abstand gewichtet werden. Laut KAHLERT ist ein Wert $\beta > 2$ empfehlenswert, da so das Verlassen einer quadratischen Senke ermöglicht wird und viele Gütelandschaften in der Umgebung eines Optimums näherungsweise quadratisch verlaufen.

Es werden so viele Nischen-Individuen ausgewählt, wie vom Anwender festgelegt wurde. Hierbei werden die Individuen mit den kleinsten Einnischungs-Kennwerten verwendet. Die Größe der Population, die aus dem Programmschritt Selektion hervorgeht, wurde ebenfalls festgelegt. Diese Population setzt sich aus den Nischen-Individuen und *den* Individuen der Menge, die dem Programmschritt Selektion zugeführt wurde, zusammen, die die günstigsten Gütewerte aufweisen.

Wie bereits oben erwähnt, weist diese Strategie zwei Nachteile auf: Die Anzahl der Nischen-Individuen muss vom Anwender vorgegeben werden, was bei einer unbekanntem Gütelandschaft schwierig ist. Des Weiteren kann der Fall auftreten, dass sich mehrere Nischen-Individuen in demselben lokalen Optimum ansammeln. Dieser Fall kann verhindern, dass eine größere Anzahl lokaler Optima besetzt wird.

Die sogenannte *Roulette-Wheel-Selektion* – auch *Proportionale Selektion* genannt – ist beim Einsatz Genetischer Algorithmen üblich [Spr95, Spr99]. Sie kann aber in geeignet modifizierter Weise auch bei Evolutionsstrategien eingesetzt werden. Hierbei geschieht die Auswahl der Individuen der Menge \mathcal{M}_{j+1} zufallsabhängig. Die Wahrscheinlichkeit, mit der ein Individuum als neues Eltern-Individuum ausgewählt wird, ist um so größer, je besser der ihm zugeordnete Gütewert ist. Im Folgenden wird die Roulette-Wheel-Selektion zur Bildung der Menge \mathcal{M}_{j+1} formal dargestellt:

Die Ausgangsmenge \mathcal{K}_j ist abhängig davon, ob eine Plus- oder eine Komma-Strategie eingesetzt wird:

$$\mathcal{K}_j = \begin{cases} \mathcal{M}_j^{mut} & \text{bei } (\mu, \lambda)\text{-Strategie,} \\ \mathcal{M}_j^{mut} \cup \mathcal{M}_j & \text{bei } (\mu + \lambda)\text{-Strategie.} \end{cases} \quad (4.20)$$

Von der Wahl einer Plus- oder Komma-Strategie hängt somit auch die Anzahl \tilde{k} der zur Menge \mathcal{K}_j gehörenden Individuen ab:

$$\tilde{k} = \begin{cases} \lambda & \text{bei } (\mu, \lambda)\text{-Strategie,} \\ \mu + \lambda & \text{bei } (\mu + \lambda)\text{-Strategie.} \end{cases} \quad (4.21)$$

Die Roulette-Wheel-Selektion läuft folgendermaßen ab:

1. $\bar{\mathcal{M}}_{j,0} = \mathcal{K}_j$ (Restmenge der noch nicht ausgewählten Individuen)

2. $\mathcal{P}_{j,0} = \{\}$ (Menge der ausgewählten Individuen)
3. Schleife 1: wiederhole für $i = 1, \dots, \mu$: (für jeden Auswahlsschritt)
 - Schleife 2: wiederhole für $\hat{j} = 1, \dots, \tilde{k} - i + 1$: (für jedes der noch nicht ausgewählten Individuen)

$$CDF(\hat{j}) = \frac{\sum_{\nu=1}^{\hat{j}-1} \frac{1}{Q(\mathbf{x}_\nu)}}{\sum_{\eta=1}^{\tilde{k}-i+1} \frac{1}{Q(\mathbf{x}_\eta)}}, \quad \mathbf{x}_\nu, \mathbf{x}_\eta \in \bar{\mathcal{M}}_{j,i-1}, \quad Q(\mathbf{x}) > 0 \quad (4.22)$$

Ende Schleife 2

$$\mathbf{x}_{Zufall} = \{\mathbf{x}_{\tilde{j}} \in \bar{\mathcal{M}}_{j,i-1} \mid \tilde{j} = \max(\hat{j} \mid CDF(\hat{j}) \leq G_i [0\dots 1])\}$$

$$\bar{\mathcal{M}}_{j,i} = \bar{\mathcal{M}}_{j,i-1} \setminus \{\mathbf{x}_{Zufall}\}$$

$$\mathcal{P}_{j,i} = \mathcal{P}_{j,i-1} \cup \{\mathbf{x}_{Zufall}\}$$

Ende Schleife 1

4. $\mathcal{M}_{j+1} = \mathcal{P}_{j,\mu}$.

Im Gegensatz zum Einsatz der Roulette-Wheel-Selektion im Rahmen der Genetischen Algorithmen [Spr95, Spr99] kann hier jedes Individuum nur *einmal* ausgewählt und zur Menge der Eltern-Individuen der nächsten Generation hinzugefügt werden. Gleichung (4.22) muss deshalb jedes mal neu ausgewertet werden, wenn ein Individuum ausgewählt wurde. Durch Gleichung (4.22) ergibt sich die sogenannte *Kumulative Dichtefunktion*. Hier wurde in der Gleichung berücksichtigt, dass Minimierungsprobleme betrachtet werden (durch Kehrwertbildung der Güterwerte in den Summen).

Im Gegensatz zur Selektion mit Ranking oder zur Roulette-Wheel-Selektion handelt es sich bei der *Selektion durch Turniere* um eine dezentrale Selektionsstrategie. Es existiert keine übergeordnete Autorität, durch die der Programmschritt Selektion überwacht und gesteuert wird. Vielmehr finden bei der Selektion durch Turniere mehrere Teilselektionen statt. Die Selektion durch Turniere ist ebenfalls aus dem Bereich der Genetischen Algorithmen bekannt [BT95, MG95]. Zur Lösung von Optimierungsproblemen, die durch Rauschen beeinflusst werden (s. Abschnitt 3.3), wird eine Selektion durch Turniere häufig eingesetzt [MG95].

Abbildung 6.1 verdeutlicht schematisch einen möglichen Ablauf einer Selektion durch Turniere. Der Programmschritt Selektion ist hier dadurch gekennzeichnet, dass aus der Ausgangsmenge \mathcal{K}_j durch \bar{j}_{max} Aussonderungsschritte $E_{j,\bar{j}}$ mit $\bar{j} = 1, \dots, \bar{j}_{max}$ solche Individuen eliminiert werden, die im Sinne des Gütemaßes

ungünstige Güterwerte aufweisen. Jeder Aussonderungsschritt $E_{j,\bar{j}}$ besteht dabei aus der Teilmengenbildung und der Elimination.

Bei der Teilmengenbildung wird für die im Rahmen dieser Arbeit herangezogene Evolutionsstrategie eine echte Teilmenge, bestehend aus mindestens zwei Individuen der Restmenge der in den vorangegangenen Aussonderungsschritten noch nicht eliminierten Individuen der Ausgangsmenge \mathcal{K}_j , gebildet. Die Restmenge ist für $\bar{j} = 0$ durch \mathcal{K}_j gegeben. Bei der Teilmengenbildung ist eine nahe liegende Vorgehensweise, aus allen Individuen der Restmenge eine bestimmte Anzahl zufällig zu ziehen, wobei jedes Individuum mit gleicher Wahrscheinlichkeit gezogen wird.

Neue Ansätze, die im Rahmen dieser Arbeit entwickelt wurden, werden in den Abschnitten 6.1 und 6.2 detailliert beschrieben. Es wird eine erste Vorgehensweise vorgestellt, bei der die Teilmengenbildung durch den mit Hilfe einer Vektornorm $\|\cdot\|$ durch $\|\mathbf{x}_h - \mathbf{x}_m\|$ definierten \mathbf{x} -Abstand mindestens eines Paares zweier Individuen $(\mathbf{x}_h, \mathbf{x}_m)$ der Restmenge beeinflusst wird. Somit wird die Turnierbildung unter Berücksichtigung des räumlichen Abstands der Individuen vorgenommen. In einer zweiten Vorgehensweise beeinflussen sowohl der \mathbf{x} -Abstand von Paaren zweier Individuen als auch der Güterwert eines Individuums die Teilmengenbildung.

Bei der anschließenden Elimination wird – wenn möglich – mindestens ein Individuum aus jeder Teilmenge $\mathcal{M}_{j,\bar{j}}$ eliminiert, das innerhalb der Teilmenge vergleichsweise schlecht im Sinne des Gütermaßes bewertet wurde. Die zuletzt erhaltene Restmenge $\bar{\mathcal{M}}_{j,\bar{j}_{max}}$ wird als Menge der Eltern-Individuen für die Generation $j + 1$ verwendet. Die Strategieparameter der Individuen aus $\bar{\mathcal{M}}_{j,\bar{j}_{max}}$ werden unverändert für die Programmschritte der nächsten Generation übernommen.

In der im Rahmen dieser Arbeit zu Vergleichszwecken verwendeten Standard-Evolutionsstrategie ES_{std} (s. Kapitel 6 und Kapitel 7) wurde eine Selektion mit Ranking ohne Einnischung eingesetzt.

Bei multikriteriellen Optimierungsproblemen besteht die wesentliche Schwierigkeit darin, eine in der Regel *kontinuierliche* Menge von Vektoren \mathbf{x} , die Pareto-Menge \mathcal{X} , durch eine *endliche* Anzahl von Individuen möglichst gleichmäßig abzudecken. Bei vielen Varianten der Evolutionsstrategien liegen die Defizite darin, dass die Individuen auf einen oder mehrere ausgezeichnete Punkte der Pareto-Menge zustreben, aber keine ausreichend gleichmäßige Abdeckung der gesamten Pareto-Menge erreicht wird.

Der Versuch, ein multikriterielles Optimierungsproblem unter Verwendung ei-

ner skalaren Ersatzfunktion als Gütemaß zu lösen, weist die in Abschnitt 3.2 beschriebenen Nachteile auf. Auch bei Variation der Gewichtungsfaktoren nach Gleichung (3.9) werden u. U. nicht alle Punkte der Pareto-Menge durch den Optimierungsprozess approximierbar.

Die in [Kah91, Kah95] vorgestellte Methode der Einnischung wird ebenfalls zur Lösung multikriterieller Optimierungsprobleme empfohlen. Dabei wird der Einnischungs-Kennwert jedes Individuums bezüglich jedes vorliegenden Gütekriteriums nach Gleichung (4.19) bestimmt. Für jedes Gütekriterium wird dann die festgelegte Anzahl an Nischen-Individuen aus der Menge der mutierten oder nicht mutierten Individuen nach dem Programmschritt Mutation und aller bei der Rekombination erzeugten Individuen ausgewählt. Hierbei werden wieder kleine Einnischungs-Kennwerte bevorzugt. Die Nischen-Individuen werden – wie auch bei Vorliegen eines skalaren Optimierungsproblems – zusammen mit einer festgelegten Anzahl weiterer Individuen, die entsprechend ihrer Bewertung mittels des Gütemaßes ausgewählt werden, dem Programmschritt Mutation der nächsten Generation zugeführt (s. Fußnote 2 auf Seite 38).

In [SV01] werden unter dem Oberbegriff *Pareto-basierte Methoden* Varianten der Evolutionsstrategie zusammengefasst, die das Ziel haben, die gesamte Pareto-Menge zu ermitteln:

Bei der *Optimierung nach wechselnden Zielen* werden in jedem Programmschritt Selektion anteilig alle einzelnen Gütekriterien berücksichtigt. Bei k Gütekriterien besteht eine neue Elternpopulation dann zu $1/k$ aus Individuen, die nach jeweils einem der Gütekriterien ausgewählt wurden. Durch diese Vorgehensweise werden in der Regel nur einzelne Punkte der Pareto-Menge, und zwar die Optima der einzelnen Gütekriterien, näherungsweise erreicht, die gesamte Pareto-Menge wird nicht approximiert.

Als weitere Pareto-basierte Methoden werden die *Pareto-Selektion*, die *Wettkampf-Pareto-Selektion* und die *Rangbasierte Selektion* erläutert.

Bei der Pareto-Selektion ist die neue Elternpopulation \mathcal{M}_{j+1} die Menge der paarweise nicht vergleichbaren Individuen der Menge \mathcal{K}_j nach Gleichung (4.20). Nachteilig ist hier, dass zur Bestimmung der paarweise nicht vergleichbaren Individuen in jedem Programmschritt Selektion ein Vergleich innerhalb aller möglichen Paarbildungen von Individuen erforderlich ist. Die Anzahl der Eltern-Individuen ist in jeder Elternpopulation unbestimmt. Ein starkes Anwachsen der Populationsgröße von \mathcal{M}_j ist bei Einsatz einer Plus-Strategie um so wahrscheinlicher, je näher man dem Ziel der Optimierung – Auffinden der Pareto-Menge – kommt. Somit steigt bei Einsatz einer Plus-Strategie auch der Aufwand für den paarweisen Vergleich aller Individuen aus \mathcal{K}_j stark an.

Bei der Wettkampf-Pareto-Selektion wird in jedem Programmschritt Selektion zunächst die Menge der paarweise nicht vergleichbaren Individuen der Menge \mathcal{K}_j bestimmt und in die neue Elternpopulation \mathcal{M}_{j+1} übernommen. Aus der Menge der restlichen Individuen aus \mathcal{K}_j wird wieder die Menge der paarweise nicht vergleichbaren Individuen bestimmt und zur neuen Elternpopulation hinzugefügt. So wird verfahren, bis die eingestellte Anzahl an Eltern-Individuen erreicht ist. Es besteht eine starke Ähnlichkeit zur Pareto-Selektion. Auch hier kann es bei Einsatz einer Plus-Strategie zu einem starken Anwachsen der Populationsgröße der Elternpopulation \mathcal{M}_j kommen.

Bei der Rangbasierten Selektion werden ebenfalls mehrere ‘Wahlgänge’ durchgeführt, bei denen aus der Menge der restlichen Individuen aus \mathcal{K}_j jeweils die paarweise nicht vergleichbaren Individuen ermittelt werden. Diese werden aber nicht unmittelbar in die neue Elternpopulation übernommen, sondern in eine Zwischengruppe eingefügt. Die Individuen erhalten je nach Wahlgang eine Selektionswahrscheinlichkeit, mit der anschließend die Individuen der neuen Elternpopulation \mathcal{M}_{j+1} aus allen Individuen der Zwischengruppe ausgewählt werden.

In [SV01] wird eine Kombination aus Pareto-Selektion und Rangbasierter Selektion eingesetzt.

In [Hor97] wird ein Überblick über aus der Literatur bekannte Vorgehensweisen bei multikriteriellen Optimierungsproblemen gegeben.

Generell muss bei der simultanen Optimierung mit mehreren Gütekriterien ein Kompromiss zwischen den beiden prinzipiell zur Verfügung stehenden Vorgehensweisen geschlossen werden. Beide Varianten weisen Vor- und Nachteile auf, die gegeneinander abzuwägen sind.

Eine mögliche Strategie kann darin bestehen, keine bereits gefundenen guten Vektoren \mathbf{x} wieder aufzugeben. Diese Strategie harmoniert gut mit der Plus-Strategie, die im Fall eines skalaren Gütemaßes gerade dazu führt, dass sich nie eine Verschlechterung des besten Individuums von einer Elternpopulation zur nächsten einstellen kann (außer durch Schwankungen in der Gütebewertung, beispielsweise durch Rauschen). Gute Individuen \mathbf{x} werden beibehalten und können nur durch besser bewertete Individuen in der Folge der Elternpopulationen abgelöst werden. Bei der multikriteriellen Optimierung legt diese Strategie nahe, nur solche Individuen im Programmschritt Selektion zu eliminieren, die von mindestens einem anderen Individuum dominiert werden. Dies kann bei Verwendung der Plus-Strategie dazu führen, dass die Größe der Elternpopulation stark ansteigt, was je nach Ausgestaltung der Strategie zu einem

stark ansteigenden Rechenaufwand im Programmschritt Selektion führen kann. Wird die Strategie verfolgt, bereits gefundene gute Vektoren \mathbf{x} aufzugeben, um dadurch leichter bessere Vektoren zu finden, so macht der Einsatz der Komma-Strategie Sinn. Die Komma-Strategie erleichtert die Anpassung der Strategieparameter der Individuen und kann durch die besser angepassten Strategieparameter möglicherweise zu einem Vordringen in Bereiche des Suchraums führen, das ohne diese gute Anpassung der Strategieparameter nicht erfolgt wäre. Andererseits werden bereits gefundene gute Einstellungen der Parametervektoren \mathbf{x} eventuell nicht an die nachfolgende Generation weitergegeben. Bei multikriteriellen Optimierungsproblemen legt diese Strategie den Ansatz nahe, unter Umständen auch Individuen zu eliminieren, die von keinem anderen Individuum dominiert werden, also Bestandteil der Pareto-Menge sein könnten. Bei Festlegung einer konstanten Populationsgröße für die Elternpopulation kann diese Elimination von nicht dominierten Individuen erforderlich werden, wenn sich mehr paarweise nicht vergleichbare Individuen in der Menge \mathcal{K}_j (s. Gleichung (4.20)) befinden, als für die Populationsgröße der Elternpopulation festgelegt sind. Bei dieser Strategie kann es dann allerdings nicht zu einem stark ansteigenden Rechenaufwand im Programmschritt Selektion kommen, der Rechenaufwand ist für jede Generation begrenzt.

4.1.5 Abbruchbedingung

Der Optimierungsprozess wird beendet, wenn erstmalig mindestens eine der folgenden Abbruchbedingungen erfüllt ist:

- Eine vom Anwender vorgegebene maximale Generationenzahl \hat{k} wurde erreicht, d. h. eine maximale Anzahl an Durchläufen des Zyklus Rekombination – Mutation – Selektion.
- Eine vom Anwender vorgegebene maximale Anzahl an Bewertungen mittels des Gütemaßes wurde durchgeführt. Wie bereits oben erläutert, kosten diese Gütebewertungen in vielen praktischen Anwendungsfällen die meiste Zeit. In vielen Fällen kann von der Generationenzahl ausgehend recht einfach auf die Anzahl an Gütebewertungen umgerechnet werden. Dies ist jedoch nicht immer möglich. Manchmal fehlen hierfür notwendige Angaben oder es wurde nicht mit einer konstanten Anzahl an Gütebewertungen pro Generation gearbeitet. Deshalb kann nur anhand der Angabe dieser Anzahl ein fairer Vergleich zwischen unterschiedlichen Optimierungsverfahren oder verschiedenen Strategieelementen vorgenommen werden.

- Wie bereits genannt, werden im Rahmen dieser Arbeit Minimierungsprobleme betrachtet. Bei der Lösung skalarer Optimierungsprobleme kann somit ein Schwellwert ρ für das Gütemaß angegeben werden, bei dessen Auftreten innerhalb des Optimierungsprozesses dieser abgebrochen werden soll und das Optimierungsproblem als gelöst angesehen wird. Wird das Gütemaß so angesetzt, dass der optimale Gütewert bei Null liegt, so kann beispielsweise ein Schwellwert von $\rho = 10^{-5}$ sinnvoll sein. Liegt hingegen ein multikriterielles Optimierungsproblem vor, so ist das Auffinden der gesamten Pareto-Menge – bzw. deren möglichst genaue Approximation – Ziel des Optimierungsprozesses. Da diese Pareto-Menge wohl kaum vorab bekannt ist, macht ein Abbruchkriterium, das auf einem Schwellwert für das Gütemaß basiert, bei multikriteriellen Optimierungsproblemen keinen Sinn.

4.2 Hooke-Jeeves-Verfahren

Die Methode von Hooke und Jeeves [HJ61] wurde zur Lösung eines skalaren Optimierungsproblems im \mathbb{R}^n ohne Nebenbedingungen mit einer unimodalen Funktion als Gütemaß konzipiert. Der Ablauf des Verfahrens ist außerdem in [Sch77, GZZ88, Bro96] beschrieben. Bei der Methode von Hooke und Jeeves – im Folgenden auch Hooke-Jeeves-Verfahren genannt – handelt es sich um ein Quasi-Gradientenverfahren. Es wird in einem zu untersuchenden Punkt \mathbf{x} im Parameterraum nicht der wirkliche Gradient bestimmt, sondern das Verfahren arbeitet mit Suchschritten entlang der Koordinatenachsen.

In [Bro96] wurde das ursprüngliche Verfahren bereits so erweitert, dass es auch für einen begrenzten Suchraum (s. Gleichung (3.3)) sinnvoll einsetzbar ist. Außerdem wurde der Schwellwert $\epsilon > 0$ eingeführt (Algorithmus s. u.). Im ursprünglichen Algorithmus wird $\epsilon = 0$ verwendet.

Das Hooke-Jeeves-Verfahren arbeitet mit zwei ineinander verschachtelten Teilalgorithmen. Die sogenannte *Exploration* (‘exploratory move’ [HJ61]) dient der Erkundung der näheren Umgebung eines Punktes. In der sogenannten *Extrapolation* (‘pattern move’ [HJ61]) werden größere Suchschritte durchgeführt.

In dem Teilalgorithmus Exploration wird, ausgehend vom Punkt $\mathbf{x} = (x_1, \dots, x_n)$, mit der skalaren Schrittweite s und mit dem Gütewert $Q(\mathbf{x})$ folgendermaßen vorgegangen:

1. Erkundung in x_1 -Richtung von \mathbf{x} aus mit s und dem Vergleichsgütewert $Q(\mathbf{x})$:

wenn

$$Q(x_1 + s, x_2, \dots, x_n) < Q(\mathbf{x})$$

dann

$$\mathbf{x}_{neu}^1 = (x_1 + s, x_2, \dots, x_n)$$

sonst

wenn

$$Q(x_1 - s, x_2, \dots, x_n) < Q(\mathbf{x})$$

dann

$$\mathbf{x}_{neu}^1 = (x_1 - s, x_2, \dots, x_n)$$

sonst

$$\mathbf{x}_{neu}^1 = \mathbf{x}$$

2. Erkundung in x_2 -Richtung von \mathbf{x}_{neu}^1 aus mit s und dem Vergleichsgütwert $Q(\mathbf{x}_{neu}^1)$:

von \mathbf{x}_{neu}^1 aus wird durch Schritte mit der Schrittweite s in x_2 -Richtung (entsprechend wie in 1.) erkundet, das Ergebnis ist \mathbf{x}_{neu}^2

3. usw., das Ergebnis der gesamten Exploration ist \mathbf{x}_{neu}^n

Für den gesamten Algorithmus wird ein Startpunkt \mathbf{x} im Suchraum S festgelegt. Die Startschrittweite s wird bestimmt, ebenso wie ein Variationsfaktor α mit $0 < \alpha < 1$ zur Verkleinerung der Schrittweite.

Das Hooke-Jeeves-Verfahren läuft wie folgt ab:

- I. (1) Exploration von \mathbf{x} aus mit s und dem Vergleichsgütwert $Q(\mathbf{x})$ liefert \mathbf{x}_{neu}^n
- (2) wenn
- $$Q(\mathbf{x}) - Q(\mathbf{x}_{neu}^n) > \epsilon \quad (\text{es konnte eine Verbesserung erreicht werden})$$
- dann
- weiter mit II.
- sonst (es konnte keine Verbesserung erreicht werden)
1. $s := \alpha \cdot s$ (Verkleinerung der Schrittweite, $\alpha = 0.5$)
 2. weiter mit I.
- II. (1) Exploration von $\mathbf{x}_{neu}^n + (\mathbf{x}_{neu}^n - \mathbf{x})$ aus mit s und dem Vergleichsgütwert $Q(\mathbf{x}_{neu}^n)$ liefert $[\mathbf{x}_{neu}^n + (\mathbf{x}_{neu}^n - \mathbf{x})]_{neu}^n$

(2) wenn

$$Q(\mathbf{x}_{neu}^n + (\mathbf{x}_{neu}^n - \mathbf{x})) - Q([\mathbf{x}_{neu}^n + (\mathbf{x}_{neu}^n - \mathbf{x})]_{neu}^n) > \epsilon$$

(Verbesserung)

dann

1. Extrapolationsschritt 1: $\mathbf{x} := \mathbf{x}_{neu}^n$
2. Extrapolationsschritt 2: $\mathbf{x}_{neu}^n := [\mathbf{x}_{neu}^n + (\mathbf{x}_{neu}^n - \mathbf{x})]_{neu}^n$
3. weiter mit II.

sonst (keine Verbesserung)

1. $\mathbf{x} := \mathbf{x}_{neu}^n$
2. weiter mit I.

Das Verfahren wird abgebrochen, wenn die Schrittweite s eine untere Grenze s_{min} unterschritten hat, wenn eine maximal zulässige Anzahl an Bewertungen mittels des Gütemaßes stattgefunden hat oder wenn ein minimaler Gütewert ρ unterschritten wurde. Der so erhaltene Punkt \mathbf{x} ist das Optimierungsergebnis.

Laut [GZZ88] besteht der Vorteil des Hooke-Jeeves-Verfahrens darin, dass es bei unimodalen Funktionen als Gütemaß schnell in die Nähe des Minimums führt.

Soll das Hooke-Jeeves-Verfahren bei einem Gütemaß in Form einer nicht-unimodalen Funktion eingesetzt werden, so hängt es von der Wahl des Startpunktes und von der Festlegung der Startschrittweite ab, ob durch das Verfahren das bzw. ein globales Optimum oder nur ein lokales Optimum gefunden wird.

Für die letzte Phase der feineren Annäherung an ein Optimum sind nach [GZZ88] hingegen andere Optimierungsverfahren besser geeignet.

Kapitel 5

Auswahl analytischer Testfunktionen

In diesem Kapitel werden die in dieser Arbeit als Gütemaß verwendeten analytischen Testfunktionen vorgestellt. Es werden unterschiedliche Testfunktionen eingesetzt, um zu vermeiden, dass die entwickelten Strategieelemente speziell an die Optimierung einer einzigen Testfunktion angepasst werden, dann aber durch diese spezielle Anpassung für andere Testfunktionen oder reale Optimierungsprobleme nicht zu guten Ergebnissen führen.

Six-Hump-Camel-Back-Funktion [DS78]

$$Q(\mathbf{x}) = 4 \cdot x_1^2 - 2.1 \cdot x_1^4 + \frac{1}{3} \cdot x_1^6 + x_1 \cdot x_2 - 4 \cdot x_2^2 + 4 \cdot x_2^4 + 1.0316. \quad (5.1)$$

Bei Verwendung dieser Testfunktion werden die einzustellenden Parameter in folgenden Bereichen variiert:

$$-2.5 \leq x_1 \leq 2.5$$

$$-1.5 \leq x_2 \leq 1.5.$$

Als Güte-Schwellwert für den Abbruch des Optimierungsprozesses wird bei dieser Funktion ein Wert von $\rho = 10^{-5}$ verwendet. Die Funktion weist zwei globale Optima (Minima) auf. Ziel des Optimierungsprozesses ist es, eines dieser beiden globalen Optima mindestens mit der angegebenen Genauigkeit ρ zu ermitteln. Ferner besitzt die Funktion 4 lokale Optima. Aufgrund der Genauigkeit von 10^{-4} des additiven Terms 1.0316 können bei dieser Funktion auch negative Funktionswerte erreicht werden, die ebenfalls zum Abbruch des Optimierungsprozesses führen. Die Funktion ist in Abbildung 5.1 dargestellt.

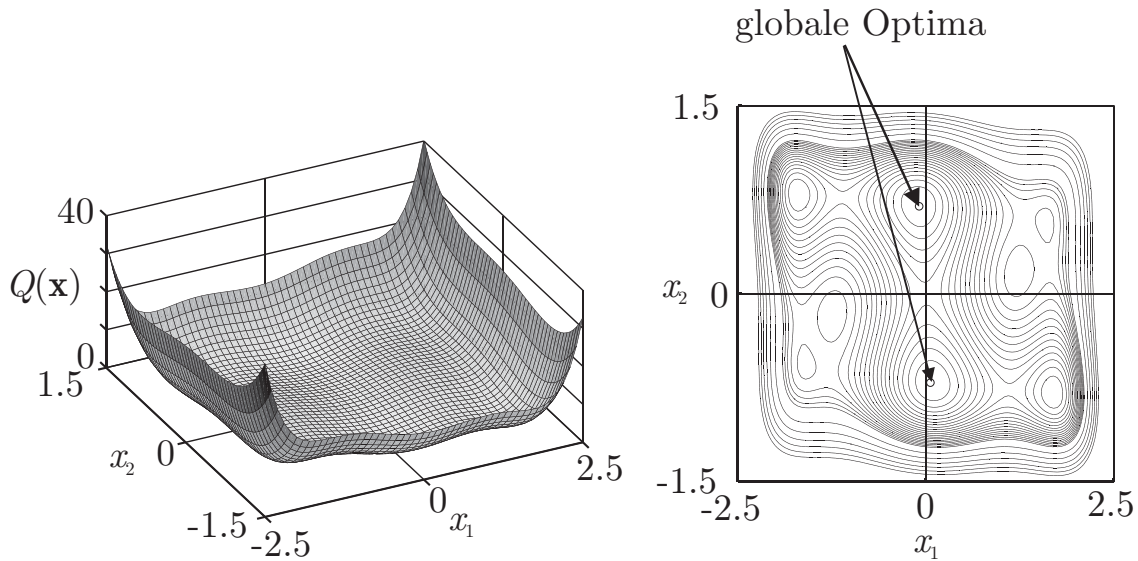


Abbildung 5.1: Testfunktion Six-Hump-Camel-Back-Funktion, links: $Q(x_1, x_2)$, rechts: Höhenlinien, d. h. Linien gleicher Funktionswerte $Q(x_1, x_2)$.

Rastrigin-Funktion [SHF94]

$$Q(\mathbf{x}) = n \cdot A + \sum_{i=1}^n (x_i^2 - A \cdot \cos(2 \cdot \pi \cdot x_i)) \quad \text{mit } A = 10, \quad (5.2)$$

für $n = 2$ ergibt sich:

$$Q(\mathbf{x}) = 20 + (x_1^2 - 10 \cdot \cos(2 \cdot \pi \cdot x_1)) + (x_2^2 - 10 \cdot \cos(2 \cdot \pi \cdot x_2)). \quad (5.3)$$

Die einzustellenden Parameter werden in folgendem Bereich variiert:

$$-5.12 \leq x_i \leq 5.12 \quad \text{für } i = 1, \dots, n.$$

Als Schwellwert für den Abbruch des Optimierungsprozesses wird bei dieser Funktion ein Wert von $\rho = 10^{-5}$ verwendet. Die Funktion weist ein globales Optimum und zahlreiche lokale Optima auf. Die Funktion ist in Abbildung 5.2 für $n = 2$ dargestellt.

Schwefel-Funktion [SHF94]

$$Q(\mathbf{x}) = \sum_{i=1}^n (-x_i \cdot \sin(\sqrt{|x_i|})), \quad (5.4)$$

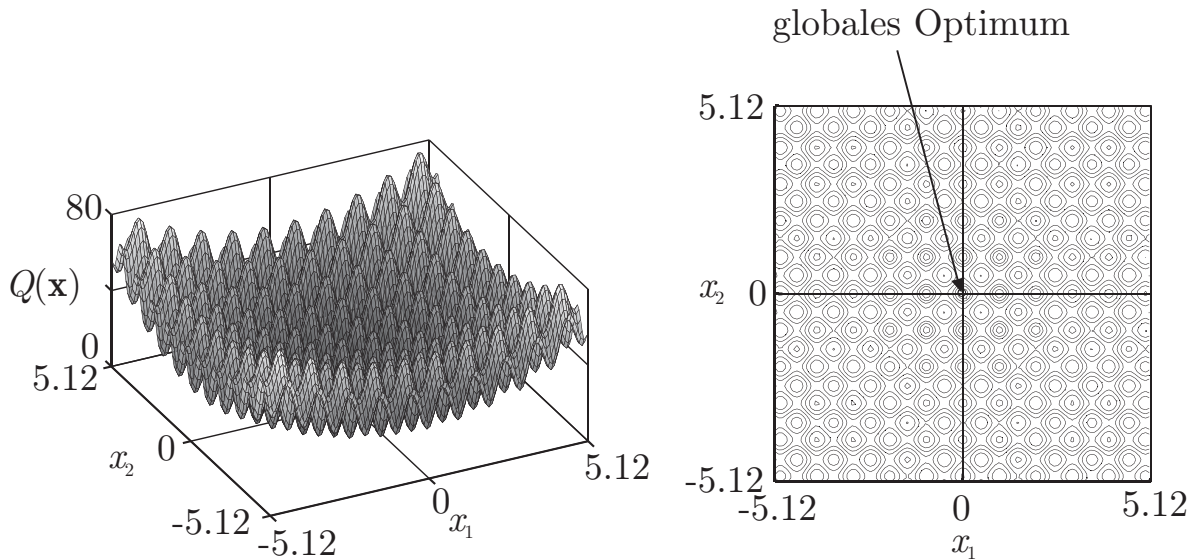


Abbildung 5.2: Testfunktion Rastrigin-Funktion bei Verwendung zweier Eingangsgrößen x_1 und x_2 , links: $Q(x_1, x_2)$, rechts: Höhenlinien, d. h. Linien gleicher Funktionswerte $Q(x_1, x_2)$.

für $n = 2$ ergibt sich:

$$Q(\mathbf{x}) = -x_1 \cdot \sin(\sqrt{|x_1|}) - x_2 \cdot \sin(\sqrt{|x_2|}). \quad (5.5)$$

Die ursprüngliche Funktion nach Gleichung (5.4) bzw. Gleichung (5.5) wurde so modifiziert, dass im globalen Optimum ein Funktionswert von Null auftritt (mit einer Genauigkeit von 10^{-6} , damit die Abbruchbedingung durch den Optimierungsprozess erreicht werden kann):

$$Q(\mathbf{x}) = n \cdot 418.982887 - \sum_{i=1}^n (x_i \cdot \sin(\sqrt{|x_i|})). \quad (5.6)$$

Die einzustellenden Parameter werden in folgendem Bereich variiert:

$$-500 \leq x_i \leq 500 \quad \text{für } i = 1, \dots, n.$$

Als Schwellwert für den Abbruch des Optimierungsprozesses wird bei dieser Funktion ein Wert von $\rho = 10^{-5}$ verwendet. Die Funktion besitzt ebenfalls ein globales Optimum und mehrere lokale Optima. Da das globale Optimum in einer Ecke des Suchraums liegt, ist das Optimierungsproblem je nach Lage der Startpunkte für den Optimierungsprozess schwierig zu lösen. Die Funktion ist in Abbildung 5.3 für $n = 2$ dargestellt.

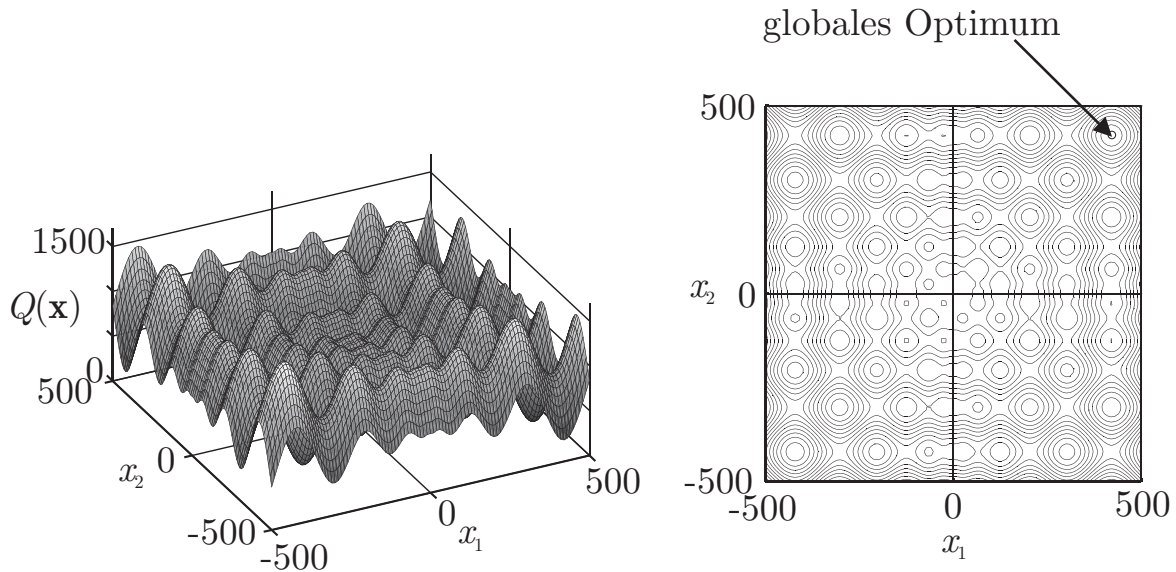


Abbildung 5.3: Testfunktion nach Schwefel bei Verwendung zweier Eingangsgrößen x_1 und x_2 , links: $Q(x_1, x_2)$, rechts: Höhenlinien, d. h. Linien gleicher Funktionswerte $Q(x_1, x_2)$.

Rosenbrock-Funktion [Poh00]

$$Q(\mathbf{x}) = \sum_{i=1}^{n-1} (100 \cdot (x_i^2 - x_{i+1})^2 + (1 - x_i)^2), \quad (5.7)$$

für $n = 2$ ergibt sich [Ros60]:

$$Q(\mathbf{x}) = 100 \cdot (x_1^2 - x_2)^2 + (1 - x_1)^2. \quad (5.8)$$

Die einzustellenden Parameter werden in folgendem Bereich variiert:

$$-2.048 \leq x_i \leq 2.048 \quad \text{für } i = 1, \dots, n,$$

für den Sonderfall $n = 2$ werden folgende Grenzen der einzustellenden Parameter verwendet:

$$\begin{aligned} -2 &\leq x_1 \leq 2 \\ -1 &\leq x_2 \leq 3. \end{aligned}$$

Als Schwellwert für den Abbruch des Optimierungsprozesses wird auch bei dieser Funktion ein Wert von $\rho = 10^{-5}$ verwendet. Die Funktion weist ein globales Optimum und keine weiteren Optima auf. Dennoch ist das durch diese Testfunktion als Gütemaß gegebene Optimierungsproblem für viele Optimierungsverfahren schwierig zu lösen, da das Optimum in einem Tal liegt, das diagonal

zu den Koordinatenachsen verläuft. Die Funktion ist in Abbildung 5.4 für $n = 2$ dargestellt.

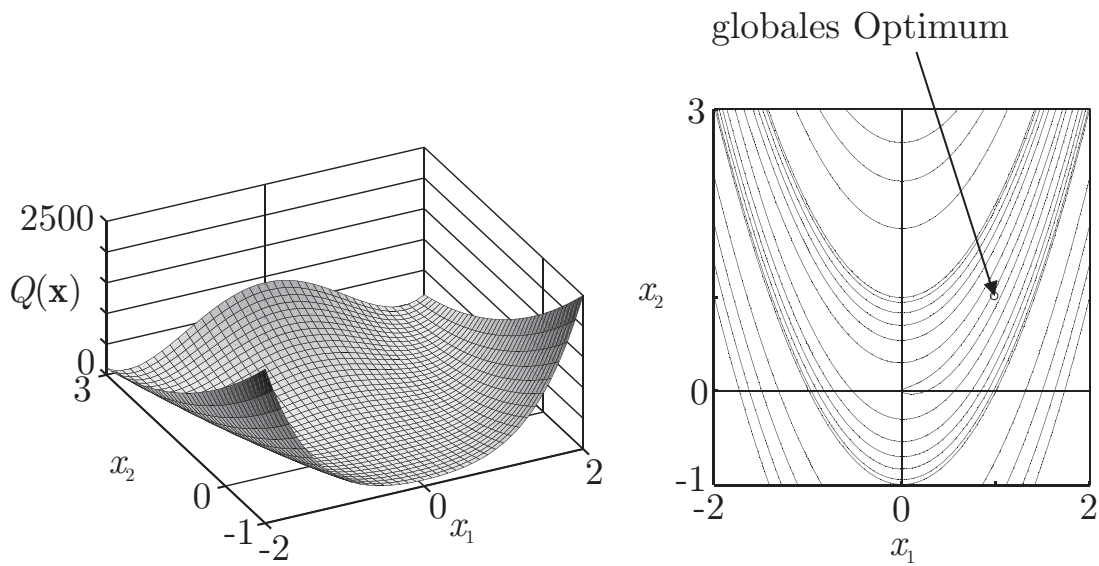


Abbildung 5.4: Testfunktion Rosenbrock-Funktion bei Verwendung zweier Eingangsgrößen x_1 und x_2 , links: $Q(x_1, x_2)$, rechts: Höhenlinien, d. h. Linien gleicher Funktionswerte $Q(x_1, x_2)$.

Kapitel 6

Neue Strategieelemente: Methodische Konzeption und Nachweis der Wirkungsweise

Herkömmliche Evolutionsstrategien zeichnen sich vielfach durch mangelnde Breitensuche aus. Sie konvergieren nach nur wenigen Generationen gegen ein Optimum. Bei einfach zu lösenden Optimierungsproblemen, die sich durch eine freundliche Gütelandschaft auszeichnen, besteht eine vergleichsweise hohe Chance, dass es sich bei dem gefundenen Optimum um ein globales Optimum handelt. In vielen praktischen Anwendungsfällen handelt es sich aber um schwieriger zu lösende Optimierungsprobleme mit dementsprechend zerklüfteter Gütelandschaft oder mit einer Gütelandschaft, die viele relative Optima aufweist.

Aus diesem Grund wurden bereits in [Nic99a] erste Strategieelemente vorgestellt, die zu einer verstärkten Breitensuche gegenüber einer herkömmlichen Evolutionsstrategie führen. Die neu eingeführten Strategieelemente betreffen zum einen die Auswahl der verwendeten Eltern-Individuen im Programmschritt Rekombination, zum anderen wurden die strategischen Neuerungen im Programmschritt Selektion eingebracht, der auf einer Selektion durch Turniere basiert.

Durch die in diesem Kapitel vorgestellten neuen Strategieelemente soll ein ausgewogener Kompromiss zwischen Breiten- und Tiefensuche erreicht werden [KNL⁺03, Nic04].

6.1 Evolutionsstrategie mit wechselnder Breiten- und Tiefensuche (ES_{WBT})

Die verstärkte Breitensuche soll durch die hier umgesetzte Strategie im Programmschritt Selektion berücksichtigt, eine stark ausgeprägte Tiefensuche

durch die im Programmschritt Rekombination eingebrachten Strategieelemente realisiert werden. Insgesamt soll sich hier eine gegenüber der Strategie ES_{std} stärker ausgeprägte Breitensuche während des Optimierungsprozesses einstellen.

Die hier vorgestellte Gesamtstrategie wird mit *Evolutionstrategie mit wechselnder Breiten- und Tiefensuche* (ES_{WBT}) bezeichnet, da sich die einzelnen Programmschritte innerhalb der Evolutionsstrategie und somit auch die Ausrichtung auf eine verstärkte Breiten- oder eine verstärkte Tiefensuche abwechseln.

Im Gegensatz zu herkömmlichen Rekombinationsstrategien, bei denen die für einen Rekombinationsschritt herangezogenen Eltern-Individuen häufig mit gleichverteilter Wahrscheinlichkeit zufallsabhängig aus allen Eltern-Individuen ausgewählt werden, wird bei der hier dargestellten Konzeption des Programmschritts Rekombination die Auswahl der zu einem Rekombinationsschritt herangezogenen Eltern-Individuen mit einer abstandsabhängigen Wahrscheinlichkeitsfunktion durchgeführt.

Es steht folgende Intention hinter dieser Wahrscheinlichkeitsfunktion: In der Natur können Individuen nur dann gemeinsame Nachkommen haben, wenn eine große genetische Ähnlichkeit der Individuen besteht.

Als räumlicher oder ‘genetischer’ Abstand zweier Individuen \mathbf{x}_h und \mathbf{x}_m wird bezüglich der Strategie ES_{WBT} das Abstandsmaß

$$d(\mathbf{x}_h, \mathbf{x}_m) = \frac{1}{n} \cdot \sum_{i=1}^n \frac{|x_{i,h} - x_{i,m}|}{x_{i,max} - x_{i,min}}, \quad (6.1)$$

also die auf die Dimension n des Suchraums normierte Betragssummennorm in einem – auf die Breite des Suchraums in jeder Dimension i – normierten Suchraum verwendet.

Der Programmschritt Selektion ist bei ES_{WBT} so ausgerichtet, dass hierdurch die Komponente der Breitensuche berücksichtigt wird. Auch hier wird eine abstandsabhängige Wahrscheinlichkeitsfunktion für die Selektion durch Turniere verwendet. Die angestrebte Wirkungsweise ist hierbei, dass Turniere hauptsächlich in der ‘näheren Umgebung’ eines Individuums ausgetragen werden. In der Natur rivalisieren Lebewesen, die den gleichen Lebensraum bzw. die gleichen Nahrungsquellen nutzen. Je größer die Ähnlichkeit von Individuen ist, desto stärker ist in der Regel der Konkurrenzkampf. Dieser natürliche Zustand wird durch den Programmschritt Selektion nachgeahmt.

6.1.1 ES_{WBT} mit variabler Laufzeit: ES_{WBT}^{T_{var}}

In einer ersten Variante von ES_{WBT} läuft ein Rekombinationsschritt folgendermaßen ab: Aus allen Individuen der Menge \mathcal{M}_j (s. Abbildung 4.1) werden mit gleichverteilter Wahrscheinlichkeit zwei Individuen \mathbf{x}_h und \mathbf{x}_m festgelegt. Anschließend wird durch einen Zufallsgenerator entschieden, ob ein Rekombinationsschritt unter Verwendung der beiden festgelegten Individuen durchgeführt wird. Es wird folgende Wahrscheinlichkeitsfunktion verwendet (s. Abbildung 6.2 (a)):

$$D_{Rek}(d) = e^{-\frac{d(\mathbf{x}_h, \mathbf{x}_m)^2}{2 \cdot \sigma_{Rek}^2}} \quad \text{mit} \quad \sigma_{Rek} = 1. \quad (6.2)$$

Der Wert $\sigma_{Rek} = 1$ wurde empirisch ermittelt.

Durch den Zufallsgenerator wird eine gleichverteilte Zufallszahl

$$z_{Rekombination} = G_{[0..1]} \quad (6.3)$$

erzeugt. Der Rekombinationsschritt wird unter Verwendung der beiden festgelegten Individuen durchgeführt, wenn $z_{Rekombination} \leq D_{Rek}(d)$ ist. Bei $z_{Rekombination} > D_{Rek}(d)$ wird der Rekombinationsschritt nicht durchgeführt und es werden zwei neue Individuen \mathbf{x}_h und \mathbf{x}_m festgelegt. Ein Rekombinationsschritt zwischen zwei festgelegten Individuen wird also mit größerer Wahrscheinlichkeit durchgeführt, wenn die Individuen zu einem kleinen Wert $d(\mathbf{x}_h, \mathbf{x}_m)$ führen. Jedes Individuum der Menge \mathcal{M}_j kann hierbei mehrfach durch den Zufallsgenerator festgelegt und auch mehrfach für einen Rekombinationsschritt verwendet werden. Bei jedem durchgeführten Rekombinationsschritt werden zwei Nachkommen-Individuen generiert (s. Abschnitt 4.1.2). So wird verfahren, bis die eingestellte Anzahl λ an Nachkommen-Individuen generiert wurde. Nachteilig an der hier beschriebenen Vorgehensweise ist, dass sich keine feste Laufzeit für den Programmschritt Rekombination ergibt. Durch die Zufallsentscheidung ‘Rekombinationsschritt durchführen \rightarrow ja/nein’ variiert die Laufzeit des Programmschritts Rekombination insbesondere in Abhängigkeit der Konstellation der Individuen der Menge \mathcal{M}_j im Suchraum.

Der Programmschritt Selektion der bereits genannten ersten Variante von ES_{WBT} wurde für den Fall eines skalaren Optimierungsproblems folgendermaßen implementiert (s. Abbildung 6.1):

In jedem Aussonderungsschritt $E_{j, \bar{j}}$ wird zunächst bei der Teilmengenbildung eine Teilmenge, bestehend aus zwei Individuen aus der Menge $\bar{\mathcal{M}}_{j, \bar{j}-1}$, gebildet

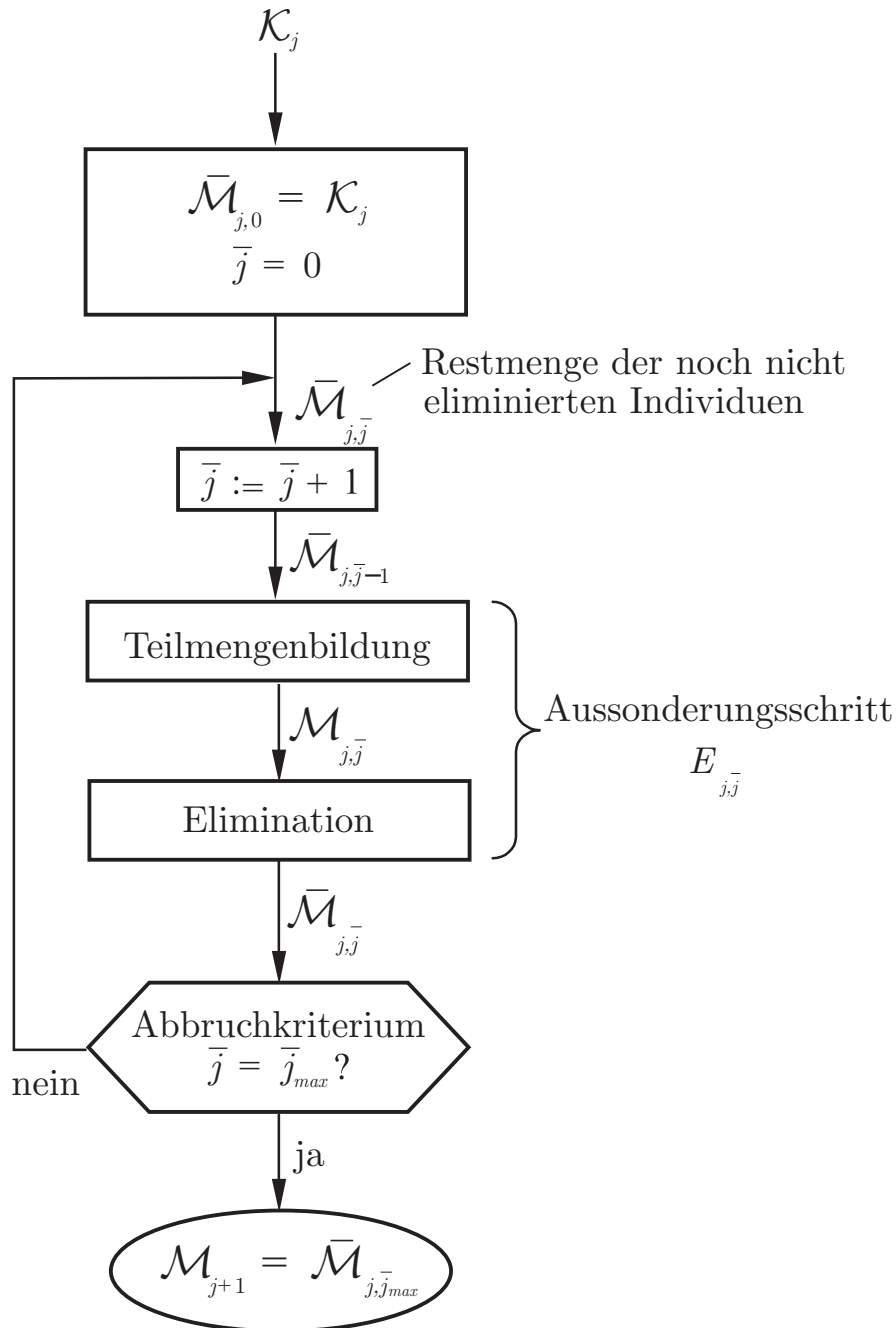


Abbildung 6.1: Schematische Darstellung der Selektion durch Turniere.

und anschließend bei der Elimination das Individuum ausgesondert, das den schlechteren Gütewert aufweist.

Die Teilmengenbildung wird folgendermaßen vorgenommen: Es werden aus allen Individuen der Menge $\bar{\mathcal{M}}_{j,\bar{j}-1}$ mit gleichverteilter Wahrscheinlichkeit zwei Individuen \mathbf{x}_h und \mathbf{x}_m festgelegt. Danach wird durch einen Zufallsgenerator entschieden, ob eine Teilmengenbildung mit den beiden festgelegten Individuen erfolgt. Hierzu wird die Wahrscheinlichkeitsfunktion aus Gleichung (6.2) mit

$\sigma_{Sel} = 0.14$ und $d(\mathbf{x}_h, \mathbf{x}_m)$ wie in Gleichung (6.1) definiert verwendet (s. Abbildung 6.2 (b)):

$$D_{Sel}(d) = e^{-\frac{d(\mathbf{x}_h, \mathbf{x}_m)^2}{2 \cdot \sigma_{Sel}^2}} \quad \text{mit} \quad \sigma_{Sel} = 0.14. \quad (6.4)$$

Der Wert $\sigma_{Sel} = 0.14$ wurde auch hier empirisch ermittelt.

Durch den Zufallsgenerator wird eine gleichverteilte Zufallszahl wie in Gleichung (6.3) generiert:

$$z_{Selektion} = G_{[0...1]}. \quad (6.5)$$

Die Teilmengenbildung wird mit den beiden festgelegten Individuen vorgenommen, wenn $z_{Selektion} \leq D_{Sel}(d)$ ist, anderenfalls erfolgt die Teilmengenbildung nicht und es werden zwei neue Individuen \mathbf{x}_h und \mathbf{x}_m bestimmt. Eine Teilmengenbildung wird also mit größerer Wahrscheinlichkeit vorgenommen, wenn die beiden festgelegten Individuen einen kleinen Wert $d(\mathbf{x}_h, \mathbf{x}_m)$ aufweisen.

Jedes bisher noch nicht eliminierte Individuum der Restmenge $\bar{\mathcal{M}}_{j,\bar{j}}$ kann hierbei mehrfach durch den Zufallsgenerator festgelegt und auch mehrfach für eine Teilmengenbildung verwendet werden. Ein eliminiertes Individuum entfällt aus der Restmenge und wird im Weiteren nicht mehr für den Optimierungsprozess berücksichtigt.

Auch im Programmschritt Selektion ergibt sich der oben genannte Nachteil, dass durch die zufallsabhängige Entscheidung ‘*Teilmengenbildung durchführen* → *ja/nein*’ keine feste Laufzeit vorliegt.

Diese erste Variante der Strategie ES_{WBT} wird aufgrund der variablen Programmtdauer der einzelnen Programmschritte mit $ES_{WBT}^{T_{var}}$ bezeichnet. [Nic99a]

6.1.2 ES_{WBT} mit konstanter Laufzeit: $ES_{WBT}^{T_{const}}$

Der dargestellte Mangel der nicht exakt festgelegten Laufzeit einzelner Programmschritte wurde in einer zweiten Variante der Strategie ES_{WBT} gemildert. Diese zweite Variante wird im Folgenden mit $ES_{WBT}^{T_{const}}$ bezeichnet. Die zweite Variante beinhaltet eine ähnliche Wirkungsweise wie die Variante $ES_{WBT}^{T_{var}}$. Auch hier soll durch den Programmschritt Rekombination eine verstärkte Tiefensuche realisiert und durch den Programmschritt Selektion eine stark ausgeprägte Breitensuche erzielt werden.

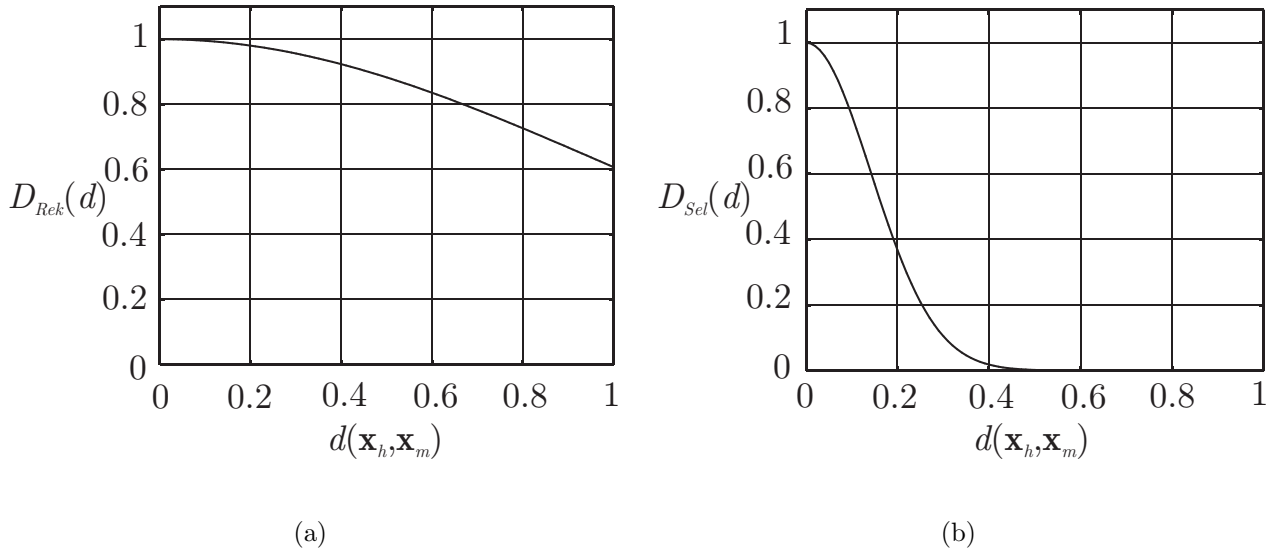


Abbildung 6.2: Wahrscheinlichkeitsfunktion mit $\sigma_{Rek} = 1$ für einen Rekombinationsschritt (a) bzw. mit $\sigma_{Sel} = 0.14$ für eine Teilmengenbildung (b) in der Variante ES_{WBT}^{Tvar} .

Hierzu wird bei einem Rekombinationsschritt folgendermaßen vorgegangen: Aus allen Individuen der Menge \mathcal{M}_j wird mit gleichverteilter Wahrscheinlichkeit ein Referenzindividuum \mathbf{x}_{ref} bestimmt. Allen restlichen Individuen \mathbf{x}_h der Menge $\{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$ wird in Abhängigkeit von ihrem räumlichen Abstand zu dem festgelegten Referenzindividuum eine sogenannte Rekombinations-Wahrscheinlichkeitsdichte $P_{Rek}(\mathbf{x}_h)$ zugewiesen, die die Rekombinationswahrscheinlichkeit wie in der Variante ES_{WBT}^{Tvar} nach Gleichung (6.2) verwendet. Auch hier wird ein Wert von $\sigma_{Rek} = 1$ verwendet:

$$D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\frac{d(\mathbf{x}_h, \mathbf{x}_{ref})^2}{2 \cdot \sigma_{Rek}^2}} \quad \text{mit } \sigma_{Rek} = 1.$$

Es wird über alle Individuen $\mathbf{x}_i \in \{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$ die Summe über $D_{Rek}(\mathbf{x}_i, \mathbf{x}_{ref})$ gebildet:

$$D_{Rek}^{Sum} = \sum_{i=1}^{\mu-1} D_{Rek}(\mathbf{x}_i, \mathbf{x}_{ref}).$$

Die Rekombinations-Wahrscheinlichkeitsdichte für ein Individuum \mathbf{x}_h ergibt sich aus dem auf D_{Rek}^{Sum} normierten Wert von $D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$:

$$P_{Rek}(\mathbf{x}_h) = \frac{D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})}{D_{Rek}^{Sum}}. \quad (6.6)$$

Anschließend wird unter Berücksichtigung der Rekombinations-Wahrscheinlichkeitsdichten $P_{Rek}(\mathbf{x}_i)$ aller Individuen der Menge $\{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$ aus allen restlichen Individuen mit Hilfe eines Zufallsgenerators ein Individuum bestimmt, das für den Rekombinationsschritt zusammen mit \mathbf{x}_{ref} verwendet wird. Diese Auswahl des zweiten Eltern-Individuums geschieht in Anlehnung an die Roulette-Wheel-Selektion (s. Abschnitt 4.1.4). Der Vorgang kann so veranschaulicht werden, dass jedem Individuum der Menge $\{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$ entsprechend seiner Rekombinations-Wahrscheinlichkeitsdichte ein bestimmtes Winkelstück auf einem virtuellen Roulette-Rad zugewiesen wird. Durch den Zufallsgenerator wird dann die Halteposition des Roulette-Rades bestimmt. Die Größe des Winkelstücks ist hierbei proportional zur Rekombinations-Wahrscheinlichkeitsdichte. Bei großer Rekombinations-Wahrscheinlichkeitsdichte ist somit die Wahrscheinlichkeit, mit der das Roulette-Rad innerhalb dieses Winkelstücks zum Stillstand gelangt und das entsprechende Individuum zu einem Rekombinationsschritt mit dem Referenzindividuum verwendet wird, ebenfalls groß. Die Zufallsentscheidung ‘Rekombinationsschritt durchführen \rightarrow ja/nein’ entfällt also, da in jedem Fall ein Rekombinationsschritt erfolgt. Somit kann eine feste Laufzeit für den Programmschritt Rekombination garantiert werden. Auch bei $ES_{WBT}^{T_{const}}$ kann jedes Individuum der Menge \mathcal{M}_j mehrfach für einen Rekombinationsschritt verwendet werden. Auch hier werden bei jedem durchgeführten Rekombinationsschritt zwei Nachkommen-Individuen erzeugt und der Programmschritt Rekombination wird durchlaufen, bis λ Nachkommen-Individuen generiert wurden.

Im Programmschritt Selektion erfolgt die Selektion durch Turniere für den Fall eines skalaren Optimierungsproblems so, dass bei jedem Aussonderungsschritt $E_{j,\bar{j}}$ aus allen Individuen der Menge $\bar{\mathcal{M}}_{j,\bar{j}-1}$ mit gleichverteilter Wahrscheinlichkeit ein Referenzindividuum \mathbf{x}_{ref} bestimmt wird. Allen restlichen Individuen \mathbf{x}_h der Menge $\{\bar{\mathcal{M}}_{j,\bar{j}-1} \setminus \mathbf{x}_{ref}\}$ wird in Abhängigkeit von ihrem räumlichen Abstand zu dem festgelegten Referenzindividuum eine sogenannte Kampf-Wahrscheinlichkeitsdichte zugewiesen, die unter Verwendung von Gleichung (6.4) folgendermaßen ermittelt wird:

$$D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\frac{d(\mathbf{x}_h, \mathbf{x}_{ref})^2}{2 \cdot \sigma_{Sel}^2}} \quad \text{mit} \quad \sigma_{Sel} = 0.2.$$

Hier wird allerdings ein Wert von $\sigma_{Sel} = 0.2$ verwendet.

Analog zu der Vorgehensweise im Programmschritt Rekombination wird auch in einem Aussonderungsschritt die Summe über die Werte $D_{Sel}(\mathbf{x}_i, \mathbf{x}_{ref})$ über

alle bisher noch nicht eliminierten Individuen $\mathbf{x}_i \in \{\bar{\mathcal{M}}_{j,\bar{j}-1} \setminus \mathbf{x}_{ref}\}$ zu Normierungszwecken gebildet

$$D_{Sel}^{Sum} = \sum_{i=1}^{card(\bar{\mathcal{M}}_{j,\bar{j}-1})-1} D_{Sel}(\mathbf{x}_i, \mathbf{x}_{ref})$$

und die Kampf-Wahrscheinlichkeitsdichte für ein Individuum \mathbf{x}_h aus dem Quotienten von $D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})$ und D_{Sel}^{Sum} gebildet:

$$P_{Sel}(\mathbf{x}_h) = \frac{D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})}{D_{Sel}^{Sum}}. \quad (6.7)$$

Danach wird unter Berücksichtigung der Kampf-Wahrscheinlichkeitsdichten $P_{Sel}(\mathbf{x}_i)$ aus allen Individuen der Menge $\{\bar{\mathcal{M}}_{j,\bar{j}-1} \setminus \mathbf{x}_{ref}\}$ mit Hilfe eines Zufalls-generators ein Individuum bestimmt, das zusammen mit \mathbf{x}_{ref} die Teilmenge $\mathcal{M}_{j,\bar{j}}$ bildet und in einem Turnier gegen \mathbf{x}_{ref} antreten muss. Das Individuum aus der Teilmenge, das den schlechteren Gütewert aufweist, wird durch die Elimination aus dem Optimierungsprozess ausgesondert. Die Auswahl des Turniergegners von \mathbf{x}_{ref} geschieht hier ebenfalls in Anlehnung an die Roulette-Wheel-Selektion. Bei großer Kampf-Wahrscheinlichkeitsdichte wird einem Individuum aus der Menge $\{\bar{\mathcal{M}}_{j,\bar{j}-1} \setminus \mathbf{x}_{ref}\}$ ein großes Winkelstück auf dem virtuellen Roulette-Rad zugewiesen, bei kleiner Kampf-Wahrscheinlichkeitsdichte ist auch das Winkelstück klein. Ist die Kampf-Wahrscheinlichkeitsdichte groß, so besteht eine große Wahrscheinlichkeit, dass das entsprechende Individuum zu einem Turnier gegen das Referenzindividuum antreten muss. Die Zufallsentscheidung ‘Teilmengebildung durchführen \rightarrow ja/nein’ entfällt, es wird in jedem Fall eine Teilmengebildung vorgenommen.

Jedes bisher noch nicht eliminierte Individuum der Restmenge $\bar{\mathcal{M}}_{j,\bar{j}}$ kann auch hier mehrfach durch den Zufallsgenerator festgelegt und somit mehrfach für eine Teilmengebildung verwendet werden. Ein eliminiertes Individuum entfällt aus der Restmenge und somit aus dem weiteren Optimierungsprozess.

Somit kann eine feste Laufzeit für den Programmschritt Selektion garantiert werden, es sei denn, es tritt der in praktischen Anwendungsfällen unwahrscheinliche Fall exakt gleicher Gütewerte mehrerer Individuen auf. Bei diesem Sonderfall kann unter Umständen bei einem Turnier kein Verlierer ausfindig gemacht und eliminiert werden. Die Teilmengebildung wird in diesem Fall so oft wiederholt, bis eine Elimination eines Individuums aus der Teilmenge $\mathcal{M}_{j,\bar{j}}$ vorgenommen werden kann. Dieser Sonderfall mehrerer exakt gleich bewerteter Individuen in einer Population kann bei jeder Variante der Evolutionsstrategie

auftreten. Wird eine Selektion durch Turniere durchgeführt, so muss er gesondert gehandhabt werden. Generell sollte bei Auftreten dieses Sonderfalls die Frage geklärt werden, ob die Festlegung des Gütemaßes in der durchgeführten Art und Weise sinnvoll ist.

Die Tatsache, dass für die Programmschritte Rekombination und Selektion eine feste Laufzeit garantiert werden kann, bedeutet nicht, dass dies auch für den gesamten Optimierungsprozess zutrifft bzw. dass innerhalb einer zulässigen Zeitdauer ein bestimmter Gütewert garantiert erreicht werden kann. Es wird lediglich garantiert, dass eine bestimmte Anzahl an Durchläufen der einzelnen Programmschritte eine feste Laufzeit benötigt und diese nicht in Abhängigkeit von Zufallsentscheidungen variiert.

6.1.3 ES_{WBT} bei multikriteriellen Optimierungsproblemen

Liegt ein multikriterielles Optimierungsproblem vor, so wird der Programmschritt Rekombination sowohl in der Variante $ES_{WBT}^{T_{var}}$ als auch in der zweiten Variante $ES_{WBT}^{T_{const}}$ analog zum Fall eines skalaren Optimierungsproblems durchgeführt.

Im Programmschritt Selektion wird die Teilmengenbildung in der Variante $ES_{WBT}^{T_{var}}$ durch eine vorläufige Teilmengenbildung ersetzt, die prinzipiell genauso durchgeführt wird wie die Teilmengenbildung bei Vorliegen eines skalaren Optimierungsproblems. Hier wird allerdings für die Bestimmung von $D_{Sel}(d)$ der empirisch ermittelte Wert von $\sigma_{Sel} = 0.44$ verwendet. Es werden also stets zwei Individuen \mathbf{x}_h und \mathbf{x}_m als vorläufige Teilmenge wie oben beschrieben bestimmt und die den Individuen zugehörigen Gütewerte miteinander verglichen. Da es bei multikriteriellen Optimierungsproblemen vorkommen kann, dass eines der Individuen bezüglich eines Gütekriteriums einen günstigeren und bezüglich eines anderen Gütekriteriums einen schlechteren Gütewert aufweist als das andere Individuum (nicht vergleichbare Vektoren, s. Abschnitt 3.2), wird in diesem Fall bei $ES_{WBT}^{T_{var}}$ keines der beiden Individuen eliminiert und es werden in einer erneuten vorläufigen Teilmengenbildung zwei neue Individuen \mathbf{x}_h und \mathbf{x}_m bestimmt und ein erneuter Vergleich der Gütewerte durchgeführt. Erst dann, wenn eines der beiden Individuen das andere dominiert (s. Abschnitt 3.2), also bezüglich aller Gütekriterien besser oder zumindest genauso gut ist wie das andere, wird die vorläufige Teilmenge als endgültige Teilmenge $\mathcal{M}_{j,\bar{j}}$ übernommen und es folgt die Elimination des dominierten Individuums.

Der Nachteil dieser Vorgehensweise ist unmittelbar ersichtlich: Es ist nicht nur so, dass hierbei durch die Zufallsentscheidung ‘*vorläufige Teilmengenbildung durchführen* \rightarrow *ja/nein*’ sowie durch die von den vorliegenden Güte-

ten abhängige Festlegung der vorläufigen Teilmenge als endgültige Teilmenge keine feste Laufzeit des Programmschritts Selektion angegeben werden kann. Möglicherweise sind auch so viele paarweise nicht vergleichbare Individuen in der Menge \mathcal{K}_j enthalten, dass gar nicht ausreichend viele Individuen eliminiert werden können und der Optimierungsprozess abgebrochen werden muss. Das Laufzeitproblem, das sich aus der Zufallsentscheidung ‘vorläufige Teilmengebildung durchführen \rightarrow ja/nein’ ergibt, wurde durch die Vergrößerung des Wertes von σ_{sel} gemildert, aber nicht behoben. [Nic99a]

Der genannte Nachteil einer nicht begrenzten Laufzeit der Variante $ES_{WBT}^{T_{var}}$ bei Vorliegen eines multikriteriellen Optimierungsproblems wird in der zweiten Variante $ES_{WBT}^{T_{const}}$ behoben.

Auch in der Variante $ES_{WBT}^{T_{const}}$ wird die bei skalaren Optimierungsproblemen vorgenommene Teilmengebildung durch eine vorläufige Teilmengebildung ersetzt. Hierbei werden aber *zwei* statt bisher nur ein Individuum zusätzlich zum Referenzindividuum \mathbf{x}_{ref} bestimmt und die Gütewerte dieser drei Individuen paarweise miteinander verglichen. Wurde also in einem ersten Schritt das Individuum \mathbf{x}_h gezogen, so wird in einem zweiten Schritt die analoge Vorgehensweise auf die Menge $\bar{\mathcal{M}}_{j,\bar{j}-1} \setminus \{\mathbf{x}_{ref}, \mathbf{x}_h\}$ angewendet. Die vorläufige Teilmengebildung wird zu einer endgültigen Teilmengebildung, wenn eines oder zwei der drei Individuen bezüglich aller Gütekriterien schlechter oder bestenfalls gleich gut sind wie mindestens ein anderes Individuum. Dann werden alle dominierten Individuen aus der Teilmenge $\mathcal{M}_{j,\bar{j}}$ in der anschließenden Elimination aus dem Optimierungsprozess ausgesondert. Werden die Gütewerte dreier Individuen paarweise miteinander verglichen, so wird die Chance erhöht, von diesen mindestens eines eliminieren zu können [Lam02].

Auch hier kann das Problem auftreten, dass aus der Menge \mathcal{K}_j nicht ausreichend viele Individuen eliminiert werden können. Deshalb wird in der Variante $ES_{WBT}^{T_{const}}$ bei multikriteriellen Optimierungsproblemen eine maximale Anzahl an vorläufigen Teilmengebildungen festgelegt. Ein Wert von $8 \cdot (\tilde{k} - \mu)$ hat sich hier als günstig erwiesen (s. Gleichung (4.21)). Somit werden für jedes zu eliminierende Individuum durchschnittlich 8 Versuche durch eine vorläufige Teilmengebildung zugelassen. Ist diese maximale Anzahl an vorläufigen Teilmengebildungen erreicht, so bilden alle dann noch nicht eliminierten Individuen die Menge \mathcal{M}_{j+1} . Die Anzahl der Eltern-Individuen μ^* kann also größer als die ursprünglich festgelegte Anzahl μ werden. Wird eine $(\mu^* + \lambda)$ -Strategie verwendet, so kann ein grenzenloses Anwachsen der Größe der Elternpopulationen im Laufe der Generationen und damit wachsender Verwaltungsaufwand die Folge sein. Trotzdem kann eine maximale Laufzeit des Programmschritts Selektion garantiert werden.

6.1.4 Ergebnisse mit ES_{WBT} bei Verwendung einer analytischen Testfunktion als Gütemaß

Im Folgenden werden die unterschiedlichen Optimierungsstrategien ES_{std} , $ES_{WBT}^{T_{var}}$ sowie $ES_{WBT}^{T_{const}}$ anhand verschiedener Optimierungsprozesse miteinander verglichen.

Jede der Optimierungsstrategien wurde in Kombination mit der verwendeten Testfunktion 10 mal eingesetzt.

Als Testfunktion für ein skalares Optimierungsproblem wird die Six-Hump-Camel-Back-Funktion nach Gleichung (5.1) verwendet. Pro Generation werden $\mu = 15$ Eltern-Individuen und $\lambda = 100$ Nachkommen-Individuen verwendet. Sämtliche Eltern-Individuen der Startpopulation werden in der linken oberen Ecke des Suchraums platziert ($\mathbf{x} = (-2.5, 1.5)$). Hierdurch wird der Optimierungsprozess absichtlich erschwert, um die unterschiedlichen Optimierungsstrategien besser miteinander vergleichen zu können.

Abbildung 6.3 zeigt die Resultate typischer Optimierungsprozesse mit den unterschiedlichen Optimierungsstrategien. Bezogen auf die Breite der Durchsuchung des Suchraums bei Verwendung der einzelnen Optimierungsstrategien werden sowohl in diesem Abschnitt als auch in den folgenden Abschnitten grafische Darstellungen der Individuen im Suchraum zweidimensionaler Testfunktionen herangezogen, um die getroffenen Aussagen zu belegen.

Zu Vergleichszwecken wurde die Optimierungsstrategie ES_{std} herangezogen (Abbildung 6.3 (a) zeigt Resultate eines typischen Optimierungsprozesses). Links sind die μ Eltern-Individuen aller Generationen in einer Grafik dargestellt. Hierdurch wird die Ausbreitung der Eltern-Individuen innerhalb des Suchraums deutlich. Alternativ wäre auch die Darstellung *aller* Individuen (also der Eltern-Individuen, der Nachkommen-Individuen sowie der mutierten Nachkommen-Individuen) innerhalb des Optimierungsprozesses möglich. Aus Gründen der besseren Übersichtlichkeit der grafischen Darstellungen wurde aber auf die alleinige Darstellung der Eltern-Individuen zurückgegriffen. Die Aussage bezüglich der Breite der Durchsuchung des Suchraums lässt sich aus beiden Darstellungsmöglichkeiten ableiten. Rechts sind die Gütewerte aller Eltern-Individuen über der Anzahl M der durchgeführten Bewertungen mittels des Gütemaßes im Laufe des Optimierungsprozesses aufgetragen. Hierbei wurden die Gütewerte einer Elternpopulation jeweils nach Durchlaufen einer kompletten Generation aufgetragen. Daher sind in der Grafik mehrere Gütewerte zu einem bestimmten Wert von M dargestellt. Wie in Kapitel 5 erläutert, können bei der Six-Hump-Camel-Back-Funktion nach Gleichung (5.1) auch negative Funktionswerte auftreten. Um dennoch eine logarithmische Darstellung der Gütewerte zeigen zu

können, wurden negative Funktionswerte bei einem Funktionswert von 10^{-7} als \circ -Symbol dargestellt. Bei Verwendung der Strategie ES_{std} wurden bei den 10 durchgeführten Optimierungsprozessen durchschnittlich 1861 Bewertungen mittels des Gütemaßes benötigt, um eines der beiden globalen Optima mit der geforderten Genauigkeit zu erreichen [Lam02].

In Abbildung 6.3 (b) sind die Resultate eines typischen Optimierungsprozesses bei Verwendung der Optimierungsstrategie $ES_{WBT}^{T_{var}}$ dargestellt. Ziel dieser Optimierungsstrategie ist es, eine insgesamt stärker ausgeprägte Breitensuche als bei Verwendung der Strategie ES_{std} zu erreichen. In der linken Darstellung in Abbildung 6.3 (b) ist zu erkennen, dass diese stärker ausgeprägte Breitensuche erreicht wurde. Bei Verwendung der Six-Hump-Camel-Back-Funktion als Gütemaß hat sich allerdings gezeigt, dass diese stärker ausgeprägte Breitensuche mit durchschnittlich längeren Optimierungsdauern – d. h. mehr Bewertungen mittels des Gütemaßes – bezahlt werden muss (s. Abbildung 6.4). Bei Verwendung der Strategie $ES_{WBT}^{T_{var}}$ wurden durchschnittlich 2821 Bewertungen mittels des Gütemaßes benötigt [Lam02].

Abbildung 6.3 (c) zeigt die Ergebnisse eines typischen Optimierungsprozesses bei Verwendung der Strategie $ES_{WBT}^{T_{const}}$. Durch diese Strategie wurde die Breitensuche der Strategie $ES_{WBT}^{T_{var}}$ etwas reduziert. Es ist aber dennoch eine deutlich breiter angelegte Durchsuchung des Suchraums als bei Verwendung der Strategie ES_{std} zu erkennen. Gleichzeitig konnte die Anzahl der durchschnittlich benötigten Bewertungen mittels des Gütemaßes gegenüber der Strategie $ES_{WBT}^{T_{var}}$ reduziert werden (s. Abbildung 6.4). Bei Verwendung der Strategie $ES_{WBT}^{T_{const}}$ wurden durchschnittlich 1881 Bewertungen mittels des Gütemaßes benötigt, näherungsweise so viele Bewertungen wie bei Verwendung der Strategie ES_{std} .

6.2 Evolutionsstrategie mit gleichzeitiger Breiten- und Tiefensuche (ES_{GBT})

Die Idee zur weiteren Verbesserung der in Abschnitt 6.1 vorgestellten Gesamtstrategie ES_{WBT} besteht darin, nicht in wechselnder Weise durch den Programmschritt Selektion eine verstärkte Breitensuche und durch den Programmschritt Rekombination eine verstärkte Tiefensuche vorzunehmen, sondern sowohl Breiten- als auch Tiefensuche innerhalb beider Programmschritte Rekombination und Selektion zu ermöglichen. Das Ziel ist hierbei keine Ausrichtung eines gesamten Programmschritts auf verstärkte Breiten- oder Tiefensuche, sondern eine individuelle Entscheidung innerhalb eines Rekombinations- bzw. innerlich eines Aussonderungsschritts.

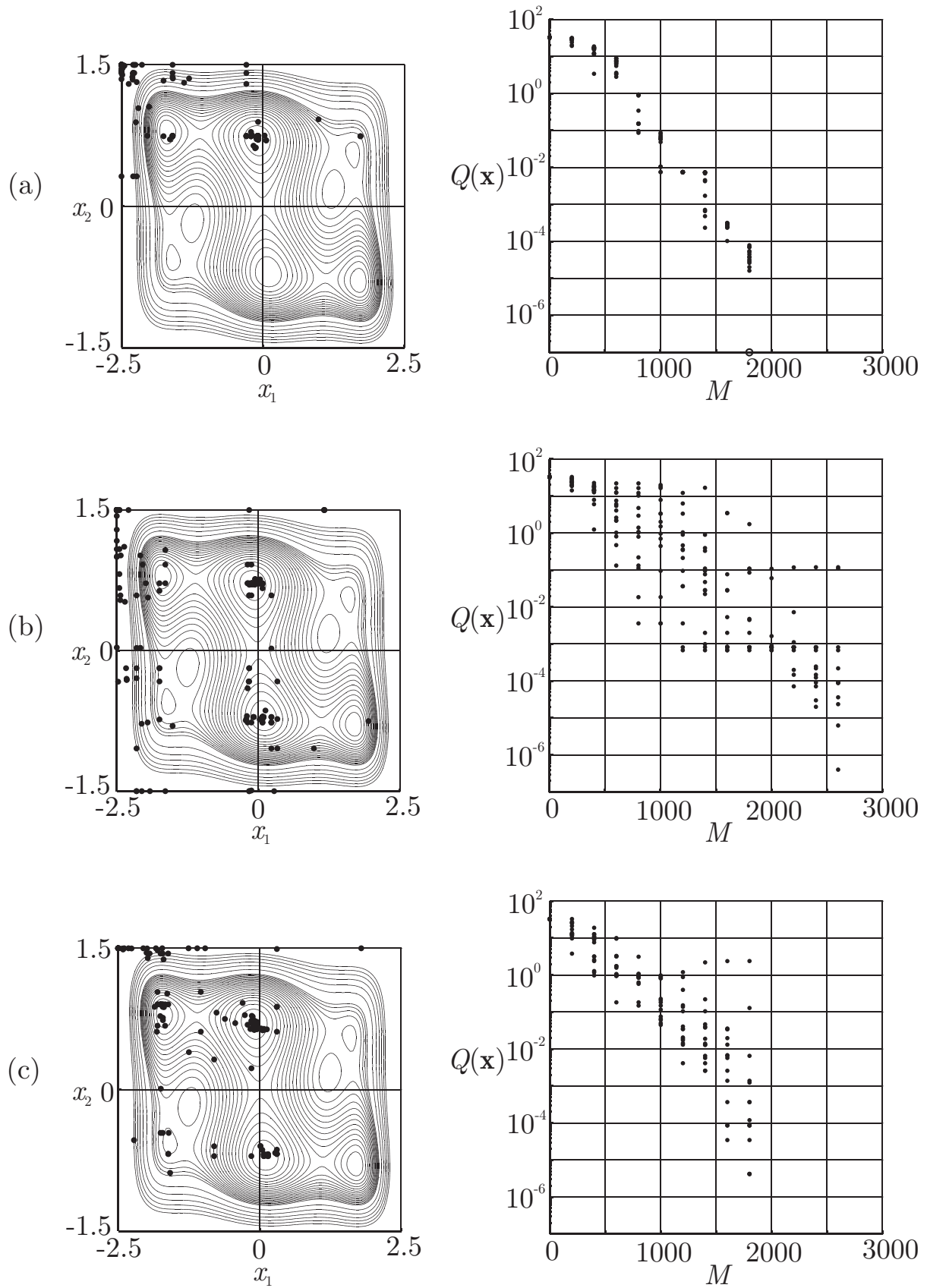


Abbildung 6.3: links: Eltern-Individuen aller Generationen im Suchraum, rechts: Gütewerte aller Eltern-Individuen über der Anzahl der Bewertungen durch das Gütemaß Six-Hump-Camel-Back-Funktion, Strategie ES_{std} (a), Strategie $ES_{WBT}^{T_{var}}$ (b), Strategie $ES_{WBT}^{T_{const}}$ (c).

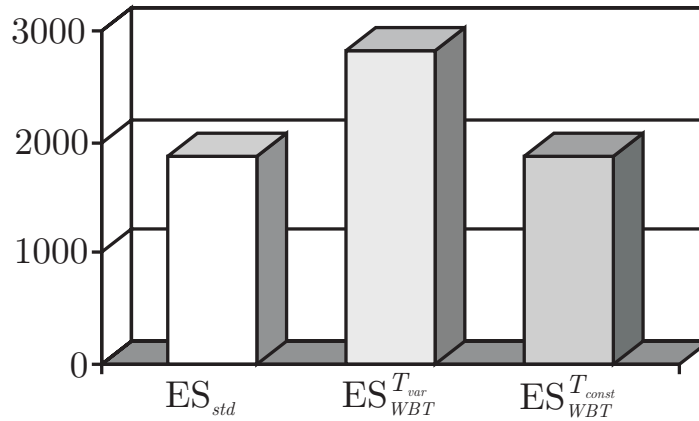


Abbildung 6.4: Durchschnittlich benötigte Anzahl an Bewertungen mittels des Gütemaßes Six-Hump-Camel-Back-Funktion bei jeweils 10 Optimierungsprozessen.

Da innerhalb eines Programmschritts Rekombination bzw. Selektion sowohl eine verstärkte Breitensuche als auch eine verstärkte Tiefensuche vorkommen kann, wird die hier vorgestellte Gesamtstrategie mit *Evolutionstrategie mit gleichzeitiger Breiten- und Tiefensuche* (ES_{GBT}) bezeichnet.

Die Vorgehensweise bei einem Rekombinationsschritt erfolgt in ähnlicher Weise wie bei der Variante $ES_{WBT}^{T_{const}}$. Der einzige Unterschied besteht in der Wahrscheinlichkeitsdichte-Funktion, mit der aus allen Individuen der Menge $\{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$ das Individuum bestimmt wird, das für den Rekombinationsschritt zusammen mit \mathbf{x}_{ref} verwendet wird. Im Gegensatz zur Variante $ES_{WBT}^{T_{const}}$ wird hier nicht eine Wahrscheinlichkeitsdichte-Funktion verwendet, die nur abstandsabhängig ist, sondern eine Wahrscheinlichkeitsdichte-Funktion, die sowohl vom Abstand eines Individuums aus der Menge $\{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$ zum Referenzindividuum \mathbf{x}_{ref} als auch von der Güte des Referenzindividuum (bei Vorliegen eines skalaren Optimierungsproblems: $Q(\mathbf{x}_{ref})$) abhängt.

Als Maß für den Abstand zweier Individuen \mathbf{x}_h und \mathbf{x}_m wird auch bei Einsatz der Strategie ES_{GBT} das Abstandsmaß entsprechend Gleichung (6.1) verwendet:

$$d(\mathbf{x}_h, \mathbf{x}_m) = \frac{1}{n} \cdot \sum_{i=1}^n \frac{|x_{i,h} - x_{i,m}|}{x_{i,max} - x_{i,min}}.$$

Für die Teilmengenbildung im Programmschritt Selektion wird ebenfalls eine Wahrscheinlichkeitsdichte-Funktion verwendet, die sowohl abstands- als auch güteabhängig ist.

Auch die Strategie ES_{GBT} wurde zunächst in einer ersten Variante implementiert und anschließend durch empirische Untersuchungen in eine zweite, verbesserte Variante überführt.

6.2.1 ES_{GBT} mit symmetrischen Wahrscheinlichkeitsdichtefunktionen: ES_{GBT}^{symm}

In der ersten Variante wurde für die zur Bestimmung der besagten Wahrscheinlichkeitsdichtefunktion benötigte Wahrscheinlichkeitsfunktion im Fall eines skalaren Gütemaßes Q für einen Rekombinationsschritt folgender Ansatz aufgestellt:

$$D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref}) = \max(D_1(\mathbf{x}_h, \mathbf{x}_{ref}), D_2(\mathbf{x}_h, \mathbf{x}_{ref})).$$

Hierbei wird für D_1 folgende Teilfunktion verwendet:

$$D_1(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{d(\mathbf{x}_h, \mathbf{x}_{ref})^2}{2 \cdot \sigma_{Rek 1}^2} + \frac{(\max(0, Q(\mathbf{x}_{ref}) - Q_{min}))^2}{2 \cdot \sigma_{Rek 2}^2}\right)} \quad (6.8)$$

$$\text{mit } \sigma_{Rek 1} = 0.2 \quad \text{und} \quad \sigma_{Rek 2} = 0.2 \cdot (Q_{max} - Q_{min}).$$

Q_{min} ist hierbei der minimale Wert, Q_{max} der maximale Wert aller bisher innerhalb des Optimierungsprozesses aufgetretenen Gütewerte. Die Werte für Q_{min} und Q_{max} werden fortlaufend bei jeder vorgenommenen Gütebewertung überprüft und ggf. aktualisiert. Ist der Wertebereich der auftretenden Gütewerte innerhalb des Suchraums bekannt, so können die Werte Q_{min} und Q_{max} auch vor Beginn des Optimierungsprozesses angegeben werden. Häufig ist dieser Wertebereich jedoch nicht oder nur unzureichend bekannt.

Die Werte Q_{min} und Q_{max} können auch so festgelegt werden, dass durch diesen Wertebereich $[Q_{min}, Q_{max}]$ ein bestimmter Teilbereich innerhalb des Bereichs der tatsächlich auftretenden Gütewerte angegeben wird. Somit können Grenzwerte angegeben werden, ab welchen ein bestimmter Gütewert als vergleichsweise günstig oder als vergleichsweise schlecht gelten soll. Für diese Vorgehensweise sind die \max -Operatoren in Gleichung (6.8) und Gleichung (6.9) erforderlich.

Für D_2 wird folgende Teilfunktion verwendet:

$$D_2(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{(1-d(\mathbf{x}_h, \mathbf{x}_{ref}))^2}{2 \cdot \sigma_{Rek 3}^2} + \frac{(\max(0, Q_{max} - Q(\mathbf{x}_{ref})))^2}{2 \cdot \sigma_{Rek 4}^2}\right)} \quad (6.9)$$

$$\text{mit } \sigma_{Rek 3} = 0.2 \quad \text{und} \quad \sigma_{Rek 4} = 0.2 \cdot (Q_{max} - Q_{min}).$$

In Abbildung 6.5 (a) ist die resultierende Funktion $D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$ dargestellt. Wurde für diesen Rekombinationsschritt ein vergleichsweise schlechtes Referenzindividuum festgelegt ($Q(\mathbf{x}_{ref}) \approx Q_{max}$), so ist in diesem Fall eine ausgeprägte

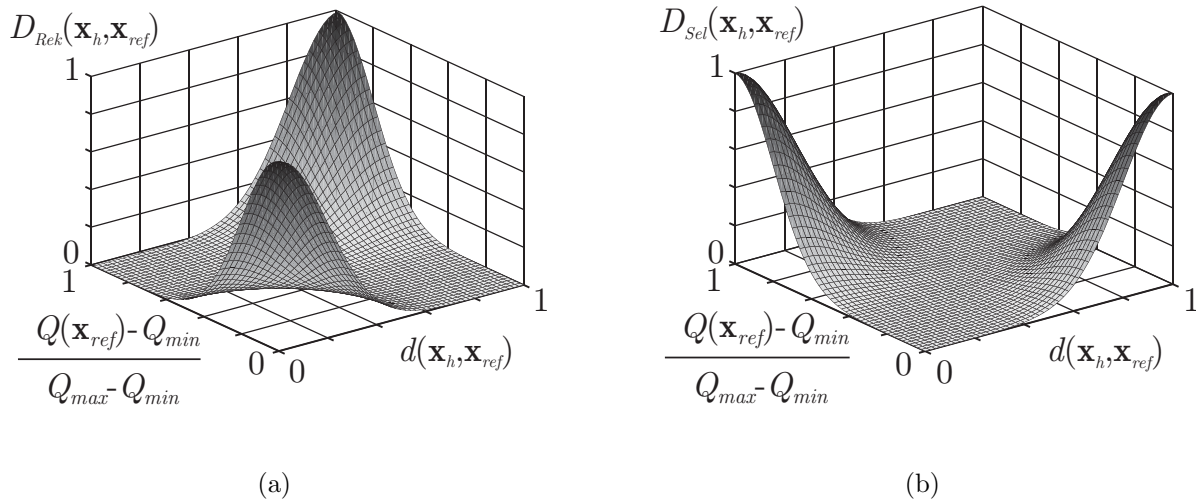


Abbildung 6.5: $D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$ für einen Rekombinationsschritt (a) bzw. $D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})$ für eine Teilmengenbildung (b) für $Q_{min} \leq Q(\mathbf{x}_{ref}) \leq Q_{max}$ in der Variante ES_{GBT}^{symm}.

Breitensuche erwünscht. Aufgrund dessen wird bevorzugt ein Partner mit großem räumlichem Abstand für den Rekombinationsschritt zusammen mit \mathbf{x}_{ref} bestimmt. Wurde hingegen für diesen Rekombinationsschritt ein Referenzindividuum mit vergleichsweise günstigem Gütewert festgelegt ($Q(\mathbf{x}_{ref}) \approx Q_{min}$), so wird eine verstärkte Tiefensuche angestrebt. Dann wird bevorzugt ein Partner bestimmt, der geringen räumlichen Abstand zu dem Referenzindividuum aufweist.

Mit der Summe über $D_{Rek}(\mathbf{x}_i, \mathbf{x}_{ref})$ über alle Individuen $\mathbf{x}_i \in \{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$

$$D_{Rek}^{Sum} = \sum_{i=1}^{\mu-1} D_{Rek}(\mathbf{x}_i, \mathbf{x}_{ref})$$

wird für jedes Individuum \mathbf{x}_h der Wert der Rekombinations-Wahrscheinlichkeitsdichte wie in Gleichung (6.6) bestimmt:

$$P_{Rek}(\mathbf{x}_h) = \frac{D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})}{D_{Rek}^{Sum}}. \quad (6.10)$$

Unter Berücksichtigung aller Rekombinations-Wahrscheinlichkeitsdichten wird durch einen Zufallsgenerator das Individuum aus der Menge $\{\mathcal{M}_j \setminus \mathbf{x}_{ref}\}$ bestimmt, das zusammen mit \mathbf{x}_{ref} für den Rekombinationsschritt herangezogen wird.

Für eine Teilmengenbildung im Programmschritt Selektion wird in der

ersten Variante für die Wahrscheinlichkeitsfunktion zur Bestimmung der Kampf-Wahrscheinlichkeitsdichte im Fall eines skalaren Gütemaßes ein programmiertechnisch ähnlicher Ansatz wie im Programmschritt Rekombination gewählt:

$$D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref}) = \max(D_3(\mathbf{x}_h, \mathbf{x}_{ref}), D_4(\mathbf{x}_h, \mathbf{x}_{ref})).$$

Hierbei wird für D_3 folgende Teilfunktion verwendet:

$$D_3(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{d(\mathbf{x}_h, \mathbf{x}_{ref})^2}{2 \cdot \sigma_{Sel 1}^2} + \frac{(\max(0, Q_{max} - Q(\mathbf{x}_{ref})))^2}{2 \cdot \sigma_{Sel 2}^2}\right)} \quad (6.11)$$

$$\text{mit } \sigma_{Sel 1} = 0.2 \quad \text{und} \quad \sigma_{Sel 2} = 0.2 \cdot (Q_{max} - Q_{min}).$$

Für D_4 wird folgende Teilfunktion verwendet:

$$D_4(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{(1-d(\mathbf{x}_h, \mathbf{x}_{ref}))^2}{2 \cdot \sigma_{Sel 3}^2} + \frac{(\max(0, Q(\mathbf{x}_{ref}) - Q_{min}))^2}{2 \cdot \sigma_{Sel 4}^2}\right)} \quad (6.12)$$

$$\text{mit } \sigma_{Sel 3} = 0.2 \quad \text{und} \quad \sigma_{Sel 4} = 0.2 \cdot (Q_{max} - Q_{min}).$$

Abbildung 6.5 (b) zeigt die Funktion $D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})$. Wurde für diese Teilmengebildung ein vergleichsweise schlechtes Referenzindividuum festgelegt ($Q(\mathbf{x}_{ref}) \approx Q_{max}$), so ist auch hier eine ausgeprägte Breitensuche erwünscht. Aufgrund dessen wird bevorzugt ein Turniergegner mit geringem räumlichem Abstand für das Turnier gegen \mathbf{x}_{ref} bestimmt. Wurde hingegen für diese Teilmengebildung ein vergleichsweise gut bewertetes Referenzindividuum festgelegt ($Q(\mathbf{x}_{ref}) \approx Q_{min}$), so wird auch im Programmschritt Selektion eine verstärkte Tiefensuche angestrebt. Dann wird bevorzugt ein Turniergegner bestimmt, der großen räumlichen Abstand zu dem Referenzindividuum aufweist. Mit der Summe über $D_{Sel}(\mathbf{x}_i, \mathbf{x}_{ref})$ über alle Individuen $\mathbf{x}_i \in \{\bar{\mathcal{M}}_{j, \bar{j}-1} \setminus \mathbf{x}_{ref}\}$

$$D_{Sel}^{Sum} = \sum_{i=1}^{card(\bar{\mathcal{M}}_{j, \bar{j}-1})-1} D_{Sel}(\mathbf{x}_i, \mathbf{x}_{ref})$$

wird für jedes Individuum \mathbf{x}_h der Wert der Kampf-Wahrscheinlichkeitsdichte

$$P_{Sel}(\mathbf{x}_h) = \frac{D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})}{D_{Sel}^{Sum}} \quad (6.13)$$

ermittelt (vgl. Gleichung (6.7)) und unter Verwendung eines Zufallsgenerators und unter Berücksichtigung der Kampf-Wahrscheinlichkeitsdichten aller Individuen aus $\{\bar{\mathcal{M}}_{j,\bar{j}-1} \setminus \mathbf{x}_{ref}\}$ das Individuum bestimmt, das in einem Turnier gegen \mathbf{x}_{ref} antreten muss.

Da die Wahrscheinlichkeitsdichte-Funktionen in Abbildung 6.5 symmetrisch angelegt sind, wird die beschriebene erste Variante von ES_{GBT} im Folgenden mit ES_{GBT}^{symm} bezeichnet.

6.2.2 ES_{GBT} mit unsymmetrischen Wahrscheinlichkeitsdichtefunktionen: ES_{GBT}^{unsymm}

In der zweiten Variante der Strategie ES_{GBT} besteht die Veränderung gegenüber ES_{GBT}^{symm} in den Teilfunktionen D_1 , D_2 , D_3 und D_4 . Durch diese Veränderungen kann ein besserer Kompromiss zwischen Breiten- und Tiefensuche erreicht werden. Durch die Verwendung der symmetrisch angelegten Wahrscheinlichkeitsdichte-Funktionen setzte in experimentellen Untersuchungen frühzeitig die Tiefensuche ein, die Breitensuche wurde nicht ausreichend lange durchgeführt. Hieraus resultierte die Überlegung, dass unsymmetrische Wahrscheinlichkeitsdichte-Funktionen sinnvoller sein könnten [Lam02].

Für einen Rekombinationsschritt werden bei Vorliegen eines skalaren Gütemaßes folgende Teilfunktionen verwendet:

$$D_1(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{d(\mathbf{x}_h, \mathbf{x}_{ref})^2}{2 \cdot \sigma_{Rek\ 1}^2} + \frac{(\max(0, Q(\mathbf{x}_{ref}) - Q_{min}))^2}{2 \cdot \sigma_{Rek\ 2}^2}\right)} \quad (6.14)$$

$$\text{mit } \sigma_{Rek\ 1} = 0.2 \quad \text{und} \quad \sigma_{Rek\ 2} = 0.01 \cdot (Q_{max} - Q_{min}).$$

Für D_2 wird hier folgende Teilfunktion verwendet:

$$D_2(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{(1-d(\mathbf{x}_h, \mathbf{x}_{ref}))^2}{2 \cdot \sigma_{Rek\ 3}^2} + \frac{(\max(0, Q_{max} - Q(\mathbf{x}_{ref})))^2}{2 \cdot \sigma_{Rek\ 4}^2}\right)} \quad (6.15)$$

$$\text{mit } \sigma_{Rek\ 3} = 0.2 \quad \text{und} \quad \sigma_{Rek\ 4} = 0.9 \cdot (Q_{max} - Q_{min}).$$

Die resultierende Funktion $D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$ für einen Rekombinationsschritt ist in Abbildung 6.6 (a) dargestellt.

Durch diese Modifikationen gegenüber der ursprünglichen Strategie ES_{GBT}^{symm} wird erreicht, dass bei vergleichsweise sehr hoher Güte des Referenzindividuum

($Q(\mathbf{x}_{ref}) \approx Q_{min}$) bevorzugt Partner für den Rekombinationsschritt bestimmt werden, die entweder einen großen räumlichen Abstand oder einen geringen räumlichen Abstand zu dem Referenzindividuum haben. Partner mit mittelgroßem Abstand zu dem Referenzindividuum werden vermieden. Somit wird in diesem Fall eines sehr guten Referenzindividuums entweder eine verstärkte Tiefensuche (wie in der Strategie ES_{GBT}^{symm}) oder eine stark ausgeprägte Breitensuche erreicht.

Ist die Güte des Referenzindividuums vergleichsweise schlecht ($Q(\mathbf{x}_{ref}) \approx Q_{max}$), so ist auch in diesem Fall eine ausgeprägte Breitensuche erwünscht. Deshalb wird auch hier bevorzugt ein Partner für den Rekombinationsschritt zusammen mit \mathbf{x}_{ref} festgelegt, der großen räumlichen Abstand zu \mathbf{x}_{ref} aufweist.

Für eine Teilmengenbildung im Programmschritt Selektion werden bei einem skalaren Optimierungsproblem die folgenden Teilfunktionen verwendet:

$$D_3(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{d(\mathbf{x}_h, \mathbf{x}_{ref})^2}{2 \cdot \sigma_{Sel 1}^2} + \frac{(\max(0, Q_{max} - Q(\mathbf{x}_{ref})))^2}{2 \cdot \sigma_{Sel 2}^2}\right)} \quad (6.16)$$

$$\text{mit } \sigma_{Sel 1} = 0.2 \quad \text{und} \quad \sigma_{Sel 2} = 0.9 \cdot (Q_{max} - Q_{min}).$$

Für D_4 wird in dieser Variante folgende Teilfunktion verwendet:

$$D_4(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{(1-d(\mathbf{x}_h, \mathbf{x}_{ref}))^2}{2 \cdot \sigma_{Sel 3}^2} + \frac{(\max(0, Q(\mathbf{x}_{ref}) - Q_{min}))^2}{2 \cdot \sigma_{Sel 4}^2}\right)} \quad (6.17)$$

$$\text{mit } \sigma_{Sel 3} = 0.2 \quad \text{und} \quad \sigma_{Sel 4} = 0.01 \cdot (Q_{max} - Q_{min}).$$

Abbildung 6.6 (b) zeigt die resultierende Funktion $D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})$ für eine Teilmengenbildung.

Bei vergleichsweise sehr hoher Güte des Referenzindividuums ($Q(\mathbf{x}_{ref}) \approx Q_{min}$) werden bei der Gegnerauswahl im Programmschritt Selektion bevorzugt Gegner festgelegt, die entweder einen großen oder einen geringen räumlichen Abstand zu dem Referenzindividuum aufweisen. Es sind keine Gegner mit mittelgroßem Abstand zu dem Referenzindividuum erwünscht. Wurde ein sehr gutes Referenzindividuum gezogen, so wird also entweder eine verstärkte Tiefensuche (wie in der Strategie ES_{GBT}^{symm}) oder eine stark ausgeprägte Breitensuche erreicht.

Ist die Güte des Referenzindividuums vergleichsweise schlecht ($Q(\mathbf{x}_{ref}) \approx Q_{max}$), so ist auch in diesem Fall eine ausgeprägte Breitensuche erwünscht.

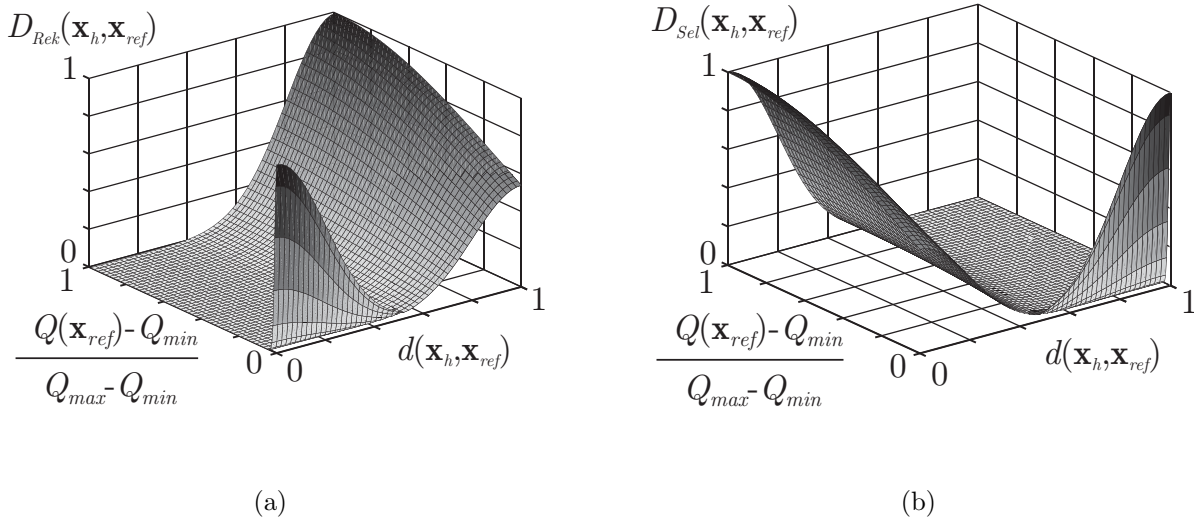


Abbildung 6.6: $D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$ für einen Rekombinationsschritt (a) bzw. $D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})$ für eine Teilmengenbildung (b) für $Q_{min} \leq Q(\mathbf{x}_{ref}) \leq Q_{max}$ in der Variante ES_{GBT}^{unsymm}.

Deshalb wird auch hier bevorzugt ein Gegner für eine Teilmengenbildung zusammen mit \mathbf{x}_{ref} festgelegt, der geringen räumlichen Abstand zu \mathbf{x}_{ref} aufweist.

Aufgrund der zu erkennenden Unsymmetrie in den Funktionen $D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$ und $D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref})$ wird die hier beschriebene Variante der Strategie ES_{GBT} mit ES_{GBT}^{unsymm} bezeichnet.

6.2.3 ES_{GBT} bei multikriteriellen Optimierungsproblemen

Liegt ein multikriterielles Optimierungsproblem mit vektoriellem Gütemaß $\mathbf{Q}(\mathbf{x}) = (Q_1(\mathbf{x}), \dots, Q_k(\mathbf{x}))$ vor, so wird sowohl in der Variante ES_{GBT}^{symm} als auch in der verbesserten Variante ES_{GBT}^{unsymm} für jeden Rekombinationsschritt anstelle der Funktion

$$D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref}) = \max(D_1(\mathbf{x}_h, \mathbf{x}_{ref}), D_2(\mathbf{x}_h, \mathbf{x}_{ref}))$$

in angepasster Weise die Funktion

$$D_{Rek}(\mathbf{x}_h, \mathbf{x}_{ref}) = \sum_{i=1}^k D_{i,Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$$

mit

$$D_{i,Rek}(\mathbf{x}_h, \mathbf{x}_{ref}) = \max(D_{i,1}(\mathbf{x}_h, \mathbf{x}_{ref}), D_{i,2}(\mathbf{x}_h, \mathbf{x}_{ref})),$$

$$D_{i,1}(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{d(\mathbf{x}_h, \mathbf{x}_{ref})^2}{2 \cdot \sigma_{Rek\ 1}^2} + \frac{(\max(0, Q_i(\mathbf{x}_{ref}) - Q_{i,min}))^2}{2 \cdot \sigma_{Rek\ 2}^2}\right)} \quad (6.18)$$

mit $\sigma_{Rek\ 1} = 0.2$ und $\sigma_{Rek\ 2} = 0.2 \cdot (Q_{i,max} - Q_{i,min})$
bzw. $\sigma_{Rek\ 2} = 0.01 \cdot (Q_{i,max} - Q_{i,min})$

und

$$D_{i,2}(\mathbf{x}_h, \mathbf{x}_{ref}) = e^{-\left(\frac{(1-d(\mathbf{x}_h, \mathbf{x}_{ref}))^2}{2 \cdot \sigma_{Rek\ 3}^2} + \frac{(\max(0, Q_{i,max} - Q_i(\mathbf{x}_{ref})))^2}{2 \cdot \sigma_{Rek\ 4}^2}\right)} \quad (6.19)$$

mit $\sigma_{Rek\ 3} = 0.2$ und $\sigma_{Rek\ 4} = 0.2 \cdot (Q_{i,max} - Q_{i,min})$
bzw. $\sigma_{Rek\ 4} = 0.9 \cdot (Q_{i,max} - Q_{i,min})$

verwendet. $Q_{i,min}$ ist hierbei der minimale Gütewert bezüglich des i -ten Gütekriteriums, $Q_{i,max}$ ist im Gegensatz zu Abschnitt 3.2 der maximale Gütewert bezüglich des i -ten Gütekriteriums.

Es wird also für jedes einzelne Gütekriterium so vorgegangen wie im Fall eines skalaren Optimierungsproblems und anschließend werden die Werte $D_{i,Rek}(\mathbf{x}_h, \mathbf{x}_{ref})$ aufsummiert.

D_{Rek}^{Sum} und $P_{Rek}(\mathbf{x}_h)$ werden wie in Abschnitt 6.1.2 bzw. Abschnitt 6.2.1 berechnet.

Für eine Teilmengenbildung in Programmschritt Selektion wird analog zum Ablauf bei einem Rekombinationsschritt vorgegangen. Die Teilfunktionen D_3 und D_4 (Gleichung (6.11) und Gleichung (6.12) bzw. Gleichung (6.16) und Gleichung (6.17)) werden für jedes einzelne Gütekriterium Q_i mit $Q_i(\mathbf{x}_{ref})$, $Q_{i,min}$ und $Q_{i,max}$ anstelle von $Q(\mathbf{x}_{ref})$, Q_{min} und Q_{max} ausgewertet. Hierdurch ergeben sich die Teilfunktionen $D_{i,3}$ und $D_{i,4}$. Für jede Teilmengenbildung wird anstelle der Funktion

$$D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref}) = \max(D_3(\mathbf{x}_h, \mathbf{x}_{ref}), D_4(\mathbf{x}_h, \mathbf{x}_{ref}))$$

für jedes einzelne Gütekriterium die Funktion

$$D_{i,Sel}(\mathbf{x}_h, \mathbf{x}_{ref}) = \max(D_{i,3}(\mathbf{x}_h, \mathbf{x}_{ref}), D_{i,4}(\mathbf{x}_h, \mathbf{x}_{ref})),$$

und schließlich die Summenbildung

$$D_{Sel}(\mathbf{x}_h, \mathbf{x}_{ref}) = \sum_{i=1}^k D_{i,Sel}(\mathbf{x}_h, \mathbf{x}_{ref})$$

verwendet.

D_{Sel}^{Sum} und $P_{Sel}(\mathbf{x}_h)$ werden ebenfalls wie in Abschnitt 6.1.2 bzw. Abschnitt 6.2.1 berechnet.

Auch hier werden, wie in Abschnitt 6.1.3 beschrieben, zwei Individuen zusätzlich zu \mathbf{x}_{ref} bestimmt und die Güterwerte dieser drei Individuen paarweise miteinander verglichen.

6.2.4 Ergebnisse mit ES_{GBT} bei Verwendung einer analytischen Testfunktion als Gütemaß

In diesem Abschnitt wird ein Vergleich der Optimierungsstrategien ES_{GBT}^{symm} und ES_{GBT}^{unsymm} mit den Strategien ES_{std} , $ES_{WBT}^{T_{var}}$ sowie $ES_{WBT}^{T_{const}}$ präsentiert.

Auch hier wurde jede Optimierungsstrategie 10 mal mit der verwendeten Testfunktion kombiniert.

Als Testfunktion für ein skalares Gütemaß wird die Rastrigin-Funktion nach Gleichung (5.3) für $n = 2$ eingesetzt. Pro Generation werden wie in Abschnitt 6.1.4 $\mu = 15$ Eltern-Individuen und $\lambda = 100$ Nachkommen-Individuen verwendet. Alle Eltern-Individuen der Startpopulation werden hier auf dem Punkt $\mathbf{x} = (5.0, 5.0)$ platziert. Somit können die unterschiedlichen Optimierungsergebnisse sowie die Ausbreitungen der Individuen im Suchraum gut miteinander verglichen werden.

In Abbildung 6.7 sind die Ergebnisse typischer Optimierungsprozesse mit den Optimierungsstrategien ES_{std} , ES_{GBT}^{symm} sowie ES_{GBT}^{unsymm} dargestellt.

Auch hier wurde zu Vergleichszwecken die Optimierungsstrategie ES_{std} herangezogen (Abbildung 6.7 (a)). Links sind die μ Eltern-Individuen aller Generationen innerhalb des Suchraums zu sehen. Rechts sind wiederum die Güterwerte aller Eltern-Individuen über der Anzahl M der durchgeführten Bewertungen mittels des Gütemaßes im Laufe des Optimierungsprozesses dargestellt. Anhand der linken Grafik ist erkennbar, dass keine besonders breite Durchsuchung des Suchraums stattfindet. Zunächst erfolgt die Bewegung der Individuen in x_2 -Richtung, bevor eine Wanderung in x_1 -Richtung einsetzt. Bei Verwendung der Strategie ES_{std} wurden durchschnittlich 8641 Bewertungen mittels des Gütemaßes benötigt, um das globale Optimum mit der geforderten Genauigkeit zu

lokalisieren [Lam02]. Bei Verwendung der Strategien $ES_{WBT}^{T_{var}}$ und $ES_{WBT}^{T_{const}}$ wurden durchschnittlich 7401 [Lam02] bzw. 10201 Bewertungen benötigt (s. Abbildung 6.8). Bei diesen beiden neuen Optimierungsstrategien wurde eine bessere Breitensuche erreicht als bei der Strategie ES_{std} . Hier wurde aber auf eine grafische Darstellung der Individuen im Suchraum bzw. der Gütewerte im Laufe des Optimierungsprozesses verzichtet.

In Abbildung 6.7 (b) sind die Ergebnisse eines typischen Optimierungsprozesses bei Verwendung der Optimierungsstrategie ES_{GBT}^{symm} visualisiert. Das Ziel dieser Strategie besteht darin, einen besseren Kompromiss zwischen Breiten- und Tiefensuche und damit ein durchschnittlich schnelleres Auffinden des Optimierungsziels – des bzw. eines globalen Optimums – zu erreichen. Abbildung 6.8 verdeutlicht, dass dieses Ziel bei Anwendung der Strategie ES_{GBT}^{symm} bei der Rastrigin-Funktion als Gütemaß nicht erreicht wurde. Die durchschnittlich benötigte Anzahl an Bewertungen mittels des Gütemaßes liegt bei 19081. Das liegt daran, dass in drei der 10 durchgeführten Optimierungsprozesse das globale Optimum der Funktion auch nach 50000 Bewertungen mittels des Gütemaßes nicht mit der verlangten Genauigkeit erreicht wurde und diese drei Optimierungsprozesse danach abgebrochen wurden [Lam02].¹ Abbildung 6.7 (b) verdeutlicht den Mangel der Strategie ES_{GBT}^{symm} : Es wird zwar eine breiter angelegte Suche als bei der Strategie ES_{std} durchgeführt (s. linke Grafik), die Tiefensuche setzt allerdings bereits bei Gütewerten ein, die noch zu schlecht sind. Somit setzt sich der Optimierungsprozess in einem lokalen Optimum fest (s. rechte Grafik) und in einigen Fällen ist es danach nicht mehr möglich, dieses lokale Optimum wieder zu verlassen und das globale Optimum zu finden.

Der Kompromiss zwischen Breiten- und Tiefensuche konnte durch die Strategie ES_{GBT}^{unsymm} deutlich verbessert werden. Die Breitensuche wird auch bei vergleichsweise guten Gütewerten noch beibehalten und erst bei sehr guten Gütewerten setzt die Tiefensuche ein. Abbildung 6.7 (c) zeigt die Resultate eines typischen Optimierungsprozesses bei Verwendung der Strategie ES_{GBT}^{unsymm} . Es ist eine breit angelegte Durchsuchung des Suchraums zu erkennen. Durch Abbildung 6.8 wird deutlich, dass bei Verwendung der Optimierungsstrategie ES_{GBT}^{unsymm} bei der Rastrigin-Funktion als Gütemaß so wenig Bewertungen mittels des Gütemaßes benötigt werden wie mit keiner anderen der bisher vorgestellten Strategien. Es werden durchschnittlich 5501 Bewertungen durchgeführt [Lam02].

¹Wurde die maximal zugelassene Anzahl von 50000 Gütebewertungen erreicht, so wurde diese Anzahl für die Durchschnittsbildung verwendet.

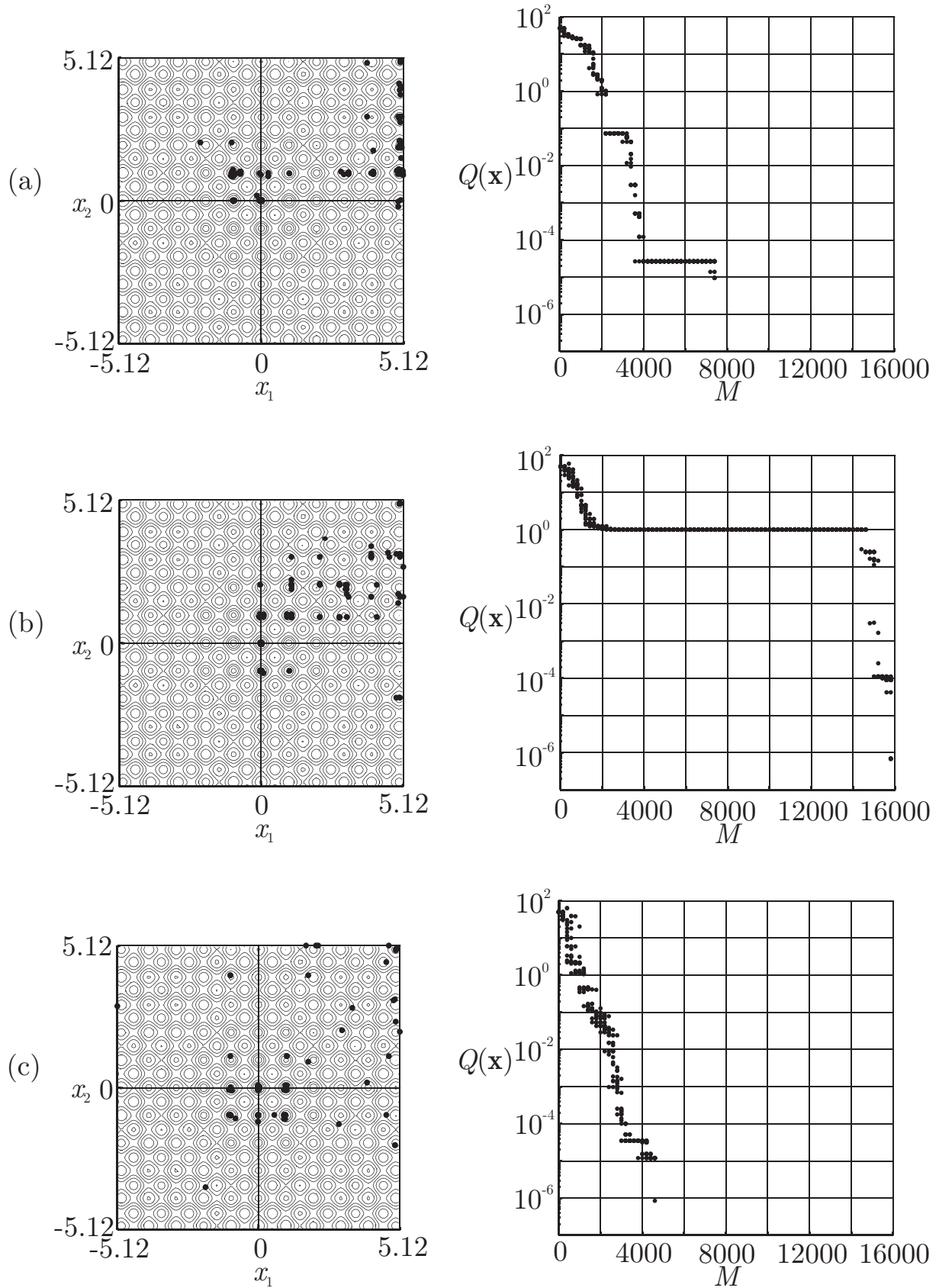


Abbildung 6.7: links: Eltern-Individuen aller Generationen im Suchraum, rechts: Güteverläufe aller Eltern-Individuen über die Anzahl der Bewertungen durch das Gütemaß Rastrigin-Funktion, Strategie ES_{std} (a), Strategie ES_{GBT}^{symm} (b), Strategie ES_{GBT}^{unsymm} (c).

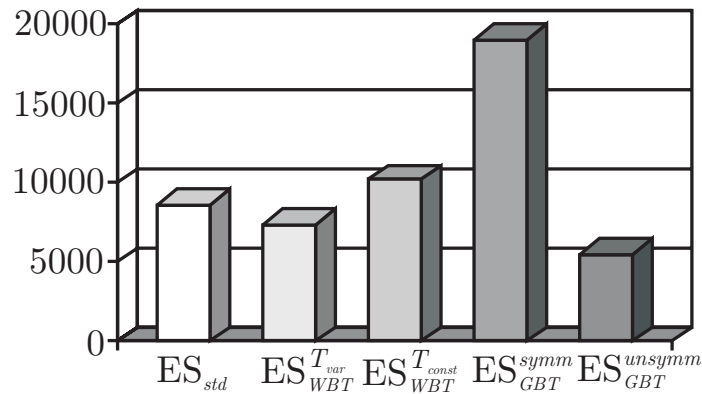


Abbildung 6.8: Durchschnittlich benötigte Anzahl an Bewertungen mittels des Gütemaßes Rastrigin-Funktion bei jeweils 10 Optimierungsprozessen.

6.3 Kombination der Evolutionsstrategie mit einem modifizierten Hooke-Jeeves-Verfahren (ES_{hybrid})

Die Grundidee der hier vorgestellten neuen Strategie besteht darin, die Robustheit einer Evolutionsstrategie mit der hohen Konvergenzgeschwindigkeit des Hooke-Jeeves-Verfahrens zu verkoppeln. Bei unbekannter Gütelandschaft können beide Optimierungsverfahren Vorteile bieten. Durch die vorgenommene Kombination wird dem Anwender die Entscheidung für das eine oder das andere dieser beiden Verfahren abgenommen. Durch eine intelligente Verknüpfung werden die Vorteile beider Verfahren miteinander vereint.

Eine gegenüber dem ursprünglichen Hooke-Jeeves-Verfahren [HJ61] vorgenommene geringfügige Modifikation wurde bereits in [Bro96] vorgestellt. Dabei wurde das ursprüngliche Verfahren so verändert, dass es auch in einem begrenzten Suchraum zur Lösung eines Optimierungsproblems eingesetzt werden kann.

Als weitere Modifikation wurde in dieser Arbeit die bisher verwendete skalare Schrittweite s durch eine vektorielle Schrittweite \mathbf{s} ersetzt. Dadurch werden die Suchschritte in x_i -Richtung ($i = 1, \dots, n$) innerhalb des Programmschritts Exploration mit der Schrittweite s_i ($\mathbf{s} = (s_1, \dots, s_n)$) durchgeführt. Der Vorteil, der sich durch eine vektorielle Schrittweite ergibt, besteht in einer besseren Anpassung der Suchschritte an die Beschaffenheit der Gütelandschaft. Außerdem wird innerhalb des Programmschritts Exploration ein Vorzeichenwechsel $s_i := -s_i$ vorgenommen, wenn der Suchschritt in x_i -Richtung durch Subtraktion von s_i eine Güteverbesserung erbracht hat (vgl. Abschnitt 4.2). Dadurch kann die Anzahl der Bewertungen mittels des Gütemaßes reduziert werden, da im nächsten Programmschritt Exploration der erste Suchschritt in x_i -Richtung mit dem Vorzeichen erfolgt, das bisher bereits erfolgreich war. Das durch diese

Modifikationen leicht veränderte Verfahren wird im Folgenden als *modifiziertes Hooke-Jeeves-Verfahren* bezeichnet.

Die Kombination der Evolutionsstrategie mit dem modifizierten Hooke-Jeeves-Verfahren ist in Abbildung 6.9 dargestellt. Dieses hybride Optimierungskonzept, das im Folgenden mit ES_{hybrid} bezeichnet wird, kann nur zur Lösung skalarer Optimierungsprobleme eingesetzt werden, da das modifizierte Hooke-Jeeves-Verfahren nicht für multikriterielle Optimierungsprobleme konzipiert ist.

Die Verknüpfung beider Verfahren wird folgendermaßen durchgeführt:

Der Optimierungsprozess wird mit der Evolutionsstrategie gestartet. Nachdem die Startpopulation festgelegt und die erste Generation durchlaufen wurde, wird das Individuum aus der Menge \mathcal{M}_1 kopiert, das den besten Gütewert aufweist. Der zugehörige Schrittweitenvektor dieses Individuums wird ebenfalls kopiert. Anschließend wird das kopierte Individuum als Startpunkt für das modifizierte Hooke-Jeeves-Verfahren verwendet, wobei der kopierte Schrittweitenvektor des Individuums als Schrittweitenvektor für das modifizierte Hooke-Jeeves-Verfahren eingesetzt wird. Nun wird das modifizierte Hooke-Jeeves-Verfahren mit einer festgelegten Anzahl an Bewertungen mittels des Gütemaßes durchgeführt (s. Tabelle 6.1). Wurde diese festgelegte Anzahl an Bewertungen innerhalb des modifizierten Hooke-Jeeves-Verfahrens erreicht, so wird die Teiloptimierung mit dem modifizierten Hooke-Jeeves-Verfahren beendet und der durch die Hooke-Jeeves-Teiloptimierung ermittelte Parametervektor wird zur Population \mathcal{M}_1 hinzugefügt. Der innerhalb der Hooke-Jeeves-Teiloptimierung veränderte Schrittweitenvektor wird nicht weiter verwendet, da er in der Regel für eine anschließende ES-Generation zu kleine Werte aufweist. Stattdessen wird dem Parametervektor der ursprüngliche Schrittweitenvektor des aus \mathcal{M}_1 kopierten Individuums zugewiesen. So wird die Menge \mathcal{M}_2 aus der Vereinigungsmenge von \mathcal{M}_1 und dem ermittelten Parametervektor gebildet (s. Abbildung 6.10). Anschließend wird auf \mathcal{M}_2 wieder der Programmschritt Rekombination angewendet (s. Abbildung 6.9). Prinzipiell ist es auch möglich, mehrere Individuen aus der Menge \mathcal{M}_1 als Startpunkte für das modifizierte Hooke-Jeeves-Verfahren zu verwenden.

Nach der zweiten ES-Generation wird die Hooke-Jeeves-Teiloptimierung wie in Abbildung 6.11 dargestellt durchgeführt.

Eine sogenannte *Blockierung* ist zunächst ausgeschaltet. Aus der Menge \mathcal{M}_j wird wieder das beste Individuum und dessen Schrittweitenvektor kopiert.² Anschließend wird überprüft, ob eine *Veränderung* gegenüber dem besten Pa-

²Auch hier wäre es möglich, mehrere Individuen aus \mathcal{M}_j zu kopieren und als Startpunkte für das modifizierte Hooke-Jeeves-Verfahren zu verwenden.

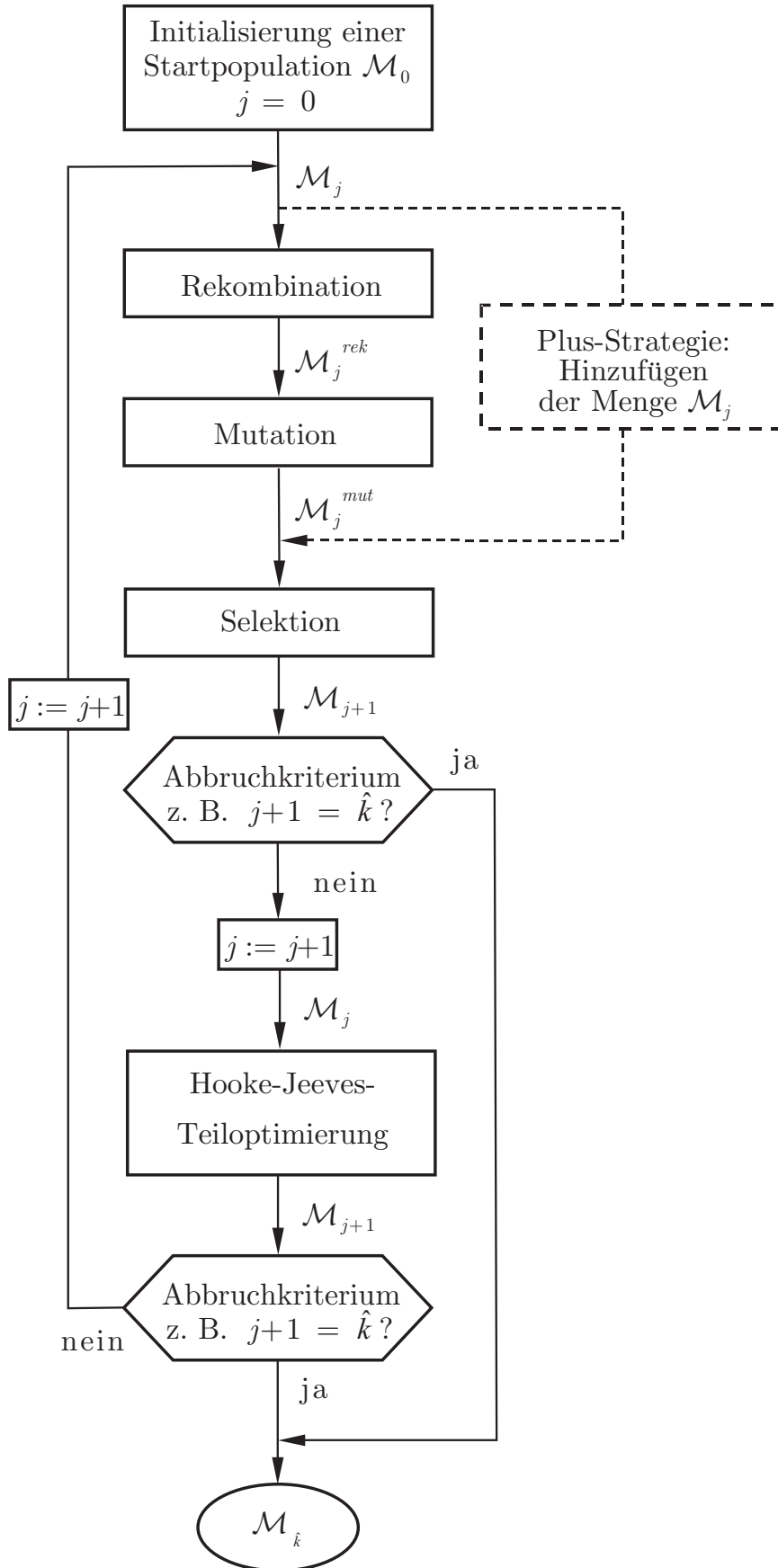


Abbildung 6.9: Kombination der Evolutionsstrategie mit dem modifizierten Hooke-Jeeves-Verfahren.

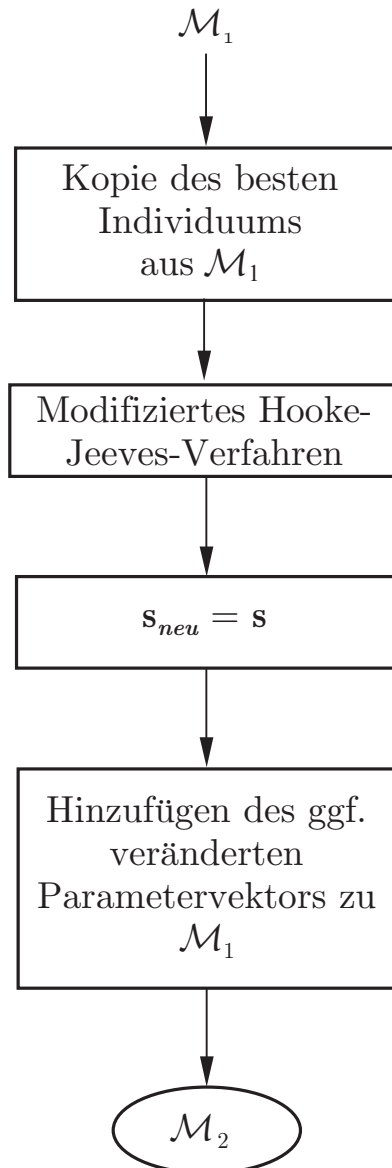


Abbildung 6.10: Hooke-Jeeves-Teiloptimierung mit dem besten Individuum aus \mathcal{M}_1 als Startpunkt.

parametervektor aus \mathcal{M}_{j-1} besteht (s. Tabelle 6.1). Da auch hier in jeder ES-Generationen die Plus-Strategie eingesetzt wird, kann eine Veränderung des besten Individuums im Parameterraum nur einer Verbesserung entsprechen.³

Wird eine Veränderung und damit eine Verbesserung festgestellt (Fall 1, s. Abbildung 6.11), so ist es vielversprechend, das modifizierte Hooke-Jeeves-Verfahren zu starten, um in dem Fall, dass das kopierte Individuum sich in der Nähe eines globalen Optimums befindet, dieses lokalisieren zu können. In

³Durch das modifizierte Hooke-Jeeves-Verfahren kann es ebenfalls nie zu einer Verschlechterung des ermittelten Parametervektors gegenüber dem Parametervektor, der als Startpunkt für das modifizierte Hooke-Jeeves-Verfahren fungierte, kommen.

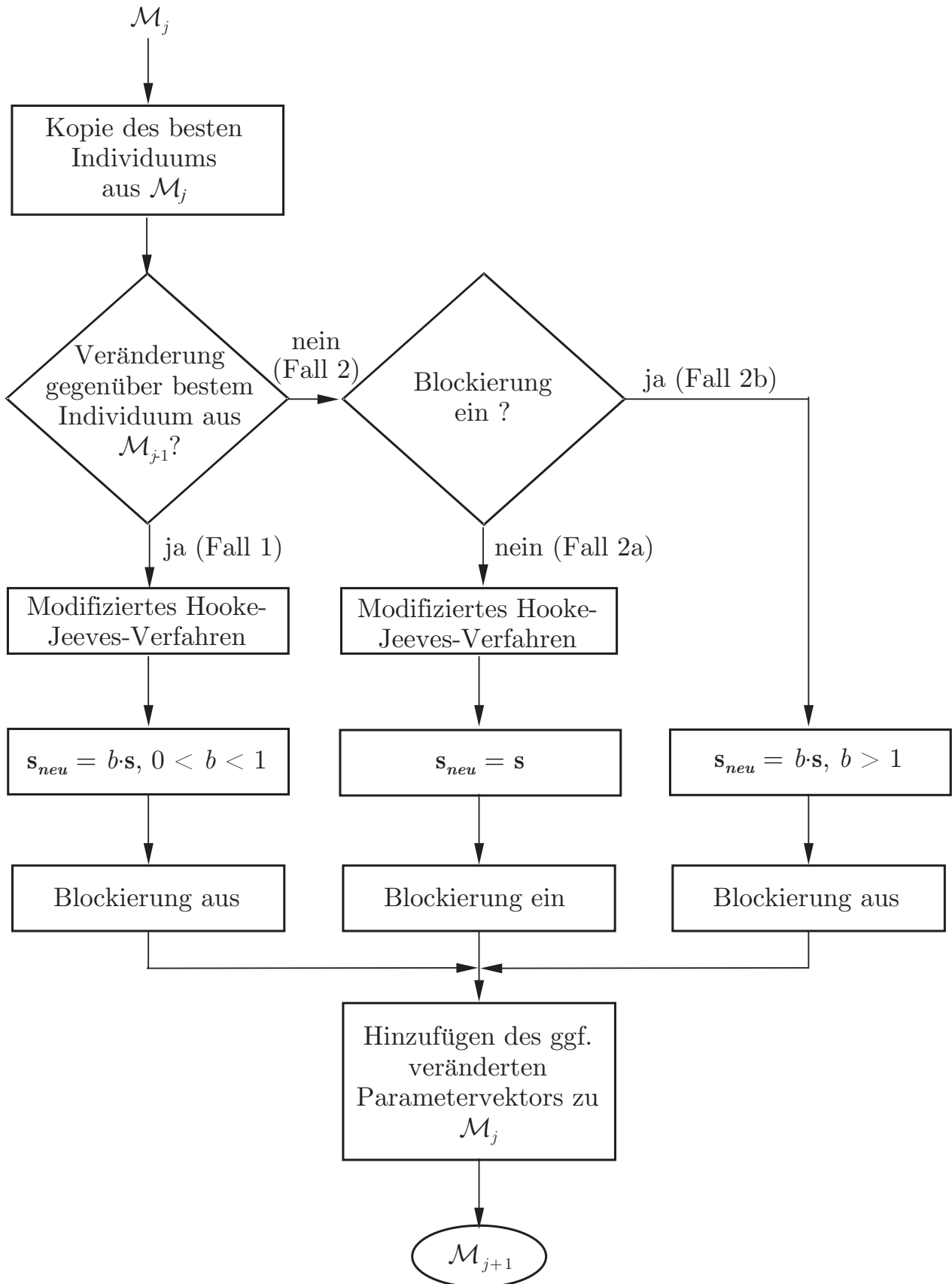


Abbildung 6.11: Hooke-Jeeves-Teiloptimierung mit dem besten Individuum aus \mathcal{M}_j (für $j > 2$) als Startpunkt.

diesem Fall wird dem durch das modifizierte Hooke-Jeeves-Verfahren ermittelten Parametervektor anschließend ein Schrittweitenvektor zugewiesen, der sich aus dem ursprünglichen Schrittweitenvektor des aus \mathcal{M}_j kopierten Individuums durch Multiplikation mit dem Faktor b ergibt, wobei durch $0 < b < 1$ eine Verkleinerung des Schrittweitenvektors erreicht wird. Hier wird ein Wert von $b = 0.5$ verwendet. Befindet sich der Parametervektor in der Nähe eines globalen Optimums, so ist ein verkleinerter Schrittweitenvektor für die anschließende ES-Generation sinnvoll. Die Blockierung bleibt ausgeschaltet.

Wird keine Veränderung festgestellt (Fall 2, s. Abbildung 6.11), so ist es möglich, dass bereits ein globales oder lokales Optimum ermittelt wurde. In diesem Fall hängt es von der Vorgeschichte des Optimierungsprozesses ab, ob das modifizierte Hooke-Jeeves-Verfahren gestartet wird.

Ist die Blockierung ausgeschaltet (Fall 2a), so wird das modifizierte Hooke-Jeeves-Verfahren gestartet. In diesem Fall ist eine Aussage über einen sinnvollen Schrittweitenvektor schwierig. Deshalb wird dem durch das modifizierte Hooke-Jeeves-Verfahren ermittelten Parametervektor der Schrittweitenvektor des aus \mathcal{M}_j kopierten Individuums unverändert zugewiesen. Die Blockierung wird eingeschaltet, um nach der nächsten ES-Generation den Aufruf des modifizierten Hooke-Jeeves-Verfahrens zu verhindern, wenn anhaltend keine Veränderung des besten Individuums besteht.

Wenn keine Veränderung des kopierten Individuums gegenüber dem besten Individuum aus \mathcal{M}_{j-1} festgestellt wird und die Blockierung eingeschaltet ist (Fall 2b), so wird das modifizierte Hooke-Jeeves-Verfahren *nicht* gestartet. Tritt dieser Fall ein, so wurde zuvor bereits zwei mal durch die ES-Generation keine Veränderung des besten Individuums hervorgerufen. Die Vermutung ist nahe liegend, dass bereits ein Optimum ermittelt wurde. Es ist jedoch nicht bekannt, ob es sich bei diesem Optimum um ein globales oder nur um ein lokales Optimum handelt. Für den Fall, dass es sich bei dem gefundenen Optimum um ein lokales handelt, wird nun dem aus \mathcal{M}_j kopierten Individuum für die nächste ES-Generation ein Schrittweitenvektor zugewiesen, der sich aus dem ursprünglichen Schrittweitenvektor des aus \mathcal{M}_j kopierten Individuums durch Multiplikation mit einem Faktor $b > 1$ ergibt. Hier wird ein Wert von $b = 10$ verwendet. Es wird also eine Vergrößerung des Schrittweitenvektors erreicht. Hierdurch soll es leichter möglich sein, das lokale Optimum in der nächsten ES-Generation wieder zu verlassen. Die Blockierung weiterer Aufrufe des modifizierten Hooke-Jeeves-Verfahrens wird aufgehoben.

In den Fällen 1 und 2a wird der durch das modifizierte Hooke-Jeeves-Verfahren ermittelte Parametervektor mit dem ihm zugewiesenen Schrittweitenvektor zur

Menge \mathcal{M}_j hinzugefügt und so die Menge \mathcal{M}_{j+1} gebildet. Im Fall 2b wird das aus \mathcal{M}_j kopierte Individuum mit dem ihm zugewiesenen Schrittweitenvektor ebenfalls zur Menge \mathcal{M}_j hinzugefügt und dadurch die Menge \mathcal{M}_{j+1} gebildet.

Innerhalb des modifizierten Hooke-Jeeves-Verfahrens werden folgende Abbruchkriterien verwendet (s. Tabelle 6.1):

- Unterschreiten des Schwellwertes für das Gütemaß ρ
- Unterschreiten einer minimalen Schrittweite s_{min} in allen Dimensionen
- Erreichen einer maximal zulässigen Anzahl an Bewertungen mittels des Gütemaßes entweder für die Hooke-Jeeves-Teiloptimierung oder für den gesamten Optimierungsprozess

Bei Eintreten eines dieser Abbruchkriterien wird das modifizierte Hooke-Jeeves-Verfahren beendet. Hierbei wird eine Exploration nicht komplett durchlaufen, wenn die maximal zulässige Anzahl an Bewertungen mittels des Gütemaßes sonst überschritten würde. Wird der Schwellwert für das Gütemaß unterschritten, so wird der gesamte Optimierungsprozess beendet.

Im Anschluss an die Hooke-Jeeves-Teiloptimierung wird wieder die nächste ES-Generation, bestehend aus Rekombination, Mutation und Selektion, durchlaufen (s. Abbildung 6.9). Besteht über mehrere Durchläufe der ES-Generation und der Hooke-Jeeves-Teiloptimierung keine Veränderung des jeweils aus \mathcal{M}_j kopierten Individuums gegenüber dem besten Individuum aus \mathcal{M}_{j-1} , so wird in der Hooke-Jeeves-Teiloptimierung bei jedem zweiten Durchlauf der Aufruf des modifizierten Hooke-Jeeves-Verfahrens verhindert und bei den anderen Durchläufen das modifizierte Hooke-Jeeves-Verfahren aufgerufen. Wird der Aufruf verhindert, so ist es durch die dann vorgenommene Schrittweitenvergrößerung leichter möglich, in einer der nächsten ES-Generationen ein lokales Optimum zu verlassen. Der Aufruf des modifizierten Hooke-Jeeves-Verfahrens bei jedem zweiten Durchlauf der Hooke-Jeeves-Teiloptimierung führt dazu, dass ein möglicherweise aufgefundenen globales Optimum exakt lokalisiert werden kann.

Durch die beschriebene Kombination der Evolutionsstrategie mit dem modifizierten Hooke-Jeeves-Verfahren erfolgt eine Anpassung des Optimierungsprozesses an die Beschaffenheit der Gütelandschaft. Je nach vorliegender Gütelandschaft stellt sich eine stärker ausgeprägte Suche entlang eines bestimmten Pfades im Suchraum durch das modifizierte Hooke-Jeeves-Verfahren oder eine stärker ausgeprägte Suche ohne bevorzugte Richtung durch die Evolutionsstrategie ein [Lam02, KNL⁺03, Nic04].

Tabelle 6.1: Einstellungen bei der Optimierungsstrategie ES_{hybrid} .

	$n = 2$	$n = 4$	$n = 24$ (Reg- leroptimierung Hydraulikventil, s. Kapitel 7)
Definition: <i>Veränderung</i> eines Individuums gegenüber einem anderen Individuum besteht	Differenz der Individuen beträgt $>1\%$ der Breite $(\mathbf{x}_{max} - \mathbf{x}_{min})$ in mindestens einer Dimension	Differenz der Individuen beträgt $>1\%$ der Breite $(\mathbf{x}_{max} - \mathbf{x}_{min})$ in mindestens einer Dimension	Gütedifferenz der Individuen beträgt $>2.5\%$ der Differenz $Q_{max} - Q_{min}$
(Anzahl Gütebewertungen innerhalb eines Aufrufs des modifizierten Hooke-Jeeves-Verfahrens) \div (Anzahl Gütebewertungen innerhalb einer ES-Generation)	0.25	0.56	1
ϵ (Schwellwert für das modifizierte Hooke-Jeeves-Verfahren)	10^{-6}	10^{-6}	10^{-6}
s_{min} für das modifizierte Hooke-Jeeves-Verfahren	$10^{-8} \cdot (\mathbf{x}_{max} - \mathbf{x}_{min})$	$10^{-8} \cdot (\mathbf{x}_{max} - \mathbf{x}_{min})$	$10^{-8} \cdot (\mathbf{x}_{max} - \mathbf{x}_{min})$
ρ (Schwellwert für den Gütewert als Abbruchkriterium, Exploration wird noch komplett durchlaufen)	10^{-5}	10^{-5}	–

6.3.1 Ergebnisse mit ES_{hybrid} bei Verwendung einer analytischen Testfunktion als Gütemaß

Dieser Abschnitt beinhaltet einen Vergleich der Optimierungsstrategie ES_{hybrid} mit den Strategien ES_{std} , $ES_{WBT}^{T_{var}}$, $ES_{WBT}^{T_{const}}$, ES_{GBT}^{symm} sowie ES_{GBT}^{unsymm} .

Die ES-Generationen innerhalb der Optimierungsstrategie ES_{hybrid} basieren auf der Strategie ES_{GBT}^{unsymm} . Es kann aber auch jede andere der vorgestellten Varianten der Evolutionsstrategie mit dem modifizierten Hooke-Jeeves-Verfahren kombiniert werden.

Auch hier wurde jede Optimierungsstrategie 10 mal mit der verwendeten Testfunktion durchlaufen.

Als Testfunktion wird die Schwefel-Funktion nach Gleichung (5.5) für $n = 2$ verwendet. In jeder Generation werden, wie in den vorherigen Abschnitten 6.1.4

und 6.2.4, $\mu = 15$ Eltern-Individuen und $\lambda = 100$ Nachkommen-Individuen verwendet. Alle Eltern-Individuen der Startpopulation werden hier zur besseren Vergleichbarkeit der unterschiedlichen Optimierungsprozesse auf dem Punkt $\mathbf{x} = (-500, -500)$ platziert.

In Abbildung 6.12 sind die Ergebnisse typischer Optimierungsprozesse mit den Optimierungsstrategien ES_{std} und ES_{hybrid} dargestellt.

Hier wurde ebenfalls die Optimierungsstrategie ES_{std} zu Vergleichszwecken eingesetzt (Abbildung 6.12 (a)). Links sind wiederum die μ Eltern-Individuen aller Generationen innerhalb des Suchraums zu sehen. Rechts sind auch hier die Gütwerte aller Eltern-Individuen über der Anzahl M der durchgeführten Gütebewertungen im Laufe des Optimierungsprozesses dargestellt. Bei diesem Optimierungsprozess ist auffällig, dass das globale Optimum, das sich im Punkt $\mathbf{x} = (420.97, 420.97)$ befindet, nicht gefunden wurde. Dies ist typisch für die eingesetzte Testfunktion mit dem angegebenen Startpunkt. Bei sämtlichen getesteten Optimierungsstrategien außer der Strategie ES_{hybrid} konnte das globale Optimum nicht bei allen der durchgeführten 10 Optimierungsprozesse ermittelt werden. Dies ist in der Beschaffenheit der Gütelandschaft und in dem absichtlich ungünstig gewählten Startpunkt begründet. Nur bei Einsatz der Optimierungsstrategie ES_{hybrid} konnte das globale Optimum dieser Testfunktion zuverlässig in allen 10 Optimierungsprozessen ermittelt werden.

Bei Verwendung der Strategie ES_{std} wurde das globale Optimum drei mal mit der geforderten Genauigkeit gefunden. Die restlichen 7 der 10 Optimierungsprozesse wurden nach 50000 Bewertungen mittels des Gütemaßes abgebrochen, da nach dieser hohen Anzahl an Gütebewertungen keine wesentliche weitere Verbesserung durch den Optimierungsprozess mehr zu erwarten ist. Durchschnittlich wurden somit 40660 Bewertungen mittels des Gütemaßes durchgeführt (s. Abbildung 6.13). Bei Einsatz der Strategien ES_{WBT}^{Tvar} konnte das globale Optimum ebenfalls nur drei mal in 10 Optimierungsprozessen ermittelt werden. Mit dieser Optimierungsstrategie wurden durchschnittlich 36086 Bewertungen mittels des Gütemaßes durchgeführt. Mit der Strategie ES_{WBT}^{Tconst} gelang es in 5 von 10 Optimierungsprozessen, das globale Optimum ausreichend genau zu finden. Es wurden durchschnittlich 32461 Gütebewertungen benötigt. Bei Verwendung der Strategie ES_{GBT}^{symm} wurde das globale Optimum in 4 der 10 durchgeführten Optimierungsprozesse ermittelt. Bei dieser Strategie wurde eine durchschnittliche Anzahl von 32000 Gütebewertungen ermittelt. Beim Einsatz der Optimierungsstrategie ES_{GBT}^{unsymm} konnte das globale Optimum in drei von 10 Optimierungsprozessen ermittelt werden. Hier wurden durchschnittlich 35600 Bewertungen mittels des Gütemaßes durchgeführt. Allein bei der Optimierungsstrategie ES_{hybrid} konnte das globale Optimum durch alle 10 Optimierungsprozesse

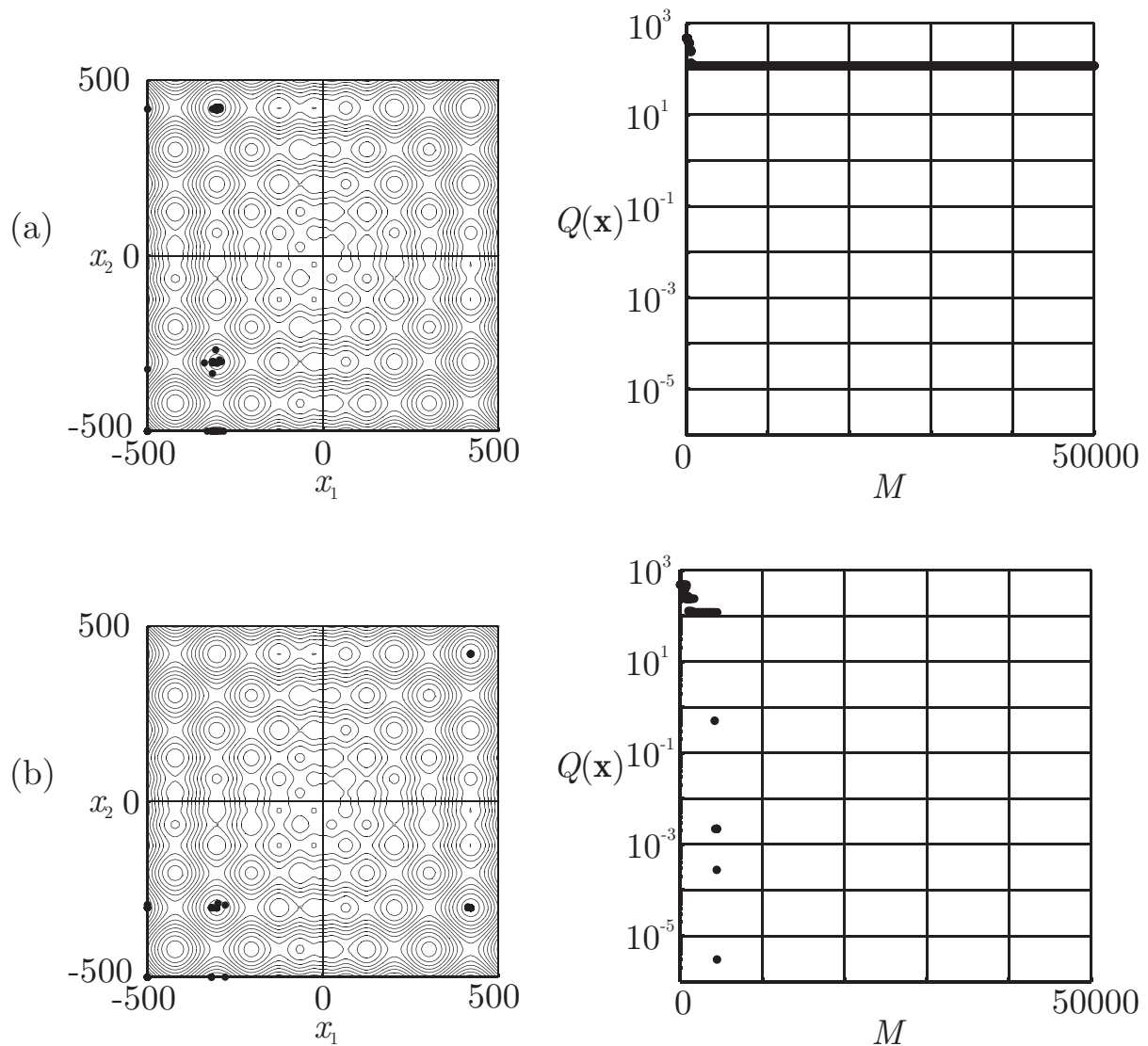


Abbildung 6.12: links: Eltern-Individuen aller Generationen (a) bzw. Eltern-Individuen aller ES-Generationen und Endpunkte aller Hooke-Jeeves-Teiloptimierungen im Suchraum (b), rechts: Gütewerte aller Eltern-Individuen (a) bzw. Gütewerte aller Eltern-Individuen und der Endpunkte aller Hooke-Jeeves-Teiloptimierungen (b) über der Anzahl der Bewertungen durch das Gütemaß Schwefel-Funktion, Strategie ES_{std} (a), Strategie ES_{hybrid} (b).

se gefunden werden. Durchschnittlich wurden hierfür 7767 Gütebewertungen benötigt (s. Abbildung 6.13). Diese Anzahl durchschnittlich benötigter Gütebewertungen ist deutlich geringer als die ermittelten Anzahlen bei Verwendung der restlichen bereits vorgestellten Optimierungsstrategien [Lam02].

Die in Abbildung 6.12 dargestellten Optimierungsprozesse stellen somit typische Optimierungsprozesse für die Strategien ES_{std} und ES_{hybrid} dar. Für die restlichen Optimierungsstrategien wurde auf eine grafische Darstellung der Individuen im Suchraum bzw. der Gütewerte im Laufe des Optimierungsprozesses verzichtet.

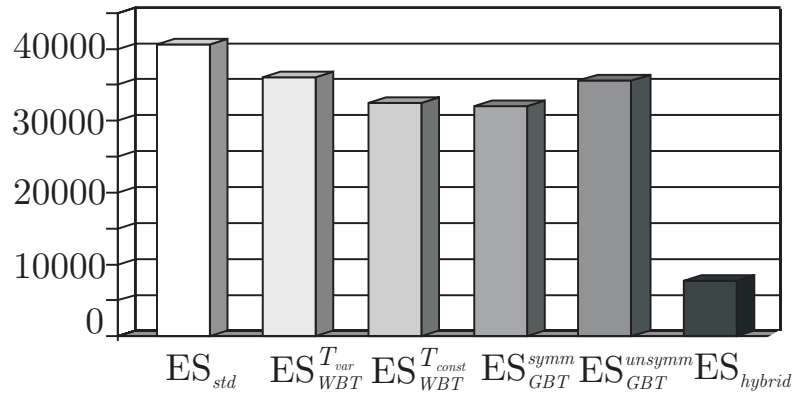


Abbildung 6.13: Durchschnittlich benötigte Anzahl an Bewertungen mittels des Gütemaßes Schwefel-Funktion bei jeweils 10 Optimierungsprozessen.

6.4 Modifiziertes Hooke-Jeeves-Verfahren mit Neustart ($HoJe_{Neustart}$)

Dieses Optimierungsverfahren wurde implementiert, um die Evolutionsstrategie mit den beschriebenen neuen Strategieelementen mit einem herkömmlichen deterministischen Optimierungsverfahren zu vergleichen.

Ein Nachteil des herkömmlichen und auch des modifizierten Hooke-Jeeves-Verfahrens ist, dass bei einem ungünstigen Startpunkt im Suchraum der Optimierungsprozess möglicherweise nur zum Auffinden eines lokalen Optimums führt. Diese Gefahr besteht ebenso bei Evolutionsstrategien. Sie ist beim Einsatz des Hooke-Jeeves-Verfahrens bei Optimierungsproblemen mit zahlreichen lokalen Optima aber deutlich höher. Auch die Wahl einer ungünstigen Start-Schrittweite – bzw. beim modifizierten Hooke-Jeeves-Verfahren die Festlegung eines ungünstigen Start-Schrittweitenvektors – kann zu diesem suboptimalen Ergebnis führen.

Als Abhilfe wurde das modifizierte Hooke-Jeeves-Verfahren mit Neustart – im Folgenden mit $HoJe_{Neustart}$ abgekürzt – implementiert, wobei die Startpunkte der einzelnen Teiloptimierungen und die Schrittweitenvektoren, mit denen die Teiloptimierungen starten, zufallsabhängig sind. Die Startpunkte werden hierbei gleichverteilt im Suchraum generiert. Die einzelnen Komponenten jedes Start-Schrittweitenvektors werden unabhängig voneinander erzeugt. Die Schrittweitenkomponenten werden ebenfalls gleichverteilt generiert. Die Schwankungsbreite liegt hier zwischen 5% und 20% der Breite des Suchraums in jeder Dimension [Liu03].

Der Optimierungsprozess läuft folgendermaßen ab: Nach der Generierung des ersten Startpunktes wird die erste Teiloptimierung mit dem modifizierten

Hooke-Jeeves-Verfahren vollzogen, bis ein Abbruchkriterium für diese Teiloptimierung erfüllt ist.

Hierbei werden in Anlehnung an Abschnitt 6.3 folgende Abbruchkriterien verwendet (s. Tabelle 6.2):

- Unterschreiten des Schwellwertes für das Gütemaß ρ
- Erreichen einer maximal zulässigen Anzahl an Bewertungen mittels des Gütemaßes entweder für die Teiloptimierung oder für den gesamten Optimierungsprozess
- Unterschreiten einer minimalen Schrittweite \mathbf{s}_{min} in allen Dimensionen
- optional: Erreichen einer maximal zulässigen Anzahl an Verkleinerungen des Schrittweitenvektors durch Multiplikation mit α (s. Abschnitt 4.2), ohne zwischen den Verkleinerungszyklen eine Extrapolation vorzunehmen, und anschließender Explorationen

Tritt eines dieser Abbruchkriterien ein, so wird die Teiloptimierung mit dem modifizierten Hooke-Jeeves-Verfahren beendet. Hierbei wird eine Exploration nicht komplett durchlaufen, wenn die maximal zulässige Anzahl an Gütebewertungen sonst überschritten würde. Wird der Schwellwert für das Gütemaß unterschritten oder die maximal zulässige Anzahl an Gütebewertungen für den gesamten Optimierungsprozess erreicht, so wird der Optimierungsprozess beendet. Wird die maximal zulässige Anzahl an Bewertungen mittels des Gütemaßes für die Teiloptimierung erreicht, die minimale Schrittweite in allen Dimensionen unterschritten oder gegebenenfalls die maximal zulässige Anzahl an Verkleinerungen des Schrittweitenvektors ohne Extrapolation erreicht, so wird ein neuer Startpunkt und ein neuer Start-Schrittweitenvektor erzeugt und eine neue Teiloptimierung gestartet.

Das Optimierungsverfahren $HoJe_{Neustart}$ weist bei Vorliegen eines Optimierungsproblems mit nur einem globalen Optimum und keinen oder wenigen lokalen Optima Vorteile gegenüber einer herkömmlichen Evolutionsstrategie auf. Analog zum Einsatz des Hooke-Jeeves-Verfahrens, wo eine gerichtete Suche nach einem optimalen Gütewert vorgenommen wird, werden durch die Strategie $HoJe_{Neustart}$ mehrere gerichtete Suchprozesse durchgeführt, wohingegen der Optimierungsprozess bei Einsatz einer Evolutionsstrategie stärker zufallsabhängig ist und nicht wie das Hooke-Jeeves-Verfahren einer festgelegten Suchrichtung folgt.

Tabelle 6.2: Einstellungen bei der Optimierungsstrategie $\text{HoJe}_{\text{Neustart}}$.

	$n = 2, n = 4$	$n = 24$ (Regleroptimierung Hydraulikventil, s. Kapitel 7)
ρ (Schwellwert für den Gütewert als Abbruchkriterium, Exploration wird noch komplett durchlaufen)	10^{-5}	–
Anzahl Gütebewertungen innerhalb eines Aufrufs des modifizierten Hooke-Jeeves-Verfahrens (Teiloptimierung)	$75 \cdot n$	$75 \cdot n$
s_{\min} für das modifizierte Hooke-Jeeves-Verfahren	$10^{-12} \cdot (\mathbf{x}_{\max} - \mathbf{x}_{\min})$	$5 \cdot 10^{-3} \cdot (\mathbf{x}_{\max} - \mathbf{x}_{\min})$
Anzahl Verkleinerungen des Schrittweitenvektors und anschließender Explorationen	–	6
ϵ (Schwellwert für das modifizierte Hooke-Jeeves-Verfahren, s. Abschnitt 4.2)	10^{-8}	$5 \cdot 10^{-3}$

Nachteile des Verfahrens $\text{HoJe}_{\text{Neustart}}$ liegen in einer stärkeren Empfindlichkeit gegenüber Störungen und Rauschen in der Gütebewertung oder in Schwierigkeiten bei einer stark zerklüfteten Gütelandschaft bei realen Optimierungsproblemen. Auch bei sehr vielen lokalen Optima reicht die Anzahl der Neustarts gegebenenfalls nicht aus, um in die Umgebung des globalen Optimums zu gelangen. Hier hat eine Evolutionsstrategie, vor allem mit Strategieelementen, die zu einer stark ausgeprägten Breitensuche führen, Vorzüge.

Das Ziel beim Einsatz der beschriebenen Strategie $\text{HoJe}_{\text{Neustart}}$ besteht darin, die Nachteile eines möglicherweise ungünstigen Startpunktes oder einer ungünstigen Startschrittweite bei dem modifizierten Hooke-Jeeves-Verfahren zu beheben und somit einen fairen Vergleich zwischen den auf einer Evolutionsstrategie basierenden Optimierungsverfahren ES_{std} , $\text{ES}_{\text{WBT}}^{\text{Tvar}}$, $\text{ES}_{\text{WBT}}^{\text{Tconst}}$, $\text{ES}_{\text{GBT}}^{\text{symm}}$ und $\text{ES}_{\text{GBT}}^{\text{unsymm}}$, dem hybriden Optimierungsverfahren $\text{ES}_{\text{hybrid}}$ und der allein auf dem Hooke-Jeeves-Verfahren basierenden Strategie $\text{HoJe}_{\text{Neustart}}$ zu ermöglichen.

Ergebnisse bei der Verwendung analytischer Testfunktionen als Gütemaße bei Einsatz der Strategie $\text{HoJe}_{\text{Neustart}}$ sind im folgenden Abschnitt 6.5 zu finden.

6.5 Ergebnisse bei Verwendung höherdimensionaler Testfunktionen

In diesem Abschnitt werden wesentliche Ergebnisse bei Verwendung der vorgestellten neuen Strategieelemente zur Optimierung höherdimensionaler Testfunktionen präsentiert [Liu03].

Auch hier wurde jede der 7 vorgestellten Optimierungsstrategien (ES_{std} , $ES_{WBT}^{T_{var}}$, $ES_{WBT}^{T_{const}}$, ES_{GBT}^{symm} , ES_{GBT}^{unsymm} , ES_{hybrid} sowie $HoJe_{Neustart}$) 10 mal zur Optimierung mit jeder in diesem Abschnitt verwendeten Testfunktion als Gütemaß eingesetzt.

Die Optimierungsstrategie ES_{hybrid} wird auch hier durch eine Kombination des modifizierten Hooke-Jeeves-Verfahrens mit der Strategie ES_{GBT}^{unsymm} realisiert.

In jeder Generation werden hier aufgrund der höheren Dimension der Optimierungsprobleme $\mu = 40$ Eltern-Individuen und $\lambda = 270$ Nachkommen-Individuen verwendet. Alle Eltern-Individuen der Startpopulationen bzw. alle Startpunkte für die Teiloptimierungen bei der Strategie $HoJe_{Neustart}$ werden zufällig gleichverteilt im Suchraum generiert.

Tabelle 6.3 zeigt die am besten geeigneten Optimierungsstrategien zur Optimierung bei $n = 4$ mit den angegebenen Testfunktionen. Die Beurteilung der untersuchten Optimierungsstrategien erfolgt anhand von drei Kriterien. Zum einen wird die durchschnittliche Anzahl an Bewertungen mittels des Gütemaßes \bar{M} , die erfolgt, bis eines der Abbruchkriterien erfüllt ist, herangezogen. Hierbei ist eine möglichst geringe Anzahl, d. h. möglichst geringer zeitlicher Aufwand, gewünscht. Außerdem wird berücksichtigt, wie häufig das globale Optimum der Testfunktion bei den jeweils 10 durchgeführten Optimierungsprozessen mit der in Kapitel 5 angegebenen Genauigkeit ρ gefunden wurde. Schließlich werden die Durchschnittswerte der jeweils erreichten besten Gütewerte der jeweiligen Optimierungsprozesse verglichen. Hierbei ist wieder ein minimaler Wert erwünscht.

Für die Optimierung mit allen hier untersuchten Testfunktionen als Gütemaß liefert die Strategie ES_{hybrid} die besten Ergebnisse bezogen auf alle oben angegebenen Beurteilungskriterien.

Für die Optimierung mit der Rastrigin-Funktion als Gütemaß liefert auch die Strategie $ES_{WBT}^{T_{var}}$ gute Ergebnisse. Dies ist auf die Beschaffenheit der durch die Testfunktion nach Gleichung (5.2) aufgespannten Gütelandschaft mit

Tabelle 6.3: Beste Optimierungsstrategien zur Optimierung mit den angegebenen Testfunktionen als Gütemaß bei $n = 4$.

Testfunktion $\hat{=}$ Gütemaß	M minimal	Optimum am häufigsten gefunden (mit Genauigkeit ρ)	Durchschnittswert der besten Güterwerte minimal
Rastrigin-Funktion	1.: ES_{hybrid} 2.: ES_{WBT}^{Tvar} 3.: ES_{GBT}^{symm}	1.: ES_{hybrid} 2.: ES_{WBT}^{Tvar} 3.: ES_{GBT}^{symm}	1.: ES_{hybrid} 2.: ES_{std} 3.: ES_{WBT}^{Tconst}
Schwefel-Funktion	1.: ES_{hybrid} 2.: $HoJe_{Neustart}$ 3.: ES_{GBT}^{symm}	1.: ES_{hybrid} 2.: $HoJe_{Neustart}$ 3.: –	1.: ES_{hybrid} 2.: $HoJe_{Neustart}$ 3.: ES_{WBT}^{Tconst}
Rosenbrock-Funktion	–	–	1.: ES_{hybrid} 2.: $HoJe_{Neustart}$ 3.: ES_{WBT}^{Tvar}

sehr vielen lokalen Optima zurückzuführen. Eine breite Durchsuchung des zur Verfügung stehenden Suchraums zahlt sich hier aus.

Wird die Schwefel-Funktion nach Gleichung (5.4) als Gütemaß verwendet, so schneidet auch die Strategie $HoJe_{Neustart}$ gut ab. Die Funktion weist zwar auch mehrere lokale Optima auf, aber die Anzahl der lokalen Optima ist nicht so groß wie bei der Rastrigin-Funktion.

Bei der Optimierung mit der Rosenbrock-Funktion nach Gleichung (5.7) als Gütemaß konnte mit den vorgenommenen Einstellungen bei keiner der eingesetzten Strategien das globale Optimum in der festgelegten Anzahl an Bewertungen mittels des Gütemaßes mit der geforderten Genauigkeit gefunden werden. Bezogen auf den Mittelwert der erreichten besten Güterwerte erweist sich aber auch hier die Strategie ES_{hybrid} als am erfolgreichsten. Die Schwierigkeiten bei der Optimierung mit dieser Testfunktion als Gütemaß sind durch die Beschaffenheit der Gütelandschaft mit einem langgezogenen, gekrümmten, diagonal zu den Koordinatenrichtungen gerichteten Tal erklärbar. Diese Testfunktion weist keine lokalen Optima auf.

Insgesamt bietet die Kombination einer Evolutionsstrategie mit dem modifizierten Hooke-Jeeves-Verfahren bei der Lösung aller hier untersuchten Optimierungsprobleme mit $n = 4$ deutliche Vorteile gegenüber dem alleinigen Einsatz einer der beiden Vorgehensweisen.

Kapitel 7

Regleroptimierung für ein industrielles Hydraulikventil

Der Einsatzbereich industrieller Hydraulikventile erstreckt sich über weite Bereiche des Maschinen- und Anlagenbaus. Hydraulikventile werden beispielsweise in Produktionsstraßen der Holz- und Papierindustrie oder in Kunststoffspritzguss-Maschinen eingesetzt.

Im Rahmen dieser Arbeit wird die *Regelung* eines industriellen Hydraulikventils untersucht und optimiert (s. Abbildung 7.2). Bei fest implementierter Reglerstruktur werden darin enthaltene noch freie Parameter durch einen Optimierungsprozess so eingestellt, dass das Regelungsverhalten möglichst günstig ist.

Für den Optimierungsprozess ist es notwendig, ein geeignetes Gütemaß zur Beurteilung des Regelungsverhaltens zu verwenden. Es wurde ein Gütemaß entwickelt, das ausschließlich im *Zeitbereich* arbeitet. Dieses Gütemaß Q_{Zeit} bewertet die Reaktion eines Systems auf ein vorgegebenes Eingangssignal in Abhängigkeit von der Zeit (Abschnitt 7.4). Es wird als skalares Gütemaß für den Optimierungsprozess eingesetzt.

7.1 Hardware

Bei dem untersuchten Hydraulikventil handelt es sich um ein Regelventil NG6 der Bosch Rexroth AG, ehemals Robert Bosch GmbH. Abbildung 7.1 zeigt den Querschnitt des Ventils. Das dargestellte Ventil gehört zur Gruppe der Schieberventile. Das Ventil enthält einen Kolben (Ventilschieber), der in Längsrichtung verschoben wird. Der Ventilschieber befindet sich in einer Längsbohrung in dem Gehäuse des Ventils. In die Längsbohrung sind Ringkanäle eingearbeitet, die jeweils mit einem äußeren Anschluss am Gehäuse verbunden sind (Anschlüsse A, B, P und T, s. Abbildung 7.1). Durch die Verschiebung des Ventilschiebers

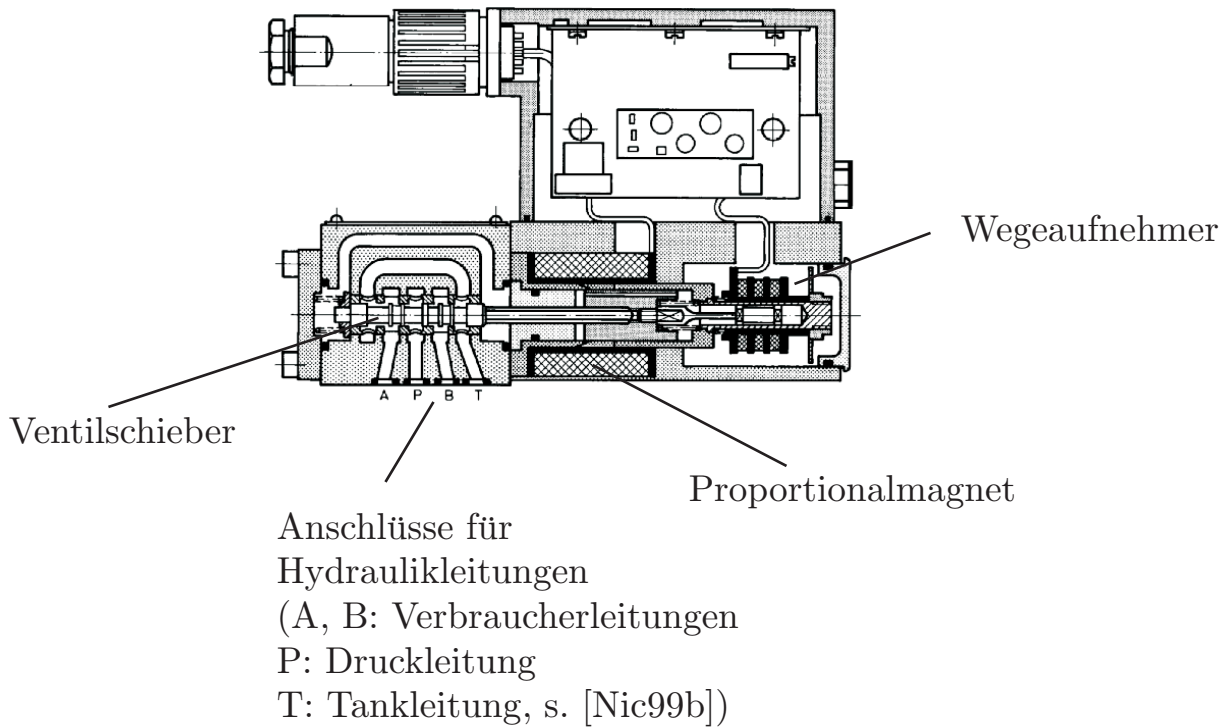


Abbildung 7.1: Querschnittsdarstellung des eingesetzten Hydraulikventils [BLSK00].

werden Durchflusswege für eine Hydraulikflüssigkeit geöffnet oder geschlossen, indem durch den Ventilschieber die einzelnen Ringkanäle verbunden oder getrennt werden. Die Position des in Abbildung 7.1 markierten Ventilschiebers soll mit Hilfe der eingesetzten *Lageregelung* geregelt werden (s. Abbildung 7.2). Die Position $x(t)$ des Ventilschiebers wird im Folgenden auch als *Ventilhub* bezeichnet. Detailliertere Informationen zur Wirkungsweise und zum Einsatz von Hydraulikventilen sind in [Rob99, Sch78, EHK⁺88] zu finden.

Die Struktur des gesamten Regelungssystems ist in Abbildung 7.2 dargestellt. Es wird zusätzlich zur Lageregelung ein unterlagerter Regelkreis zur Stromregelung eingesetzt (s. Abbildung 7.3). Dieser unterlagerte Regelkreis sorgt dafür, dass die vom Lageregelkreis angeforderte Stellgröße möglichst schnell zur Verfügung gestellt wird. Der Stromregelkreis ist jedoch nicht Gegenstand dieser Arbeit.

Die Ansteuerung des Ventils wird mit Hilfe eines handelsüblichen PC (Pentium IV, 1.4 GHz) und einer Echtzeit-Simulationskarte (DS1103, dSPACE) vorgenommen. Durch die eingesetzte Echtzeit-Simulationskarte wird es ermöglicht, die benötigte Abtastzeit von $t_a = 100\mu\text{s}$ für die Lageregelung einzuhalten.

Oftmals wird die Lageregelung eines Hydraulikventils im praktischen Einsatz

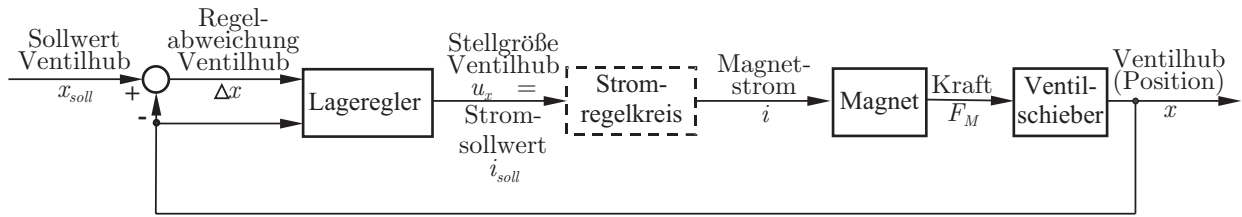


Abbildung 7.2: Regelungssystem bestehend aus Lageregelung und unterlagerter Stromregelung (s. Abbildung 7.3).

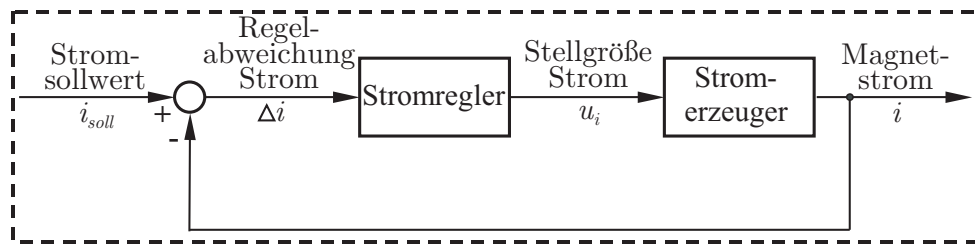


Abbildung 7.3: Unterlagerter Regelkreis zur Stromregelung.

durch einen analogen Regler vorgenommen. In den letzten Jahren hat jedoch das Interesse an digital realisierten Reglern zugenommen. Der Vorteil der Analogtechnik liegt in der größeren Schnelligkeit der Regelung, da der Schnelligkeit eines digitalen Reglers durch die realisierbare Abtastzeit Grenzen gesetzt sind. In den nächsten Jahren ist jedoch aufgrund besserer Rechenleistungen mit einem deutlichen ‘Vorsprung’ der digitalen Regelung zu rechnen. Die großen Vorteile eines digital realisierten Reglers liegen zum einen in der automatisierten Einstellbarkeit günstiger Reglerparameter¹, zum anderen ist in Kombination mit der digital realisierten Regelung die CAN-Bus-Technologie² zur zentralen Ansteuerung der Ventile einsetzbar [Mül00].

Bei Verwendung eines digitalen Lagereglers sind prinzipiell zwei Varianten für die Kommunikation zwischen dem PC und dem Ventil möglich.

In der ersten Variante ist der Lageregler als ‘On-Board-Elektronik’ (OBE-D2)³ direkt auf dem Ventil angebracht. Dieser Aufbau ist für den praktischen Einsatz der digitalen Regelung vorgesehen. Bei einer fest in der OBE-D2 implementierten Reglerstruktur können über einen CAN-Bus die Parameter des Lagereglers

¹Bei analog geregelten Ventilen musste bisher eine günstige Einstellung der Reglerparameter durch interaktive Optimierung ‘von Hand’ durch einen Prozessexperten mit Hilfe von Widerstands-Dekaden vorgenommen werden. Da die einzelnen Reglerparameter nicht unabhängig voneinander wirken, ist dieser Vorgang sehr zeitaufwendig (Dauer: mehrere Tage) und erfordert viel Erfahrung und ‘Fingerspitzengefühl’.

²CAN-Bus: Controller-Area-Network-Bus (engl.): Bus in einem regelungstechnischen Netzwerk

³OBE-D2: On-Board-Elektronik, digital, 2. Generation

verändert werden. Außerdem können zu Analyse Zwecken verschiedene Signale aus der OBE-D2 über den CAN-Bus ausgelesen werden [Bos01b, Bos01a].

In der zweiten Variante ist der digitale Lageregler als Software-Regler auf dem PC implementiert. Die Reglerstruktur wird auf die Echtzeit-Simulationskarte geschrieben. Die Parameter des Lagereglers können über den PC verändert werden, ohne dass die Reglerstruktur nochmals auf die Karte übertragen werden muss. Vorteilhaft ist hierbei, dass nicht nur verschiedene Reglerparameter, sondern im Gegensatz zur ersten vorgestellten Variante auch unterschiedliche Reglerstrukturen getestet und analysiert werden können. Aus diesem Grund wird im Rahmen dieser Arbeit die zweite Variante herangezogen.

7.2 Software

Die Ansteuerung der Echtzeit-Simulationskarte erfolgt mit speziellen Software-Komponenten, die von dSPACE bereitgestellt werden und die in das Programm-Paket MATLAB eingebunden werden können [dSP01a, dSP01b, dSP01c, Mat99b].

Der digitale Lageregler ist unter MATLAB / SIMULINK implementiert [Mat99c, Mat99a].

Für das Gütemaß (s. Abschnitt 7.4) wurden zwei Fuzzy-Module entwickelt. Hierzu wurde das Software-Paket DORA FÜR WINDOWS verwendet [KK97]. Die Eingangsgrößen der Fuzzy-Module werden mit Hilfe von MATLAB ermittelt.

Ebenso erfolgt der Optimierungsprozess zur Einstellung günstiger Reglerparameter unter MATLAB.

7.3 Reglerstruktur: Kombination PI-Regler und Zustandsregler

Bei der eingesetzten Reglerstruktur handelt es sich um eine Kombination aus einem erweiterten PI-Regler und einem Zustandsregler (s. Abbildung 7.4).

Im P- und im I-Zweig des Reglers wird jeweils durch Kennlinien dafür gesorgt, dass Regelabweichungen $\Delta x(t) = x_{soll}(t) - x(t)$ im Klein-, im Mittel- und im Großsignalbereich⁴ unterschiedlich verarbeitet werden. Die Verstärkungsfaktoren beider Zweige $p_p(\Delta x)$ und $p_i(\Delta x)$ sind abhängig von der Größe der Regelabweichung.

⁴Die Bezeichnungen Klein-, Mittel- und Großsignal beziehen sich auf die Amplitudenänderung des jeweiligen Signals.

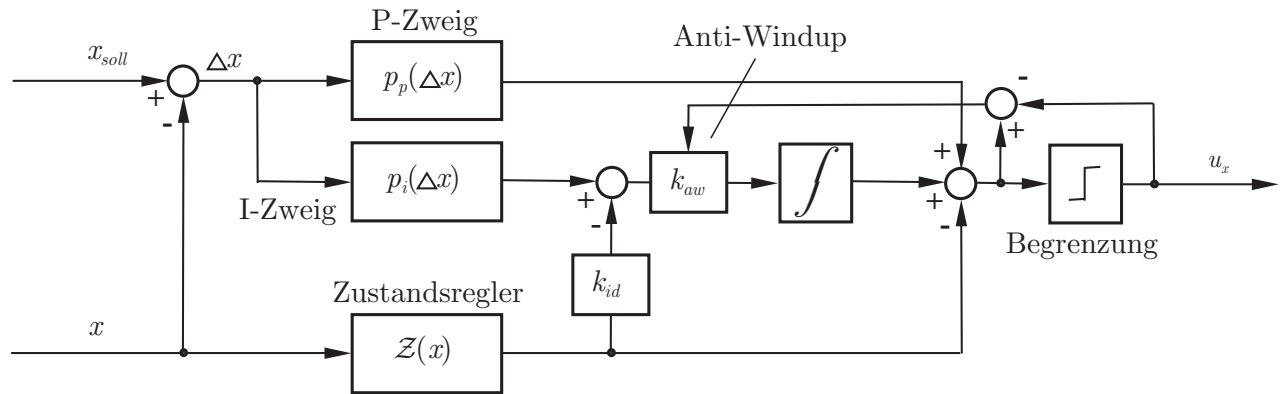


Abbildung 7.4: Struktur des Lagreglers.

Der Zustandsregler erhält als Eingangsgröße den Ventilhub $x(t)$. Dieser Zweig des Reglers weist zweifach differenzierenden Charakter auf und bringt dadurch ein Maß für die Geschwindigkeit $\dot{x}(t)$ und die Beschleunigung $\ddot{x}(t)$ des Ventilchiebers in die Stellgröße des Lagreglers u_x ein. Im Zweig des Zustandsreglers sind ebenfalls Verstärkungsfaktoren und Verzögerungs-Zeitkonstanten enthalten.

Der P-Zweig weist 8 zu optimierende Parameter auf, der I-Zweig enthält 11 einstellbare Parameter und der Zustandsregler kann durch 5 einstellbare Parameter beeinflusst werden. Insgesamt enthält die in Abbildung 7.4 dargestellte Reglerstruktur also 24 zu optimierende Parameter.

7.4 Bewertung der Regelungsgüte

Es wurde ein Gütemaß zur Bewertung der Regelungsgüte im Zeitbereich entwickelt. Hierbei werden Systemantworten $x(t)$ des Regelungssystems auf verschiedene Eingangsgrößen-Sprünge bewertet. Dies geschieht, indem aus jeder Sprungantwort Merkmale extrahiert und durch ein Fuzzy-System bewertet werden.

Die Gütebewertung durch ein Fuzzy-System, bestehend aus zwei Fuzzy-Modulen, erweist sich als besonders günstig, da durch ein *zweisträngiges Fuzzy-System* (s. u.: **Grundlagen von Fuzzy-Logik**) die teilweise widersprüchlichen Güteaspekte berücksichtigt werden können und durch einen entsprechenden Wert eines der Merkmale eine globale Abwertung der Güte einer Sprungantwort leicht realisiert werden kann.

Durch das erste Fuzzy-Modul (FM 1) wird hierbei eine Bewertung in Form von Schulnoten (Note 1 $\hat{=}$ sehr gut, ... , Note 6 $\hat{=}$ ungenügend) vorgenommen.

Das zweite Fuzzy-Modul (FM 2) dient der weiteren Differenzierung aller durch FM 1 mit 6 benoteten Sprungantworten, um eine identische Benotung bei unterschiedlichen Sprungantworten und somit verschiedenen Merkmalen zu vermeiden. Diese Vorgehensweise ist für den erfolgreichen Einsatz eines Optimierungsverfahrens erforderlich, um Plateaus in der Gütelandschaft zu vermeiden. Durch FM 2 wird das Spektrum der Gütewerte auf den Bereich $[1, 26]$ ausgedehnt.

Auf das Regelungssystem wird eine Folge von Eingangsgrößen-Sprüngen geschaltet. Jede einzelne Sprungantwort wird durch das Fuzzy-Modul FM 1 bzw. durch beide Fuzzy-Module FM 1 und FM 2 benotet. Anschließend wird aus den ermittelten Einzelnoten eine Gesamtnote berechnet. Diese Gesamtnote stellt das Gütemaß Q_{Zeit} dar.

Grundlagen von Fuzzy-Logik

In der klassischen Logik sind nur die Wahrheitswerte 1 (wahr) und 0 (falsch) bekannt. Bereits 1965 wurde von L. ZADEH die sogenannte *Fuzzy-Logik* eingeführt, die auch für Wahrheitswerte zwischen 0 und 1 definiert ist [Zad65]. An dieser Stelle kann nur ein grober Überblick über Fuzzy-Logik gegeben werden, für weiterführende Einzelheiten sei auf [Kie97] verwiesen.

Die Anwendungsfelder der Fuzzy-Logik sind vielfältig. Fuzzy-Logik kann in der Regelungstechnik in Form eines Fuzzy-Reglers eingesetzt werden. In diesem Fall spricht man von *Fuzzy Control*. Weitere Einsatzmöglichkeiten bestehen in der Ereigniserkennung sowie in der Gütebewertung. Auch auf dem Gebiet der Modellierung ist der Einsatz der Fuzzy-Logik in Form von Fuzzy-Modellen weit verbreitet.

Ein Fuzzy-Modul basiert auf *linguistischen Regeln*. Diese linguistischen Regeln weisen folgende Struktur auf:

$$\text{WENN } \langle \text{Prämisse} \rangle \quad \text{DANN } \langle \text{Konklusion} \rangle . \quad (7.1)$$

In einem Fuzzy-Modul sind meist mehrere linguistische Regeln implementiert. Die Menge aller linguistischen Regeln eines Fuzzy-Moduls wird *Regelbasis* genannt. In der Prämisse und der Konklusion einer jeden Regel werden aus dem Sprachgebrauch bekannte Ausdrücke wie beispielsweise ‘Temperatur = niedrig’ oder ‘Regelabweichung = groß’ verwendet. In dem Beispiel ‘Temperatur = niedrig’ ist die Temperatur die Prozessvariable, sie wird auch als *linguistische Variable* bezeichnet. Der *linguistische Wert* dieser Variablen kann zum Beispiel

‘niedrig’, ‘mittel’ oder ‘hoch’ sein. In einem Fuzzy-Modul werden zum einen die verwendeten linguistischen Regeln hinterlegt, zum anderen wird auch die Zuordnung zwischen linguistischen Werten (z. B. ‘niedrig’) und reellen Werten (z. B. $[0^{\circ}C, 25^{\circ}C]$) festgelegt.

Bei der Zuordnung

$$\begin{aligned} [0^{\circ}C, 25^{\circ}C] &\hat{=} \text{‘niedrig’}, \\]25^{\circ}C, 65^{\circ}C] &\hat{=} \text{‘mittel’}, \\]65^{\circ}C, 100^{\circ}C] &\hat{=} \text{‘hoch’} \end{aligned}$$

wird jeder reelle Wert für die Temperatur im Bereich $[0^{\circ}C, 100^{\circ}C]$ genau einem linguistischen Wert zugeordnet. Hierdurch entstehen ‘scharfe’ Übergänge zwischen den einzelnen linguistischen Werten an den Bereichsgrenzen. Diese Zuordnung kann durch sogenannte *Zugehörigkeitsfunktionen* (s. Abbildung 7.5 (a)) dargestellt werden.

Es kann aber auch eine unscharfe Zuordnung mit fließenden Übergängen realisiert werden. Hierzu werden häufig trapez- oder dreieckförmige Zugehörigkeitsfunktionen verwendet (s. Abbildung 7.5 (b)). Abbildung 7.5 zeigt beispielhaft Zugehörigkeitsfunktionen zu den linguistischen Werten ‘niedrig’, ‘mittel’ und ‘hoch’.⁵ Der Sonderfall ‘scharfer’ Übergänge wird hierbei durch rechteckige Zugehörigkeitsfunktionen realisiert (Abbildung 7.5 (a)).

Der Funktionswert μ gibt den Grad an, in dem eine Prozessvariable einem bestimmten linguistischen Wert zugeordnet ist. Durch den fließenden Übergang zwischen den Zugehörigkeitsfunktionen bei Variation der Prozessvariablen ergibt sich eine Verschiebung der Zugehörigkeitsgrade zu den einzelnen linguistischen Werten.

Aus der klassischen Logik sind die Operatoren \neg (NICHT), \wedge (UND) und \vee (ODER) zur Verarbeitung der booleschen Wahrheitswerte 0 und 1 bekannt. Auch im Rahmen der Fuzzy-Logik sind entsprechende Operatoren definiert, die für die Wahrheitswerte 0 und 1 mit den klassischen Operatoren übereinstimmen.

- Fuzzy-NICHT-Operator:

$$\neg\mu = 1 - \mu$$

- Fuzzy-UND-Operator:

– Algebraisches Produkt:

$$\mu_1 \wedge \mu_2 = \mu_1 \cdot \mu_2$$

⁵Aus Gründen der Übereinstimmung mit der Literatur wird für die Zugehörigkeitsfunktionen die Variable μ verwendet. Die Abgrenzung zu der Anzahl der Eltern-Individuen im Rahmen der Evolutionären Algorithmen, die ebenfalls durch die Variable μ gekennzeichnet ist, ergibt sich aber aus dem Zusammenhang.

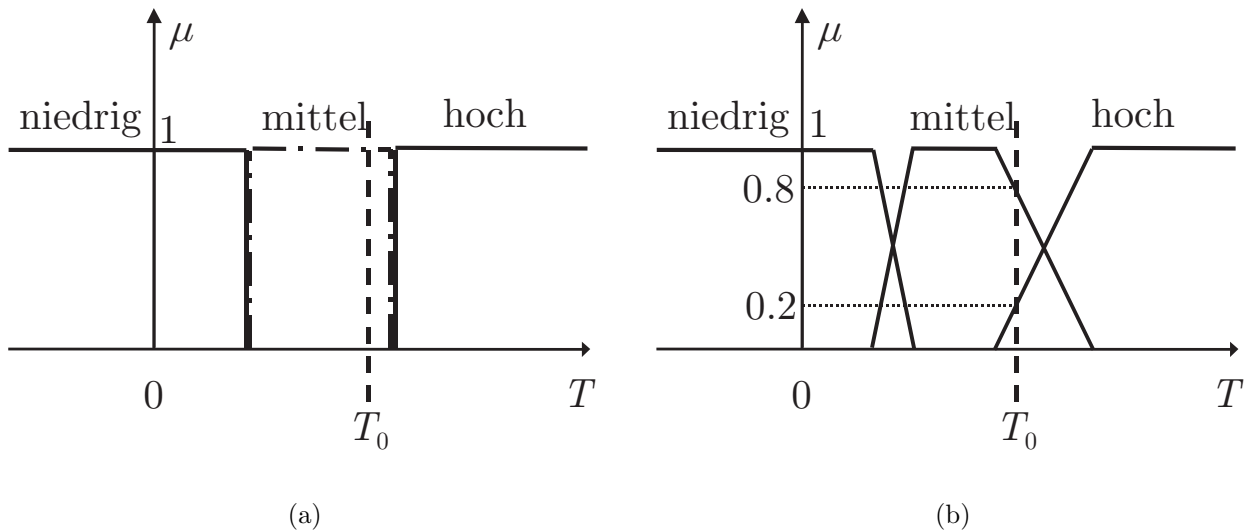


Abbildung 7.5: Zugehörigkeitsfunktionen μ zu den linguistischen Werten 'niedrig', 'mittel' und 'hoch'; (a) rechteckige Zugehörigkeitsfunktionen, die Temperatur T_0 ist im Grade $\mu = 1$ dem linguistischen Wert 'mittel' zugeordnet; (b) trapezförmige Zugehörigkeitsfunktionen, die Temperatur T_0 ist im Grade $\mu = 0.8$ dem linguistischen Wert 'mittel' und im Grade $\mu = 0.2$ dem linguistischen Wert 'hoch' zugeordnet.

– Minimum:

$$\mu_1 \wedge \mu_2 = \min(\mu_1, \mu_2)$$

• Fuzzy-ODER-Operator:

– Algebraische Summe:

$$\mu_1 \vee \mu_2 = \mu_1 + \mu_2 - \mu_1 \cdot \mu_2$$

– Maximum:

$$\mu_1 \vee \mu_2 = \max(\mu_1, \mu_2)$$

Es existieren noch weitere Fuzzy-Operatoren (UND: Hamacher-Produkt, Einstein-Produkt, begrenzte Differenz; ODER: Hamacher-Summe, Einstein-Summe, begrenzte Summe [Kie97]). Die oben genannten Operatoren werden jedoch am häufigsten verwendet.

Die Zuordnung einer reellen Prozessvariablen (z. B. $T_0 = 58^\circ\text{C}$) zu Zugehörigkeitsgraden zu linguistischen Werten (z. B. $\mu(T_0 = \text{mittel}) = \mu_{\text{mittel}}(T_0) = 0.8$, $\mu(T_0 = \text{hoch}) = \mu_{\text{hoch}}(T_0) = 0.2$) wird *Fuzzifizierung* genannt.

Die Prämisse einer linguistischen Regel besteht entweder aus einer Elementaraussage (z. B. Temperatur = niedrig) oder aus der Verknüpfung mehrerer Elementaraussagen mit Hilfe von Fuzzy-Operatoren (z. B. (Temperatur = niedrig) \wedge (Wind = stark)). Die Bestimmung des Erfülltheitsgrades der Prämisse wird *Aggregation* genannt (z. B. $(\mu(\text{Temperatur} = \text{niedrig}) \wedge \mu(\text{Wind} = \text{stark})) = 0.2$).

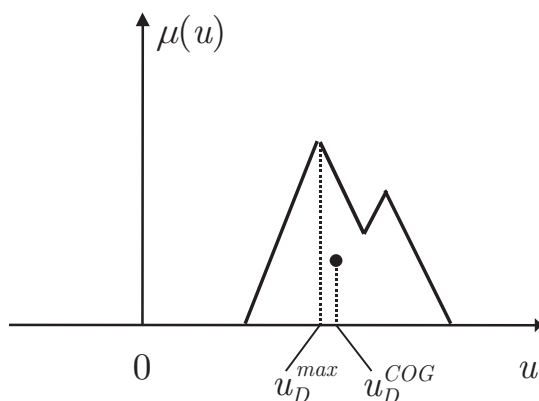


Abbildung 7.6: Beispiel für die Akkumulation der durch die Aktivierung gebildeten Zugehörigkeitsfunktionen für die Ausgangsgröße u des Fuzzy-Moduls und resultierender reeller Ausgangsgrößenwert bei Defuzzifizierung mit der Maximum-Mittelwert-Methode (u_D^{max}) und mit der Schwerpunktmethod (u_D^{COG}).

Je nach Erfülltheitsgrad der Prämisse wird die zugehörige linguistische Regel in einem bestimmten Maße *aktiviert*. Ist der Erfülltheitsgrad der Prämisse 1, so erfolgt die Aktivierung der zugehörigen Regel in vollem Maße. Bei geringerem Erfülltheitsgrad der Prämisse wird die durch die Konklusion gegebene Empfehlung der Regel abgeschwächt. Hierzu wird der Fuzzy-UND-Operator eingesetzt. Weist beispielsweise eine im Grade $\mu = 0.4$ aktivierte Regel die Konklusion ‘Stellgröße = groß’ auf, so ergibt sich durch $(\mu_{\text{groß}} \wedge 0.4)$ die Empfehlung der Regel.

Unter *Akkumulation* wird die Überlagerung der Empfehlungen aller linguistischen Regeln verstanden. Bei der Akkumulation wird der Fuzzy-ODER-Operator verwendet. Die Aktivierung aller Regeln und die Akkumulation werden unter dem Begriff *Inferenz* zusammengefasst.

Durch die Akkumulation ergibt sich ein ‘Empfehlungsgebirge’ für die Ausgangsgröße des Fuzzy-Moduls (s. Abbildung 7.6). Das Empfehlungsgebirge $\mu(u)$ gibt für jeden möglichen Ausgangsgrößenwert u an, wie attraktiv er als Ausgangsgröße ist. Aus dem Empfehlungsgebirge $\mu(u)$ wird schließlich ein reeller Ausgangsgrößenwert u_D bestimmt. Dieser Vorgang wird *Defuzzifizierung* genannt. Mögliche Defuzzifizierungs-Methoden sind die Maximum-Mittelwert-Methode oder die Schwerpunktmethod. Bei der Maximum-Mittelwert-Methode wird der Mittelwert aller Werte u gebildet, bei denen $\mu(u)$ maximal ist. Dieser Wert wird als Ausgangsgrößenwert u_D^{max} verwendet. Bei der Schwerpunktmethod wird der Schwerpunkt der Fläche bestimmt, die sich zwischen dem Funktionsgraphen $\mu(u)$ und der horizontalen Achse (u -Achse) ergibt. Es wird der Wert u , der sich durch das Lot des Flächenschwerpunktes auf die horizontale Achse ergibt, als

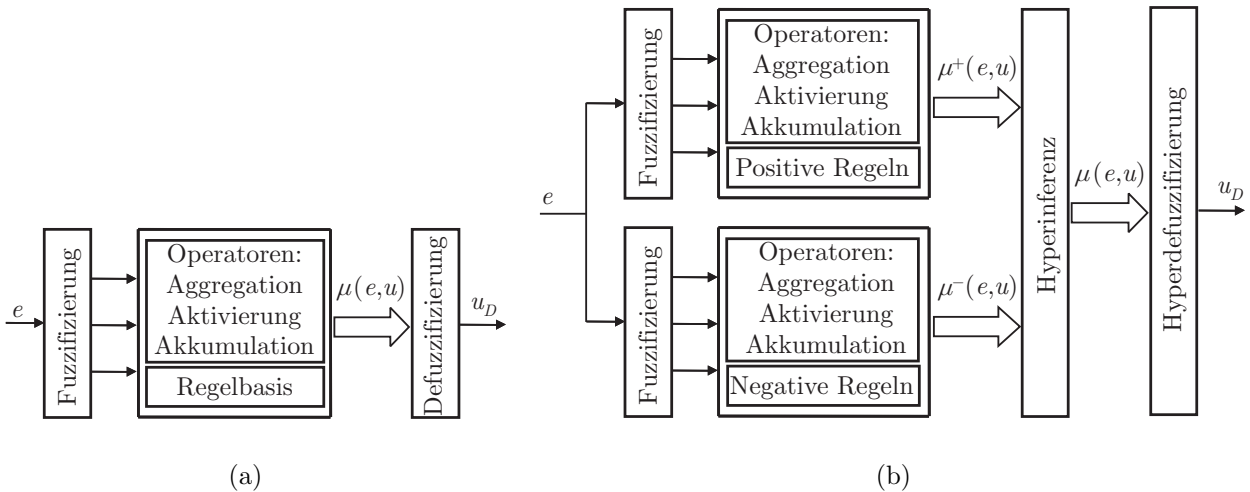


Abbildung 7.7: Struktur eines einsträngigen Fuzzy-Moduls (a) und eines zweisträngigen Fuzzy-Moduls (b) mit der Prozessvariablen e als Eingangsgröße (es ist auch möglich, einen Eingangsgrößenvektor \mathbf{e} zu verwenden).

Ausgangsgrößenwert verwendet (u_D^{COG}).⁶

Die bereits genannten Verarbeitungsschritte werden in Abbildung 7.7 (a) verdeutlicht.

Mit Hilfe eines erweiterten Fuzzy-Moduls können nicht nur linguistische Regeln nach Ausdruck (7.1) verarbeitet werden. Es ist auch möglich, Regeln in der Form

$$\text{WENN } \langle \text{Prämisse} \rangle \quad \text{DANN } \langle \text{Konklusion} \rangle \quad \text{VERBOTEN} \quad (7.2)$$

zu verwenden. Linguistische Regeln nach Ausdruck (7.1) drücken Empfehlungen bezüglich der Ausgangsgröße aus. Diese Regeln werden auch *positive Regeln* genannt. Im Gegensatz hierzu werden durch linguistische Regeln nach Ausdruck (7.2) Warnungen oder Verbote bezüglich der Ausgangsgröße des Fuzzy-Moduls ausgesprochen. Diese Regeln werden als *negative Regeln* bezeichnet.

Aufgrund der negativen Regeln ergeben sich Bereiche der Ausgangsgröße, vor denen vollständig oder in einem bestimmten Grade gewarnt wird. Für die Verarbeitung von negativen Regeln existieren dieselben Methoden wie für die Verarbeitung von positiven Regeln.

Ein Fuzzy-Modul, dessen Regelbasis nur positive Regeln beinhaltet, wird *einsträngiges Fuzzy-Modul* genannt. Ein Fuzzy-Modul, dessen Regelbasis sowohl positive als auch negative Regeln enthält, wird als *zweisträngiges Fuzzy-Modul* bezeichnet. In einem zweisträngigen Fuzzy-Modul können für die Verarbeitung

⁶COG: Center of Gravity (engl.): Schwerpunkt

der positiven Regeln andere Methoden (Fuzzy-Operatoren) verwendet werden als für die Verarbeitung der negativen Regeln.

Abbildung 7.7 (b) zeigt schematisch den Aufbau eines zweisträngigen Fuzzy-Moduls. Die Verrechnung des ‘Empfehlungsgebirges’ der positiven und des ‘Verbotsgebirges’ der negativen Regeln wird *Hyperinferenz* genannt. Es existieren verschiedene Hyperinferenz-Strategien [Kie97].

Aus dem aus der Hyperinferenz resultierenden ‘Ausgangsgebirge’ wird durch die *Hyperdefuzzifizierung* ein sinnvoller Ausgangsgrößenwert u_D ermittelt [Kie97].

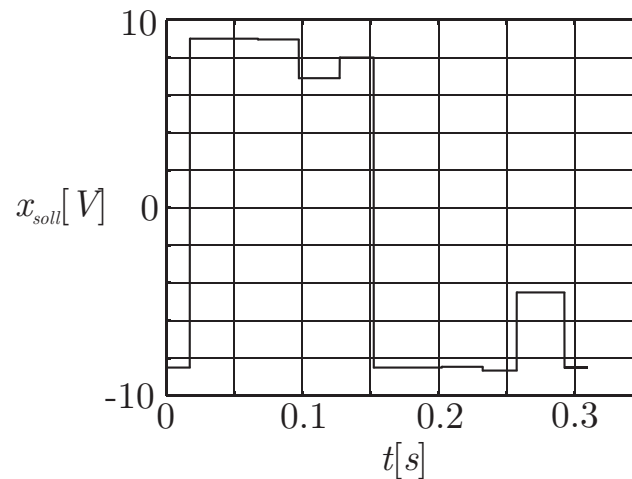
Bewertungsszenario

Der Eingangsgrößen-Verlauf, der dem Sollwert für den Ventilhub $x_{soll}(t)$ entspricht, besteht aus einer Folge von 9 verschiedenen Sprüngen (s. Abbildung 7.8). Der verwendete Sollwert-Verlauf enthält Klein-, Mittel- und Großsignalsprünge, um den Bereich der im industriellen Einsatz auftretenden Sollwert-Verläufe möglichst gut abzudecken.

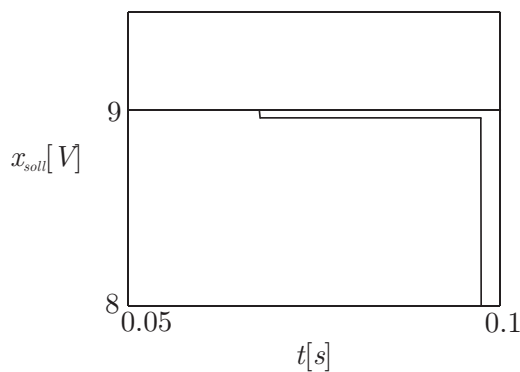
Aufgrund der Nichtlinearitäten des Ventils (magnetische Hysterese, Reibung, etc.) und der daraus resultierenden Nichtlinearitäten des Lagereglers sind die Sprungantworten des Regelungssystems auf unterschiedliche Eingangsgrößen-Sprünge von der jeweiligen Sprunghöhe Δx_{soll} , der Richtung des jeweiligen Sprunges (positiv oder negativ) sowie von den zugehörigen Absolutwerten $x_{soll \text{ vor Sprung}}$ und $x_{soll \text{ nach Sprung}}$ des Eingangsgrößen-Sprunges abhängig.

Um den Einfluss von Rauschen und Störungen zu reduzieren, wird der in Abbildung 7.8 (a) dargestellte Eingangsgrößen-Verlauf für jeden zu bewertenden Satz von Reglerparametern \mathbf{x} des Lagereglers mehrfach aufgeschaltet [BW00]. Hierbei hat sich eine 5- bzw. 10-fache Aufschaltung des dargestellten Eingangsgrößen-Verlaufs als ausreichend erwiesen. Für jeden der 9 Eingangsgrößen-Sprünge wird der Mittelwert der 5 bzw. 10 Einzelnoten gebildet. Anschließend wird die Gesamtnote wie in Tabelle 7.1 dargestellt berechnet.

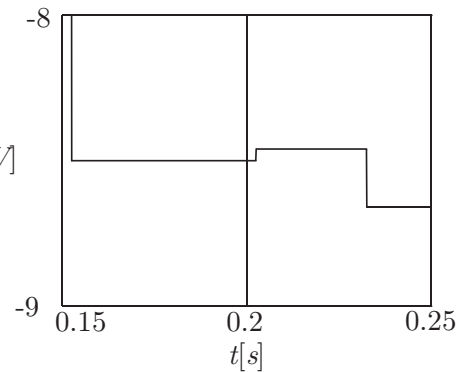
Die Aufzeichnung des Ventilhubes $x(t)$ erfolgt in zeitlichen Abständen von $100\mu s$ (Abtastzeit). Um den zeitlichen Aufwand der Gütebewertung zu reduzieren, wird für die Ermittlung des Gütewertes Q_{Zeit} eine Unterabtastung vorgenommen. Für die Gütebestimmung wird nur jeder zweite Datenpunkt verwendet (Unterabtastung $r = 2$). Durch diesen Vorgang wird der zeitliche Aufwand für die Datenaufnahme nicht reduziert, jedoch wird die Berechnungsdauer für die anschließende Güteermittlung anhand der aufgezeichneten Daten durch die kleinere zu betrachtende Datenmenge verkleinert. Durch eine Unterabtastung



(a)



(b)



(c)

Abbildung 7.8: Sprungförmiger Eingangsgrößenverlauf (Stimulus-Funktion) $x_{soll}(t)$ (a) und Zoom, um Kleinsignalsprünge sichtbar zu machen ((b) und (c)).

wird die Gütebewertung ungenauer, ein Wert von $r = 2$ hat sich jedoch als akzeptabel erwiesen. Hierdurch wird die Gütebewertung nur geringfügig verfälscht [Nic99a].

Gütemaß zur Bewertung von Sprungantworten

Die Bewertung ist modular aufgebaut. Es können auf Wunsch leicht weitere Merkmale aus den Sprungantworten extrahiert und bei der Gütebestimmung mitbewertet werden.

Die Bewertungsgrundlage stellte eine Bewertung einer Vielzahl unterschiedlicher Sprungantworten durch einen Prozessexperten dar. Diese Bewertung wurde von dem Prozessexperten durch visuelle Inspektion vorgenommen.

Tabelle 7.1: Berechnung der Gesamtgüte $Q_{Zeit}(\mathbf{x})$ eines Satzes von Reglerparametern \mathbf{x} aus den Einzelnoten aller Sprungantworten $Q_{Zeit(i,j)}(\mathbf{x})$ bei 5 Aufschaltungen des Eingangsgrößenverlaufs, bestehend aus jeweils 9 Sprüngen; $\sigma_{Q_{Zeit(j)}} = \sqrt{\frac{1}{9-1} \cdot \sum_{j=1}^9 (Q_{Zeit(j)}(\mathbf{x}) - \bar{Q}_{Zeit}(\mathbf{x}))^2}$ ist die Standardabweichung aller $Q_{Zeit(j)}(\mathbf{x})$ vom Mittelwert $\bar{Q}_{Zeit}(\mathbf{x}) = \frac{1}{9} \cdot \sum_{j=1}^9 Q_{Zeit(j)}(\mathbf{x})$.

	Sprung 1	...	Sprung 9	
Lauf 1	$Q_{Zeit(1,1)}(\mathbf{x})$...	$Q_{Zeit(1,9)}(\mathbf{x})$	
⋮	⋮	⋱	⋮	
Lauf 5	$Q_{Zeit(5,1)}(\mathbf{x})$...	$Q_{Zeit(5,9)}(\mathbf{x})$	
	$Q_{Zeit(1)}(\mathbf{x}) =$ $\frac{1}{5} \cdot \sum Q_{Zeit(i,1)}(\mathbf{x})$...	$Q_{Zeit(9)}(\mathbf{x}) =$ $\frac{1}{5} \cdot \sum Q_{Zeit(i,9)}(\mathbf{x})$	$Q_{Zeit}(\mathbf{x}) =$ $0.5 \cdot \sigma_{Q_{Zeit(j)}} + \bar{Q}_{Zeit}(\mathbf{x})$

Bei der Entwicklung des Fuzzy-Moduls FM 1 wurde auf eine möglichst große Übereinstimmung der durch das Fuzzy-Modul erhaltenen Noten mit den durch den Prozessexperten ermittelten Noten geachtet. Das Fuzzy-Modul FM 2 stellt eine Erweiterung der Expertenbewertung dar.

Das Fuzzy-Gütemaß wurde wissensbasiert – d. h. auf der Grundlage von Expertenwissen – erstellt. Diese wissensbasierte Realisierung ist aufwendig, da eine Vielzahl von – teilweise widersprüchlichen – Bewertungsaspekten in das Fuzzy-Gütemaß integriert werden müssen. Durch die Verwendung eines zweisträngigen Fuzzy-Moduls mit positiven und negativen Regeln ist es jedoch leichter möglich, den unterschiedlichen Güteaspekten gerecht zu werden. Prinzipiell ist auch eine datenbasierte Erstellung eines Fuzzy-Gütemaßes auf der Grundlage der Expertenbewertung möglich. Hierfür kann beispielsweise das Software-Tool WINROSA eingesetzt werden [Kie97, Kro99, Sla01, Kra02].

Folgende Gütekriterien wurden durch den Prozessexperten festgelegt:

- Anregelzeit
- Überschwingweite
- Ausregelzeit
- Schwingneigung

Die Bewertung der einzelnen Gütekriterien soll jedoch nicht bezogen auf die absoluten Werte, sondern abhängig von der Sprunghöhe erfolgen. Hierzu wird eine geeignete Normierung vorgenommen. Durch eine Umnormierung kann das Gütemaß somit auch vergleichsweise einfach an unterschiedliche Ventiltypen, an verschiedene Anforderungen oder an neue Einsatzbereiche angepasst werden.

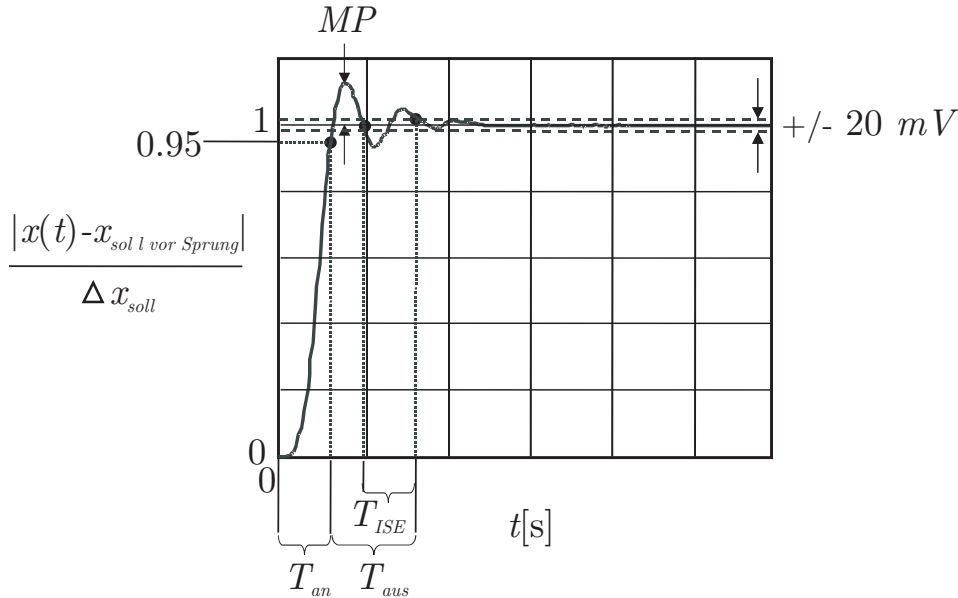


Abbildung 7.9: Normierter Ventilhub $x(t)$ (zum Zeitpunkt $t = 0$ wird ein Sollwertsprung aufgeschaltet); Darstellung der Anregelzeit T_{an} , der maximalen Überschwingweite MP , der Ausregelzeit T_{aus} sowie der zur Ermittlung der Schwingneigung benötigten Zeitspanne T_{ISE} .

Unter der Anregelzeit T_{an} wird die Zeitspanne von der Aufschaltung eines Eingangsgrößen-Sprunges bis zum Erreichen von 95 % der Sprunghöhe $\Delta x_{soll} = |x_{soll \text{ nach Sprung}} - x_{soll \text{ vor Sprung}}|$ durch den Ventilhub verstanden (s. Abbildung 7.9). Die aus der Sprungantwort ermittelte Anregelzeit T_{an} wird durch die von der Sprunghöhe Δx_{soll} abhängige, experimentell bestimmte schnellstmögliche Anregelzeit $T_{an \text{ opt}}(\Delta x_{soll})$ dividiert:

$$T_{an \text{ norm}} = \frac{T_{an}}{T_{an \text{ opt}}(\Delta x_{soll})}.$$

$T_{an \text{ opt}}(\Delta x_{soll})$ wird experimentell folgendermaßen bestimmt: Zunächst wird $i_{soll} = 0$ (s. Abbildung 7.2) aufgeschaltet. Der Ventilschieber befindet sich dabei – bedingt durch die Bauart des Ventils⁷ – an der einen Grenze des möglichen Bewegungsbereichs. Nun wird auf das Ventil bei deaktiviertem Regler $i_{soll} = i_{soll \text{ max}}$ aufgeschaltet und der Ventilhub über der Zeit aufgezeichnet. Die Zeitdauer bis zum Erreichen eines bestimmten Ventilhubes kann als schnellstmögliche Anregelzeit für eine dem erreichten Ventilhub entsprechende Sprunghöhe Δx_{soll} verstanden werden.

⁷Im Ventil befindet sich eine Rückstellfeder, die ohne Magnetstrom den Ventilschieber im Anschlag hält.

$T_{an\ opt}(\Delta x_{soll})$ wird in Form einer Kennlinie hinterlegt. $T_{an\ norm}$ stellt eine der vier Eingangsgrößen des Fuzzy-Systems zur Gütebewertung dar.

Als Überschwingweite MP wird die maximale Abweichung des Ventilhubes $x(t)$ vom Sollwert $x_{soll}(t)$ nach dem durch T_{an} festgelegten Anregelzeitpunkt und ab dem erstmaligen Richtungswechsel des Ventilhubes (Vorzeichenwechsel von $\dot{x}(t)$) bezeichnet (s. Abbildung 7.9). Diese betraglich maximale Differenz zwischen $x(t)$ und $x_{soll}(t)$ wird in vielen Fällen, aber nicht immer, während des ersten Überschwingens erreicht. Die Überschwingweite MP wird auf eine von der Sprunghöhe Δx_{soll} abhängige maximal zulässige Überschwingweite MP_{max} normiert:

$$MP_{norm} = \frac{MP}{MP_{max}(\Delta x_{soll})}.$$

Hierbei wurden die von Δx_{soll} abhängigen maximal zulässigen Überschwingweiten MP_{max} für einzelne ausgewählte Sprunghöhen anhand der vorliegenden Expertenbewertungen empirisch ermittelt [Bül98, Nic99b]. Für Zwischenwerte von Δx_{soll} wird in geeigneter Weise interpoliert. Auch hier wird $MP_{max}(\Delta x_{soll})$ in Form einer Kennlinie implementiert. MP_{norm} ist die zweite Eingangsgröße des Fuzzy-Systems.

Die dritte Eingangsgröße des Fuzzy-Systems stellt die normierte Ausregelzeit $T_{aus\ norm}$ dar. Als Ausregelzeit T_{aus} wird die Zeitdauer vom durch T_{an} definierten Anregelzeitpunkt bis zu dem Zeitpunkt verstanden, ab welchem der Betrag der Regelabweichung $|\Delta x(t)| = |x_{soll}(t) - x(t)|$ ständig kleiner als $20mV$ ist (s. Abbildung 7.9). Diese Ausregelzeit wird mit Hilfe eines Anpassungsfaktors γ , der ebenfalls von der Sprunghöhe Δx_{soll} abhängt, normiert:

$$T_{aus\ norm} = \frac{T_{aus}}{\gamma(\Delta x_{soll})}.$$

$\gamma(\Delta x_{soll})$ ist im Gegensatz zu $T_{an\ opt}(\Delta x_{soll})$ und $MP_{max}(\Delta x_{soll})$ keine interpretierbare Größe. Dieser Wert dient lediglich dazu, einer in Abhängigkeit von der Sprunghöhe Δx_{soll} unterschiedlichen Bewertung der Ausregelzeit gerecht zu werden [Nic99b]:

$$\gamma(\Delta x_{soll}) = 1 + \frac{\Delta x_{soll}}{20V}.$$

Zur Ermittlung der Schwingneigung als vierte Eingangsgröße des Fuzzy-Systems wird zunächst die Zeitdauer T_{ISE} vom zweiten Nulldurchgang der Regelabweichung $\Delta x(t)$ bis zu dem Zeitpunkt ermittelt, ab welchem $|\Delta x(t)|$ ständig kleiner

ist als $20mV$ (s. Abbildung 7.9). Das herkömmliche Maß ISE als Integral über die quadrierte Regelabweichung wird folgendermaßen modifiziert⁸:

$$ISE_{mod} = \delta \cdot ISE,$$

wobei

$$ISE = \int_{T_1}^{T_2} \left(\frac{\Delta x(t)}{\Delta x_{soll}} \right)^2 dt$$

verwendet wird.

δ ist ein normiertes Maß für die Kurvenlänge von $x(t)$ zwischen dem Zeitpunkt $T_1 = T_{an} + T_{aus} - T_{ISE}$ des zweiten Nulldurchgangs der Regelabweichung $\Delta x(t)$ und dem Zeitpunkt $T_2 = T_{an} + T_{aus}$:

$$\delta = \frac{1}{T_{ISE}} \cdot \int_{T_1}^{T_2} (\sqrt{1 + q \cdot \dot{x}(t)^2}) dt.$$

δ kann als Oszillationsmaß interpretiert werden. Abweichend von der Berechnungsvorschrift für die Kurvenlänge wurde ein Faktor q eingeführt. Für diesen wurde ein Wert von $q = 0.1 \frac{s^2}{V^2}$ empirisch festgelegt. Durch den Faktor q kann die Größenordnung des Oszillationsmaßes δ an die Größenordnung von ISE angepasst und die Wirkungsweise des Oszillationsmaßes beeinflusst werden [Ste01].

Die Verrechnung der einzelnen vier Gütekriterien zu dem skalaren Gütemaß Q_{Zeit} erfolgt durch die Fuzzy-Module FM 1 und FM 2.

Die vier Gütekriterien dienen als Eingangsgrößen der Fuzzy-Module. Die verwendeten Zugehörigkeitsfunktionen des Fuzzy-Moduls FM 1 sind in Abbildung 7.10 dargestellt. Die Zugehörigkeitsfunktionen des zweiten Fuzzy-Moduls FM 2 sind in Anhang B in Abbildung B.1 dargestellt.

Bei beiden Fuzzy-Modulen besteht die Regelbasis aus Regeln, die jeweils nur eine Elementaraussage in der Prämisse aufweisen. Die Regelbasis des Fuzzy-Moduls FM 1 ist in Anhang B in Tabelle B.1 zu finden, die Regelbasis von FM 2 ist in Tabelle B.2 aufgelistet.

Die Zugehörigkeitsfunktionen der Ausgangsgröße Q_{Zeit} sind für FM 1 in Abbildung 7.11 dargestellt. Abbildung 7.11 (a) zeigt die in den positiven Regeln

⁸ ISE : integral of squared error (engl): Integral über die quadrierte Regelabweichung

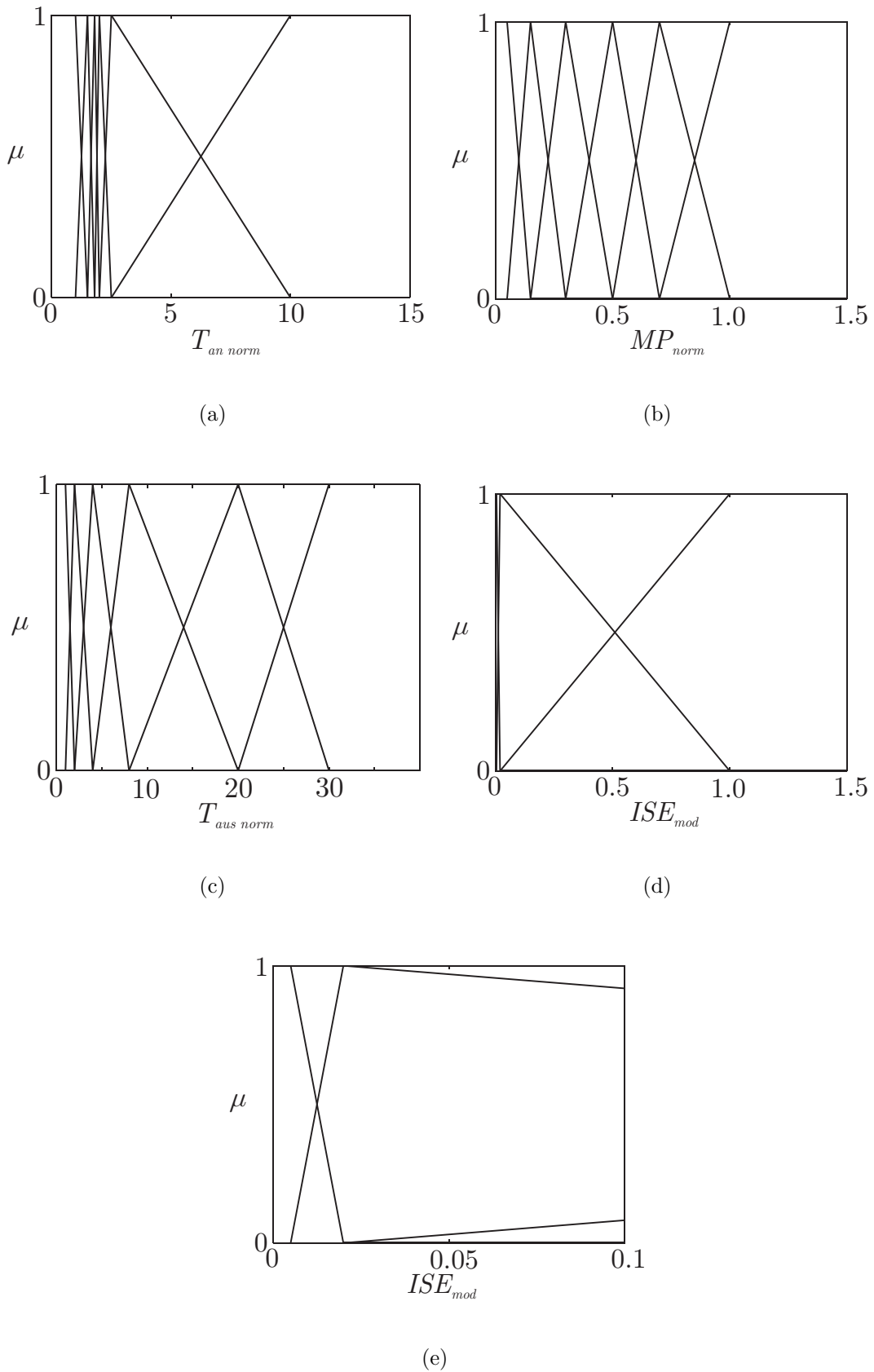


Abbildung 7.10: Zugehörigkeitsfunktionen der Eingangsgrößen $T_{an\ norm}$ (a), MP_{norm} (b), $T_{aus\ norm}$ (c), ISE_{mod} (d) und (e) für FM 1.

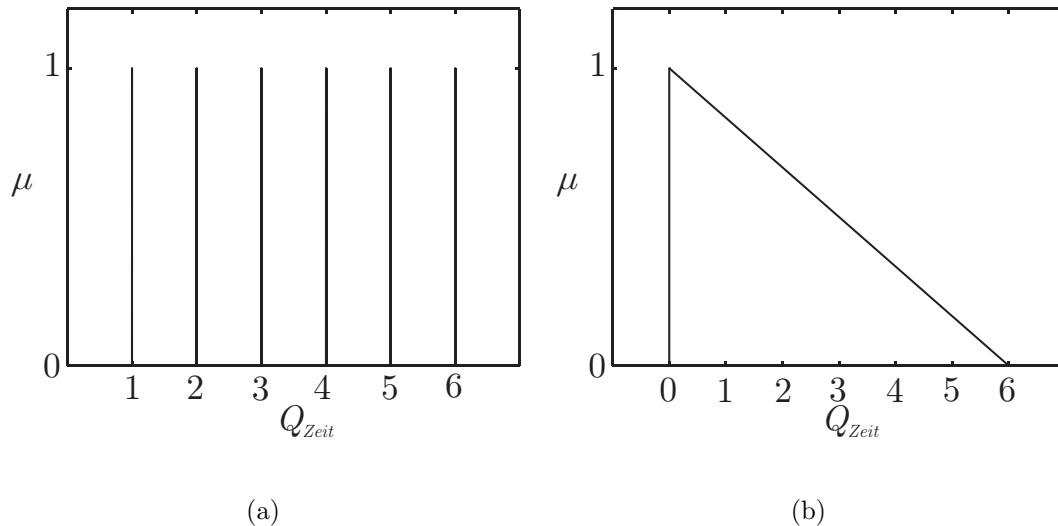


Abbildung 7.11: Zugehörigkeitsfunktion(en) der Ausgangsgröße Q_{Zeit} für positive Regeln (a) und negative Regeln (b) für FM 1.

verwendeten Zugehörigkeitsfunktionen, in Abbildung 7.11 (b) ist die in den negativen Regeln verwendete Zugehörigkeitsfunktion dargestellt. Für FM2 sind die Zugehörigkeitsfunktionen der Ausgangsgröße Q_{Zeit} in Anhang B in Abbildung B.2 dargestellt. Im Fuzzy-Modul FM 2 werden nur positive Regeln verarbeitet.

Für die positiven Regeln werden Singletons als ausgangsseitige Zugehörigkeitsfunktionen verwendet. Dies sind Zugehörigkeitsfunktionen, die nur an einer Stelle einen Funktionswert $\mu = 1$ aufweisen und an allen anderen Stellen $\mu = 0$ als Funktionswert haben.

Die Aktivierung der einzelnen Regeln (positive und negative Regeln) erfolgt durch das algebraische Produkt. Die Akkumulation der Konklusionen der positiven Regeln und der negativen Regeln wird zunächst getrennt voneinander durch die für reelle Zahlen erklärte Summe vorgenommen.

Die aktivierten Singletons werden bei FM 1 zur Ermittlung einer Parabel verwendet, wobei der Scheitelpunkt der Parabel von der Lage und vom Aktivierungsgrad der aktivierten Singletons abhängt. Die Breite w der Parabel ist konstant. Die Parabel wird folgendermaßen angesetzt:

$$\mu(u) = -\left(\frac{u - u_{COS}}{0.5 \cdot w}\right)^2 + 1, \quad (7.3)$$

wobei für die Ausgangsgröße $u = Q_{Zeit}$ gilt und für w ein Wert von 10 empirisch ermittelt wurde. Die Parabel $\mu(u)$ ist in Abbildung 7.12 dargestellt.

Der Scheitelpunkt der Parabel fällt mit dem Schwerpunkt u_{COS} der Singletons

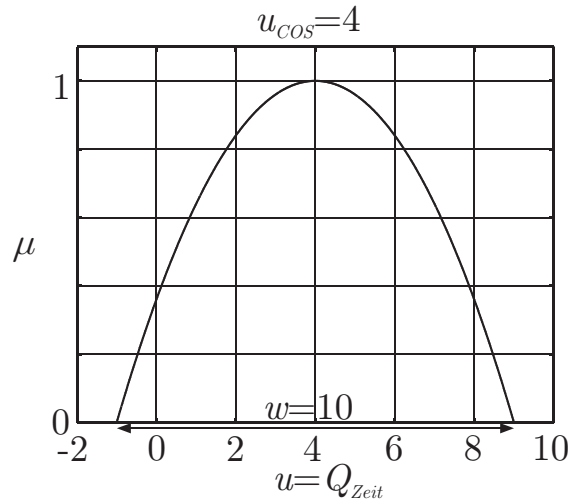


Abbildung 7.12: Parabel abhängig von der Lage und Aktivierung der Singletons für die positiven Regeln von FM 1; beispielhafte Darstellung für den Fall $u_{COS} = 4$.

zusammen:⁹

$$u_{COS} = \frac{\sum_{k_s=1}^N \mu_{k_s}^+ \cdot u_{k_s}^+}{\sum_{k_s=1}^N \mu_{k_s}^+}. \quad (7.4)$$

Hierbei ist N die Anzahl der ausgangsseitigen Singletons, hier ist also $N = 6$. $\mu_{k_s}^+$ sind die Aktivierungsgrade der Singletons, ‘+’ verdeutlicht, dass es sich um die Zugehörigkeitsfunktionen für die positiven Regeln handelt.

Bei alleiniger Verwendung positiver Regeln, wie dies bei FM 2 der Fall ist, ist der Einsatz einer solchen Parabel überflüssig: Dort kann der Schwerpunkt der Singletons nach Gleichung (7.4) als Ausgangsgrößenwert verwendet werden. Da für FM 1 aber ein zweisträngiges Fuzzy-Modul zum Einsatz kommt, wird die ermittelte Parabel durch die Hyperinferenz (s. Abbildung 7.7 (b)) mit den Warnungen bzw. Verboten der negativen Regeln verrechnet. Als Hyperinferenzstrategie wird die begrenzte Differenz (Fuzzy-Veto) verwendet. Die Hyperdefuzzifizierung erfolgt nach der Schwerpunktmethod (COG).

Die ursprüngliche Idee, auch die Breite w der Parabel von der Lage und vom Aktivierungsgrad der aktivierten Singletons abhängig zu machen [KK97], wurde hier nicht aufgegriffen. Der Grund hierfür liegt in der Feststellung, dass bei variabler Breite kein monotonen Verhalten des Fuzzy-Moduls erreicht werden kann [Nic99b]. Es ist aber ein monotonen Verhalten der Benotung in Abhängigkeit der Eingangsgrößen des Fuzzy-Moduls erforderlich, um ein eingesetztes Optimierungsverfahren durch die Beschaffenheit des Gütemaßes zu unterstützen und

⁹COG: Center of Singletons (engl.): Schwerpunkt der Singletons

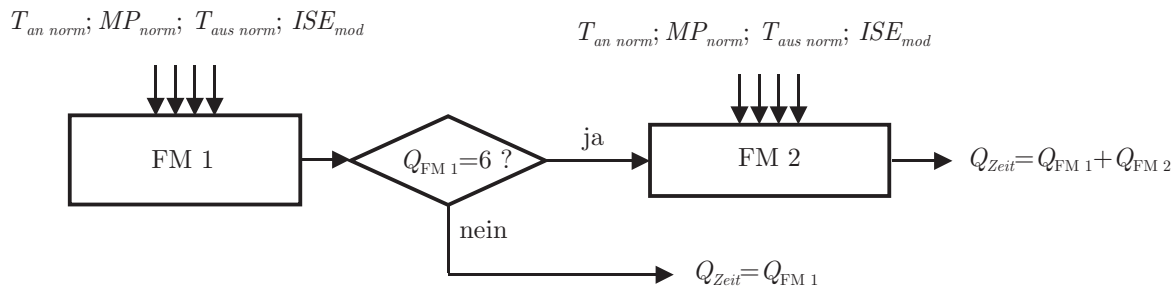


Abbildung 7.13: Flussdiagramm zur Ermittlung des Gütewertes Q_{Zeit} anhand der einzelnen Gütekriterien.

den Optimierungsprozess nicht durch nichtmonotones Verhalten des Gütemaßes zusätzlich zu erschweren.

Abbildung 7.13 zeigt, wie aus den vier Gütekriterien mit Hilfe von FM 1 und FM 2 der Wert des Gütemaßes Q_{Zeit} ermittelt wird.

Zusammenfassend sei darauf hingewiesen, dass bei der Erstellung eines Gütemaßes Monotonie und Glattheit¹⁰ besonders wichtig sind, damit durch ein Optimierungsverfahren ein Satz von Reglerparametern \mathbf{x} gefunden werden kann, der zu einem besonders günstigen Verhalten des Regelungssystems führt. Die dargestellte Bewertung des Verhaltens des Regelungssystems kann als Bewertung des eingesetzten Reglers aufgefasst werden. Es wird innerhalb des Optimierungsprozesses jeder Parametersatz \mathbf{x} durch das beschriebene Gütemaß $Q_{Zeit}(\mathbf{x})$ bewertet.

Monotonie des Gütemaßes wird durch die verwendete Regelbasis (Anhang B, Tabellen B.1 und B.2), durch die Wahl der verwendeten Zugehörigkeitsfunktionen (Abbildungen 7.10 und 7.11 sowie Abbildungen B.1 und B.2) sowie durch den Einsatz einer Parabel mit konstanter Breite bei FM 1 erreicht. Diese Monotonie des Gütemaßes führt jedoch nicht zwingend zu einer monotonen Güte-landschaft (s. Kapitel 2).

Die Streuung des Verhaltens des Regelungssystems stellt eine zusätzliche Schwierigkeit dar, der durch mehrfache Aufschaltung des Eingangsgrößen-Verlaufs (s. Tabelle 7.1) entgegengewirkt wird. Somit wird diese Schwierigkeit gemildert und für eine größere Glattheit der Gütebewertung gesorgt.

Die Bewertung eines Parametersatzes \mathbf{x} dauert unter Verwendung der in Abschnitt 7.1 beschriebenen Hardware bei 5-facher Aufschaltung des Eingangs-

¹⁰Glattheit bedeutet hierbei, dass sich bei geringfügiger Variation einer oder mehrerer Eingangsgrößen des Gütemaßes der Gütewert ebenfalls nur geringfügig ändert. Zur Beurteilung der Glattheit ist die Verwendung des in [Kie00] vorgestellten Rauigkeitsmaßes denkbar.

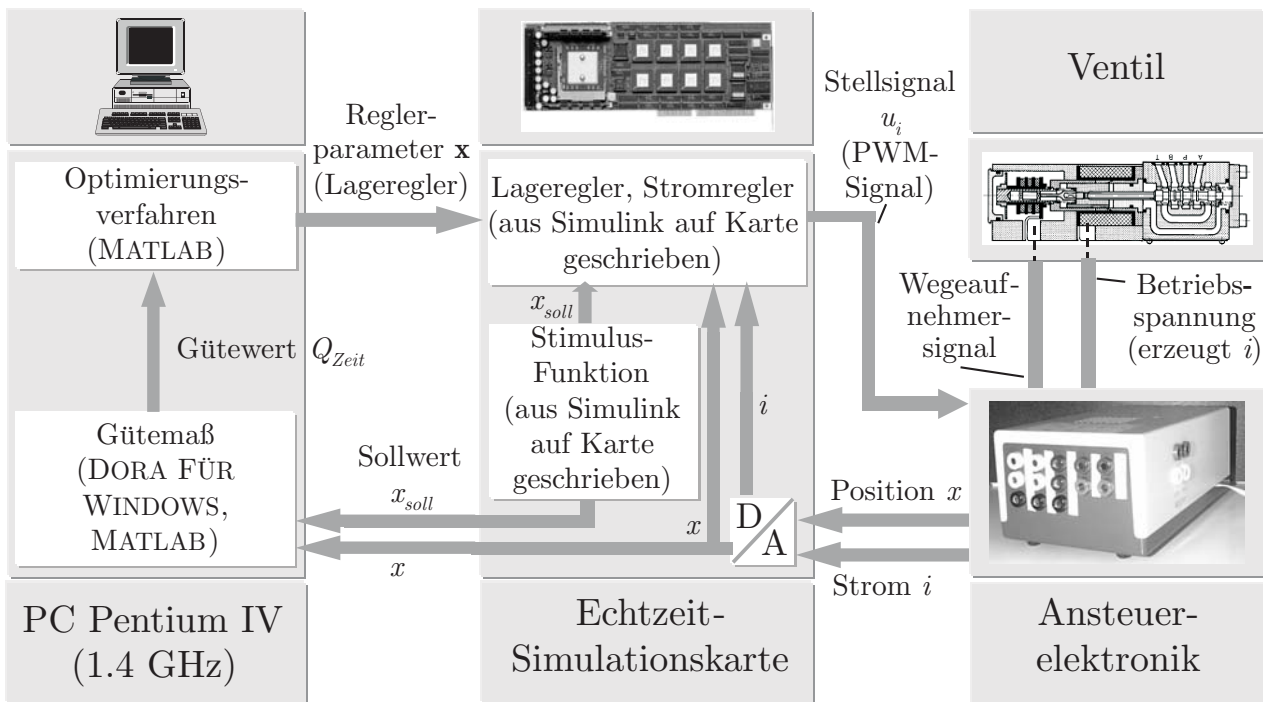


Abbildung 7.14: Schematische Darstellung des Laborstands.

größen-Verlaufs ca. 2.4s. Wird als Optimierungsverfahren eine Evolutionsstrategie eingesetzt, so ergibt sich durch weitere Rechenzeiten eine Dauer von ca. 25min für jede Generation (bei 430 Gütebewertungen innerhalb der Generation). Somit dauert ein Optimierungsprozess bestehend aus beispielsweise 50 Generationen insgesamt ca. 21 Stunden [Bre02].

Die Gütebewertung soll mit möglichst wenig zeitlichem Aufwand auskommen. Sie soll aber trotzdem die Gütekriterien möglichst genau ermitteln und bei gleichem Parametersatz \mathbf{x} trotz Variationen im Verhalten des Regelungssystems möglichst reproduzierbare Gütewerte liefern. Diesen Anforderungen wurde durch das vorgestellte Gütemaß Q_{Zeit} Rechnung getragen.

7.5 Optimierungsergebnisse

Im Folgenden werden einige ausgewählte Optimierungsergebnisse bei Verwendung des Gütemaßes Q_{Zeit} vorgestellt.

Hierbei wird ein Vergleich zwischen den Optimierungsstrategien ES_{std} , ES_{GBT}^{unsymm} sowie ES_{hybrid} vorgenommen. Außerdem werden Ergebnisse bei Verwendung der Strategie $HoJe_{Neustart}$ vorgestellt und mit den zuvor genannten Ergebnissen verglichen.

In jeder ES-Generation werden $\mu = 30$ Eltern-Individuen und $\lambda = 200$ Nachkommen-Individuen verwendet. Alle Eltern-Individuen der Startpopulation werden hier im gesamten Suchraum zufällig gleichverteilt platziert, um für den Optimierungsprozess möglichst gute Voraussetzungen zu schaffen.

Abbildung 7.15 zeigt die Güterwerte der einzelnen Optimierungsprozesse. Abbildung 7.16 zeigt die Parameterwerte der Individuen im Laufe der Optimierungsprozesse.

Bei Verwendung der Strategie ES_{std} liegt der beste erreichte Güterwert nach insgesamt 16000 Bewertungen mittels des Gütermaßes bei $Q_{Zeit} = 2.13$. Um ein qualitatives Maß für die Schnelligkeit des Optimierungsprozesses zu erhalten, wird ein Güter-Schwellwert von $\Delta Q_{Zeit} = 0.5$ zu diesem Wert hinzu addiert. Dieser Güterwert von $Q_{Zeit} = 2.63$ ist nach 2610 Bewertungen mittels des Gütermaßes erreicht (s. Abbildung 7.15 (a)).

Bei Verwendung der Strategie ES_{GBT}^{unsymm} liegt der beste erreichte Güterwert nach insgesamt 16000 Bewertungen mittels des Gütermaßes bei $Q_{Zeit} = 2.14$. Auch hier wird ein Güter-Schwellwert von $\Delta Q_{Zeit} = 0.5$ zu diesem Wert hinzu addiert. Der Güterwert von $Q_{Zeit} = 2.64$ ist nach 2180 Bewertungen mittels des Gütermaßes Q_{Zeit} erreicht (s. Abbildung 7.15 (b)). Gegenüber der Strategie ES_{std} konnte zwar kein besserer Güterwert erreicht werden, aber durch eine breitere Durchsuchung des Suchraums ist die Wahrscheinlichkeit kleiner, einen existierenden noch günstigeren Parametersatz für die verwendete Reglerstruktur übersehen zu haben. Die länger andauernde Breitensuche wird durch Abbildung 7.16 (b), verglichen mit Abbildung 7.16 (a), deutlich. Bei 14 der 24 zu optimierenden Parameter kann bei Einsatz der Strategie ES_{GBT}^{unsymm} eine gegenüber der Strategie ES_{std} länger andauernde Breitensuche beobachtet werden. In Abbildung 7.17 ist für einen der Parameter beispielhaft der Mittelwert und die Streuung während des Optimierungsprozesses dargestellt. Auch durch diese Abbildung lässt sich die getroffene Aussage untermauern. Außerdem wurde eine Verringerung der Optimierungsdauer (bei Verwendung des o. g. qualitativen Maßes für die Schnelligkeit des Optimierungsprozesses) um 16.5% erreicht.

Bei Einsatz der Strategie ES_{hybrid} wurde als bester Güterwert nach insgesamt 16000 Bewertungen mittels des Gütermaßes ein Wert von $Q_{Zeit} = 2.11$ erreicht. Auch hier wird ein additiver Güter-Schwellwert von $\Delta Q_{Zeit} = 0.5$ verwendet. Der Güterwert von $Q_{Zeit} = 2.61$ ist nach 890 Bewertungen mittels des Gütermaßes erreicht. Auch hier konnte kein wesentlich besserer Güterwert als beim Einsatz der Strategien ES_{std} und ES_{GBT}^{unsymm} erreicht werden, aber die – wie bei der Strategie ES_{GBT}^{unsymm} – länger andauernde Breitensuche kann auch hier beobachtet werden (s. Abbildung 7.16 (c)). Außerdem wurde der

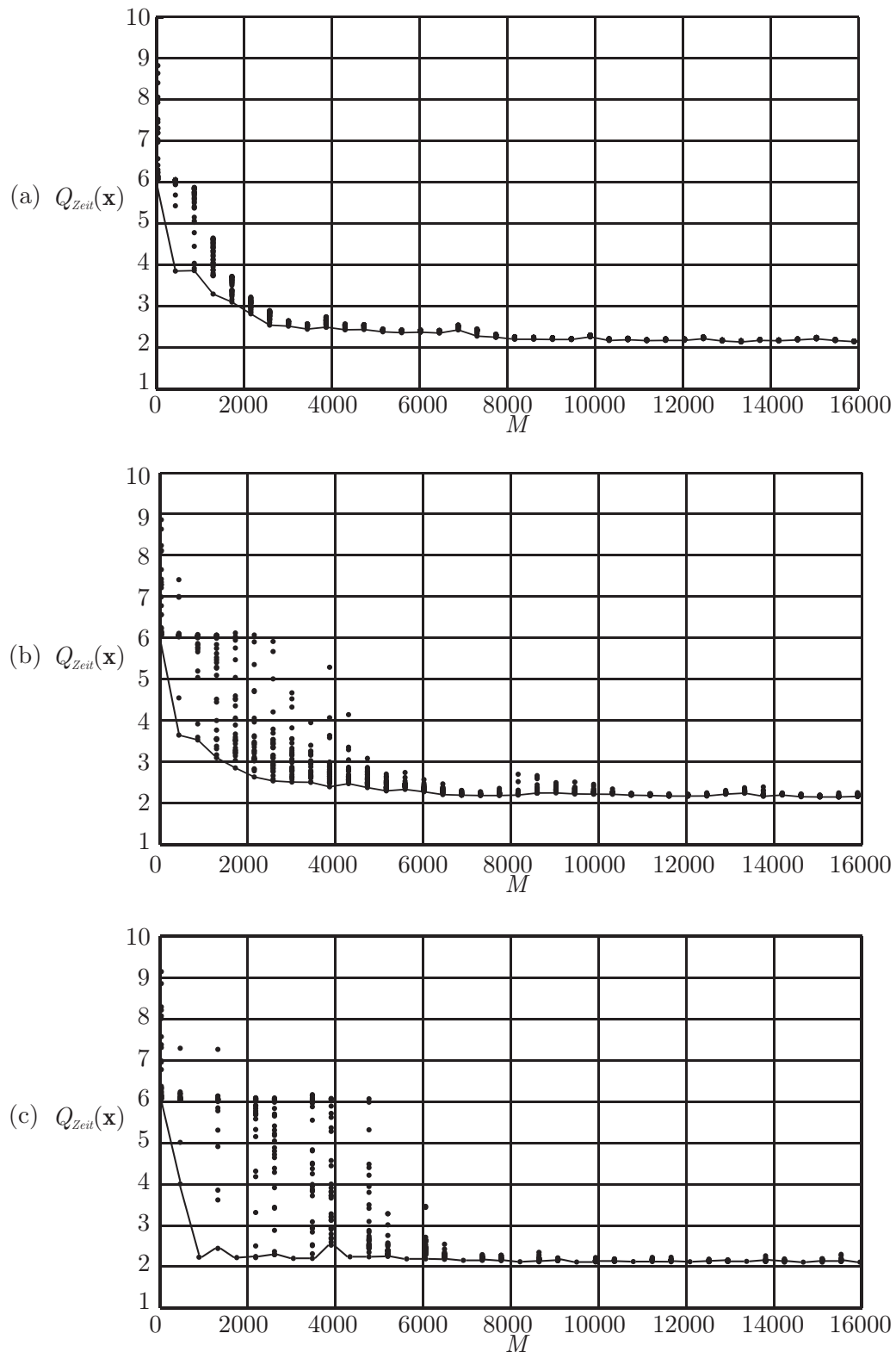


Abbildung 7.15: Gütewerte aller Eltern-Individuen (a) und (b) bzw. Gütewerte aller Eltern-Individuen und der Endpunkte aller Hooke-Jeeves-Teiloptimierungen (c) über der Anzahl der Bewertungen durch das Gütemaß Q_{Zeit} , Strategie ES_{std} (a), Strategie ES_{GBT}^{unsymm} (b), Strategie ES_{hybrid} (c).

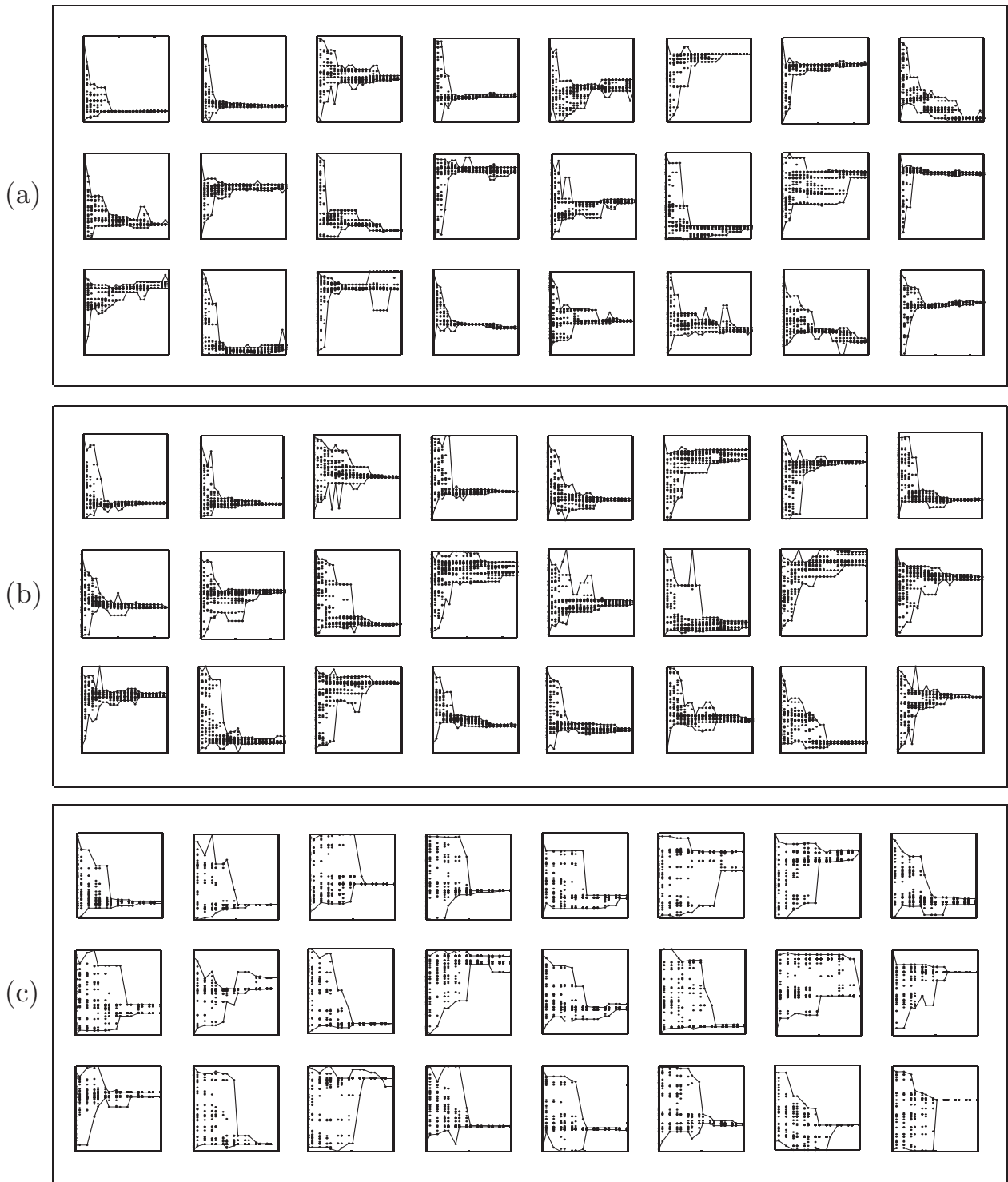


Abbildung 7.16: Darstellung der 24 einzelnen Parameter der Individuen in einem jeweils normierten Bereich über der Anzahl der Bewertungen durch das Gütemaß Q_{Zeit} , Strategie ES_{std} (a), Strategie ES_{GBT}^{unsymm} (b), Strategie ES_{hybrid} (c).

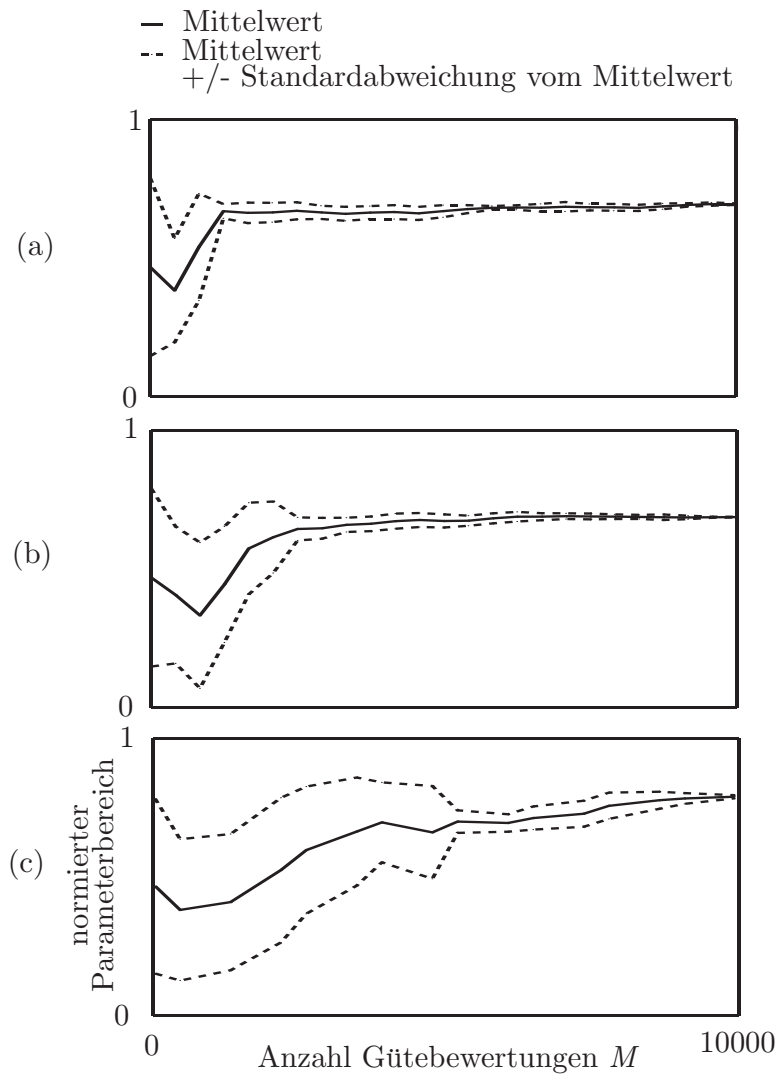


Abbildung 7.17: Beispielhafte Darstellung von Mittelwert und Streuung für den 7. zu optimierenden Parameter (vgl. Abbildung 7.16, jeweils 1. Zeile, 7. Parameter v. links) in einem jeweils normierten Bereich über der Anzahl der Bewertungen durch das Gütemaß Q_{Zeit} , Strategie ES_{std} (a), Strategie ES_{GBT}^{unsymm} (b), Strategie ES_{hybrid} (c).

Optimierungsprozess durch die Kombination des modifizierten Hooke-Jeeves-Verfahrens mit der Strategie ES_{GBT}^{unsymm} wesentlich beschleunigt. Gegenüber der Strategie ES_{GBT}^{unsymm} ergibt sich eine Verringerung der Optimierungsdauer um 59%. Verglichen mit der Strategie ES_{std} wurde sogar eine Verringerung um 66% erreicht (s. Abbildung 7.15 (c)).

Bei Verwendung der Strategie $HoJe_{Neustart}$ liegt der beste erreichte Gütewert nach insgesamt 16000 Gütebewertungen bei $Q_{Zeit} = 2.71$ (s. Abbildung 7.18). Auch hier wird ein Güte-Schwellwert von $\Delta Q_{Zeit} = 0.5$ zu diesem Wert hinzu addiert. Dieser Gütewert von $Q_{Zeit} = 3.21$ wird innerhalb der 16000 Bewertun-

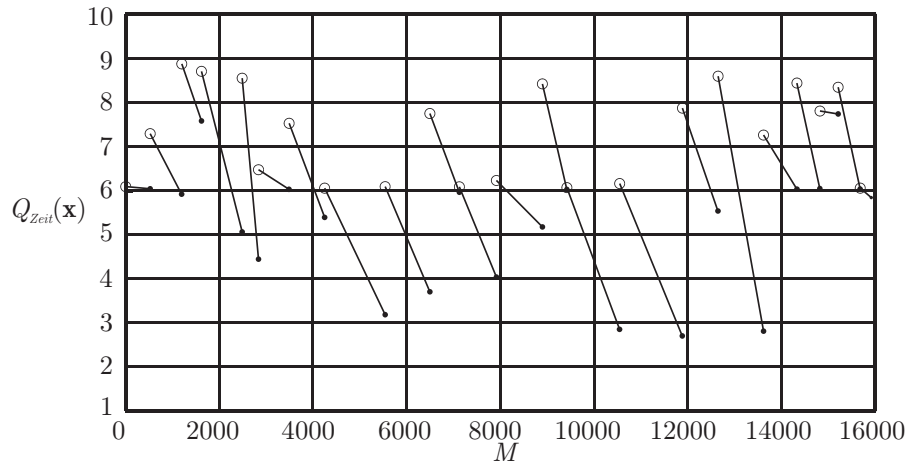
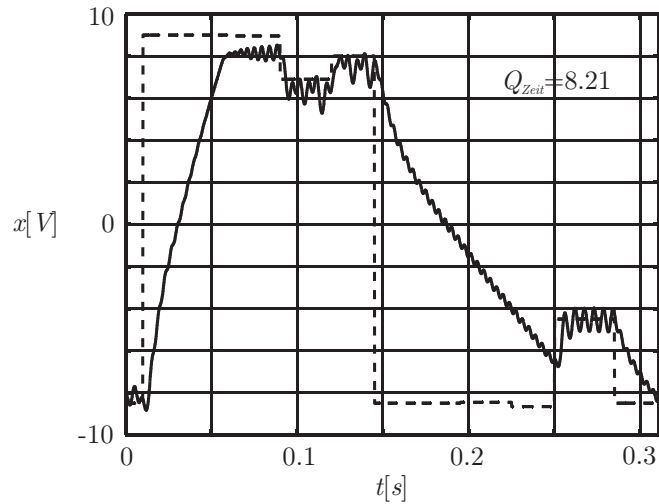


Abbildung 7.18: Güterwerte der Anfangs- und Endpunkte aller Hooke-Jeeves-Teiloptimierungen über der Anzahl der Bewertungen mittels des Gütemaßes Q_{Zeit} , Strategie $HoJe_{Neustart}$; \circ : Anfangspunkte, \bullet : Endpunkte (Hinweis: Die Verbindungslinien zwischen Anfangs- und Endpunkten dienen lediglich der grafischen Veranschaulichung und bedeuten *nicht*, dass zwischen Anfangs- und Endpunkten ein linearer Güteverlauf vorliegt).

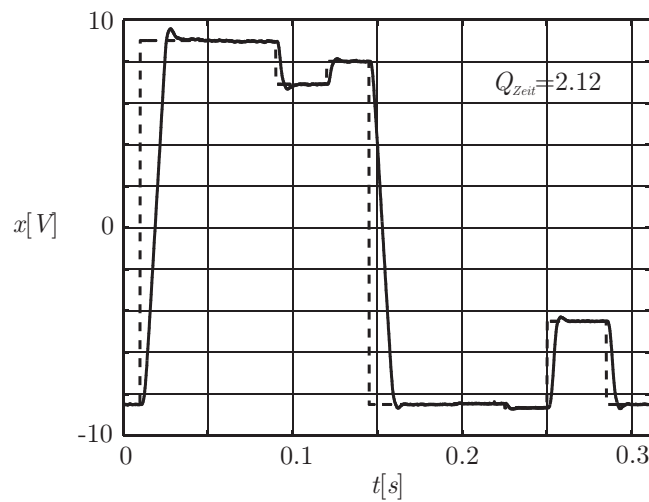
gen mittels des Gütemaßes insgesamt durch 4 Hooke-Jeeves-Teiloptimierungen erreicht, also durchschnittlich nach 4000 Gütebewertungen. Gegenüber der Strategie ES_{std} ergibt sich damit eine Verlängerung der Optimierungsdauer um den Faktor 1.53, gegenüber der Strategie ES_{GBT}^{unsymm} wird eine Verlängerung der Dauer um den Faktor 1.83 und, verglichen mit der Strategie ES_{hybrid} , sogar eine Verlängerung der Optimierungsdauer um den Faktor 4.49 ermittelt. Der beste erreichte Gütewert ist nach 16000 Gütebewertungen deutlich schlechter als bei Verwendung der Strategien ES_{std} , ES_{GBT}^{unsymm} und ES_{hybrid} . Bei längerer Laufzeit wurden aber auch vergleichbar gute Parametersätze gefunden. Um einen Gütewert von 2.2 zu erreichen, wurden beim Einsatz der Strategie $HoJe_{Neustart}$ im Schnitt ca. 18100 Bewertungen mittels des Gütemaßes benötigt.

Es ist also möglich, mit Hilfe der Strategie $HoJe_{Neustart}$ eine Regleroptimierung für das gegebene Optimierungsproblem durchzuführen. Die hierfür benötigte Dauer, d. h. die aufzubringende Anzahl an Gütebewertungen, ist aber deutlich größer als bei den Strategien ES_{std} , ES_{GBT}^{unsymm} und ES_{hybrid} .

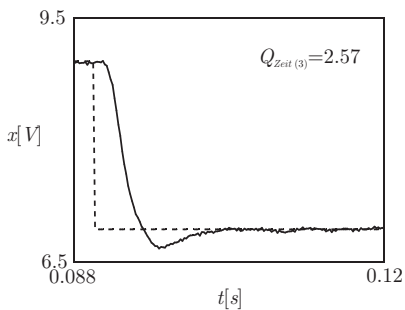
Die in Abschnitt 6.4 aufgestellte Vermutung, dass sich beim Einsatz des Optimierungsverfahrens $HoJe_{Neustart}$ zur Lösung realer Optimierungsprobleme Schwierigkeiten durch Rauschen in der Gütebewertung oder durch eine stark zerklüftete Gütelandschaft ergeben könnten, kann für das hier vorliegende Optimierungsproblem bestätigt werden.



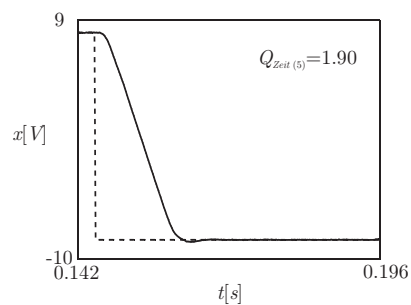
(a)



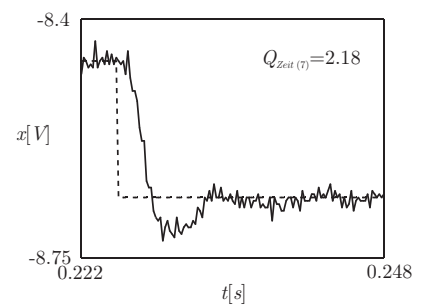
(b)



(c)



(d)



(e)

Abbildung 7.19: Beispiele für Sprungantworten unter Verwendung zweier Reglerparametersätze aus dem in Abbildung 7.15 (c) dargestellten Optimierungsprozess zu Beginn (a) und gegen Ende der Optimierung (b) (Zoom aus (b): Mittel-, Groß- und Kleinsignalsprung (c-e)); —: Ventilhub $x(t)$, - -: Sollwert $x_{soll}(t)$.

Kapitel 8

Zusammenfassung und Ausblick

Die Optimierung stellt einen wichtigen Ansatz zur Lösung zahlreicher wissenschaftlicher und technischer Problemstellungen dar. Zur Lösung eines solchen Optimierungsproblems steht eine Vielzahl von bekannten Optimierungsverfahren zur Verfügung. Eine Gruppe dieser Optimierungsverfahren sind die Evolutionären Algorithmen, wobei Evolutionsstrategien eine Variante der Evolutionären Algorithmen darstellen.

Im Rahmen dieser Arbeit werden neue Strategieelemente für die Evolutionsstrategie zur Milderung einiger Nachteile einer herkömmlichen Standard-Evolutionsstrategie vorgestellt.

Bei Standard-Evolutionsstrategien wird üblicherweise so vorgegangen, dass für die *Rekombination* (Fortpflanzung) die verwendeten Eltern-Individuen aus allen zur Verfügung stehenden Individuen der betrachteten Population zufällig gezogen werden. Bei der herkömmlichen *Selektion* wird in der Regel zur Lösung eines skalaren Optimierungsproblems ein Ranking aller Individuen entsprechend ihrer Gütwerte vorgenommen. Anschließend werden die besten aller Individuen für den weiteren Optimierungsprozess ausgewählt, die restlichen Individuen werden verworfen. Diese Vorgehensweise führt zu einer relativ schnellen Konzentration der Individuen im Suchraum der einzustellenden Parameter innerhalb weniger Generationen.

Beide Ansätze innerhalb der Standard-Evolutionsstrategien – sowohl für die Rekombination als auch für die Selektion – entsprechen nicht dem Vorbild der Natur. Dort ist die Rekombinations-Wahrscheinlichkeit ähnlichkeitsabhängig, d. h. es können nur Individuen miteinander gekreuzt werden, die eine hohe genetische Ähnlichkeit aufweisen. Auch eine übergeordnete Autorität, die alle Individuen bzw. alle Gütwerte kennt und dann die besten aller Individuen für den weiteren Optimierungsprozess auswählt, entspricht nicht dem Vorbild der Natur, wo eine dezentrale Selektion und somit keine gleich starke Konkurrenz

‘jeder gegen jeden’ auftritt. In dieser Arbeit werden neue, aus diesen ‘Naturpatienten’ entstandene Strategie-Ideen zur Verbesserung der Evolutionsstrategie entwickelt. Ferner wird ein vorteilhaftes hybrides Optimierungskonzept unter Einbeziehung eines lokal arbeitenden Optimierungsverfahrens vorgestellt.

Bei der hier entwickelten Evolutionsstrategie mit wechselnder Breiten- und Tiefensuche (ES_{WBT}) werden für die *Rekombination* bevorzugt Individuen verwendet, die im Suchraum nur einen geringen räumlichen Abstand, also eine hohe genetische Ähnlichkeit, aufweisen. Dadurch wird eine Tiefensuche in der näheren Umgebung bereits vorhandener Individuen realisiert. Bei der *Selektion* wird eine Selektion durch Turniere durchgeführt. Hierbei wird für ein Individuum bevorzugt ein Turniergegner mit großer Ähnlichkeit bestimmt. Bei jedem Turnier wird ein Gütevergleich der Turnierteilnehmer vorgenommen und der ‘Verlierer’ wird aus dem Optimierungsprozess eliminiert. Durch die Selektion wird also eine Breitensuche durchgeführt, denn Individuen, die im Suchraum von einem Individuum weit entfernt liegen, werden mit großer Wahrscheinlichkeit nicht als Turniergegner für dieses Individuum herangezogen und bleiben somit zunächst in der Population. Insgesamt wird durch diese Strategie ES_{WBT} eine gegenüber einer Standard-Evolutionsstrategie stärkere Breitensuche erreicht und somit die Gefahr einer zu frühen Konvergenz des Optimierungsprozesses gegen ein lokales Optimum verringert.

Eine verbesserte Variante der neu konzipierten Evolutionsstrategie – Evolutionsstrategie mit gleichzeitiger Breiten- und Tiefensuche (ES_{GBT}) – besteht darin, dass in Abhängigkeit von der bereits erreichten relativen Güte eine Umschaltung von Breiten- auf Tiefensuche vorgenommen wird. Dabei wird bei jedem *Rekombinationsschritt* zunächst zufällig eines der Eltern-Individuen gezogen. Bei einem vergleichsweise schlechten Individuum wird damit bevorzugt in der Breite des Suchraums nach besseren Lösungen gesucht. Dann wird bevorzugt ein Partner für die Rekombination mit geringer genetischer Ähnlichkeit, also mit großem räumlichen Abstand, bestimmt. Wurde hingegen als erstes ein relativ gutes Eltern-Individuum gezogen, so wird auf Tiefensuche umgeschaltet, um in der näheren Umgebung dieses guten Individuums nach noch besseren Lösungen zu suchen. Dann wird mit großer Wahrscheinlichkeit ein Rekombinationspartner mit großer Ähnlichkeit zu dem ersten Eltern-Individuum gezogen. Bei der *Selektion* wird für jedes Turnier ebenfalls zufällig der erste Turnierteilnehmer bestimmt. Weist dieser einen relativ schlechten Gütewert auf, so wird ebenfalls bevorzugt eine Breitensuche durchgeführt. Es wird dann bevorzugt ein Turniergegner mit großer Ähnlichkeit gezogen. Dadurch bleiben weiter entfernte Individuen mit größerer Wahrscheinlichkeit in der Population. Hat der erste

Turnierteilnehmer einen vergleichsweise guten Gütewert, so wird auf Tiefensuche umgeschaltet, indem bevorzugt ein Individuum mit geringer Ähnlichkeit mit dem ersten Turnierteilnehmer zum Turnier herangezogen wird. Durch die Strategie ES_{GBT} wird eine – vor allem zu Beginn eines Optimierungsprozesses – gegenüber einer Standard-Evolutionsstrategie länger andauernde Breitensuche im Suchraum erzielt.

Durch eine geeignete Kombination der Evolutionsstrategie mit dem Hooke-Jeeves-Verfahren und eine darin integrierte Schrittweitensteuerung wurde ein Kompromiss zwischen einem global suchenden Optimierungsverfahren und einem lokal arbeitenden Verfahren geschlossen. Dieses hybride Optimierungskonzept (ES_{hybrid}) zur Lösung eines skalaren Optimierungsproblems wurde für ein breites Anwendungsfeld ausgelegt. Häufig ist vorab nicht bekannt, ob zur Lösung eines Optimierungsproblems ein global oder ein lokal suchendes Optimierungsverfahren besser geeignet ist. Ein global arbeitendes Verfahren hat den Nachteil einer vergleichsweise langen Optimierungsdauer, der Vorteil besteht aber in der global angelegten Suche nach einem Optimum. Bei einem lokal suchenden Verfahren besteht der Nachteil hingegen in einer nur lokalen Suche in einem kleinen Bereich des Suchraums, der Vorteil liegt aber in einer zielgerichteten und somit schnelleren Suche. Die Schlüsselidee besteht darin, abwechselnd eine Generation der Evolutionsstrategie und einen Optimierungszyklus des besten Individuums mit dem Hooke-Jeeves-Verfahren vorzunehmen, bis sich keine weitere Verbesserung mehr einstellt. Der Grund einer solchen Stagnation des Optimierungsprozesses liegt vermutlich darin begründet, dass sich eine Konvergenz gegen ein lokales oder ein globales Optimum eingestellt hat. Hierbei ist normalerweise nicht bekannt, ob es sich um ein globales oder nur um ein lokales Optimum handelt. Deshalb wechselt sich *nun* eine Generation der Evolutionsstrategie jeweils einmal mit dem Hooke-Jeeves-Verfahren und einmal mit einer Vergrößerung der Schrittweite des besten Individuums ab. Falls tatsächlich ein globales Optimum gefunden wird, so kann dieses eventuell durch das Hooke-Jeeves-Verfahren genauer ermittelt werden. Wird hingegen nur ein lokales Optimum erreicht, so kann die Schrittweitenvergrößerung dafür sorgen, dass dieses lokale Optimum wieder verlassen werden kann. Das Optimierungsverfahren ES_{hybrid} weist gegenüber den jeweils einzelnen Verfahren deutlich erweiterte Anwendungsgrenzen auf. Das Verfahren stellt ebenfalls einen wichtigen Schritt in Richtung des übergeordneten Ziels dar, ein Optimierungsverfahren zu schaffen, das für einen Anwender möglichst einfach und universell einsetzbar ist, ohne tief gehende Vorkenntnisse über Optimierungsverfahren oder die Beschaffenheit der zu lösenden Optimierungsaufgabe bzw. über die Topologie der Gütelandschaft zu erfordern.

Die Leistungsfähigkeit der in dieser Arbeit entwickelten Strategieelemente wurde zunächst anhand ausgewählter Testfunktionen als skalare Gütemaße nachgewiesen.

Zur Lösung der praxisbezogenen Optimierungsaufgabe der Regleroptimierung für ein industrielles Hydraulikventil wurden die Optimierungsstrategien ES_{GBT} und ES_{hybrid} erfolgreich zur skalaren Optimierung eingesetzt. Das Ziel eines besseren Kompromisses zwischen Breiten- und Tiefensuche wurde durch die Konzeption der neuen Strategieelemente erreicht. Bei Verwendung der Optimierungsstrategie ES_{hybrid} konnte außerdem eine wesentliche Beschleunigung des Optimierungsprozesses erzielt werden. Dies führt zu einer deutlichen Zeitersparnis.

Weitere hier durchgeführte Arbeiten sind zunächst umfangreiche Untersuchungen bezüglich Strukturmodifikationen der Reglerstruktur für die Lageregelung des Hydraulikventils. Die in Abschnitt 7.3 vorgestellte Reglerstruktur wurde in [Bre02] zu Analyse Zwecken geringfügig modifiziert und zunächst durch einen Block zur Vorsteuerung erweitert. Diese Vorsteuerung berücksichtigt die Hysterese des Ventils und dient der Entkopplung der statischen Genauigkeit bei der Positionierung des Ventilschiebers. Ferner wurde ein Block zur Generierung eines zusätzlichen Stellsignals, das zum Reglerausgang addiert wird, in geeigneter Weise parametrisiert und optimiert. Diese Maßnahmen verfolgen den Zweck, das Potenzial der Reglerstruktur aus Abschnitt 7.3 zu analysieren. Sowohl für die Optimierung der Reglerparameter als auch für die Optimierung des additiven Stellsignals wurde die neue Strategie ES_{GBT} erfolgreich eingesetzt.

In [Sch03] wurde das bisher eingesetzte skalare Gütemaß zur Bewertung der Regelgüte Q_{Zeit} dahingehend erweitert, dass neben der Gütebewertung im Zeitbereich auch eine Bewertung im Frequenzbereich erfolgt. Beide Gütekriterien Q_{Zeit} und $Q_{Frequenz}$ wurden zunächst nicht zu einem skalaren Gütemaß vereint, stattdessen wurden multikriterielle Optimierungen durchgeführt. Hierbei zeigte sich aber, dass beide Gütekriterien nicht gegenläufig sind, sondern sich offenbar gegenseitig unterstützen. Deshalb können in Zukunft Q_{Zeit} und $Q_{Frequenz}$ auch sinnvoll zu einem skalaren Gütemaß kombiniert werden.

Die hier vorgestellten Ergebnisse zur Regleroptimierung für das Hydraulikventil basieren auf einer Optimierung ohne hydraulische Last. Die dabei erzielten Ergebnisse sind zwar näherungsweise auf den im späteren industriellen Einsatz auftretenden Lastfall übertragbar, zur besseren Übertragbarkeit wurden aber die unter Last auftretenden Strömungskräfte in [Ste01] durch ein Zellenmodell nachgebildet. Zur Partitionierung des Zellenmodells (Festlegung

der Lage und Größe der einzelnen Zellen) wurde bei zunächst vorgegebenen Einflussgrößen eine Standard-Evolutionsstrategie eingesetzt. Für eine solche Modellierungsaufgabe sind die Vorteile der in dieser Arbeit vorgestellten neuen Strategieelemente ebenfalls nutzbar. In einem weiterführenden Ansatz wurden nicht die zuvor vorgegebenen, sondern neue, als relevant eingestufte Einflussgrößen für diese Strömungskraftnachbildung verwendet. Für diese Merkmalsselektion wurde ein Verfahren zur evolutionären Auswahl bzw. Generierung relevanter Einflussgrößen [SNK01b, SNK01a] eingesetzt. Auch im Rahmen dieser Arbeiten kam eine Standard-Evolutionsstrategie zum Einsatz. Auch hier können die neuen Strategieelemente der vorliegenden Arbeit Vorteile bieten.

In zukünftigen Arbeiten lassen sich die hier vorgestellten neuen Strategieelemente für die Evolutionsstrategie noch in vielerlei Hinsicht verfeinern und somit deren Anwendungsmöglichkeiten erweitern. Beispielsweise ist es möglich, das Konzept des Simulated Heating (s. Kapitel 4) auch für das Optimierungsverfahren ES_{hybrid} zu nutzen und somit die Anzahl der Gütebewertungen für das Hooke-Jeeves-Verfahren im Laufe des Optimierungsprozesses zu steigern. Eine Weiterentwicklung der vorgestellten Strategieelemente hinsichtlich Mechanismen, die zu einer größeren Robustheit der gefundenen Lösungen führen, ist ebenfalls wünschenswert. Ein interessanter Ansatz diesbezüglich wurde in [Sch02] erarbeitet.

Bezogen auf die dieser Arbeit zugrunde liegende praxisbezogene Optimierungsaufgabe der Regleroptimierung für ein Hydraulikventil erscheinen noch weitere Ziele lohnenswert. Bisher wurden skalare Optimierungen unter Verwendung des vorgestellten Gütemaßes Q_{Zeit} präsentiert. Hier bietet sich die Möglichkeit, Teilkriterien dieses Gütemaßes zur multikriteriellen Optimierung einzusetzen.¹ Insbesondere für diese Optimierungsaufgabe mit teilweise gegenläufigen Gütegesichtspunkten ist es sinnvoll, zu analysieren, inwieweit die neuen Strategieelemente auch bei einer größeren Anzahl an Gütekriterien erfolgreich zur multikriteriellen Optimierung eingesetzt werden können.

¹Multikriterielle Optimierungen anhand unterschiedlicher Ventile derselben Baureihe, die aber dennoch einige Unterschiede im Verhalten aufweisen, wurden seitens des Industriepartners bereits erfolgreich durchgeführt. Hierzu wurde jeweils das Gütemaß Q_{Zeit} verwendet. Somit kann ein auch für Grenztypen in der Ventilherstellung geeigneter, d. h. möglichst robuster Regler als Optimierungsergebnis ausgewählt werden.

Anhang A

Mathematische Symbole, Definitionsgleichungen und Abkürzungen

Tabelle A.1: Mathematische Symbole und deren Bedeutung.

Symbol	Kurzbeschreibung
\mathbf{a}	Vektor von Drehwinkeln für die Optimierung mit Korrelierter Mutation, \mathbf{a} und \mathbf{s} beinhalten die Strategieparameter eines Individuums \mathbf{x}
α	Variationsfaktor zur Verkleinerung der Schrittweite beim Hooke-Jeeves-Verfahren, $0 < \alpha < 1$
b	Faktor zur Variation des Schrittweitenvektors bei der Strategie ES_{hybrid}
$\gamma(\Delta x_{soll})$	Anpassungsfaktor zur Normierung der Ausregelzeit
$D(d)$	Wahrscheinlichkeitsfunktion für den Programmschritt Rekombination und für den Programmschritt Selektion in Abhängigkeit von $d(\mathbf{x}_h, \mathbf{x}_m)$
$d(\mathbf{x}_h, \mathbf{x}_m)$	Maß für den Abstand zweier Vektoren \mathbf{x}_h und \mathbf{x}_m
δ	Oszillationsmaß, Multiplikationsfaktor zur Modifizierung des ISE -Maßes
$E_{j,\bar{j}}$	Aussonderungsschritt bei der Selektion durch Turniere mit $j = 0, \dots, \hat{k} - 1$ und $\bar{j} = 1, \dots, \bar{j}_{max}$
e_i	Exponent zur Bildung einer skalaren Ersatzfunktion Q_{Ersatz} mit $i = 1, \dots, k$
ϵ	Schwellwert für das Hooke-Jeeves-Verfahren
F_M	Kraft des Magneten auf den Ventilschieber
$G_{[0...1]}$	gleichverteilte skalare Zufallsvariable im Bereich $[0 \dots 1]$
g_i	Gewichtungsfaktor zur Bildung einer skalaren Ersatzfunktion Q_{Ersatz} mit $i = 1, \dots, k$
H_j	kleinstmöglicher Hyperquader, der alle Individuen der Menge \mathcal{M}_j umschließt
ISE_{mod}	modifiziertes ISE -Maß: $ISE_{mod} = \delta \cdot ISE$
i	Magnetstrom
i_{soll}	Sollwert für den Magnetstrom i
Δi	Regelabweichung $\Delta i = i_{soll} - i$
\hat{j}	Index der Generation, $j = 0, \dots, \hat{k}$
$\hat{\bar{j}}$	Index für die Individuen der Menge $\bar{\mathcal{M}}_{j,\hat{\bar{j}}}$, $\hat{\bar{j}} = 1, \dots, \hat{k} - i + 1$
\tilde{j}	Index für das ausgewählte Individuum bei der Roulette-Wheel-Selektion
\bar{j}	Index für die Aussonderungsschritte $E_{j,\bar{j}}$ im Programmschritt Selektion bei der Selektion durch Turniere, $\bar{j} = 1, \dots, \bar{j}_{max}$

\bar{j}_{max}	Anzahl der Aussonderungsschritte $E_{j,\bar{j}}$ im Programmschritt Selektion bei der Selektion durch Turniere
\mathcal{K}_j	Menge von Vektoren \mathbf{x} vor dem Programmschritt Selektion mit $j = 0, \dots, \hat{k} - 1$
k	Anzahl der Gütekriterien bei vektoriellem Gütemaß $\mathbf{Q}(\mathbf{x}) = (Q_1(\mathbf{x}), \dots, Q_k(\mathbf{x}))$
\hat{k}	maximale Anzahl an Generationen bei der Evolutionsstrategie bzw. maximale Anzahl an Generationen und Hooke-Jeeves-Teiloptimierungen bei der Strategie ES_{hybrid}
\tilde{k}	Anzahl der Individuen in der Menge \mathcal{K}_j
l	Anzahl der Ungleichungen $u_i(\mathbf{x})$
λ	Anzahl der Individuen in der Menge \mathcal{M}_j^{rek} bzw. in der Menge \mathcal{M}_j^{mut}
M	Anzahl der Bewertungen mittels des Gütemaßes innerhalb eines Optimierungsprozesses
MP	Überschwingweite
$MP_{max}(\Delta x_{soll})$	maximal zulässige Überschwingweite
MP_{norm}	normierte Überschwingweite $MP_{norm} = \frac{MP}{MP_{max}(\Delta x_{soll})}$
\mathcal{M}_0	Startpopulation für die Evolutionsstrategie
\mathcal{M}_j	Menge von Vektoren \mathbf{x} nach dem Programmschritt Selektion mit $j = 1, \dots, \hat{k}$
\mathcal{M}_j^{mut}	Menge von Vektoren \mathbf{x} nach dem Programmschritt Mutation mit $j = 0, \dots, \hat{k} - 1$
\mathcal{M}_j^{rek}	Menge von Vektoren \mathbf{x} nach dem Programmschritt Rekombination mit $j = 0, \dots, \hat{k} - 1$
$\mathcal{M}_{j,\bar{j}}$	Menge von Vektoren \mathbf{x} nach der Teilmengenbildung bei der Selektion durch Turniere mit $j = 0, \dots, \hat{k} - 1$ und $\bar{j} = 1, \dots, \bar{j}_{max}$
$\bar{\mathcal{M}}_{j,i}$	Restmenge der noch nicht ausgewählten Individuen aus der Menge \mathcal{K}_j mit $j = 0, \dots, \hat{k} - 1$ und $i = 0, \dots, \mu$ bei Roulette-Wheel-Selektion und $i = \bar{j} = 0, \dots, \bar{j}_{max}$ bei Selektion durch Turniere
μ, μ^*	Anzahl der Eltern-Individuen in der Menge \mathcal{M}_j
μ, μ_1, μ_2	Zugehörigkeitsfunktion eines Fuzzy-Moduls
$N(a, b)$	normalverteilte, skalare Zufallsvariable mit Erwartungswert a und Standardabweichung b
n	Anzahl der einzustellenden Parameter bzw. Dimension des Suchraums S
P	Wahrscheinlichkeitsdichte-Funktion
$\mathcal{P}_{j,i}$	Menge der durch die Roulette-Wheel-Selektion ausgewählten Individuen \mathbf{x} mit $j = 0, \dots, \hat{k} - 1$ und $i = 1, \dots, \mu$
$Q(\mathbf{x})$	Gütwert des Vektors \mathbf{x} bei Verwendung eines skalaren Gütemaßes Q
$\mathbf{Q}(\mathbf{x})$	Gütwerte des Vektors \mathbf{x} bei Verwendung eines vektoriiellen Gütemaßes \mathbf{Q}
$Q_{Ersatz}(\mathbf{x})$	Gütwert des Vektors \mathbf{x} bei Verwendung einer skalaren Ersatzfunktion Q_{Ersatz} anstelle eines vektoriiellen Gütemaßes \mathbf{Q}
$Q_{Ersatz,min}$	optimaler Gütwert bei Verwendung einer skalaren Ersatzfunktion Q_{Ersatz} anstelle eines vektoriiellen Gütemaßes \mathbf{Q}
Q_{max}	maximaler Wert aller bisher innerhalb eines Optimierungsprozesses aufgetretenen Gütwerte bei Verwendung eines skalaren Gütemaßes Q
Q_{min}	minimaler Wert aller bisher innerhalb eines Optimierungsprozesses aufgetretenen Gütwerte bei Verwendung eines skalaren Gütemaßes Q
Q_{opt}	derzeit bester Gütwert bei Verwendung eines skalaren Gütemaßes Q
$Q_{Zeit}(\mathbf{x})$	Gütwert im Zeitbereich
$Q_{Zeit(i,j)}(\mathbf{x})$	Gütwert im Zeitbereich des j . Sprunges und der i . Aufschaltung des Eingangsgößenverlaufs
$Q_{Zeit(j)}(\mathbf{x})$	Mittelwert aller Gütwerte im Zeitbereich des j . Sprunges des Eingangsgößenverlaufs

q	Faktor zur Berechnung des Oszillationsmaßes δ
ρ	Schwellwert für das Gütemaß, Erreichen dieses Wertes ist Abbruchkriterium für den Optimierungsprozess
S	Suchraum: Raum, in dem zulässige Vektoren \mathbf{x} liegen
s	skalare Schrittweite für den Optimierungsprozess, s wird \mathbf{x} zugeordnet
$\mathbf{s}, \mathbf{s}_{neu}$	Schrittweitenvektor für den Optimierungsprozess, \mathbf{s} wird \mathbf{x} zugeordnet
s_{min}	untere Grenze der skalaren Schrittweite s
\mathbf{s}_{max}	Vektor der oberen Grenzen der Schrittweiten
\mathbf{s}_{min}	Vektor der unteren Grenzen der Schrittweiten
σ	Breite der Wahrscheinlichkeitsfunktion $D(d)$
$\sigma_{NORM\ i}$	normierte Standardabweichung zur Bestimmung der Start-Schrittweite eines Individuums in der i -ten Komponente mit $i = 1, \dots, n$
$\sigma_{Q_{Zeit\ (j)}}$	Standardabweichung aller $Q_{Zeit\ (j)}$
t	Zeit
T_{an}	Anregelzeit
$T_{an\ norm}$	normierte Anregelzeit $T_{an\ norm} = \frac{T_{an}}{T_{an\ opt}(\Delta x_{soll})}$
$T_{an\ opt}(\Delta x_{soll})$	für die Sprunghöhe Δx_{soll} schnellstmögliche Anregelzeit
T_{aus}	Ausregelzeit
$T_{aus\ norm}$	normierte Ausregelzeit $T_{aus\ norm} = \frac{T_{aus}}{\gamma(\Delta x_{soll})}$
T_{ISE}	Zeitspanne zwischen dem 2. Nulldurchgang der Regelabweichung $\Delta x(t)$ und dem Ausregelzeitpunkt $T_{an} + T_{aus}$
$u_i(\mathbf{x})$	Grenzen der einzustellenden Parameter sind durch Ungleichungen $u_i(\mathbf{x}) \leq 0$ mit $i = 1, \dots, l$ gegeben
u	Ausgangsgröße eines Fuzzy-Moduls
u_{COS}	Schwerpunkt der Singletons eines Fuzzy-Moduls
$u_D, u_D^{COG}, u_D^{max}$	Ausgangsgrößenwert eines Fuzzy-Moduls
u_i	Stellgröße für den Magnetstrom i
u_x	Stellgröße für den Ventilhub $x(t)$
\mathcal{X}	Menge von Vektoren \mathbf{x} , insbesondere Pareto-Menge im Raum der einzustellenden Parameter
$x(t)$	Position des Ventilschiebers (Ventilhub)
$x_{soll}(t)$	Sollwert für die Position des Ventilschiebers $x(t)$
$\Delta x(t)$	Regelabweichung $\Delta x(t) = x_{soll}(t) - x(t)$
Δx_{soll}	Sprunghöhe $\Delta x_{soll} = x_{soll\ nach\ Sprung} - x_{soll\ vor\ Sprung} $
\mathbf{x}	Vektor der einzustellenden Parameter bzw. Individuum bei der Evolutionsstrategie $\mathbf{x} = (x_1, \dots, x_n)$
\mathbf{x}^*	globales Optimum bzw. Element der Pareto-Menge \mathcal{X}
\mathbf{x}_{max}	Vektor der oberen Grenzen der einzustellenden Parameter
\mathbf{x}_{min}	Vektor der unteren Grenzen der einzustellenden Parameter
\mathbf{x}_{opt}	derzeit am besten bewertetes Individuum
\mathbf{x}_{ref}	Referenzindividuum in einem Rekombinationsschritt bzw. in einem Aussonderungsschritt
z	skalare Zufallsvariable
\mathbf{z}	vektorielle Zufallsvariable

Tabelle A.2: Definitionsgleichungen und deren Bedeutung.

Definitionsgleichung	Kurzbeschreibung
$\ \mathbf{x}_h - \mathbf{x}_m\ _1 = \sum_{i=1}^n x_{i,h} - x_{i,m} $	Betragssummennorm als Abstand der Vektoren \mathbf{x}_h und \mathbf{x}_m
$\ \mathbf{x}_h - \mathbf{x}_m\ _2 = \sqrt{\sum_{i=1}^n x_{i,h} - x_{i,m} ^2}$	Euklidischer Abstand der Vektoren \mathbf{x}_h und \mathbf{x}_m

Tabelle A.3: Abkürzungen und deren Bedeutung.

Abkürzung	Kurzbeschreibung
CAN-Bus	Controller-Area-Network-Bus (engl.): Bus in einem regelungstechnischen Netzwerk
CDF	Kumulative Dichtefunktion
ES	Evolutionsstrategie
ES_{hybrid}	Hybrides Optimierungsverfahren aus ES und Hooke-Jeeves-Verfahren
ES_{std}	Standard-ES, zu Vergleichszwecken herangezogen
ES_{GBT}	ES mit gleichzeitiger Breiten- und Tiefensuche
ES_{GBT}^{symm}	ES mit gleichzeitiger Breiten- und Tiefensuche und symmetrischer Wahrscheinlichkeitsdichte-Funktion für die Auswahl der Individuen bei der Rekombination und bei der Selektion
ES_{GBT}^{unsymm}	ES mit gleichzeitiger Breiten- und Tiefensuche und unsymmetrischer Wahrscheinlichkeitsdichte-Funktion für die Auswahl der Individuen bei der Rekombination und bei der Selektion
ES_{WBT}	ES mit wechselnder Breiten- und Tiefensuche
$ES_{WBT}^{T_{const}}$	ES mit wechselnder Breiten- und Tiefensuche und fester Laufzeit
$ES_{WBT}^{T_{var}}$	ES mit wechselnder Breiten- und Tiefensuche und variabler Laufzeit
FM	Fuzzy-Modul, FM 1: Fuzzy-Modul zur Bewertung einer Sprungantwort mit der Schulnoten-Skala, FM 2: Fuzzy-Modul zur zusätzlichen Differenzierung aller mit 6 benoteten Sprungantworten
$HoJe_{Neustart}$	modifiziertes Hooke-Jeeves-Verfahren mit Neustart
ISE	integral of squared error (engl): Integral über die quadrierte Regelabweichung
OBE-D2	On-Board-Elektronik, digital, 2. Generation

Anhang B

Regelbasis und Zugehörigkeitsfunktionen Fuzzy-Gütemaß

Tabelle B.1: Regelbasis FM 1.

WENN <Prämisse>	DANN <Konklusion>
WENN <Prämisse>	DANN <Konklusion> VERBOTEN
positive Regeln:	
WENN $T_{an\ norm}$ = sehr schnell	DANN Ausgang = Note 1
WENN $T_{an\ norm}$ = schnell	DANN Ausgang = Note 2
WENN $T_{an\ norm}$ = mittel	DANN Ausgang = Note 3
WENN $T_{an\ norm}$ = langsam	DANN Ausgang = Note 4
WENN $T_{an\ norm}$ = sehr langsam	DANN Ausgang = Note 5
WENN $T_{an\ norm}$ = zu langsam	DANN Ausgang = Note 6
WENN MP_{norm} = kein	DANN Ausgang = Note 1
WENN MP_{norm} = sehr gering	DANN Ausgang = Note 2
WENN MP_{norm} = gering	DANN Ausgang = Note 3
WENN MP_{norm} = mittel	DANN Ausgang = Note 4
WENN MP_{norm} = hoch	DANN Ausgang = Note 5
WENN MP_{norm} = sehr hoch	DANN Ausgang = Note 6
WENN $T_{aus\ norm}$ = sehr schnell	DANN Ausgang = Note 1
WENN $T_{aus\ norm}$ = schnell	DANN Ausgang = Note 2
WENN $T_{aus\ norm}$ = mittel	DANN Ausgang = Note 3
WENN $T_{aus\ norm}$ = langsam	DANN Ausgang = Note 4
WENN $T_{aus\ norm}$ = sehr langsam	DANN Ausgang = Note 5
WENN $T_{aus\ norm}$ = zu langsam	DANN Ausgang = Note 6
WENN ISE_{mod} = kaum	DANN Ausgang = Note 1
WENN ISE_{mod} = mittel	DANN Ausgang = Note 4
WENN ISE_{mod} = sehr hoch	DANN Ausgang = Note 6
negative Regeln:	
WENN $T_{an\ norm}$ = sehr langsam	DANN Ausgang = zu gut VERBOTEN
WENN $T_{an\ norm}$ = zu langsam	DANN Ausgang = zu gut VERBOTEN
WENN $T_{an\ norm}$ = zu langsam	DANN Ausgang = zu gut VERBOTEN
WENN MP_{norm} = hoch	DANN Ausgang = zu gut VERBOTEN

WENN MP_{norm} = sehr hoch	DANN Ausgang = zu gut VERBOTEN
WENN MP_{norm} = sehr hoch	DANN Ausgang = zu gut VERBOTEN
WENN $T_{aus\ norm}$ = sehr langsam	DANN Ausgang = zu gut VERBOTEN
WENN $T_{aus\ norm}$ = zu langsam	DANN Ausgang = zu gut VERBOTEN
WENN $T_{aus\ norm}$ = zu langsam	DANN Ausgang = zu gut VERBOTEN
WENN ISE_{mod} = sehr hoch	DANN Ausgang = zu gut VERBOTEN
WENN ISE_{mod} = sehr hoch	DANN Ausgang = zu gut VERBOTEN

Tabelle B.2: Regelbasis FM 2.

WENN <Prämisse>	DANN <Konklusion>
positive Regeln:	
WENN $T_{an\ norm}$ = nicht zu schlecht	DANN Ausgang = 0 Strafpunkte
WENN $T_{an\ norm}$ = zu langsam	DANN Ausgang = 6 Strafpunkte
WENN $T_{an\ norm}$ = viel zu langsam	DANN Ausgang = 10 Strafpunkte
WENN $T_{an\ norm}$ = ultra langsam	DANN Ausgang = 20 Strafpunkte
WENN MP_{norm} = nicht zu schlecht	DANN Ausgang = 0 Strafpunkte
WENN MP_{norm} = sehr hoch	DANN Ausgang = 6 Strafpunkte
WENN MP_{norm} = zu hoch	DANN Ausgang = 10 Strafpunkte
WENN MP_{norm} = viel zu hoch	DANN Ausgang = 20 Strafpunkte
WENN $T_{aus\ norm}$ = nicht zu schlecht	DANN Ausgang = 0 Strafpunkte
WENN $T_{aus\ norm}$ = zu langsam	DANN Ausgang = 6 Strafpunkte
WENN $T_{aus\ norm}$ = viel zu langsam	DANN Ausgang = 10 Strafpunkte
WENN $T_{aus\ norm}$ = ultra langsam	DANN Ausgang = 20 Strafpunkte
WENN ISE_{mod} = nicht zu schlecht	DANN Ausgang = 0 Strafpunkte
WENN ISE_{mod} = sehr hoch	DANN Ausgang = 8 Strafpunkte
WENN ISE_{mod} = zu hoch	DANN Ausgang = 20 Strafpunkte

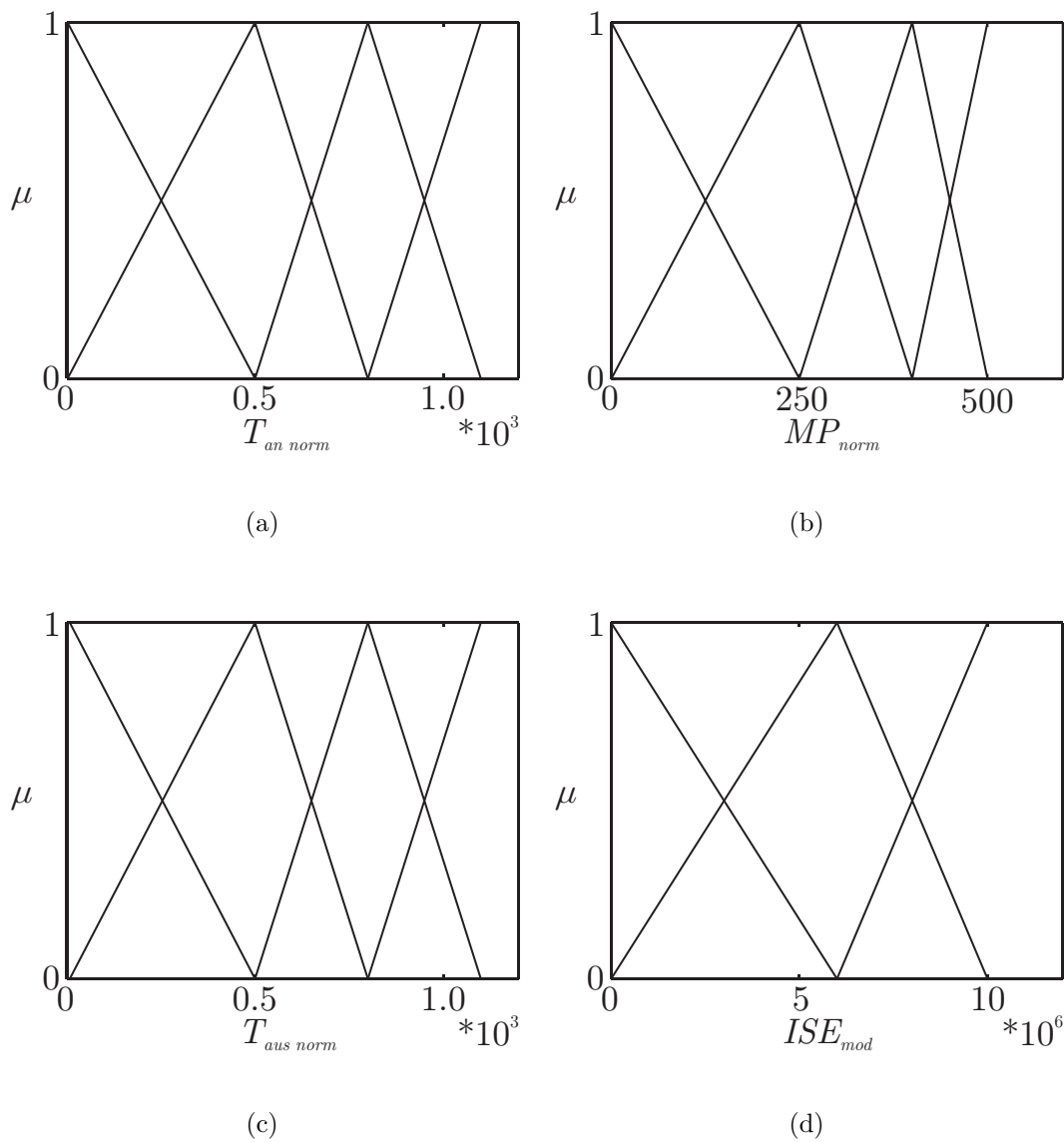


Abbildung B.1: Zugehörigkeitsfunktionen der Eingangsgrößen $T_{an\ norm}$ (a), MP_{norm} (b), $T_{aus\ norm}$ (c), ISE_{mod} (d) für FM 2.

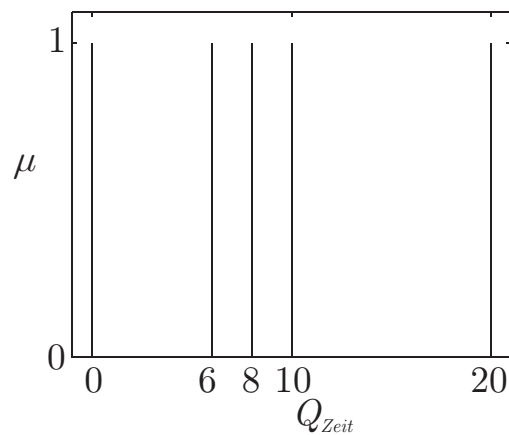


Abbildung B.2: Zugehörigkeitsfunktionen der Ausgangsgröße Q_{Zeit} für FM 2.

Literaturverzeichnis

- [AL95] ARNOLD, E. ; LINKE, H.: Anwendung Genetischer Algorithmen zur optimalen Steuerung von Pumpen in Trinkwasserversorgungssystemen. In: *at – Automatisierungstechnik* 43 (1995), Nr. 3, S. 142–147
- [BBTZ04] BAMBHA, N. K. ; BHATTACHARYYA, S. S. ; TEICH, J. ; ZITZLER, E.: Systematic Integration of Parameterized Local Search into Evolutionary Algorithms. In: *IEEE Transactions on Evolutionary Computation* 8 (2004), Nr. 2, S. 137–155
- [Bäc94] BÄCK, T.: *Evolutionary Algorithms in Theory and Practice*, Fachbereich Informatik, Universität Dortmund, Diss., 1994
- [BHM⁺98] BEIERSDÖRFER, S. ; HESSER, J. ; MÄNNER, R. ; SCHULZ, A. ; WOLFRUM, J.: Optimierung von fluoreszierenden Farbstoffen. In: HAFNER, S. (Hrsg.): *Industrielle Anwendungen Evolutionärer Algorithmen*. München : Oldenbourg, 1998, S. 25–35
- [BHS97] BÄCK, T. ; HAMMEL, U. ; SCHWEFEL, H.-P.: Evolutionary Computation: Comments on the History and Current State. In: *IEEE Transactions on Evolutionary Computation* 1 (1997), Nr. 1, S. 3–17
- [Bül98] BÜLTERS, O.: *Optimierung der Lageregelung von Ventilschiebern in hydraulischen Regelventilen*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Studienarbeit, 1998
- [BL04] BUTTELMANN, M. ; LOHMANN, B.: Optimierung mit Genetischen Algorithmen und eine Anwendung zur Modellreduktion. In: *at – Automatisierungstechnik* 52 (2004), Nr. 4, S. 151–163
- [BLSK00] BLUMENDELLER, W. ; LEUTNER, V. ; SCHWANE, U. ; KIENDL, H.: Regleroptimierung mit Evolutionären Algorithmen. In: *Tagungsband 2. Internationales Fluidtechnisches Kolloquium in Dresden*

Bd. 1. Dresden : Dresdner Verein zur Förderung der Fluidtechnik e. V., 2000, S. 437–452

- [BNSW00] BÄCK, T. ; NAUJOKS, B. ; SCHÜTZ, M. ; WILLMES, L.: Tragflächen-Optimierung mittels Evolutionärer Algorithmen. In: *Tagungsband zur VDI/VDE-GMA- und GI-Tagung Computational Intelligence im industriellen Einsatz, Baden-Baden*. Düsseldorf : VDI Verlag, 2000. – (VDI Berichte 1526), S. 67–84
- [Bos01a] Elektrohydraulische Antriebe ziehen mit elektrischen Lösungen gleich. In: *O+P "Ölhydraulik und Pneumatik"* 45 (2001), Nr. 4. – Sonderdruck durch Robert Bosch GmbH, Geschäftsbereich Automationstechnik – Industriehydraulik, Stuttgart
- [Bos01b] Mit und ohne Bus. In: *fluid* (2001), April. – Sonderdruck durch Robert Bosch GmbH, Geschäftsbereich Automationstechnik – Industriehydraulik, Stuttgart
- [Bre02] BREITNER, T.: *Analyse und CI-basierte Weiterentwicklung der Reglerstruktur eines digital geregelten Hydraulikventils*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Diplomarbeit, 2002
- [Bro96] BROECKMANN, S.: *Implementierung von Optimierungsverfahren für eine blockorientierte Simulation in C++ unter MS-Windows*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Diplomarbeit, 1996
- [BS96] BÄCK, T. ; SCHWEFEL, H.-P.: Evolutionary Computation: An Overview. In: *Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC '96), Nayoya, Japan*. Piscataway, NJ, USA : IEEE-Press, 1996, S. 20–29
- [BT95] BLICKLE, T. ; THIELE, L.: A Comparison of Selection Schemes used in Genetic Algorithms / Computer Engineering and Communication Networks Lab, Swiss Federal Institute of Technology, ETH Zürich, Schweiz. 1995 (TIK Report Nr. 11). – Forschungsbericht
- [BW00] BURKHARDT-WEITZ, I.: *Untersuchung einer evolutionären Hardware-in-the-loop-Optimierung der Lageregelung für ein industrielles Hydraulikventil*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Diplomarbeit, 2000

- [CC99] COELLO COELLO, C. A.: A Comprehensive Survey of Evolutionary-Based Multiobjective Optimization Techniques. In: *Knowledge and Information Systems. An International Journal* 1 (1999), Nr. 3, S. 269–308
- [CCVVL02] COELLO COELLO, C. A. ; VAN VELDHUIZEN, D. A. ; LAMONT, G. B.: *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York, NY, USA : Kluwer Academic Publishers, 2002
- [CDFK03] CORNE, D. W. ; DEB, K. ; FLEMING, P. J. ; KNOWLES, J. D.: The Good of the Many Outweighs the Good of the One: Evolutionary Multi-Objective Optimization. In: *Connections. The Newsletter of the IEEE Neural Networks Society* 1 (2003), Nr. 1, S. 9–13
- [Deb01] DEB, K.: *Multi-Objective Optimization using Evolutionary Algorithms*. Chichester, Großbritannien : John Wiley & Sons, 2001
- [DS78] DIXON, L. C. W. ; SZEGÖ, G. P.: The Global Optimization Problem: An Introduction. In: DIXON, L. C. W. (Hrsg.) ; SZEGÖ, G. P. (Hrsg.): *Towards Global Optimization II*. Amsterdam, Niederlande : North-Holland Publishing Company, 1978
- [dSP01a] dSPACE GmbH: *DS 1103 Installation and Configuration Guide*. 2001. – (Handbuch)
- [dSP01b] dSPACE GmbH: *MLIB/MTRACE MATLAB-dSPACE Interface and Trace Libraries*. 2001. – (Handbuch)
- [dSP01c] dSPACE GmbH: *RTI and RTI-MP Implementation Guide*. 2001. – (Handbuch)
- [EHK⁺88] EWALD, R. ; HUTTER, J. ; KRETZ, D. ; LIEDHEGENER, F. ; SCHENKEL, W. ; SCHMITT, A.: *Der Hydraulik Trainer Band 2, Proportional- und Servoventil-Technik, "Lehr- und Informationsbuch über hydraulische Proportional- und Servoventile sowie elektronische Komponenten eingesetzt in Steuerungen und Regelkreisen"*. Lohr a. Main : Mannesmann Rexroth GmbH, 1988
- [Fle95] FLEISCHER, M.: Simulated Annealing: Past, Present and Future. In: *Proceedings of the 27th Conference on Winter Simulation (WSC '95), Arlington, Virginia, USA, 1995*, S. 155–161
- [FOW66] FOGEL, L. J. ; OWENS, A. J. ; WALSH, M. J.: *Artificial Intelligence through Simulated Evolution*. New York, NY, USA : John Wiley & Sons, 1966

- [Gol89] GOLDBERG, D. E.: *Genetic algorithms in search, optimization and machine learning*. Reading, USA : Addison-Wesley, 1989
- [GZZ88] GROSCHE, G. (Hrsg.) ; ZIEGLER, V. (Hrsg.) ; ZIEGLER, D. (Hrsg.): *Ergänzende Kapitel zu Bronstein Semendjajew Taschenbuch der Mathematik*. 5. Auflage. Thun : Harri Deutsch, 1988
- [GZZ91] GROSCHE, G. (Hrsg.) ; ZIEGLER, V. (Hrsg.) ; ZIEGLER, D. (Hrsg.): *Bronstein Semendjajew Taschenbuch der Mathematik*. 25. Auflage. Stuttgart : Teubner, 1991
- [Her98] HERDY, M.: Evolutionsstrategie mit subjektiver Selektion zur Optimierung von Kaffeemischungen. In: HAFNER, S. (Hrsg.): *Industrielle Anwendungen Evolutionärer Algorithmen*. München : Oldenbourg, 1998, S. 15–23
- [HJ61] HOOKE, R. ; JEEVES, T. A.: 'Direct Search' Solution of Numerical and Statistical Problems. In: *Journal of the Association for Computing Machinery* 8 (1961), Nr. 2, S. 212–229
- [Hol75] HOLLAND, J. H.: *Adaptation in Natural and Artificial Systems*. Ann Arbor, Michigan, USA : University of Michigan Press, 1975
- [Hor97] HORN, J.: Multicriterion decision making. In: BÄCK, T. (Hrsg.) ; FOGEL, D. B. (Hrsg.) ; MICHALEWICZ, Z. (Hrsg.): *Handbook of Evolutionary Computation*. New York, NY, USA and Bristol, Großbritannien : IOP Publishing Ltd and Oxford University Press, 1997, S. F1.9: 1–15
- [Kah91] KAHLERT, J.: *Vektorielle Optimierung mit Evolutionsstrategien und Anwendungen in der Regelungstechnik*. Düsseldorf : VDI Verlag, 1991. – (Fortschritt-Berichte VDI, Reihe 8, Nr. 234)
- [Kah95] KAHLERT, J.: Globale vektorielle Optimierung mit Evolutionsstrategien. In: *at – Automatisierungstechnik* 43 (1995), Nr. 3, S. 118–126
- [KGV83] KIRKPATRICK, S. ; GELATT, C. D. ; VECCHI, M. P.: Optimization by Simulated Annealing. In: *Science* 220 (1983), Nr. 4598, S. 671–680
- [Kie97] KIENDL, H.: *Fuzzy Control Methodenorientiert*. München : Oldenbourg, 1997

- [Kie99] KIENDL, H.: Design of Advanced Fuzzy Systems. In: YLINIEMI, L. (Hrsg.) ; JUUSO, E. (Hrsg.): *Proceedings of TOOLMET '99 Symposium – Tool Environments for Intelligent Systems, University of Oulu, Oulun, Finnland, 1999*, S. 57–76
- [Kie00] KIENDL, H.: Implizite Modellierung, inkrementeller Relevanzindex und Rauigkeitsmaß: neue Strategieelemente für die datenbasierte Modellierung. In: *Proceedings 10. Workshop Fuzzy Control des GMA-FA 5.22, Dortmund*. Karlsruhe : Forschungszentrum Karlsruhe, 2000. – Wissenschaftliche Berichte FZKA 6509, S. 1–14
- [KK97] KIENDL, H. ; KNICKER, R.: DORA FÜR WINDOWS 6.2. Dortmund: Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, 1997. – (Handbuch)
- [KNL⁺03] KIENDL, H. ; NICOLAUS, B. ; LAMMERS, K. ; BLUMENDELLER, W. ; STERNKE, M.: *Verfahren zur Erzeugung günstiger Einstellwerte für technische Prozesse*. 2003. – Patentanmeldung, DE 10302898.6
- [Kra02] KRAUSE, P.: *Datenbasierte Generierung von transparenten und genauen Fuzzy-Modellen für mehrdeutige Daten und komplexe Systeme*. Düsseldorf : VDI Verlag, 2002. – (Fortschritt-Berichte VDI, Reihe 10, Nr. 691)
- [Kro99] KRONE, A.: *Datenbasierte Generierung von relevanten Fuzzy-Regeln zur Modellierung von Prozesszusammenhängen und Bedienstrategien*. Düsseldorf : VDI Verlag, 1999. – (Fortschritt-Berichte VDI, Reihe 10, Nr. 615)
- [KRO⁺03] KÖNIG, W. (Hrsg.) ; ROMMELFANGER, H. (Hrsg.) ; OHSE, D. (Hrsg.) ; HOFMANN, M. (Hrsg.) ; SCHÄFER, K. (Hrsg.) ; KUHNLE, H. (Hrsg.) ; PFEIFER, A. (Hrsg.): *Taschenbuch der Wirtschaftsinformatik und Wirtschaftsmathematik*. 2. Auflage. Harri Deutsch, 2003
- [KS98] KURSAWE, F. ; SCHWEFEL, H.-P.: Optimierung mit Evolutionären Algorithmen. In: HAFNER, S. (Hrsg.): *Industrielle Anwendungen Evolutionärer Algorithmen*. München : Oldenbourg, 1998, S. 1–13
- [Lam02] LAMMERS, K.: *Erprobung und Weiterentwicklung einer dezentralen Evolutionsstrategie und Anwendung auf die Regelung von Hydraulikventilen*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Studienarbeit, 2002

- [Liu03] LIU, W.: *Analyse und experimentelle Untersuchung unterschiedlicher Optimierungsstrategien sowie Implementierung und praktischer Einsatz einer neuen Optimierungsstrategie*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Studienarbeit, 2003
- [Mat99a] The MathWorks, Inc.: *Real-Time Workshop*. 1999. – (Handbuch)
- [Mat99b] The MathWorks, Inc.: *MATLAB*. 1999. – (Handbuch)
- [Mat99c] The MathWorks, Inc.: *SIMULINK*. 1999. – (Handbuch)
- [MG95] MILLER, B. L. ; GOLDBERG, D. E.: *Genetic Algorithms, Tournament Selection and the Effects of Noise / Department of General Engineering, University of Illinois, Il, USA. 1995 (IlliGAL Report No. 95006)*. – Forschungsbericht
- [Mic04] *Evolution*. Microsoft Encarta Online-Enzyklopädie 2004. Mai 2004. – <http://encarta.msn.com>
- [Mül00] MÜLLER, U.: *Bussysteme in der Fluidtechnik*. In: *O+P "Ölhydraulik und Pneumatik"* 44 (2000), Nr. 6. – Sonderdruck durch Robert Bosch GmbH, Geschäftsbereich Automationstechnik – Industriehydraulik, Stuttgart
- [MRR⁺53] METROPOLIS, N. ; ROSENBLUTH, A. ; ROSENBLUTH, M. ; TELLER, A. ; TELLER, E.: *Equation of State Calculations by Fast Computing Machines*. In: *Journal of Chemical Physics* 21 (1953), S. 1087–1092
- [Nic99a] NICOLAUS, B.: *Analyse von Evolutionären Strategien zur Optimierung der Lageregelung von Ventilschiebern in Hydraulikventilen*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Diplomarbeit, 1999
- [Nic99b] NICOLAUS, B.: *Entwurf eines Fuzzy-Gütemaßes zur Bewertung und Optimierung der Regelungseigenschaften von Hydraulikventilen*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Studienarbeit, 1999
- [Nic04] NICOLAUS, B.: *Dezentrale Strategieelemente für eine Evolutionsstrategie und Anwendung auf die Regleroptimierung eines industriellen Hydraulikventils*. In: *38. Regelungstechnisches Kolloquium, Boppard*, 2004

- [NKBS01] NICOLAUS, B. ; KIENDL, H. ; BLUMENDELLER, W. ; SCHWANE, U.: Evolutionary Optimization of an Industrial Hydraulic Valve with the Help of a Fuzzy Performance-Index. In: *Proceedings of the 10th IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2001, Melbourne, Australien* Bd. 1. Piscataway, NJ, USA : IEEE-Press, 2001, S. 139–142
- [OGH94a] OSTERMEIER, A. ; GAWELCZYK, A. ; HANSEN, N.: A Derandomized Approach to Self Adaptation of Evolution Strategies. In: *Evolutionary Computation 2* (1994), Nr. 4, S. 369–380
- [OGH94b] OSTERMEIER, A. ; GAWELCZYK, A. ; HANSEN, N.: Step-size adaptation based on non-local use of selection information. In: DAVIDOR, Y. (Hrsg.) ; SCHWEFEL, H.-P. (Hrsg.) ; MÄNNER, R. (Hrsg.): *Parallel Problem Solving from Nature – PPSN IV*. Berlin : Springer, 1994, S. 189–198
- [Pap91] PAPAGEORGIOU, M.: *Optimierung*. München : Oldenbourg, 1991
- [Poh00] POHLHEIM, H.: *GEATbx: Genetic and Evolutionary Algorithm Toolbox for use with MATLAB*, 2000. – (Online-Handbuch, <http://www.geatbx.com/docu/index.html>)
- [Rec73] RECHENBERG, I.: *Evolutionsstrategie – Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Stuttgart : Frommann-Holzboog, 1973
- [Rob99] Robert Bosch GmbH, Geschäftsbereich Automationstechnik: *Elektrohydraulische Proportional- und Regelungssysteme*. 1999
- [Ros60] ROSENBROCK, H. H.: An Automatic Method for Finding the Greatest or Least Value of a Function. In: *The Computer Journal* 3 (1960), Nr. 3, S. 175–184
- [Rud92] RUDOLPH, G.: On Correlated Mutations in Evolution Strategies. In: MÄNNER, R. (Hrsg.) ; MANDERICK, B. (Hrsg.): *Parallel Problem Solving from Nature 2*. Amsterdam, Niederlande : Elsevier, 1992, S. 105–114
- [Sch77] SCHWEFEL, H.-P.: *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie*. Basel und Stuttgart : Birkhäuser, 1977
- [Sch78] SCHMITT, A.: *Der Hydraulik Trainer Band 1, Lehr- und Informationsbuch über die Hydraulik*. Würzburg : Vogel-Verlag, 1978

- [Sch87] SCHWEFEL, H.-P.: Collective Phenomena in Evolutionary Systems. In: *Preprints of the 31st Annual Meeting of the International Society for General System Research, Budapest, Ungarn* Bd. 2, 1987, S. 1025–1033
- [Sch95a] SCHMIDT, G.: Heuristische Such- und Optimierungsmethoden in der Automatisierungstechnik – Editorial des Herausgebers. In: *at – Automatisierungstechnik* 43 (1995), Nr. 3, S. 109
- [Sch95b] SCHWEFEL, H.-P.: *Evolution and optimum seeking*. New York, NY, USA : John Wiley & Sons, 1995
- [Sch02] SCHÄFER, S.: *Entwicklung und Erprobung von Methoden zur adaptiven Diskretisierung von Evolutionsstrategien zur Effizienzsteigerung und Bestimmung robuster Lösungen*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Diplomarbeit, 2002
- [Sch03] SCHEER, A.: *Entwurf eines Gütemaßes im Frequenzbereich zur Bewertung und Optimierung der Lageregelung von Hydraulikventilen*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Studienarbeit, 2003
- [SHF94] SCHÖNEBURG, E. ; HEINZMANN, F. ; FEDDERSEN, S.: *Genetische Algorithmen und Evolutionsstrategien*. Bonn : Addison-Wesley, 1994
- [Sla01] SLAWINSKI, T.: *Analyse und effiziente Generierung von relevanten Fuzzy-Regeln in hochdimensionalen Suchräumen*. Düsseldorf : VDI Verlag, 2001. – (Fortschritt-Berichte VDI, Reihe 10, Nr. 686)
- [SNK⁺00] SCHWANE, U. ; NICOLAUS, B. ; KIENDL, H. ; BLUMENDELLER, W. ; LEUTNER, V.: Fuzzy-Gütemaß zur evolutionären Optimierung eines industriellen hydraulischen Regelventils. In: *Tagungsband zur VDI/VDE-GMA- und GI-Tagung Computational Intelligence im industriellen Einsatz, Baden-Baden*. Düsseldorf : VDI Verlag, 2000. – (VDI Berichte 1526), S. 229–234
- [SNK01a] SCHAUTEN, D. ; NICOLAUS, B. ; KIENDL, H.: An Evolutionary Concept for Selecting Relevant Sets of Input Variables for Data-based Fuzzy Modeling. In: *Proceedings Eunate 2001, CD-ROM, Teneriffa, Spanien*, Elite Foundation, Aachen, Verlag Mainz, Aachen, 2001

- [SNK01b] SCHAUTEN, D. ; NICOLAUS, B. ; KIENDL, H.: Evolutionäres Konzept zur Selektion relevanter Merkmalsätze für die datenbasierte Fuzzy-Modellierung. In: *Proceedings 11. Workshop Fuzzy Control des GMA-FA 5.22, Universität Dortmund, 2001*, VDI/VDE GMA-FA 5.22, Wissenschaftliche Berichte Forschungszentrum Karlsruhe, 2001, S. 133–147
- [Spr95] SPRAVE, J.: Evolutionäre Algorithmen zur Parameteroptimierung. In: *at – Automatisierungstechnik* 43 (1995), Nr. 3, S. 110–117
- [Spr99] SPRAVE, J.: *Ein einheitliches Modell für Populationsstrukturen in Evolutionären Algorithmen*, Fachbereich Informatik, Universität Dortmund, Diss., 1999
- [Ste01] STERNKE, M.: *Datenbasierte Modellierung der Strömungskräfte eines industriellen Hydraulikventils für einen fuzzybasierten Reglerentwurf*, Lehrstuhl für Elektrische Steuerung und Regelung, Universität Dortmund, Diplomarbeit, 2001
- [SV01] SCHMITT, A. ; VERSTEGE, J.: Multikriterielle Optimierung auf Basis von Evolutionsstrategien zur Bereitstellung von Systemdienstleistungen. In: *Tagungsband zur VDI-Tagung Optimierung in der Energieversorgung, Veitshöchheim*. Düsseldorf : VDI Verlag, 2001. – (VDI Berichte 1627), S. 225–234
- [WM97] WOLPERT, D. ; MACREADY, W.: No Free Lunch Theorems for Optimization. In: *IEEE Transactions on Evolutionary Computation* 1 (1997), Nr. 1, S. 67–82
- [Zad65] ZADEH, L. A.: Fuzzy Sets. In: *Information and Control* 8 (1965), Nr. 3, S. 338–353
- [ZDT00] ZITZLER, E. ; DEB, K. ; THIELE, L.: Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. In: *Evolutionary Computation* 8 (2000), Nr. 2, S. 173–195
- [ZT99] ZITZLER, E. ; THIELE, L.: Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. In: *IEEE Transactions on Evolutionary Computation* 3 (1999), Nr. 4, S. 257–271

Lebenslauf

Persönliche Daten

Name: Barbara Ottersbach, geb. Nicolaus
Familienstand: verheiratet, ein Kind

Schulbildung

1985 – 1994 Galilei-Gymnasium, Hamm
05/1994 Abitur

Studium

1994 – 1999 Diplomstudiengang Elektrotechnik
an der Universität Dortmund
vor dem Studium bzw. während des Studiums
Praktika bei den Firmen
– Deutsche Bahn AG, Hamm
– Krupp Uhde GmbH, Dortmund
sowie Tätigkeiten als studentische Hilfskraft am
Lehrstuhl für Elektrische Steuerung und Regelung
12/1999 Diplom-Hauptprüfung

Berufliche Tätigkeit

01/2000 – 03/2000 Stipendiatin im Graduiertenkolleg
*Modellierung und modellbasierte Entwicklung
komplexer technischer Systeme*
am Lehrstuhl für Elektrische Steuerung und Regelung
der Universität Dortmund (Prof. Dr. rer. nat. H. Kiendl)
04/2000 – 03/2005 wissenschaftliche Angestellte am Lehrstuhl für
Elektrische Steuerung und Regelung
der Universität Dortmund
(Prof. em. Dr. rer. nat. H. Kiendl,
Lehrstuhlvertretung Dr. rer. nat. F. Hoffmann)