

**Dimensionierung von Analogschaltungen mit formalen Methoden
auf der Basis affiner Arithmetik**

Vom Fachbereich Informatik
der Universität Hannover
zur Erlangung des akademischen Grades
Doktor-Ingenieur
genehmigte Dissertation

von Dipl.-Ing. Andreas Lemke

geboren am 2. März 1974 in Gehrden

2004

1. Referent: Prof. Dr.-Ing. Erich Barke
2. Referent: Prof. Dr.-Ing. Hermann Eul
3. Referent: Prof. Dr.-Ing. Lars Hedrich
Tag der Promotion: 24. Februar 2004

Kurzfassung

Der Entwurf immer komplexerer Systeme auf einem Chip erfordert den Einsatz automatischer, rechnergestützter Werkzeuge. Besondere Bedeutung kommt heute dem Entwurf von Schaltungen mit analogen und digitalen Teilen zu. Trotz seines kleinen Anteils an der Gesamttransistorzahl erfordert der Analogteil meist eine deutlich höhere Entwurfszeit als der Digitalteil. Um diese Entwurfsproduktivitätslücke zu schließen, werden leistungsfähige Synthesewerkzeuge für den Analogschaltungsentwurf benötigt.

Die Dimensionierung ist ein wichtiges Teilproblem der Synthese von Analogschaltungen. Diese Arbeit stellt ein neuartiges Dimensionierungsverfahren vor, das einen optimierungsbasierten Ansatz mit Berücksichtigung von Betriebs- und Herstellungsprozessparameterschwankungen verwendet. Mit Hilfe formaler Methoden auf der Basis von Intervallarithmetik und affiner Arithmetik werden garantierte Schranken des Worst-Case-Verhaltens bestimmt und das globale Optimum des Dimensionierungsproblems deterministisch mit einem Branch-and-Bound-Verfahren berechnet. Zur Lösung der dabei auftretenden nichtlinearen Gleichungssysteme mit Parameterschwankungen wird ein neuartiger Newton-Operator in affiner Arithmetik verwendet.

Im Gegensatz zu anderen Ansätzen, die dem Stand der Technik entsprechen, garantiert das Verfahren dieser Arbeit durch den Einsatz formaler Methoden Ergebnisse mit globaler Optimalität und Einhaltung der Spezifikation bei Parameterschwankungen. Die Struktur der zur Modellierung verwendbaren Gleichungen wird dabei nicht eingeschränkt. Da kein Startwert benötigt wird, ist das Verfahren für die vollautomatische Synthese von Analogschaltungen geeignet.

Mit Hilfe einer Prototyp-Implementierung wird die Funktion des Verfahrens anhand von Beispielen demonstriert. Diese umfassen Dimensionierungen hinsichtlich des statischen Großsignalverhaltens und des dynamischen Kleinsignalverhaltens sowie des dynamischen Großsignalverhaltens unter Zuhilfenahme einer Analysegleichung. Es werden Beispiele mit Betriebs- und Herstellungsprozesspara-

meterschwankungen dargestellt und sowohl flächenminimale Entwürfe als auch eine Entwurfszentrierung durchgeführt. Die Effizienzverbesserung durch den neu entwickelten Newton-Operator in affiner Arithmetik wird durch Vergleich mit einer intervallarithmetischen Implementierung gezeigt.

Schlagwörter: Schaltungsdimensionierung, Intervallarithmetik, Affine Arithmetik

Abstract

Coping with the design challenges of complex systems on a chip requires computer-based automatic tools. Circuits consisting of analog and digital parts are of particular importance. In spite of its small share of the overall transistor count the analog part in most cases takes a significantly higher design time than the digital part. To close this design productivity gap, powerful synthesis tools for analog circuit design are needed.

Sizing is an important subproblem of analog circuit synthesis. This contribution presents a novel optimization-based sizing method taking into account operating and fabrication process parameter variations. By means of formal methods based on interval arithmetic and affine arithmetic guaranteed bounds of the worst case behavior are calculated and the global optimum of the sizing problem is computed deterministically using a branch-and-bound method. To solve the nonlinear systems of equations with parameter variations a novel Newton operator using affine arithmetic is introduced.

As opposed to other state-of-the-art approaches the method presented in this contribution guarantees results that are globally optimal and comply with the specification in the presence of parameter variations. At the same time, the structure of the equations used for modeling is not restricted. As no starting point is needed, this method is suitable for fully automatic analog circuit synthesis.

Using a prototype implementation, the functionality of the method is demonstrated for example problems. These comprise sizing with respect to static large-signal behavior and dynamic small-signal behavior as well as dynamic large-signal behavior with the help of a performance equation. Examples with operating and fabrication process parameter variations are described. Both area-minimizing designs and design centering are carried out. The gain in efficiency that is achieved by the newly developed Newton operator in affine arithmetic compared to an interval arithmetic implementation is demonstrated.

Key words: circuit sizing, interval arithmetic, affine arithmetic

Inhaltsverzeichnis

1	Einleitung	1
1.1	Entwurf integrierter Schaltungen	2
1.2	Werkzeuge für den Entwurf integrierter analoger Schaltungen	3
1.2.1	Synthesewerkzeuge	3
1.2.2	Analysewerkzeuge	5
1.3	Intervallarithmetik und affine Arithmetik	5
1.4	Motivation und Ziele	6
2	Automatischer Entwurf analoger Schaltungen	9
2.1	Spezifikation	9
2.1.1	Schaltungseigenschaftsraum	11
2.1.2	Akzeptanzgebiet	11
2.2	Analyse	12
2.2.1	Parameterräume	12
2.2.2	Analyseabbildung	14
2.2.3	Analysegleichungen	15
2.2.4	Simulation	18
2.2.5	Eigenschaftsextraktion	26
2.2.6	Worst-Case-Analyse	27
2.2.7	Charakterisierung	31
2.2.8	Ausbeuteanalyse	32
2.2.9	Formale Verifikation	35
2.3	Synthese	35
2.3.1	Topologieselektion	36
2.3.2	Konstruktive Topologiesynthese	39
2.3.3	Dimensionierung	39
2.3.4	Wissensbasierte Verfahren zur Dimensionierung	41
2.3.5	Gleichungslöserbasierte Dimensionierungsansätze	45
2.3.6	Optimierungsbasierte Ansätze	46
2.3.7	Zielkonflikt von Dimensionierungsverfahren	52

Inhaltsverzeichnis

2.3.8 Ausbeuteoptimierung	53
2.3.9 Entwurfszentrierung	55
2.3.10 Gleichzeitige Topologiesynthese und Dimensionierung	57
3 Dimensionierung mit formalen Methoden	59
3.1 Problemdefinitionen	59
3.1.1 Definition des Dimensionierungsproblems als Optimierungsproblem	59
3.1.2 Definition des Entwurfszentrierungsproblems	61
3.2 Gebietsarithmetiken	64
3.2.1 Intervallarithmetik	65
3.2.2 Affine Arithmetik	70
3.2.3 Vergleich und Bewertung	76
3.3 Dimensionierungsalgorithmus mit formalen Methoden	77
3.3.1 Globale Optimierung	77
3.3.2 Worst-Case-Analyse	88
3.3.3 Lösung der nichtlinearen Systemgleichungen	97
3.3.4 Laufzeitkomplexität	105
4 Implementierung und Ergebnisse	111
4.1 Implementiertes Programmsystem	111
4.2 Testschaltungen und Ergebnisse	111
4.2.1 Operationsverstärker	113
4.2.2 Differenzverstärker	121
4.2.3 Inverter I (Entwurfszentrierung)	124
4.2.4 Inverter II (flächenminimal)	127
5 Einordnung und Bewertung	129
6 Zusammenfassung	133

Abbildungsverzeichnis

1.1	Y-Diagramm für den Digitalschaltungsentwurf	2
1.2	Y-Diagramm für den Analogschaltungsentwurf	4
2.1	Schaltungseigenschaftsraum und Akzeptanzgebiet	12
2.2	Analyseabbildung für den zweidimensionalen Fall	14
2.3	Verstärkerschaltung	16
2.4	Ergebnis der Gleichstrom-Transfer-Simulation	21
2.5	Ergebnis der Wechselstromsimulation	23
2.6	Ergebnis der Transientensimulation	25
2.7	Histogramm der Verstärkung	34
2.8	Stromspiegeltopologien	37
2.9	Entscheidungsbaum für Stromspiegeltopologien	38
2.10	Beispiel einer Dimensionierungsrelation	41
2.11	Entwurfzentrierung im Prozessparameterraum P_S	56
3.1	Herstellungsprozessparametergebiet $P_{S,max}$ und Akzeptanz- gebiet A_{P_E}	63
3.2	Korrelation bei Multiplikation mit reeller Zahl a	72
3.3	Korrelation bei $f(x) = \sqrt{x}$	75
3.4	Minimierung mit dem Branch-and-Bound-Verfahren	79
3.5	Funktion mit zwei lokalen Minima	83
3.6	Minimierung mit dem gebietsarithmetischen Branch-and- Bound-Verfahren	86
3.7	Anwendung der Prädikate falsifiziert und verifiziert	91
3.8	Schranken der Zielfunktion f_u und relativer Abstand ϵ	95
3.9	Verbesserung von $[U_e]$ mit dem Einschließungstest	100
3.10	Newton-Operator	102
4.1	CMOS-Operationsverstärker	114
4.2	Entwurfparameterraum P_E nach Ablauf des Dimensionie- rungsalgorithmus für $C_K = 4$ pF	118
4.3	Wechselstromsimulation des Operationsverstärkers	119

Abbildungsverzeichnis

4.4	Transientensimulation des Operationsverstärkers	120
4.5	Differenzverstärker	121
4.6	Gleichstrom-Transfer-Simulation des Differenzverstärkers .	124
4.7	Inverter	124
4.8	Simulation der Ausgangsspannung des Inverters	126

Tabellenverzeichnis

2.1	Spezifikation eines Operationsverstärkers	10
2.2	Parameter der Verstärkerschaltung	20
2.3	Extrahierte Schaltungseigenschaftsgrößen	26
2.4	Ergebnis des Parameterdurchlaufs	30
2.5	Vergleich der Dimensionierungsansätze	54
3.1	Vergleich von Gebietsarithmetiken	76
4.1	Beispielübersicht	112
4.2	Spezifikation des Operationsverstärkers	115
4.3	Dimensionierungsergebnisse des Operationsverstärkers . .	117
4.4	Simulationsergebnisse des Operationsverstärkers	118
4.5	Spezifikation des Differenzverstärkers	122
4.6	Dimensionierungsergebnisse des Differenzverstärkers . . .	123
4.7	Spezifikation des Inverters (Entwurfzentrierung)	125
4.8	Entwurfzentrierungsergebnisse	126
4.9	Spezifikation des Inverters (flächenminimal)	127
4.10	Dimensionierungsergebnisse des Inverters	128
5.1	Vergleich der optimierungsbasierten Dimensionierungsverfahren	130

Tabellenverzeichnis

Symbolverzeichnis

Seitenzahlen verweisen auf Definitionen.

Allgemeine Vereinbarungen

argmax	$\operatorname{argmax}_{\mathbf{x} \in \Omega} f(\mathbf{x}) = \operatorname{argmin}_{\mathbf{x} \in \Omega} (-f(\mathbf{x}))$
argmin	$\mathbf{x}_{\min} = \operatorname{argmin}_{\mathbf{x} \in \Omega} f(\mathbf{x}) \Rightarrow \mathbf{x}_{\min} \in \{\mathbf{x} : f(\mathbf{x}) = \min_{\xi \in \Omega} f(\xi)\}$
\mathcal{C}	Kovarianzmatrix einer mehrdimensionalen Gauß-Verteilung
$[F]$	Intervallerweiterung von f , Seite 66
\hat{f}_{ε}	Affine Erweiterung von f , Seite 72
$[F_m]$	Mittelwertsform-Intervallerweiterung von f , Seite 69
f_{oben}	Obere Schrankenfunktion von f , Seite 82
f_{unten}^*	Untere Schranke des globalen Minimums von f , Seite 81
f_{unten}	Untere Schrankenfunktion von f , Seite 82
$[F_v]$	Vereinigte Intervallerweiterung von f , Seite 66
\mathfrak{M}	Matrix
$[\mathfrak{M}]$	Intervallwertige Matrix
\max	$\max \mathbf{t} = -\min(-\mathbf{t})$
\min	Komponentenweise: $\min \mathbf{t} = (\min t_1, \min t_2, \dots, \min t_n)$
∇	Nabla-Operator
$\ \cdot\ $	Beliebige Norm

Symbolverzeichnis

\mathcal{NP}	Klasse der Entscheidungsprobleme mit nichtdeterministisch-polynomialer Laufzeitkomplexität
\mathcal{O}	Landau-Symbol für Komplexitätsordnung
\mathbb{R}	Menge der reellen Zahlen
\mathbb{R}_+	Menge der positiven reellen Zahlen
R	Raum oder Gebiet
$[T]$	Intervalltupel $[T] \equiv ([T_1], [T_2], \dots, [T_n])$, Seite 65
\mathbf{t}	Tupel $\mathbf{t} \equiv (t_1, t_2, \dots, t_n)$
$ \mathbf{t} $	Komponentenweise: $ \mathbf{t} = (t_1 , t_2 , \dots, t_n)$
$\mathbf{t} \geq 0$	Komponentenweise: $\mathbf{t} \geq 0 \Leftrightarrow t_1 \geq 0 \wedge t_2 \geq 0 \wedge \dots \wedge t_n \geq 0$ (entsprechend auch für andere Relationen anzuwenden)
\mathbf{v}	Vektor
$[\mathbf{v}]$	Intervallwertiger Vektor
$[X]$	Intervall $[X] \equiv [\underline{x}, \bar{x}] \equiv [\underline{x}; \bar{x}]$, Seite 65
$[x]$	Punktintervall, Seite 65
\hat{x}_ε	Affine Form, Seite 70
$[X] \cup [Y]$	Vereinigungsmenge $[X] \cup [Y] = \{\xi : \underline{x} \leq \xi \leq \bar{x} \vee \underline{y} \leq \xi \leq \bar{y}\}$

Lateinische Buchstaben

\mathcal{A}	Systemmatrix der linearisierten Systemgleichungen, Seite 105
\mathbf{a}	Analyseabbildung, Seite 14
$\hat{\mathcal{A}}_\varepsilon$	Umwandlungsoperator von Intervall in affine Form, Seite 71
\mathbb{A}_ε	Menge der reellen affinen Formen mit ε , Seite 70
A_E	Akzeptanzgebiet in E , Seite 11
\mathbf{a}_E	Extraktionsabbildung, Seite 27

A_{P_E}	Akzeptanzgebiet in P_E , Seite 40
A_{P_S}	Akzeptanzgebiet in P_S , Seite 55
$A_{P_{SE}}$	Akzeptanzgebiet in $P_{S,Rad} \times P_E$, Seite 64
\mathbf{a}_S	Simulationsabbildung, Seite 27
\mathbf{a}_{WC}	Worst-Case-Analyse-Abbildung, Seite 28
$\mathbf{a}_{WC,S}$	Worst-Case-Analyse-Abbildung mit \mathbf{p}_S als Parameter, Seite 28
\mathbf{b}	Vektor der rechten Seite der linearisierten Systemgleichungen, Seite 105
bisect	Bisektionsoperator, Seite 81
\mathbf{c}	Ungleichungsnebenbedingungsfunktionen, Seite 47
\mathbf{d}	Gleichungsnebenbedingungsfunktionen, Seite 50
D	Dimensionierungsrelation, Seite 39
$d([\mathbf{X}])$	Durchmesser von $[\mathbf{X}]$, Seite 66
E	Schaltungseigenschaftsraum, Seite 11
\mathbf{e}	Schaltungseigenschaftstupel, Seite 11
$\mathbf{e}_{WC,max}$	Durch Worst-Case-Analyse gegebene Maxima für \mathbf{e} , Seite 30
$\mathbf{e}_{WC,min}$	Durch Worst-Case-Analyse gegebene Minima für \mathbf{e} , Seite 30
E_S	Simulationsergebnisraum, Seite 27
$\mathbf{e}_{Spez,max}$	Durch die Spezifikation gegebene Maxima für \mathbf{e} , Seite 11
$\mathbf{e}_{Spez,min}$	Durch die Spezifikation gegebene Minima für \mathbf{e} , Seite 11
\mathbf{f}	Zielfunktionstupel, Seite 60
\mathbf{f}_{AE}	Tupel der Systemgleichungsfunktionen ohne zeitliche Ableitungen, Seite 98
falsifiziert	Falsifiziert-Prädikat, Seite 84
falsifiziert'	Falsifiziert-Hilfsprädikat, Seite 89

Symbolverzeichnis

f_{DAE}	Tupel der Systemgleichungsfunktionen, Seite 98
f^Z	Zielfunktion bei Entwurfszentrierung, Seite 62
$f_{u,\text{unten}}^*$	Untere Schranke des globalen Minimums von f_u , Seite 85
f_u	Skalare vereinigte Zielfunktion, Seite 61
\mathbf{g}	Gewichtstupel der Zielfunktionen, Seite 60
$[I]$	Umwandlungsoperator von affiner Form in Intervall, Seite 70
\mathbb{I}_P	Menge aller Intervalle in P , Seite 98
\mathbb{I}_{P_E}	Menge aller Intervalle in P_E , Seite 84
\mathbb{I}_R	Menge der reellen Intervalle, Seite 65
n_c	Zahl der Ungleichungsnebenbedingungen, Seite 47
n_d	Zahl der Gleichungsnebenbedingungen, Seite 50
n_e	Dimensionszahl von E , Seite 11
n_f	Zahl der Zielfunktionen, Seite 60
n_{p_B}	Dimensionszahl von P_B , Seite 13
$n_{p_{BS}}$	Dimensionszahl von P_{BS} , Seite 60
n_{p_E}	Dimensionszahl von P_E , Seite 12
n_{p_S}	Dimensionszahl von P_S , Seite 13
n_u	Dimensionszahl von \mathbf{u} , Seite 98
P	Zusammengefasster Parameterraum, Seite 14
\mathbf{p}	Zusammengefasstes Parametertupel $\mathbf{p} \in P$, Seite 14
P_B	Betriebsparameterraum, Seite 13
\mathbf{p}_B	Betriebsparametertupel, Seite 13
P_{BS}	Zusammengefasster Betriebs- und Herstellungsprozessparameterraum, Seite 60

\mathbf{p}_{BS}	Zusammengefasstes Betriebs- und Herstellungsprozessparameter-tupel, Seite 60
$P_{B,Spez}$	Betriebsparametergebiet, Seite 13
$P_{BS,Spez}$	Zusammengefasstes Betriebs- und Herstellungsprozessparametergebiet, Seite 60
P_E	Entwurfparameterraum, Seite 12
\mathbf{p}_E	Entwurfparameter-tupel, Seite 12
\mathbf{p}_E^*	Lösung des Dimensionierungsproblems, Seite 61
\mathbf{p}_E^{**}	Lösung des Entwurfszentrierungsproblems, Seite 63
P_{NF}	Menge aller nicht falsifizierten Teilräume von P_E , Seite 84
$\mathbf{p}_{nom,BS}$	Nominalpunkt der Prozess- und Betriebsparameter, Seite 14
P_S	Herstellungsprozessparameterraum, Seite 13
\mathbf{p}_S	Herstellungsprozessparameter-tupel, Seite 13
$P_{S,max}$	Herstellungsprozessparametergebiet bei maximaler Abweichung $\mathbf{p}_{S,Rad}$, Seite 62
$P_{S,Rad}$	Herstellungsprozessparameterradienraum, Seite 61
$\mathbf{p}_{S,Rad}$	Herstellungsprozessparameterradien-tupel, Seite 61
$P_{S,Spez}$	Herstellungsprozessparametergebiet, Seite 13
$S_{\mathbf{p}_E}$	Relation zur Bestimmung von A_{P_S} , Seite 55
U	Exakte Lösungsmenge von Gleichungssystem 3.75, Seite 98
\mathbf{u}	Tupel der unbekanntten Spannungen und Ströme, Seite 98
\mathbf{u}	Unbekannttenvektor der linearisierten Systemgleichungen, Seite 105
U_{BE}	Basis-Emitter-Spannung
U_{CE}	Kollektor-Emitter-Spannung
U_{DS}	Drain-Source-Spannung

Symbolverzeichnis

$[U_e]$	Intervalleinschließung von U , Seite 98
U_{GS}	Gate-Source-Spannung
$[U_N]$	Newton-Operator in affiner Arithmetik, Seite 102
verifiziert	Verifiziert-Prädikat, Seite 84
verifiziert'	Verifiziert-Hilfsprädikat, Seite 89
w	Wahrscheinlichkeitsverteilung in P_S , Seite 32
Y	Herstellungsausbeute, Seite 33

Griechische Buchstaben

α	Akzeptanzfunktion, Seite 33
β	Abstandsmaß zu $\mathbf{p}_{\text{nom},s}$ bei Gauß-Verteilung, Seite 55
β	Stromverstärkung
Δ	Absolute Überschätzung, Seite 69
ϵ	Obere Schranke der relativen Abweichung vom globalen Optimum, Seite 81
ϵ_{Tol}	Maximale relative Abweichung vom globalen Optimum, Seite 81
ϵ	Das globale Epsilontupel der affinen Arithmetik, Seite 70
Ω	Allgemeiner Suchraum
Ω	Einheit des elektrischen Widerstandes
σ	Standardabweichung einer Gauß-Verteilung

1 Einleitung

Der rasante Fortschritt in der Halbleiterfertigungstechnik ermöglicht es, integrierte Schaltungen mit immer mehr Bauelementen herzustellen. Dadurch lassen sich zunehmend komplexere Systeme auf einem Chip realisieren. Ihr Entwurf erfordert den Einsatz automatischer, rechnergestützter Werkzeuge, um die große Zahl an benötigten Elementen erzeugen und verarbeiten zu können. Dieses Vorgehen wird als Electronic Design Automation (EDA) bezeichnet.

Bei der heute verbreiteten Integration ganzer Systeme auf einem Chip, so genannter Systems-on-a-Chip (SOC), kommt dem Entwurf von Schaltungen mit analogen und digitalen Teilen (Mixed-Signal-Schaltungen) wachsende Bedeutung zu. Analoge Schaltungsteile werden vor allem als Schnittstellen zur Außenwelt benötigt, z. B. in der Kommunikations- und der Automobilelektronik. Während zur Erstellung der digitalen Schaltungsteile einer Mixed-Signal-Schaltung leistungsfähige Synthesewerkzeuge zur Verfügung stehen, werden die analogen noch weitgehend manuell unter Zuhilfenahme von Simulatoren entworfen. Dadurch liegt die Entwurfszeit pro Transistor des Analogteils um mehrere Größenordnungen über der des Digitalteils. Ferner kann der manuelle Entwurf nur von erfahrenen Anlogschaltungsentwicklern geleistet werden. Der Entwurf des Analogteils stellt damit trotz seines kleinen Anteils an der Gesamttransistorzahl meist den Flaschenhals eines Mixed-Signal-Entwurfs dar.

Um die Entwurfsproduktivitätslücke zwischen Analog- und Digital-schaltungsentwurf zu schließen, müssen Synthesewerkzeuge für den Entwurf von Anlogschaltungen entwickelt werden, die eine dem Digital-schaltungsentwurf vergleichbare Automatisierung bieten. Für die Akzeptanz eines solchen Werkzeuges ist es wichtig, dass es automatisch ein korrektes Ergebnis liefert, ohne dass der Benutzer in den Ablauf eingreifen oder die Richtigkeit des Ergebnisses überprüfen muss. Dies kann durch den Einsatz formaler Methoden gewährleistet werden. Mit ihrer Hilfe ist es möglich, Ergebnisse mit der Aussagekraft eines mathematischen Beweises zu erhalten.

In den beiden folgenden Abschnitten wird zunächst ein Überblick über die Vorgänge beim Entwurf integrierter Schaltungen und über die Automatisierung im Anlogschaltungsentwurf gegeben. Danach werden

1 Einleitung

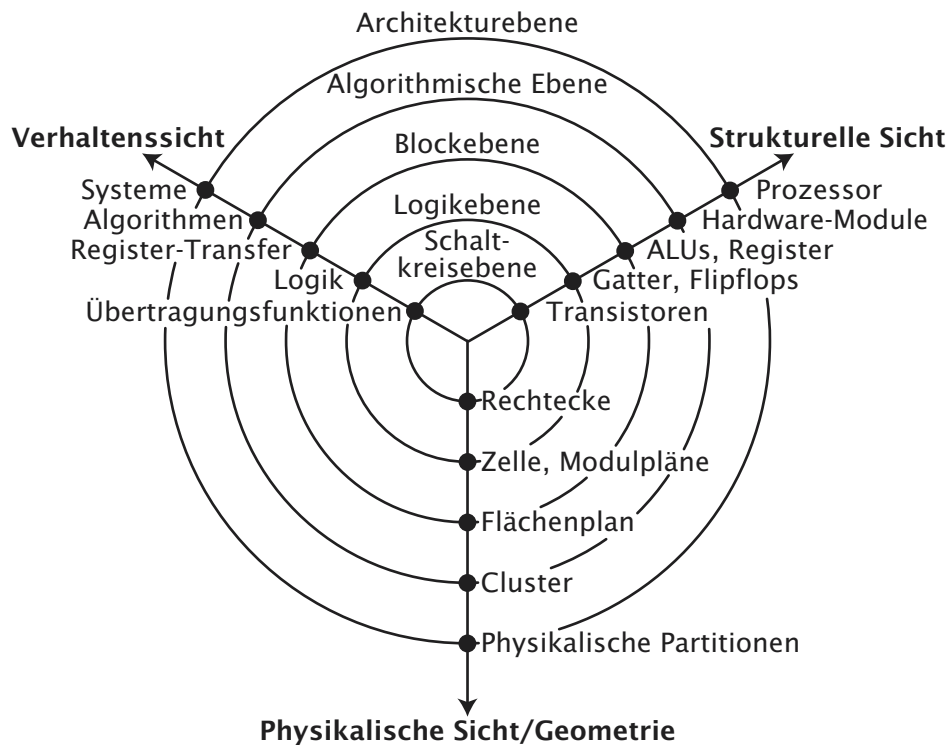


Abbildung 1.1: Y-Diagramm für den Digitalschaltungsentwurf (Gajski und Kuhn 1983, vom Verfasser übersetzt)

die den formalen Methoden in dieser Arbeit zugrunde liegenden mathematischen Werkzeuge vorgestellt. Abschließend werden die Ziele der Arbeit definiert.

1.1 Entwurf integrierter Schaltungen

Integrierte Schaltungen werden aus Komplexitätsgründen durch Lösen einer Reihe von Teilaufgaben entworfen. Zur Darstellung und Systematisierung von Entwurfsprozessen wird das Y-Diagramm verwendet. Die ursprüngliche, für den Digitalschaltungsentwurf vorgestellte Fassung (Gajski und Kuhn 1983) zeigt Abbildung 1.1. Anpassungen für den Anlogschaltungsentwurf werden in Abschnitt 1.2 beschrieben. Die konzentrischen Kreise des Y-Diagramms stellen Abstraktionsebenen beim Entwurf dar, wobei der Abstraktionsgrad von innen nach außen ansteigt. Die Achsen bezeichnen verschiedene Sichten des Entwurfs. Jedem Entwurfzustand entspricht ein Schnittpunkt zwischen einer Achse und einem Kreis. In Abbildung 1.1 sind an den Schnittpunkten Beispiele für

1.2 Werkzeuge für den Entwurf integrierter analoger Schaltungen

die Art von Elementen, aus denen der Entwurf im jeweiligen Zustand besteht, angegeben.

Der Ausgangszustand zu Beginn eines Entwurfsprozesses ist die so genannte Spezifikation in der Verhaltenssicht der Architekturebene. Der Zielzustand ist die physikalische Sicht der Schaltungsebene, also ein Maskenlayout für die Herstellung der integrierten Schaltung.

Entwurfsschritte lassen sich mit Hilfe des Y-Diagramms in zwei Kategorien einteilen:

Syntheseschritte fügen dem Entwurf Informationen hinzu und sind auf den Entwurfszielzustand, also das Zentrum des Y-Diagramms, hin gerichtet. Dabei sinkt der Abstraktionsgrad und der Detaillierungsgrad steigt. Die Synthese ist ein kreativer Prozess.

Analyseschritte dienen der Überprüfung von Syntheseergebnissen durch Abstraktion oder Extraktion und weisen damit im Y-Diagramm in die entgegengesetzte Richtung, d. h. vom Entwurfszielzustand weg.

1.2 Werkzeuge für den Entwurf integrierter analoger Schaltungen

Für den Analogschaltungsentwurf muss das Y-Diagramm an die dabei verwendeten Abstraktionsebenen angepasst werden. Abbildung 1.2 zeigt eine Variante mit vier Ebenen (Hosticka u. a. 1994). Dieses Y-Diagramm unterscheidet sich von dem in Abbildung 1.1 für den Digital-schaltungsentwurf gezeigten durch die Auslassung der Logikebene, für die es in der Analogschaltungstechnik keine Entsprechung gibt. In einem anderen Ansatz mit sieben Ebenen (Kampe und Scarbata 1993) werden die untersten zwei Ebenen des Y-Diagramms in Abbildung 1.1 durch Operationsblock-, Anpassungs-, Bauelemente- und elektrische Ebene ersetzt. Im Folgenden wird ausschließlich die in Abbildung 1.2 dargestellte Variante mit vier Ebenen verwendet.

1.2.1 Synthesewerkzeuge

Im Gegensatz zum Digital-schaltungsentwurf werden automatische Syntheseverfahren beim Entwurf von Analogschaltungen in der Praxis kaum eingesetzt (Ochotta u. a. 1998). Die Forschung konzentriert sich bisher

1 Einleitung

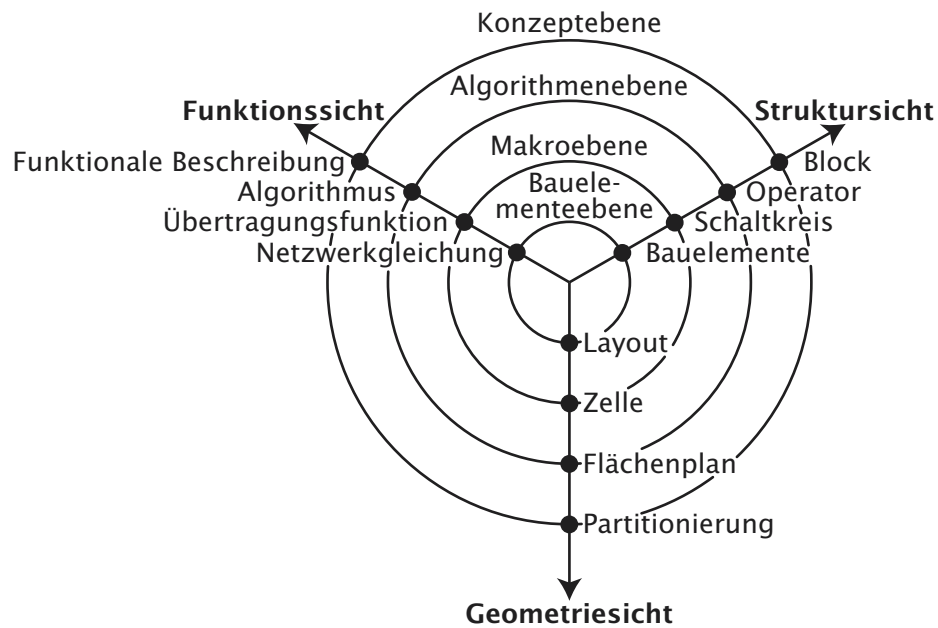


Abbildung 1.2: Y-Diagramm für den Analogschaltungsentwurf (Hosticka u. a. 1994, vom Verfasser übersetzt)

hauptsächlich auf die unteren Abstraktionsebenen. Hier wird zwischen Schaltungssynthese und Layoutsynthese unterschieden.

Bei der Schaltungssynthese wird ausgehend von einer Verhaltensbeschreibung in der Funktionssicht auf Makroebene eine dimensionierte Schaltung in der Struktursicht auf Bauelementebene erzeugt. Dazu werden die Teilprobleme Topologiegenerierung bzw. -selektion und Dimensionierung meist getrennt gelöst. Nur für die Dimensionierung gibt es erste in kommerzielle Produkte umgesetzte Ansätze (Phelps u. a. 1999; del Mar Hershenson u. a. 2001; Schwencker u. a. 2002). Die meisten dieser Ansätze verwenden ein Optimierungsverfahren in Verbindung mit einem Analysewerkzeug. Gründe dafür sind die Flexibilität und die hohe erreichbare Genauigkeit sowie der vergleichsweise geringe manuelle Vorbereitungsanfang.

Automatische Topologiegenerierung und -selektion sind bisher ausschließlich Gegenstand der Forschung (z. B. Kruiskamp und Leenaerts 1995; Veselinovic u. a. 1995; Sripramong und Toumazou 2002). Nach der Standardisierung analoger Verhaltensbeschreibungssprachen (IEEE 1999; Accellera 2002) wird die automatische Synthese mittlerweile auch auf höheren Abstraktionsebenen erforscht (Doboli u. a. 1999). Eine weitergehende Beschreibung der Verfahren zur automatischen Schaltungs-

synthese erfolgt in Abschnitt 2.3.

Bei der Layoutsynthese wird im Anschluss an die Schaltungssynthese mit einem Sichtenwechsel von der Struktursicht zur Geometriesicht ein Maskenlayout erzeugt. Verfahren zur automatischen Layoutsynthese sind sowohl für die Platzierung als auch für die Verdrahtung vorgeschlagen worden (Cohn u. a. 1991; Basaran u. a. 1993). Auch hier existieren erste kommerzielle Umsetzungen.

1.2.2 Analysewerkzeuge

Anders als für die Synthese gibt es für die automatische Analyse von Analogschaltungen eine Vielzahl kommerziell erhältlicher Werkzeuge, die zum Standard in der Entwurfspraxis geworden sind. Die zentrale Stellung nimmt dabei der Schaltungssimulator ein, auf dessen Basis weitere Werkzeuge entstanden sind, z. B. zur automatischen Charakterisierung und Monte-Carlo-Simulation. Der klassische Schaltungssimulator ist SPICE (Nagel 1975). Er führt auf Basis einer Netzlistenbeschreibung in der Struktursicht auf Bauelementeebene verschiedene Analysen durch. Von besonderem Interesse für die Dimensionierung ist dabei die Worst-Case-Analyse, die untersucht, wie sich der Entwurf bei variierenden Parametern verhält. Moderne Simulatoren verarbeiten auch Netzlisten mit Elementen, die in einer Verhaltensbeschreibungssprache modelliert sind. Sie lassen sich damit auf jeder Abstraktionsebene einsetzen.

Ein Analyseverfahren auf der Schwelle zur praktischen Einsetzbarkeit ist die formale Verifikation (Hedrich 1997; Hartong u. a. 2002). Im Gegensatz zur Validierung durch Simulationsexperimente werden mit diesem Verfahren Ergebnisse mit der Aussagekraft eines mathematischen Beweises erzielt.

Abschnitt 2.2 gibt eine systematische Beschreibung der Verfahren zur Analyse von Analogschaltungen. Da viele Syntheseverfahren auf Analyseverfahren zurückgreifen, ist er - in Umkehrung der Reihenfolge beim Entwurf - dem Abschnitt zur Synthese vorangestellt.

1.3 Intervallarithmetik und affine Arithmetik

Die Intervallarithmetik wurde ursprünglich als Mittel zur Verfolgung der Fortpflanzung von Messungenauigkeiten und Rundungsfehlern in numerischen Rechnungen entwickelt (Moore 1966). Ihr liegt das Prinzip zugrunde, Größen durch untere und obere Schranken darzustellen.

1 Einleitung

Diese Schranken definieren Intervalle, die die betrachteten Größen garantiert einschließen. Durch pessimistische Abschätzungen werden bei allen Rechenoperationen Ergebnisintervalle erreicht, die die Einschließungseigenschaft ebenfalls garantieren. Dadurch stand mit der Intervallarithmetik erstmals ein Werkzeug zur Verfügung, das trotz numerischer Rechnung Ergebnisse mit der Aussagekraft eines mathematischen Beweises liefert.

Die Methoden der Intervallarithmetik eignen sich auch zur Behandlung von Problemen, zu deren Lösung Gebiete endlicher Größe geschlossen behandelt werden müssen. Auch dabei garantiert die Einschließungseigenschaft formal richtige Ergebnisse. Nach Moores Veröffentlichung wurden verschiedene Algorithmen entwickelt, die diese Möglichkeit nutzen. Dazu zählen eine Variante des Newton-Verfahrens zur Nullstellensuche (Neumaier 1990), die globale Optimierung (Hansen 1980), Verfahren zur formalen Verifikation von Analogschaltungen (Hedrich und Barke 1998; Hartong u. a. 2002, vgl. Abschnitt 1.2.2), eine Methode zum Entwurf von linear modellierten Analogschaltungen (Leenaerts 1990) u. a. Die Effizienz dieser Verfahren wird jedoch durch den Effekt der Überschätzung (s. Abschnitt 3.2.1) stark herabgesetzt, da dieser in der Regel zu exponentieller Laufzeitkomplexität führt.

Der Überschätzungseffekt wird durch das Ignorieren der Korrelation von Größen in der Intervallarithmetik verursacht. Verschiedene Ansätze (z. B. Hansen 1975; de Figueiredo und Stolfi 1997) integrieren die teilweise Berücksichtigung von Korrelationsinformation in die Intervallarithmetik. Von diesen Ansätzen ist der neueste die von Stolfi entwickelte affine Arithmetik, die in dieser Arbeit verwendet werden soll. Ein Ansatz zur Worst-Case-Analyse mit affiner Arithmetik ist bereits veröffentlicht worden (Femia und Spagnuolo 2000). Dieser berücksichtigt jedoch nur Analysegleichungen in explizit aufgelöster Form.

Die Intervallarithmetik und die affine Arithmetik werden ausführlich in den Abschnitten 3.2.1 und 3.2.2 vorgestellt.

1.4 Motivation und Ziele

Für die Praxistauglichkeit eines Dimensionierungswerkzeuges ist es entscheidend, dass automatisch korrekte Ergebnisse geliefert werden, ohne dass der Benutzer in den Ablauf eingreifen oder die Richtigkeit der Ergebnisse überprüfen muss. Die üblichen, auf Optimierer-Simulator-Kopplung basierenden Werkzeuge leisten dies jedoch nicht. Dies zeigt sich an zwei Aspekten, die im Folgenden erläutert werden: automati-

sche globale Optimierung und Berücksichtigung von Parameterschwankungen.

Das globale Optimum des Dimensionierungsproblems muss automatisch und sicher gefunden werden. Der hierfür in der Praxis am häufigsten eingesetzte Algorithmus ist Simulated Annealing. Es kann jedoch nicht bewiesen werden, dass dieser das globale Optimum in einer endlichen Zahl an Schritten findet, sondern nur mit asymptotisch gegen 100% gehender Wahrscheinlichkeit, wenn die Zahl der Schritte gegen unendlich geht (Romeo und Sangiovanni-Vincentelli 1991). In der Praxis bedeutet dies, dass sehr viele Schritte für quasi-globale Konvergenz erforderlich sind und dass das Finden des globalen Optimums dennoch nicht sicher ist. In dieser Arbeit soll ein Dimensionierungsalgorithmus entworfen werden, der das globale Optimum des Dimensionierungsproblems mit Hilfe formaler Methoden deterministisch findet und eine formal garantierte Aussage über Lage und Wert des Optimums liefert.

Damit die zu entwerfende Schaltung auch bei Parameterschwankungen funktioniert (Variationstoleranz), müssen diese bei der Dimensionierung berücksichtigt werden. Dazu muss das Worst-Case-Verhalten analysiert werden (vgl. Abschnitt 1.2.2). Die meisten Ansätze berücksichtigen jedoch nur den Nominalfall, was zu schlechter Robustheit und niedriger Fertigungsausbeute führt. Einige Werkzeuge berechnen zwar eine formal nicht garantierte Abschätzung für den Worst Case, die Funktionsfähigkeit der entworfenen Schaltung ist damit aber nicht sichergestellt. Der zu entwerfende Dimensionierungsalgorithmus soll mit formalen Methoden garantierte Schranken für das Worst-Case-Verhalten der betrachteten Schaltung berechnen und so eine Garantie dafür liefern, dass die erzielten Ergebnisse die Spezifikation einhalten oder das Dimensionierungsproblem nicht lösbar ist. In Verbindung mit der o. a. Eigenschaft, das globale Optimum deterministisch zu finden, soll damit ein Werkzeug erstellt werden, das die zu Beginn dieses Abschnittes aufgestellte Forderung, automatisch korrekte Ergebnisse zu liefern, für Dimensionierungsprobleme erfüllen kann.

Die Flexibilität des optimierungsbasierten Ansatzes hinsichtlich der Freiheit der Wahl der zu verwendenden Modelle soll erhalten bleiben, da nur so ausreichend hohe Genauigkeit und niedriger Vorbereitungs- und Aufwand beim praktischen Einsatz gewährleistet werden können.

Der skizzierte Ansatz soll in dieser Arbeit unter Verwendung affiner Arithmetik zu einem Algorithmus entwickelt werden. Dies ist in Kapitel 3 dargestellt. Erste Arbeiten dazu sind bereits veröffentlicht worden (Lemke u. a. 2002, 2003). Auf Basis einer Prototypenimplementierung soll anschließend an einer Auswahl von Beispielen aus praxisrelevanten

1 Einleitung

Schaltungsklassen die Funktionsfähigkeit des Algorithmus gezeigt. Die Ergebnisse hieraus werden in Kapitel 4 zusammengefasst. Abschließend wird der Dimensionierungsalgorithmus im Vergleich zu anderen Ansätzen eingeordnet und seine Leistungsfähigkeit bewertet.

2 Automatischer Entwurf analoger Schaltungen

2.1 Spezifikation

Zu Beginn des Entwurfs einer analogen Schaltung werden die geforderten Eigenschaften in der Spezifikation festgelegt. Diese befindet sich im Y-Diagramm für den Analogschaltungsentwurf (vgl. Abbildung 1.2) in der Funktionssicht auf der Konzeptebene. Die Spezifikation wird in den meisten Fällen von Hand erstellt.

Das grundsätzliche Verhalten sowie die elektrischen Schnittstellen der spezifizierten Schaltung werden durch die Angabe einer Schaltungs-klasse festgelegt: z. B. wird der Entwurf eines Verstärkers oder einer Spannungsreferenz verlangt. Im Zentrum der Spezifikation einer Analo-gschaltung stehen Anforderungen an ihre charakteristischen Eigen-schaftsgrößen. Diese Anforderungen können sowohl als Ungleichungs- nebenbedingungen (constraints) als auch in Gestalt von Minimierungs- und Maximierungszielen (objectives) formuliert werden. Der Begriff *Ne-benbedingung* hat seinen Ursprung in der formalen Darstellung als Op-timierungsproblem und impliziert keine Aussage über den Rang einer Anforderung innerhalb der Spezifikation.

Zu einer vollständigen Spezifikation gehören ferner die statistische Verteilung der Parameter des vorgegebenen Herstellungsprozesses und die Betriebsparameterbereiche, in denen die Schaltung funktionieren, d. h. die geforderten Eigenschaften aufweisen muss. Oft werden die-se Parameterbereiche und -verteilungen zur Vereinfachung auf den so genannten Nominalparameterpunkt reduziert. Dies kann jedoch zu ho-hem Ausschuss in der Fertigung und mangelhafter Robustheit im Ein-satz führen, da die spezifizierten Eigenschaften bei Abweichung vom Nominalparameterpunkt nicht gewährleistet sind.

Beispiel 2.1 In Tabelle 2.1 ist die Spezifikation einer Schaltung aus der Klasse der Operationsverstärker dargestellt. Die charakteristischen Eigenschaftsgrößen sind: Verstärkung, Verstärkungsbandbreiteprodukt,

2 Automatischer Entwurf analoger Schaltungen

Eigenschaft	Spezifikation
Herstellungsprozess	0,5 μm CMOS ^{a, b}
Versorgungsspannung	4,5 bis 5,5 V ^b
Umgebungstemperatur	-40 bis 85 °C
Verstärkung	> 80 dB
Verstärkungsbandbreiteprodukt	> 50 MHz
Phasenreserve	> 60°
Anstiegsgeschwindigkeit	> 10 V/ μs
Offsetspannung	< 1 mV
Eingangsbezogene Rauschspannung bei 1 kHz	< 10 nV/ $\sqrt{\text{Hz}}$
Eingangsbezogener Rauschstrom bei 1 kHz	< 10 pA/ $\sqrt{\text{Hz}}$
Stromaufnahme	< 1 mA
Fläche	minimieren

^aHerstellerspezifische Prozessbezeichnung ist hinzuzufügen.

^bEntspricht nicht mehr dem Stand der Technik, ist aber aus Kostengründen noch sehr verbreitet. Die genaue Wahl ist für Überlegungen zur automatischen Synthese unbedeutend.

Tabelle 2.1: Spezifikation eines Operationsverstärkers

Phasenreserve, Anstiegsgeschwindigkeit, Offsetspannung, Rauschspannung und -strom, Stromaufnahme und Fläche. Bis auf die Fläche sind für alle genannten Eigenschaftsgrößen Ungleichungsnebenbedingungen gegeben, für die Fläche wird eine Minimierung verlangt. Mit der Bezeichnung des vorgesehenen Herstellungsprozesses können seine Parameter nachgeschlagen bzw. beim Hersteller erfragt werden. Ferner sind als Betriebsparameterbereiche der Versorgungsspannungs- und der Umgebungstemperaturbereich spezifiziert.

Manuell erstellte Spezifikationen sind oft ungenau und unvollständig oder sogar fehlerhaft. Als Eingabe für automatische Entwurfswerkzeuge ist jedoch eine eindeutige, maschinenlesbare Formulierung erforderlich. Derartige, in einer formalen Sprache erstellte Spezifikationen werden als ausführbar bezeichnet, wenn ein automatisches Werkzeug ihre Einhaltung für eine gegebene Schaltung überprüfen kann. Im AnaloSchaltungsentwurf werden dazu meist Charakterisierungsskripte (s. Abschnitt 2.2.7) eingesetzt, mit denen die Einhaltung der Nebenbedingungen der Spezifikation überprüft wird. Dazu werden die berechneten Werte der Eigenschaftsgrößen mit der Spezifikation verglichen. Selten werden in einer Verhaltensbeschreibungssprache formulierte Modelle verwendet (s. Äquivalenzüberprüfung, Abschnitt 2.2.9). Ausführbare

Spezifikationen werden beim Schaltungsentwurf zur automatischen Verifikation und zur Synthese eingesetzt.

Aus der Spezifikation ergeben sich der *Schaltungseigenschaftsraum* und das darin liegende *Akzeptanzgebiet*, die im Folgenden definiert werden.

2.1.1 Schaltungseigenschaftsraum

Alle in der Spezifikation vorkommenden Eigenschaftsgrößen e_i lassen sich formal zu einem Tupel \mathbf{e} zusammenfassen. Zusammen mit den Wertebereichen, die für die Komponenten e_i je nach ihrer Definition in der Realität gelten, lässt sich ein Schaltungseigenschaftsraum definieren.

Definition 2.1 *Der Schaltungseigenschaftsraum E wird durch das Schaltungseigenschaftstupel \mathbf{e} aufgespannt, wobei $\mathbf{e} \in E$ und $E \subseteq \mathbb{R}^{n_e}$.*

2.1.2 Akzeptanzgebiet

Die Ungleichungsnebenbedingungen der Spezifikation definieren in E das so genannte Akzeptanzgebiet.

Definition 2.2 *Das Akzeptanzgebiet A_E im Schaltungseigenschaftsraum E ist die Schnittmenge aller Gebiete, die in E durch je eine der in der Spezifikation gegebenen Ungleichungsnebenbedingungen für Schaltungseigenschaftsgrößen definiert werden.*

Bei praktischen Problemen treten ausschließlich Spezifikationen mit Ungleichungsnebenbedingungen der Form $e_i < e_{i,Spez,max}$ bzw. $e_i > e_{i,Spez,min}$ mit $e_{i,Spez,min}, e_{i,Spez,max} \in \mathbb{R}$ auf. Nebenbedingungen der Form $x < y$, wobei x und y beliebige Ausdrücke sind, lassen sich ggf. durch Einführen einer künstlichen Eigenschaft $e_m = x - y$ in $e_m < 0$ umformen. Damit ergeben sich Akzeptanzgebiete, die von achsenparallelen Hyperebenen begrenzt werden.

Beispiel 2.2 *Abbildung 2.1 zeigt den Schaltungseigenschaftsraum E für zwei Schaltungseigenschaftsgrößen e_1 und e_2 . Durch die Spezifikation seien folgende Nebenbedingungen gegeben:*

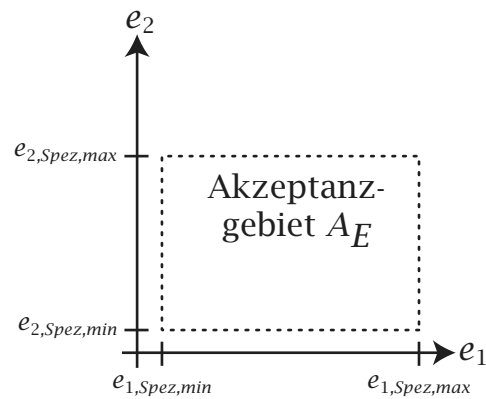
$$e_1 > e_{1,Spez,min} , \quad (2.1)$$

$$e_1 < e_{1,Spez,max} , \quad (2.2)$$

$$e_2 > e_{2,Spez,min} , \quad (2.3)$$

$$e_2 < e_{2,Spez,max} . \quad (2.4)$$

2 Automatischer Entwurf analoger Schaltungen



Schaltungseigenschaftsraum E

Abbildung 2.1: Schaltungseigenschaftsraum und Akzeptanzgebiet

Die Schnittmenge der durch diese vier Ungleichungsnebenbedingungen definierten vier Halbräume in E ist das in Abbildung 2.1 dargestellte Akzeptanzgebiet A_E .

2.2 Analyse

Die meisten Synthesewerkzeuge greifen im Kern auf Analyseverfahren zurück (s. Abschnitt 2.3). In diesem Abschnitt werden daher die für die Betrachtungen zur Synthese benötigten Analysewerkzeuge und -verfahren eingeführt. Es sind Verfahren zur Extraktion von Schaltungseigenschaftsgrößen, die der Überprüfung von Synthesergebnissen dienen (vgl. Abschnitt 1.1).

2.2.1 Parameterräume

Zur formalen Definition von Analyseverfahren müssen zunächst die Begriffe: Entwurfs-, Betriebs- und Herstellungsprozessparameterraum eingeführt werden.

Die beim Analogschaltungsentwurf festzulegenden Schaltungsparameter werden als Entwurfparameter bezeichnet. Sie lassen sich formal zu einem Tupel zusammenfassen, das mit den zulässigen Wertebereichen einen Raum definiert:

Definition 2.3 Der Entwurfparameterraum P_E wird durch das Entwurfparametertupel \mathbf{p}_E aufgespannt, wobei $\mathbf{p}_E \in P_E$ und $P_E \subseteq \mathbb{R}^{n_{pE}}$.

Entwurfsparameter sind typischerweise Bauelementgrößen wie Transistorweiten und -längen, Widerstands- und Kapazitätswerte.

Über die im Entwurf festlegbaren Parameter hinaus gibt es weitere Parameter, die in die Analyse eingehen: Betriebsparameter und Herstellungsprozessparameter. Sie spannen ebenfalls Räume auf:

Definition 2.4 *Der Betriebsparametererraum P_B wird durch das Betriebsparameterertupel \mathbf{p}_B aufgespannt, wobei $\mathbf{p}_B \in P_B$ und $P_B \subseteq \mathbb{R}^{n_{PB}}$.*

Zu den Betriebsparametern zählen die Umgebungstemperatur, die Versorgungsspannung u. ä. Meist muss nicht ein einzelner Punkt im Umgebungs- und Betriebsparametererraum, sondern ein durch die Spezifikation definiertes Gebiet $P_{B,Spez}$ bei der Analyse betrachtet werden.

Definition 2.5 *Das Betriebsparametergebiet $P_{B,Spez} \subseteq P_B$ ist das durch den spezifizierten Betriebsparameterbereich gegebene Gebiet in P_B .*

Wie das Akzeptanzgebiet A_E ist auch $P_{B,Spez}$ in praktischen Problemen ein durch achsenparallele Hyperebenen begrenztes Gebiet (Begründung entsprechend Abschnitt 2.1.2).

Definition 2.6 *Der Herstellungsprozessparametererraum P_S wird durch das Herstellungsprozessparameterertupel \mathbf{p}_S aufgespannt, wobei $\mathbf{p}_S \in P_S$ und $P_S \subseteq \mathbb{R}^{n_{PS}}$.*

Herstellungsprozessparameter umfassen Oxidschichtdicken, Dotierungsdichten, Mismatch-Größen etc. In der Regel ist es wie bei den Umgebungs- und Betriebsparametern erforderlich, nicht einen Punkt, sondern ein Gebiet zur Analyse anzusetzen:

Definition 2.7 *Das Herstellungsprozessparametergebiet $P_{S,Spez} \subseteq P_S$ ist das durch die spezifizierten Herstellungsprozessparameterbereiche gegebene Gebiet in P_S .*

Auch $P_{S,Spez}$ ist in der Praxis ein durch achsenparallele Hyperebenen begrenztes Gebiet (Begründung entsprechend Abschnitt 2.1.2). Oft wird allerdings $P_{S,Spez} = P_S$ gesetzt und eine Wahrscheinlichkeitsdichtefunktion auf P_S gegeben, die die statistische Verteilung der Parameter beschreibt (s. Abschnitt 2.2.8).

Der Nominalpunkt der Prozess- und Betriebsparameter (vgl. Abschnitt 2.1) lässt sich mit P_B und P_S formal wie folgt definieren:

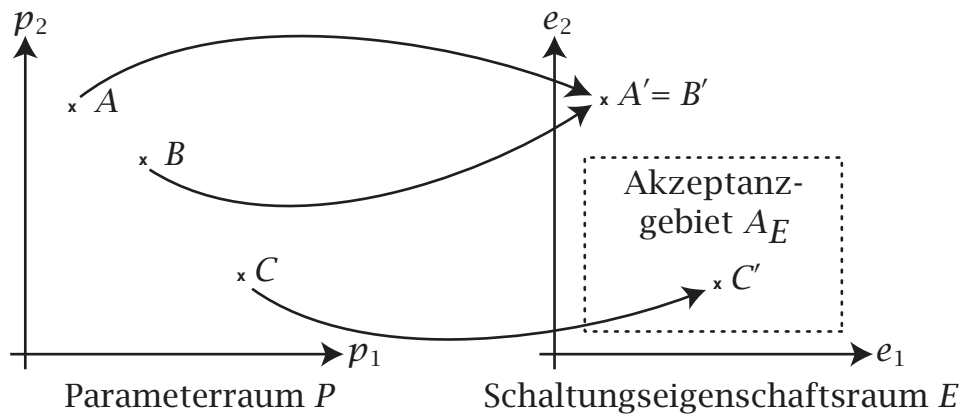


Abbildung 2.2: Analyseabbildung für den zweidimensionalen Fall

Definition 2.8 Der Nominalpunkt der Prozess- und Betriebsparameter ist ein Tupel $\mathbf{p}_{\text{nom,BS}} = (\mathbf{p}_{\text{nom,B}}, \mathbf{p}_{\text{nom,S}}) \in P_B \times P_S$.

Zur kompakteren Darstellung werden die bisher definierten Parameterräume zusammengefasst:

Definition 2.9 Der zusammengefasste Parameterraum P ist gegeben durch $P = P_E \times P_B \times P_S$. Das zusammengefasste Parametertupel \mathbf{p} ist damit als $\mathbf{p} \in P$ gegeben.

2.2.2 Analyseabbildung

Die Analyse liefert für alle praxisrelevanten Fälle eindeutige Ergebnisse. Daher lässt sich eine Analyseabbildung definieren.

Definition 2.10 Eine Analyseabbildung ist eine Abbildung $\mathbf{a} : P \rightarrow E$ vom Parameterraum P in den Schaltungseigenschaftsraum E .

Die Abbildung \mathbf{a} ist im Allgemeinen nicht eindeutig umkehrbar.

Beispiel 2.3 In Abbildung 2.2 sind ein zweidimensionaler Parameterraum P und ein ebenfalls zweidimensionaler Schaltungseigenschaftsraum E mit Akzeptanzgebiet A_E dargestellt. Die Analyseabbildung dreier Punkte A , B und C ist exemplarisch gezeigt. Dabei liegen die Bilder A' und B' von A bzw. B außerhalb des Akzeptanzgebietes, d. h. A und B verstoßen gegen mindestens eine Ungleichungsnebenbedingung der Spezifikation. Das Bild C' von C liegt innerhalb des Akzeptanzgebietes. Der Punkt C erfüllt also die Ungleichungsnebenbedingungen der Spezifikation. Die

Bildpunkte A' und B' sind identisch, d. h. die dargestellte Abbildung ist nicht eindeutig umkehrbar.

Im Gegensatz zu Beispiel 2.3, wo zur besseren Darstellbarkeit zweidimensionale Parameter- und Eigenschaftsräume P bzw. E dargestellt sind, ist die Zahl der Parameter bei praktischen Problemen meist deutlich höher als die der Eigenschaften. Dies ist ein Hauptgrund dafür, dass die eindeutige Umkehrbarkeit der Analyseabbildung \mathbf{a} in der Regel nicht gegeben ist.

Analyseabbildungen nach Definition 2.10 sind nur für jeweils eine spezielle Schaltungstopologie gültig, da sich die Entwurfparameter \mathbf{p}_E erst nach Wahl einer Topologie definieren lassen. Bei der Synthese wird die Topologie meist getrennt von den Schaltungsparametern festgelegt (s. Abschnitt 2.3).

Definition 2.11 *Ein Analyseverfahren ist eine Implementierung einer Analyseabbildung.*

In den folgenden Abschnitten werden die gebräuchlichen Analyseverfahren auf der Grundlage der hier eingeführten Begriffe dargestellt.

2.2.3 Analysegleichungen

Die einfachste Möglichkeit, die Analyseabbildung \mathbf{a} zu definieren, ist die Verwendung expliziter Analysegleichungen. Dies ist jedoch nur in folgenden Fällen möglich:

- Die zu analysierende Schaltung und die verwendete Modellierung sind so einfach, dass sich ein explizit nach den gesuchten Eigenschaftsgrößen auflösbares Gleichungssystem ergibt.
- Es werden gezielte Vereinfachungen dieses Gleichungssystems vorgenommen, um die Auflösbarkeit nach den Eigenschaftsgrößen zu erreichen. Durch die Vereinfachungen treten Abweichungen im Analyseergebnis auf.
- Auf Basis einer bereits implementierten Analyseabbildung oder mit Hilfe von Messdaten wird ein empirisches, numerisches Modell erzeugt, das auflösbar ist. Auch hierbei entstehen in der Regel Fehler.

2 Automatischer Entwurf analoger Schaltungen

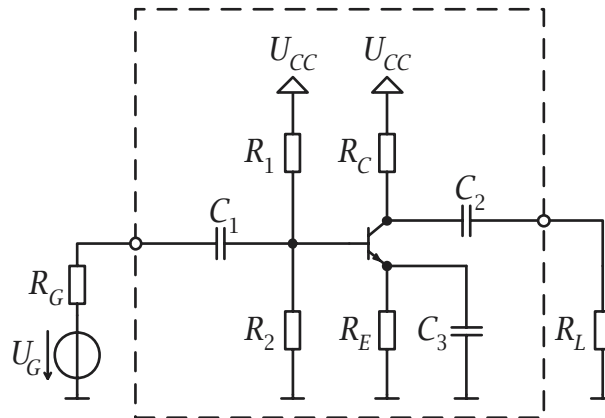


Abbildung 2.3: Verstärkerschaltung

In den ersten beiden Fällen wird eine physikalische Modellierung verwendet. Dies ergibt Gleichungen, die die physikalischen Zusammenhänge widerspiegeln, die Grundlage für die Funktion der Schaltung sind. Das Aufstellen, Umformen und Vereinfachen der Gleichungen kann entweder manuell oder durch rechnergestützte Werkzeuge zur so genannten symbolischen Analyse erfolgen (Gielen und Sansen 1991; Fernández u. a. 1998; ITWM 2002).

Bei der empirischen Modellierung wird die Struktur der Modellgleichungen vorgegeben. Die Koeffizienten der Gleichungen werden numerisch auf das zu modellierende Verhalten angepasst. Dazu werden auf nichtlinearer Regressionsanalyse basierende automatische Verfahren verwendet (Box u. a. 1978; Daems u. a. 2001).

Beispiel 2.4 Die Ableitung von Analysegleichungen soll an dem in Abbildung 2.3 gezeigten Verstärker gezeigt werden. Außerhalb der gestrichelten Umrandung ist eine Testbeschaltung mit einem Generator U_G mit Innenwiderstand R_G am Eingang und einer Last R_L am Ausgang des Verstärkers dargestellt. Als Verhaltensmodell für den Bipolartransistor werde das für den vorwärts-aktiven Betriebsbereich und niedrige Frequenzen vereinfachte Ebers-Moll-Modell (Muller und Kamins 1977) verwendet. Die Herstellungsprozessparameter des Modells sind die Stromverstärkung β , die Early-Spannung U_{Early} und der Sättigungsstrom I_S . Damit ist das Prozessparametertupel durch

$$\mathbf{ps} = (\beta, U_{Early}, I_S) \quad (2.5)$$

gegeben. Ferner verwendet das Modell die Temperatur T als Betriebsparameter. Da auch U_G , R_G und R_L veränderlich sind, werden diese ebenso

zu den Betriebsparametern gezählt. Damit ergibt sich für das Betriebsparametertupel

$$\mathbf{p}_B = (T, U_G, R_G, R_L) . \quad (2.6)$$

Zur besseren Übersichtlichkeit werden an Stelle von \mathbf{p}_S und \mathbf{p}_B nachfolgend weiterhin die üblichen Bezeichnungen ihrer Komponenten: β , U_{Early} , I_S , T , U_G , R_G bzw. R_L verwendet. Die Modellgleichungen sind damit

$$U_T = \frac{kT}{q} , \quad (2.7)$$

$$I_{CC} = I_S \left(e^{\frac{U_{BE}}{U_T}} - 1 \right) , \quad (2.8)$$

$$I_C = I_{CC} \left(1 + \frac{U_{CB}}{U_{\text{Early}}} \right) , \quad (2.9)$$

$$I_B = \frac{I_{CC}}{\beta} . \quad (2.10)$$

Die Entwurfsparameter der Verstärkerschaltung sind R_1 , R_2 , R_C und R_E sowie C_1 bis C_3 , so dass sich das Entwurfsparametertupel

$$\mathbf{p}_E = (R_1, R_2, R_C, R_E, C_1, C_2, C_3) \quad (2.11)$$

ergibt. Gesucht sind in diesem Beispiel die Kleinsignalverstärkung V sowie die differentiellen Ein- und Ausgangswiderstände r_{ein} bzw. r_{aus} . Das Schaltungseigenschaftstupel ist damit

$$\mathbf{e} = (V, r_{\text{ein}}, r_{\text{aus}}) . \quad (2.12)$$

Ebenso wie für \mathbf{p}_B und \mathbf{p}_E werden nachfolgend auch für \mathbf{p}_E und \mathbf{e} die gebräuchlichen Bezeichnungen ihrer Komponenten verwendet.

Trotz der vergleichsweise einfachen Modellierung lässt sich keine explizite Gleichung für den Kollektorstrom I_C im Arbeitspunkt angeben, ohne dass Näherungen durchgeführt werden. Wird U_{BE} als fest angenommen (z. B. 0,7V) und gilt weiterhin $U_{\text{Early}} \gg U_{CB}$, dann ergibt sich:

$$I_C = \frac{\beta (R_2 U_{CC} - (R_1 + R_2) U_{BE})}{(1 + \beta) (R_1 + R_2) R_E + R_1 R_2} . \quad (2.13)$$

Durch Linearisierung der Gleichungen 2.8 bis 2.10 werden die Kleinsignalparameter Steilheit S , Kollektor-Emitter-Widerstand r_{CE} und Basis-Emitter-Widerstand r_{BE} des Transistors bestimmt, wobei $e^{U_{BE}/U_T} \gg 1$ an-

2 Automatischer Entwurf analoger Schaltungen

genommen wird:

$$S = \frac{I_C}{U_T}, \quad (2.14)$$

$$r_{CE} = \frac{U_{Early}}{I_C}, \quad (2.15)$$

$$r_{BE} = \frac{\beta}{S}. \quad (2.16)$$

Unter der Voraussetzung, dass sich die Kapazitäten C_1 bis C_3 bei Signalfrequenzen wie ideale Kurzschlüsse verhalten, lassen sich damit Eingangs- und Ausgangswiderstand r_{ein} bzw. r_{aus} sowie die Kleinsignalverstärkung V der Schaltung berechnen:

$$r_{ein} = \frac{1}{\frac{1}{r_{BE}} + \frac{1}{R_1} + \frac{1}{R_2}}, \quad (2.17)$$

$$r_{aus} = \frac{1}{\frac{1}{r_{CE}} + \frac{1}{R_C}}, \quad (2.18)$$

$$V = -\frac{S}{\left(\frac{1}{r_{aus}} + \frac{1}{R_L}\right) \left(1 + \frac{R_G}{r_{ein}}\right)}. \quad (2.19)$$

Das Beispiel zeigt, dass selbst für einfache Probleme Näherungen und Vereinfachungen erforderlich sind, um explizite Analysegleichungen zu gewinnen. Integrierte Anlogschaltungen weisen eine deutlich höhere Transistorzahl auf. Wegen der ihrem Verhalten zugrunde liegenden nichtlinearen Zusammenhänge ist die Ableitung expliziter Analysegleichungen nur bei Anwendung grober Näherungen möglich. Daraus ergeben sich meist Abweichungen, die nicht mehr akzeptabel sind.

Die Verwendung expliziter Analysegleichungen ist jedoch aufgrund des gegenüber der Schaltungssimulation (s. Abschnitt 2.2.4) deutlich verringerten Ressourcenbedarfs für Fälle, in denen eine große Anzahl von Analysen durchgeführt werden muss, sehr attraktiv. Solche Fälle sind z. B. die verschachtelte Durchführung von Analysen innerhalb eines übergeordneten Verfahrens wie der Optimierung (s. Abschnitt 2.3.6) oder der Monte-Carlo-Simulation (s. Abschnitt 2.2.8). Die Einsparung an Ressourcen muss jedoch gegenüber der im Vergleich zur numerischen Simulation geringeren Genauigkeit abgewogen werden.

2.2.4 Simulation

Das meisteingesetzte Analysewerkzeug im Anlogschaltungsentwurf ist der Schaltungssimulator. Der klassische Schaltungssimulator, der als

Standard zu Referenzzwecken herangezogen wird und zu dessen Eingabesprache die meisten heutigen kommerziellen und akademischen Simulatoren kompatibel sind, ist SPICE (Nagel 1975). Die Eingabesprache von SPICE besteht aus einer Netzlistenbeschreibung der zu simulierenden Schaltung auf Bauelementeebene gefolgt von Simulationsanweisungen. Während die Verhaltensmodelle der Bauelemente in SPICE als Teil des Simulationsprogrammes fest eingebaut sind, erlauben moderne Simulatoren wie ADMS (Mentor 2003) und Spectre (Cadence 2003c) auch die Definition von Modellen in einer Verhaltensbeschreibungssprache. Diese Simulatoren lassen sich damit auf jeder Abstraktionsebene und auch für nichtelektrische und gemischte Simulationsdomänen einsetzen.

Definition 2.12 *Simulation ist die numerische Berechnung der Größen eines Vorganges mit Hilfe eines mathematischen Modells.*

Darstellung, Nachverarbeitung und Interpretation der Ergebnisse sind demnach nicht Teil der Simulation (s. Abschnitt 2.2.5). Die im Folgenden gezeigten graphischen Aufbereitungen der Simulationsergebnisse wurden mit einem getrennten Darstellungswerkzeug (waveform viewer) erzeugt.

Aufstellen der Systemgleichungen

Ein nach dem Muster von SPICE arbeitender Schaltungssimulator stellt zunächst aus der in der Eingabe vorliegenden Netzliste mit Hilfe der *modifizierten Knotenanalyse* ein Gleichungssystem bestehend aus Systemmatrix, einem Unbekannten- und einem Erregungsvektor auf (McCalla 1988; Spiro 1990; Vladimirescu 1994). Der Unbekanntenvektor enthält die als Simulationsergebnis zu berechnenden Knotenspannungen und Zweigströme. Zeitableitungen aufgrund von Kapazitäten, Induktivitäten oder innerhalb von Verhaltensmodellen werden wie nachfolgend beschrieben in Abhängigkeit von der Simulationsart ersetzt. Nichtlineare Zusammenhänge wie die Strom-Spannungsbeziehungen an Transistoren werden durch Unterprogrammaufruf der in den Simulator integrierten Verhaltensmodelle behandelt. Ergebnis der Unterprogrammaufrufe sind Linearisierungen, die in die Systemmatrix und den Erregungsvektor eingetragen werden. Damit steht eine ableitungsfreie, linearisierte Darstellung des nichtlinearen Gesamtsystems zur Verfügung.

Auf der Grundlage dieses Vorgehens können verschiedene Arten von Simulationen durchgeführt werden, von denen die gebräuchlichsten im Folgenden beschrieben sind.

2 Automatischer Entwurf analoger Schaltungen

Parameter	Wert
Transistortyp	2N2222
Stromverstärkung β	255,9
Early-Spannung U_{Early}	74,03 V
Sättigungsstrom I_S	14,34 fA
Temperaturspannung U_T	25,9 mV
Kapazität C_1	1 μ F
Kapazitäten C_2, C_3	10 μ F
Kollektorwiderstand R_C	2,2 k Ω
Emitterwiderstand R_E	1 k Ω
Basiswiderstand R_1	100 k Ω
Basiswiderstand R_2	47 k Ω
Generatorinnenwiderstand R_G	1 k Ω
Lastwiderstand R_L	1 k Ω
Versorgungsspannung U_{CC}	10 V

Tabelle 2.2: Parameter der Verstärkerschaltung

Arbeitspunktsimulation

Ziel der Arbeitspunktsimulation ist die Berechnung des als Arbeitspunkt bezeichneten Grenzwertes des Unbekanntenvektors für $t \rightarrow \infty$ bei ausschließlicher Erregung mit konstanten Signalen. Dazu werden in den Systemgleichungen alle Zeitableitungen zu Null und alle eingepprägten Zeitveränderlichen konstant gesetzt. Zur Lösung des nichtlinearen Problems wird ein iteratives Verfahren eingesetzt, das auf die o. a. Methode zur Linearisierung zurückgreift. Meist wird das Newton-Raphson-Verfahren verwendet. Eine Verbesserung der Konvergenzeigenschaften wird durch gedämpfte Varianten, Homotopieverfahren (Nocedal und Wright 1999) o. ä. erreicht.

Beispiel 2.5 *Der Arbeitspunkt des Verstärkers in Abbildung 2.3 wird für die in Tabelle 2.2 gegebenen Parameter mit dem Simulator PSpice (Cadence 2003a) simuliert. Es ergibt sich ein Kollektorstrom I_C von 2,10 mA. Analysegleichung 2.13 liefert dagegen 2,21 mA. Die bei Verwendung der Analysegleichung auftretende Abweichung beträgt also 5,2%.*

Die Abweichung ist darauf zurückzuführen, dass bei der Ableitung der Analysegleichung Näherungen durchgeführt worden sind, während der Simulator PSpice unvereinfachte Gleichungen verwendet.

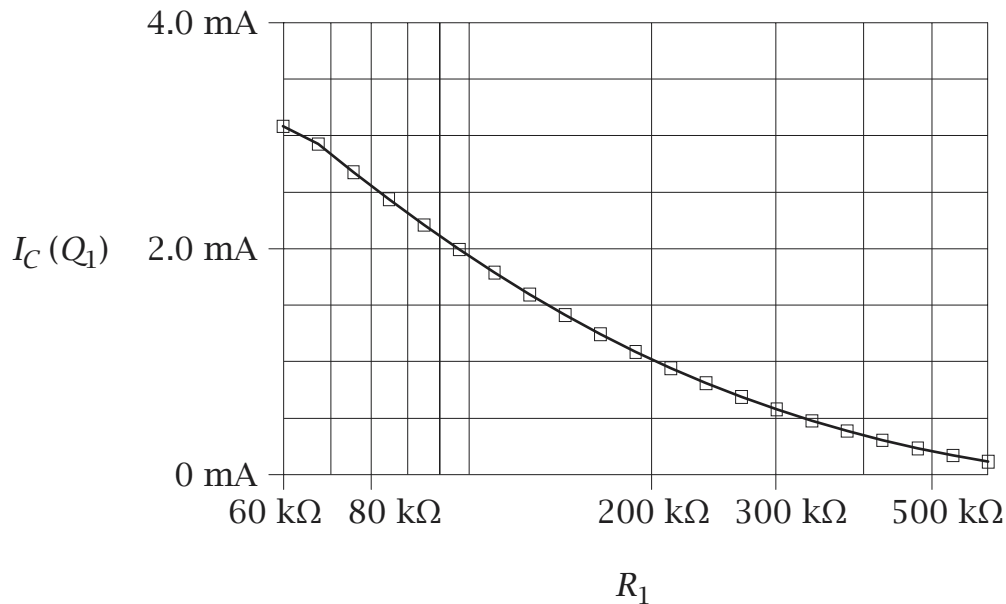


Abbildung 2.4: Ergebnis der Gleichstrom-Transfer-Simulation: Kollektorstrom I_C in Abhängigkeit von R_1

Gleichstrom-Transfer-Simulation

Die Gleichstrom-Transfer-Simulation (DC transfer, DC sweep) ist eine Erweiterung der Arbeitspunktsimulation, bei der die Änderung des Arbeitspunktes ohne Berücksichtigung dynamischer Effekte in Abhängigkeit einer Eingangsgröße oder eines Schaltungsparameters simuliert wird. Dazu werden punktweise Arbeitspunktsimulationen durchgeführt. Die Simulationsergebnisse werden anschließend graphisch in Form von Kennlinien der Unbekannten in Abhängigkeit des geänderten Parameters dargestellt. Bei der Gleichstrom-Transfer-Simulation wird ausgenutzt, dass eine solche Kennlinie in den meisten Fällen stetig verläuft. Dadurch ist beim Übergang von einem Simulationspunkt zum nächsten mit der letzten Lösung bereits ein sehr guter Startwert für den iterativen Gleichungslöser gegeben, so dass sich die Zahl der nötigen Iterationen erheblich reduziert.

Beispiel 2.6 Der Arbeitspunkt des Verstärkers in Abbildung 2.3 wird in Abhängigkeit von R_1 ($60 \text{ k}\Omega < R_1 < 600 \text{ k}\Omega$) an 21 logarithmisch verteilten Punkten für die in Tabelle 2.2 gegebenen Parameter als Gleichstrom-Transfer mit dem Simulator PSpice simuliert. Das Ergebnis für den Kollektorstrom I_C zeigt Abbildung 2.4. Die vom Simulator berechneten Punkte sind durch Markierungen hervorgehoben. Die dargestellte Kennlinie wird

2 Automatischer Entwurf analoger Schaltungen

vom Darstellungswerkzeug zwischen den Punkten interpoliert.

Wechselstromsimulation

Die Wechselstromsimulation dient der Untersuchung des frequenzabhängigen Kleinsignalverhaltens. Dazu muss grundsätzlich eine Arbeitspunktsimulation vorausgehen. An dem dort berechneten Arbeitspunkt werden wiederum nach dem o. a. Verfahren die Systemgleichungen linearisiert. Jedoch werden die Zeitableitungen dabei im Gegensatz zur Arbeitspunktberechnung nicht zu Null gesetzt. Das linearisierte System wird durch Fourier-Transformation in den Frequenzbereich übertragen. Eingeprägte Zeitveränderliche müssen daher mit Betrag und Phase im Frequenzbereich vorgegeben werden. Es ergibt sich ein komplexes lineares Gleichungssystem, das in der Frequenzvariablen ω parametrisiert ist. Dieses wird punktweise für verschiedene ω gelöst, so dass für die Unbekannten Punkte in der komplexen Ebene berechnet werden. Im Anschluss an die Simulation werden sie in Abhängigkeit der Frequenz als Bodediagramm bestehend aus Amplituden- und Phasengang oder als Ortskurve in der komplexen Ebene dargestellt.

Beispiel 2.7 Für den Verstärker in Abbildung 2.3 wird eine Wechselstromsimulation mit dem Simulator PSpice an 21 logarithmisch verteilten Punkten zwischen 100 Hz und 100 MHz durchgeführt. Dabei werden wieder die Parameter aus Tabelle 2.2 zugrunde gelegt. Der Generator liefert eine Spannung mit 1 V Amplitude und 0° Phase. Abbildung 2.5 stellt den sich ergebenden Amplituden- und Phasengang der Ausgangsspannung am Lastwiderstand R_L als Bodediagramm dar. Die Markierungen zeigen die vom Simulator berechneten Punkte. Für mittlere Frequenzen lässt sich eine Verstärkung von 31,3 dB ablesen. Analysegleichung 2.19 ergibt 32,5 dB. Das entspricht einer Abweichung von 1,2 dB im logarithmischen bzw. 15 % im linearen Maßstab.

Eine der Wechselstromsimulation verwandte Simulationsart ist die Rauschsimulation, bei der nicht mit konstanten komplexen Amplituden, sondern mit spektralen Rauschleistungsverteilungen gerechnet wird (Vladimirescu 1994).

Transientensimulation

Die Transientensimulation berechnet den Verlauf zeitlich begrenzter Vorgänge im Zeitbereich unter Berücksichtigung des dynamischen Groß-

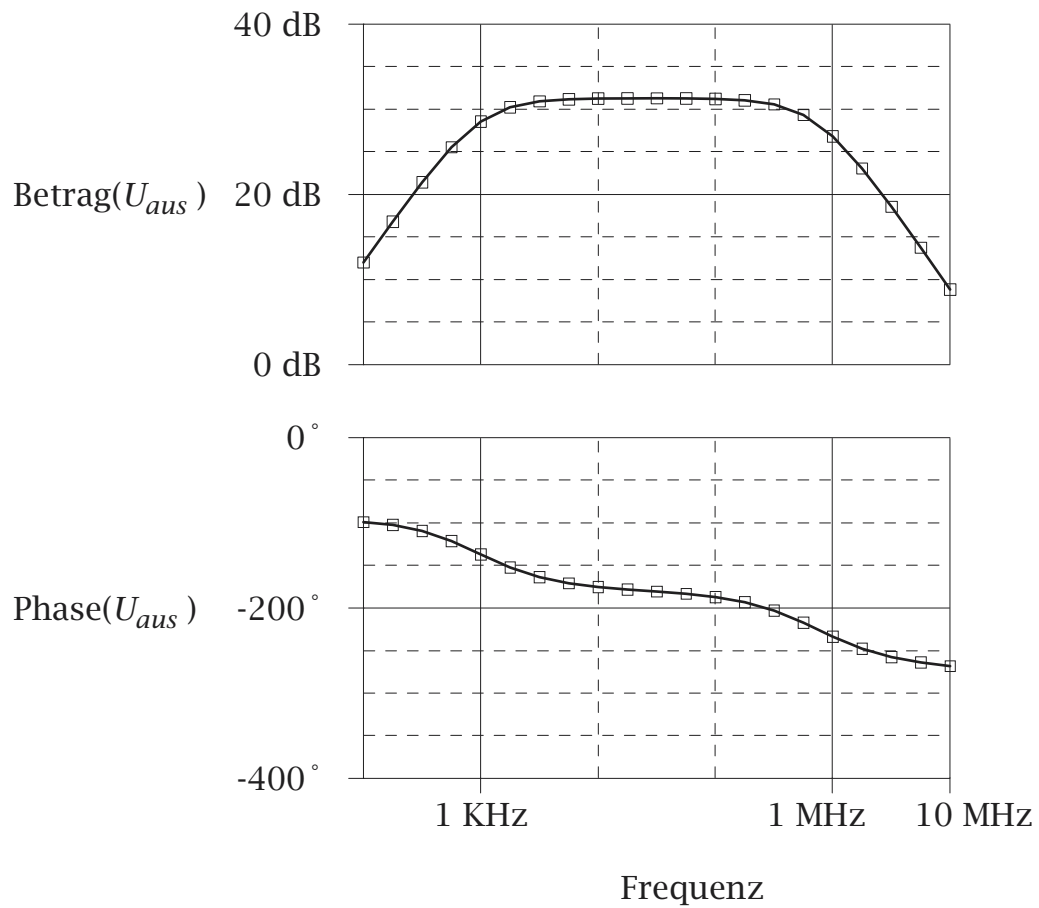


Abbildung 2.5: Ergebnis der Wechselstromsimulation: Amplituden- und Phasengang der Ausgangsspannung

2 Automatischer Entwurf analoger Schaltungen

signalverhaltens der Schaltung. Dazu wird das zu simulierende Zeitintervall durch eine Folge von Zeitpunkten ersetzt, an denen die Zeitableitungen in den Systemgleichungen mit Hilfe eines numerischen Integrationsverfahrens durch Integrationsformeln, z. B. Differenzenquotienten, ersetzt werden. Gebräuchliche Integrationsverfahren sind z. B. das implizite Euler-, das Trapez- und das Gear-Verfahren. Außerdem werden die Momentanwerte eingepprägter Zeitveränderlicher eingesetzt. Dadurch wird die Transientensimulationsaufgabe in eine Folge von mit dem Arbeitspunktsimulationsverfahren lösbaren Problemen überführt. Wie bei der Gleichstrom-Transfer-Simulation gilt auch hier, dass die Lösung des vorherigen Zeitpunktes als Startwert für den iterativen Gleichungslöser verwendet wird und sich die Zahl der nötigen Iterationen dadurch erheblich reduziert. Die Verteilung der Zeitpunkte im zu simulierenden Zeitintervall wird während der Simulation von einer Zeitschrittsteuerung vorgenommen. Die berechneten Werte der Unbekannten werden nach der Simulation graphisch als Zeitverläufe dargestellt.

Beispiel 2.8 *Es wird eine Transientensimulation von 0 bis 1 ms für den Verstärker in Abbildung 2.3 mit dem Simulator PSpice durchgeführt. Dabei werden eine symmetrische Generatorrechteckspannung von 0,1 V Amplitude mit einer Frequenz von 10 kHz und die Parameter aus Tabelle 2.2 verwendet. Der Zeitverlauf der Ausgangsspannung am Lastwiderstand R_L ist in Abbildung 2.5 dargestellt. Die Markierungen zeigen die Verteilung der simulierten Punkte aufgrund der Zeitschrittsteuerung.*

Komplexität

Zur Lösung der linearisierten Systeme werden in nach Spice-Muster arbeitenden Simulatoren Sparse-Matrix-Varianten des Gaußschen Eliminationsverfahrens eingesetzt. Die damit verbundene Rechenzeit- und Speicherbedarfskomplexität ist $\mathcal{O}(n^{1,5})$, wenn n die Anzahl der Knoten der Schaltung ist (Spiro 1990). Das hat zur Folge, dass Systemsimulationen auf Bauelementeebene, die bereits heute nicht mehr möglich sind, auch in Zukunft nicht möglich sein werden, wenn Rechenressourcen und Größe der zu simulierenden Systeme gleich schnell wachsen. Solche Systeme müssen daher bereits zur Simulation auf oberhalb der Bauelementeebene liegenden Ebenen durch abstrakte Verhaltensmodelle beschrieben werden. Aus der Komplexität folgt ferner, dass der Schaltungssimulator als Analysewerkzeug in Fällen, in denen eine große Anzahl von Simulationen durchgeführt werden muss, z. B. bei der Optimierung oder der Monte-Carlo-Simulation, auf Bauelementeebene nur

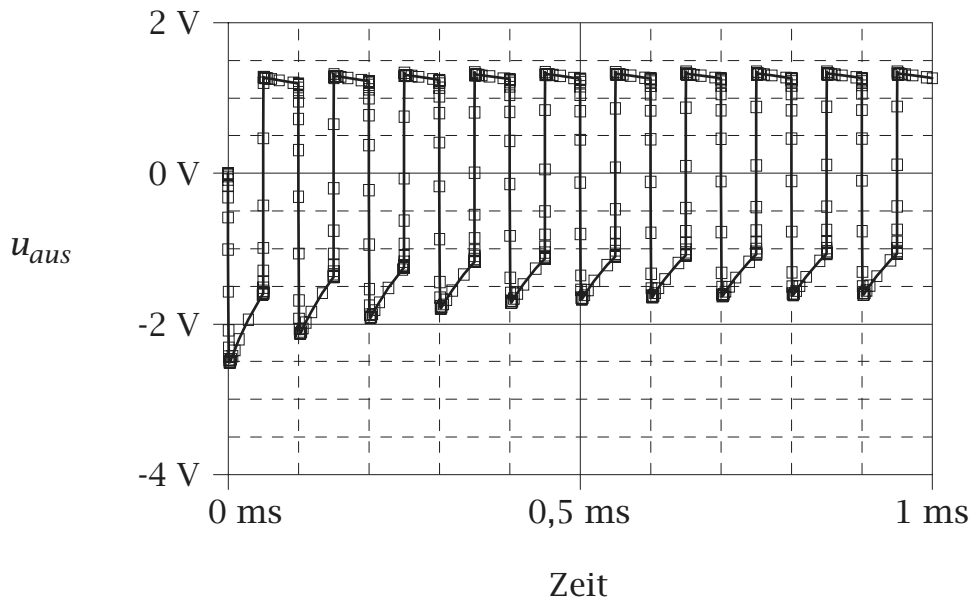


Abbildung 2.6: Ergebnis der Transientensimulation: Zeitverlauf der Ausgangsspannung

für kleine Problemgrößen einsetzbar ist. Alternativ können bei großen Problemen Analysegleichungen verwendet werden (vgl. Abschnitt 2.2.3). Ferner existieren für bestimmte Schaltungsklassen und Simulationstypen mit niedrigeren Genauigkeitsanforderungen angepasste Verfahren mit geringerer Laufzeitkomplexität, z. B. das Relaxationsverfahren für schwach gekoppelte Schaltungen mit eindeutigem Signalfluss und die Schalterebensimulation (switch-level simulation) für Digitalschaltungen in CMOS-Technik (Saleh und Newton 1990).

Simulation und Verifikation

Definition 2.13 *Verifikation ist der formale Beweis einer Aussage.*

Definition 2.14 *Falsifikation ist die formale Widerlegung einer Aussage.*

Falsifikation ist damit die Verifikation der Verneinung der Aussage.

Der Schaltungssimulator wird sehr oft als Werkzeug zur Überprüfung der Korrektheit eines Entwurfs eingesetzt, z. B. als Teil eines Charakterisierungsskripts (s. Abschnitt 2.2.7), das Zahlenwerte des Eigenschaftstupels e (vgl. Abschnitt 2.1.1) berechnet, die mit der Spezifikation verglichen werden können. Da ein nach dem beschriebenen Muster arbeitender Simulator in der Spezifikation gegebene Parameterintervalle

Simulationsart	Eigenschaftsgrößen
Wechselstromsimulation	Bandbreite Phasenreserve Verstärkung Verstärkungsbandbreiteprodukt
Transientensimulation	Anstiegsgeschwindigkeit Klirrfaktor Periodendauer Überschwingen Verlustleistung

Tabelle 2.3: Extrahierte Schaltungseigenschaftsgrößen

(Frequenz, Schaltungsparameter, Eingangsgrößen, Zeit etc.) grundsätzlich durch eine endliche Anzahl diskreter Punkte ersetzen muss, ist auch das Simulationsergebnis nur für diese Punkte gültig, nicht jedoch für das vorgegebene Intervall. Dadurch kommt es zu einer Unterschätzung der gesuchten Lösung (s. Abschnitt 2.2.6). Deshalb ermöglicht ein solches Vorgehen keine Verifikation, sondern nur eine Falsifikation der Korrektheit. Wird aus dem Misslingen einer endlichen Zahl von Falsifikationsversuchen auf Korrektheit geschlossen, so wird dies fälschlicherweise als *indirekte Verifikation* bezeichnet. Verfahren zur *formalen Verifikation* werden in Abschnitt 2.2.9 beschrieben.

2.2.5 Eigenschaftsextraktion

Wie in Abschnitt 2.2.4 dargestellt, liefert die Schaltungssimulation numerische Werte für Knotenspannungen und Zweigströme in einem Arbeitspunkt oder in Abhängigkeit von einem Parameter. Vielfach sind die gesuchten Schaltungseigenschaften e abstrakter (vgl. Beispiel 2.1) und müssen erst in einem Nachverarbeitungsschritt aus den Simulationsergebnissen extrahiert werden. Tabelle 2.3 listet für die Simulationsarten Wechselstromsimulation und Transientensimulation beispielhaft einige Schaltungseigenschaftsgrößen auf, die aus ihren Ergebnissen extrahiert werden können.

Formal müssen daher zur Durchführung der Analyseabbildung a eine *Simulationsabbildung* und eine *Extraktionsabbildung* in Folge durchgeführt werden. Zu ihrer Definition muss zunächst der *Simulationsergebnisraum* eingeführt werden:

Definition 2.15 Der Simulationsergebnisraum E_S ist die Menge aller möglichen Simulationsergebnisse.

Die Gestalt des Raumes E_S hängt vom verwendeten Simulator und der gewählten Simulationsarten ab. Mit seiner Hilfe lassen sich die Simulations- und die Extraktionsabbildung definieren:

Definition 2.16 Eine Simulationsabbildung ist eine Abbildung $\mathbf{a}_S : P \rightarrow E_S$ vom Parameterraum P in einen Simulationsergebnisraum E_S .

Definition 2.17 Eine Extraktionsabbildung ist eine Abbildung $\mathbf{a}_E : E_S \rightarrow E$ von einem Simulationsergebnisraum E_S in den Schaltungseigenenschaftsraum E .

Die durch Simulation und Extraktion gegebene Analyseabbildung ist damit $\mathbf{a} = \mathbf{a}_E \circ \mathbf{a}_S$.

Die Durchführung der Extraktionsabbildung \mathbf{a}_E ist durch graphische Darstellung des Simulationsergebnisses und manuelle Interpretation möglich (vgl. Beispiel 2.7). Automatische Extraktionswerkzeuge bieten dagegen höhere Genauigkeit und exakte Wiederholbarkeit. Außerdem ermöglichen sie eine erhebliche Reduktion des manuellen Aufwandes, wenn die Anzahl der durchzuführenden Analysen groß ist.

Beispiel 2.9 Aus dem Ergebnis der in Beispiel 2.7 durchgeführten Wechselstromsimulation wird die 3-dB-Bandbreite mit einem automatischen Extraktionswerkzeug bestimmt. Es ergibt sich eine Bandbreite von 739 kHz.

2.2.6 Worst-Case-Analyse

Eine Spezifikation gilt in den meisten Fällen nicht nur für den Nominalparameterpunkt $\mathbf{p}_{\text{nom,BS}}$ der Betriebs- und Prozessparameter \mathbf{p}_B bzw. \mathbf{p}_S , sondern für die Gebiete $P_{B,\text{Spez}}$ und $P_{S,\text{Spez}}$, die durch die Spezifikation definiert sind (vgl. Abschnitt 2.2.1). Oft wird $P_{S,\text{Spez}} = P_S$ gesetzt, eine Wahrscheinlichkeitsdichtefunktion auf P_S gegeben und die Fertigungsausbeute mit einem übergeordneten Verfahren als Eigenschaftsgröße berechnet (s. Abschnitte 2.2.8 und 2.3.9). In diesem Fall bleibt \mathbf{p}_S bei der Worst-Case-Analyse ein variabler Parameter, von dem das Ergebnis abhängt, und es wird im Folgenden ausschließlich das Gebiet $P_{B,\text{Spez}}$ betrachtet.

Zur Überprüfung der Einhaltung der Spezifikation ist es erforderlich, die Extrema der Eigenschaftsgrößen auf $P_{B,\text{Spez}} \times P_{S,\text{Spez}}$ bzw. $P_{B,\text{Spez}}$

2 Automatischer Entwurf analoger Schaltungen

durch Analyse zu bestimmen. Das so bestimmte extremale Verhalten besteht aus den Minima und Maxima der Schaltungseigenschaftsgrößen und wird als *Worst-Case-Verhalten* bezeichnet. Erfüllt das Worst-Case-Verhalten die Spezifikation, so ist bewiesen, dass die Schaltung die Spezifikation erfüllt.

Für die zwei möglichen Fälle, dass das Worst-Case-Verhalten auf $P_{S,Spez}$ oder mit \mathbf{p}_S als Parameter berechnet wird, lassen sich formal zwei entsprechende Abbildungen definieren:

Definition 2.18 Die Worst-Case-Analyse-Abbildung $\mathbf{a}_{WC} : P_E \rightarrow E^2$ zur Bestimmung des Worst-Case-Verhaltens auf $P_{B,Spez} \times P_{S,Spez}$ für eine Analyseabbildung \mathbf{a} ist gegeben durch

$$\mathbf{a}_{WC}(\mathbf{p}_E) = (\mathbf{a}_{min}, \mathbf{a}_{max}) , \quad (2.20)$$

worin

$$\mathbf{a}_{min} = \min_{\mathbf{p}_B \in P_{B,Spez}, \mathbf{p}_S \in P_{S,Spez}} \mathbf{a}(\mathbf{p}_E, \mathbf{p}_B, \mathbf{p}_S) , \quad (2.21)$$

$$\mathbf{a}_{max} = \max_{\mathbf{p}_B \in P_{B,Spez}, \mathbf{p}_S \in P_{S,Spez}} \mathbf{a}(\mathbf{p}_E, \mathbf{p}_B, \mathbf{p}_S) . \quad (2.22)$$

Definition 2.19 Die Worst-Case-Analyse-Abbildung $\mathbf{a}_{WC,S} : P_E \times P_S \rightarrow E^2$ zur Bestimmung des Worst-Case-Verhaltens auf $P_{B,Spez}$ für eine Analyseabbildung \mathbf{a} , wobei \mathbf{p}_S ein Parameter ist, ist gegeben durch

$$\mathbf{a}_{WC,S}(\mathbf{p}_E, \mathbf{p}_S) = (\mathbf{a}_{min}, \mathbf{a}_{max}) \quad (2.23)$$

mit

$$\mathbf{a}_{min} = \min_{\mathbf{p}_B \in P_{B,Spez}} \mathbf{a}(\mathbf{p}_E, \mathbf{p}_B, \mathbf{p}_S) , \quad (2.24)$$

$$\mathbf{a}_{max} = \max_{\mathbf{p}_B \in P_{B,Spez}} \mathbf{a}(\mathbf{p}_E, \mathbf{p}_B, \mathbf{p}_S) . \quad (2.25)$$

Eine exakte Bestimmung des Worst-Case-Verhaltens ist numerisch nicht möglich. Bei der Berechnung sind zwei Fälle von besonderer Bedeutung:

Unterschätzung: Das berechnete Maximum liegt *unter* dem tatsächlichen Maximum und das berechnete Minimum *über* dem tatsächlichen.

Überschätzung: Das berechnete Maximum liegt *über* dem tatsächlichen Maximum und das berechnete Minimum *unter* dem tatsächlichen.

Um zu dem Schluss gelangen zu können, dass eine in der Spezifikation gegebene Ungleichungsnebenbedingung formal erfüllt ist, sind garantierte obere Schranken der Maxima und untere Schranken der Minima des Worst-Case-Verhaltens erforderlich. Für eine Ungleichungsnebenbedingung $e_i > e_{i,Spez,min}$ und eine untere Schranke $e_{i,schranke} \leq e_{i,WC,min}$ des Worst-Case-Verhaltens der Eigenschaftsgröße e_i kann, wenn $e_{i,schranke} > e_{i,Spez,min}$, der Schluss wie folgt geführt werden:

$$\begin{aligned} e_{i,schranke} \leq e_{i,WC,min} \quad \wedge \quad e_{i,schranke} > e_{i,Spez,min} \\ \Rightarrow e_{i,WC,min} > e_{i,Spez,min} . \end{aligned} \quad (2.26)$$

Die letzte Relation besagt, dass die Spezifikation in Bezug auf die Ungleichungsnebenbedingung $e_i > e_{i,Spez,min}$ eingehalten wird. Für Ungleichungsnebenbedingungen der Form $e_i < e_{i,Spez,max}$ lässt sich das dargestellte Vorgehen entsprechend anwenden.

Ohne obere Schranken der Maxima und untere Schranken der Minima ist der Schluss 2.26 für beide Typen von Ungleichungsnebenbedingungen jedoch nicht möglich. Die Ergebnisse einer numerischen Berechnung des Worst-Case-Verhaltens sind demnach nur dann als Schranken verwendbar, wenn sie entweder exakt sind oder der Fall der Überschätzung vorliegt. Keinesfalls ist dies bei Unterschätzung möglich.

Im folgenden werden einige gebräuchliche Verfahren zur Worst-Case-Berechnung dargestellt.

Parameterdurchlauf

Beim Parameterdurchlauf (parameter sweep) wird die Analyseabbildung a für eine endliche Zahl von Punkten in $P_{B,Spez} \times P_{S,Spez}$ bzw. $P_{B,Spez}$ durchgeführt. Diese sind meist äquidistant verteilt. Von der Menge der Bildpunkte in E werden komponentenweise Maxima und Minima bestimmt und als genähertes Worst-Case-Verhalten verwendet. Dieser Ansatz führt zu einer systematischen Unterschätzung im Ergebnis, das gegen das exakte Worst-Case-Verhalten konvergiert, wenn die Anzahl der Punkte gegen unendlich geht. Die Laufzeitkomplexität ist hierbei $\mathcal{O}(b^n)$, also exponentiell, wenn n die Anzahl der variierten Parameter und b die der Punkte je Parameterichtung ist, da die Unterteilung für jeden Parameter durchgeführt wird und sich die abzubildende Punktmenge aus dem kartesischen Produkt ergibt.

$R_4/\text{k}\Omega$	I_C/mA
40	3,0818
47	3,0848
54	3,0842
61	3,0747

Tabelle 2.4: Ergebnis des Parameterdurchlaufs

Beispiel 2.10 Für den Verstärker in Abbildung 2.3 wird eine Worst-Case-Analyse der Eigenschaft $e = I_C$ mit einem Prozessparameter $p_S = R_4$ und $40 \text{ k}\Omega \leq R_4 \leq 61 \text{ k}\Omega$ durchgeführt. Für alle anderen Parameter werden die in Tabelle 2.2 gegebenen festen Werte angenommen. Dazu wird für p_S ein Parameterdurchlauf mit 4 äquidistanten Punkten mit dem Simulator PSpice simuliert. Tabelle 2.4 zeigt das Ergebnis des Parameterdurchlaufs. Hieraus lassen sich ein Minimum von $e_{WC,\min} = 3,0747 \text{ mA}$ und ein Maximum von $e'_{WC,\max} = 3,0848 \text{ mA}$ ablesen.

Eine genauere Analyse ergibt jedoch ein Maximum von $e_{WC,\max} = 3,0850 \text{ mA}$ zwischen den simulierten Punkten $p_S = 47 \text{ k}\Omega$ und $p_S = 54 \text{ k}\Omega$. Das mit dem Parameterdurchlauf erzielte Ergebnis ist also unterschätzt.

Eckenanalyse

Um beim Parameterdurchlauf die Anzahl der abzubildenden Punkte zu reduzieren, kann eine Heuristik verwendet werden, die ausnutzt, dass die Eigenschaftsgrößen auf $P_{B,\text{Spez}} \times P_{S,\text{Spez}}$ bzw. $P_{B,\text{Spez}}$ oft monoton verlaufen. Dann treten die gesuchten Extrema ausschließlich in den Ecken dieser Gebiete auf, die bei der Eckenanalyse mit \mathbf{a} abgebildet werden. Da die Zahl der Ecken 2^n ist, wenn n die Anzahl der variierten Parameter ist, bleibt die Laufzeitkomplexität gegenüber dem Parameterdurchlauf jedoch unverändert. Auch die Eckenanalyse liefert Ergebnisse mit systematischer Unterschätzung, falls die Annahme der Monotonie nicht zutrifft.

Beispiel 2.11 Die in Beispiel 2.10 dargestellte Worst-Case-Analyse-Aufgabe wird mit der Methode der Eckenanalyse behandelt. Die zu analysierenden Ecken können Tabelle 2.4 entnommen werden: $p_S = R_4 = 40 \text{ k}\Omega$ und $p_S = R_4 = 61 \text{ k}\Omega$. Die zugehörigen Werte für $e = I_C$ sind $3,0818 \text{ mA}$ bzw. $3,0747 \text{ mA}$. Gegenüber der in Beispiel 2.10 erreichten unterschätzten Näherung des Maximums $e'_{WC,\max} = 3,0848 \text{ mA}$ ergibt sich

also eine Verstärkung der Unterschätzung auf 3,0818 mA.

Eine Verringerung der Laufzeitkomplexität auf $\mathcal{O}(n)$ lässt sich erzielen, indem zunächst im Nominalparameterpunkt $\mathbf{p}_{\text{nom,BS}}$ der aus n Ableitungen bestehende Gradient für jede Komponente von \mathbf{a} bestimmt wird (Cadence 2003b). Seine Richtung gibt die Lage der Ecken an, in denen sich die Extrema befinden. Liegt keine Monotonie vor, so ist die Unterschätzung jedoch im Allgemeinen erheblich stärker als bei Betrachtung aller Ecken.

Beispiel 2.12 In Beispiel 2.11 ergibt sich für $p_{\text{nom,S}} = 49 \text{ k}\Omega$ ein Gradient $\frac{de}{dp_S} = 3,57 \cdot 10^{-10} \text{ A}/\Omega$. Demnach wird die Lage des Maximums in der Ecke $p_S = 61 \text{ k}\Omega$ angenommen, was zu einer weiteren Erhöhung der Unterschätzung auf 3,0747 mA führt.

Die Beispiele 2.11 und 2.12 zeigen, dass eine fälschliche Annahme monotoner Verläufe der betrachteten Eigenschaften \mathbf{e} in den Betriebs- und Prozessparameterbereichen $P_{BS,Spez}$ bei der Eckenanalyse zu einer deutlichen Erhöhung der Unterschätzung gegenüber der Methode des Parameterdurchlaufes führen kann.

Ansatz mit formalen Methoden

Die o. a. Methoden zur Berechnung von Näherungen für das Worst-Case-Verhalten eignen sich nicht als Grundlage zur formalen Überprüfung der Einhaltung der Spezifikation mit Folgerung 2.26, da sie systematisch unterschätzte Ergebnisse liefern. Sie werden dennoch in der Praxis verwendet, wobei es durch die Unterschätzung zu Spezifikationsverletzungen kommen kann. Bei Verwendung klassischer Schaltungssimulatoren gibt es jedoch keine Alternative, da diese grundsätzlich nur punktweise Analysen durchführen können (vgl. Abschnitt 2.2.4).

Mit formalen Methoden auf der Basis von Intervallarithmetik (s. Abschnitt 2.2.9) oder affiner Arithmetik lassen sich Schranken berechnen, die die o. a. Forderungen erfüllen. Davon wird auch in dem Verfahren dieser Arbeit Gebrauch gemacht (s. Abschnitt 3.3.2). Ein Ansatz mit affiner Arithmetik ausschließlich für explizite Analysegleichungen ist von (Femia und Spagnuolo 2000) vorgestellt worden.

2.2.7 Charakterisierung

Die in den vorhergehenden Abschnitten beschriebenen Vorgänge: Simulation, Extraktion und Worst-Case-Analyse werden im Entwurf üblicher-

2 Automatischer Entwurf analoger Schaltungen

weise verkettet ausgeführt, um die Eigenschaften einer Schaltung bei Parameterschwankungen zu erfassen und anschließend mit der Spezifikation zu vergleichen. Beim vorherrschenden Handentwurf, aber auch bei der analysebasierten Synthese von Analogschaltungen werden diese Schritte sehr oft in gleicher Weise durchlaufen. Sie können daher zu einem übergeordneten Charakterisierungsvorgang zusammengefasst werden. Der Ablauf wird dabei von einem in einer Hochsprache erstellten *Charakterisierungsskript* gesteuert. Dabei wird auch das Anschließen von verschiedenen Testschaltungen an die zu untersuchende Schaltung automatisiert. Da die durchzuführenden Analyseaufgaben bei Schaltungen der gleichen Schaltungsklasse meist identisch sind, lassen sich schaltungsklassenspezifische Charakterisierungsskripte aufstellen und in einer Bibliothek speichern.

Charakterisierungsskripte erhöhen durch die erzielte Automatisierung nicht nur die Entwurfsproduktivität. Sie sind auch Voraussetzung für die Verwendung von simulationsbasierten Analyseverfahren als Grundlage für die automatische Synthese.

2.2.8 Ausbeuteanalyse

Durch den Fertigungsvorgang werden die Herstellungsprozessparameter \mathbf{p}_S einer integrierten Schaltung mit festen Werten innerhalb von P_S belegt. Die statistische Verteilung auf P_S wird für jeden einzelnen Herstellungsprozess empirisch bestimmt und kann formal als Wahrscheinlichkeitsdichtefunktion $w : P_S \rightarrow \mathbb{R}$ angegeben werden. Es gibt zwei verschiedene Ansätze, den Einfluss der Schwankungen der Herstellungsprozessparameter auf die Schaltungseigenschaftsgrößen zu analysieren: durch Worst-Case-Analyse oder Ausbeuteanalyse.

Herstellungsprozessparameterschwankungen in der Worst-Case-Analyse

Wie in Abschnitt 2.2.6 beschrieben, können angenommene maximale Herstellungsprozessparameterschwankungen, die als Gebiet $P_{S,Spez}$ gegeben sind, zusammen mit dem Gebiet $P_{B,Spez}$ als Ausgangspunkt einer Worst-Case-Analyse verwendet werden. Aus dem Gebiet $P_{S,Spez}$ lässt sich mit Hilfe von w die korrespondierende Ausbeute Y mit folgendem Integral bestimmen:

$$Y = \int_{P_{S,Spez}} w(\mathbf{p}_S) dP_S . \quad (2.27)$$

Für eine spezifizizierte Mindestausbeute Y_{min} ist umgekehrt das zugehörige Gebiet $P_{S,Spez}$ festlegbar. Häufig wird ein Gebiet gewählt, das die 3σ -Grenzen einer durch w gegebenen Gauß-Verteilung einschließt. Dieses entspricht einem Y_{min} von 99% (Ochotta u. a. 1998). Wird durch Worst-Case-Analyse sichergestellt, dass die Spezifikation auf dem gewählten $P_{S,Spez}$ eingehalten wird, so ist auch die Herstellungsausbeute Y_{min} garantiert. Es ist jedoch nicht möglich, die Ausbeute auf diese Weise zu maximieren.

Ausbeuteanalyse

Alternativ dazu kann die Worst-Case-Analyse mit \mathbf{p}_S als Parameter durchgeführt werden. Dadurch lässt sich die Ausbeute Y als Funktion der Entwurfparameter \mathbf{p}_E bestimmen:

$$Y(\mathbf{p}_E) = \int_{P_S} w(\mathbf{p}_S) \alpha(\mathbf{p}_E, \mathbf{p}_S) dP_S, \quad (2.28)$$

wobei

$$\alpha(\mathbf{p}_E, \mathbf{p}_S) = \begin{cases} 1, & \text{falls } \mathbf{a}_{WC,S}(\mathbf{p}_E, \mathbf{p}_S) \in A_E, \\ 0 & \text{sonst.} \end{cases} \quad (2.29)$$

Hierin ist A_E das Akzeptanzgebiet im Schaltungseigenschaftsraum (vgl. Abschnitt 2.1.2), d. h. die Akzeptanzfunktion α ist 1, wenn die Spezifikation erfüllt ist.

Die so berechnete Ausbeute Y kann mit einer spezifizierten Mindestausbeute Y_{min} verglichen werden. Darüber hinaus bietet die Funktion $Y(\mathbf{p}_E)$ eine direkte Darstellung des Zusammenhanges zwischen den Entwurfparametern \mathbf{p}_E und der Ausbeute Y . Dadurch ist es möglich, die Ausbeute durch Wahl der Entwurfparameter zu maximieren (s. Abschnitt 2.3.8).

Monte-Carlo-Simulation

Die Monte-Carlo-Simulation ist das Standardverfahren zur Ausbeuteanalyse. Hierbei wird der Herstellungsprozess simuliert, indem eine endliche Zahl N Herstellungsprozessparametersätze $\mathbf{p}_{S,n}$ (samples) durch einen Zufallsgenerator entsprechend der Verteilungsfunktion w erzeugt

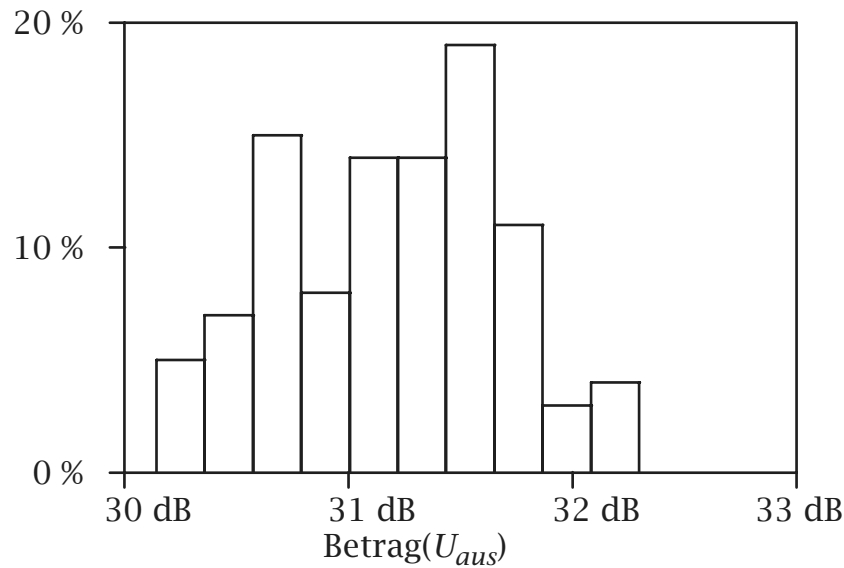


Abbildung 2.7: Histogramm der Verstärkung

werden. Durch Abbildung der $\mathbf{p}_{s,n}$ mit $\mathbf{a}_{wC,s}$ wird eine Näherung der statistischen Eigenschaftsgrößenverteilung in E berechnet. Zur Ausbeuteanalyse wird die in Formel 2.29 definierte Akzeptanzfunktion α ausgewertet und hieraus eine Näherung \tilde{Y} des Integrals 2.28 berechnet:

$$\tilde{Y}(\mathbf{p}_E) = \frac{1}{N} \sum_{n=1}^N \alpha(\mathbf{p}_E, \mathbf{p}_{s,n}) . \quad (2.30)$$

Beispiel 2.13 *Mit dem im Simulator PSpice integrierten Monte-Carlo-Simulator wird eine Simulation der Verstärkung des in Abbildung 2.3 gegebenen Verstärkers durchgeführt. Es werden die Parameter aus Tabelle 2.2 verwendet, wobei für die Widerstände R_1 bis R_4 eine Toleranz von je 5% und für die in der Tabelle aufgelisteten Transistorparameter eine Toleranz von 10% angenommen wird, jeweils mit Gauß-Normalverteilung. Die mit $N = 100$ berechnete statistische Verteilung der Verstärkung ist im Histogramm in Abbildung 2.7 dargestellt. Bei einer spezifizierten Mindestverstärkung von 31 dB ergibt sich aus Formel 2.30 eine simulierte Ausbeute von 65%.*

Bei endlichem N erlaubt das Ergebnis \tilde{Y} von Summe 2.30 jedoch keine formal garantierte Aussage über den Wert der mit Integral 2.28 definierten exakten Ausbeute Y . \tilde{Y} liegt in einem statistischen Vertrauensintervall, dessen Breite proportional zu $1/\sqrt{N}$ ist (Spence und Soin 1988).

2.2.9 Formale Verifikation

Die Schwäche der verbreiteten simulationsbasierten Analyseverfahren, grundsätzlich keine formal garantierte Aussage über die Einhaltung der Spezifikation liefern zu können (vgl. Abschnitte 2.2.4 und 2.2.6) hat zur Entwicklung einer neuen Klasse von Analyseverfahren geführt: der Verfahren zur formalen Verifikation. Diese Entwicklung hat im Bereich des Digitalschaltungsentwurfs bereits zu einer Verbreitung im praktischen Einsatz entsprechender Werkzeuge geführt.

Verfahren zur formalen Verifikation lassen sich in drei Unterklassen einteilen:

Äquivalenzüberprüfung (equivalence checking): Überprüfung vollständiger funktionaler Übereinstimmung von Implementierung und Spezifikation oder zweier Implementierungen.

Modellüberprüfung (model checking): Überprüfung einzelner Eigenschaften eines Systems.

Theorembeweis (theorem proving): Halbautomatische Führung komplexer logischer Beweise.

Nur für die Äquivalenzüberprüfung (Hedrich 1997) und die Modellüberprüfung (Hartong u. a. 2002) gibt es bereits Ansätze für den Analogschaltungsentwurf. Diese stützen sich teilweise auf Intervallarithmetik, um zu formal garantierten Aussagen zu gelangen.

Aufbauend auf formaler Verifikation wird in dieser Arbeit ein Dimensionierungsverfahren für Analogschaltungen, das Ergebnisse mit formal garantierten Eigenschaften berechnet, entwickelt (s. Kapitel 3).

2.3 Synthese

Wie in Abschnitt 1.2.1 beschrieben, werden bei der Synthese von Analogschaltungen die Schritte Schaltungs- und Layoutsynthese nacheinander ausgeführt. Das im Mittelpunkt dieser Arbeit stehende Problem der Dimensionierung ist Teil der Schaltungssynthese, die im Folgenden beschrieben wird. Der Vorgang der Layoutsynthese ist hiervon klar trennbar und wird hier nicht weiter betrachtet.

Das Problem der Schaltungssynthese mit Spezifikationen in der in Abschnitt 2.1 beschriebenen allgemeinen Form gehört zur Klasse der gemischt-kombinatorisch-kontinuierlichen nichtlinearen Optimierungsprobleme (MINLP): es muss eine Topologie erzeugt werden und ihre

Entwurfsparameter sind zu bestimmen. MINLP-Probleme gehören zur Problemklasse \mathcal{NP} (Garey und Johnson 1979). Das Problem wird daher meist nach dem Teile-und-herrsche-Prinzip in die Probleme Topologiesynthese (kombinatorisches Optimierungsproblem, IP) und Dimensionierung (nichtlineares Optimierungsproblem, NLP) aufgeteilt, die nacheinander bearbeitet werden. Dadurch muss auf Optimalität bei der Lösung des Schaltungssyntheseproblems verzichtet werden. Die entstandenen Subprobleme gehören jedoch immer noch zu \mathcal{NP} . Die in den folgenden Abschnitten dargestellten Lösungsverfahren nehmen deshalb Einschränkungen der Problemklasse oder -größe vor oder verwenden Heuristiken.

Ansätze zur Lösung des Schaltungssyntheseproblems als Ganzes werden in Abschnitt 2.3.10 beschrieben.

2.3.1 Topologieselektion

Die Topologieselektion ist ein Bibliotheksansatz zur Lösung des Topologiesyntheseproblems, der auf der Wiederverwendung (Reuse) von einmal erstellten Topologien, teilweise unter Verwendung von Hierarchie (z. B. Harjani u. a. 1989), aufbaut. Der Nachteil eines solchen Ansatzes besteht darin, dass die Bibliothek starr ist. Infolge des rasanten technologischen Fortschritts werden z. B. die Versorgungsspannungen kontinuierlich weiter abgesenkt, und die Transistoreigenschaften ändern sich durch immer kleinere Strukturgrößen. Dies erfordert neue, angepasste Topologien. Eine Bibliothek ist daher sehr schnell veraltet und unbrauchbar. Neben der kurzen Lebensdauer entsteht zusätzlich der hohe Erstellungsaufwand einer umfassenden Bibliothek.

Beispiel 2.14 *In Abbildung 2.8 sind als Ausschnitt einer Bibliothek drei verschiedene Topologien für Stromspiegel dargestellt. Topologie 2.8(a) weist einen geringen Flächenverbrauch und eine kleine minimale Ausgangsspannung auf. Demgegenüber ist der Ausgangswiderstand in 2.8(b) durch die Hinzufügung einer Kaskodestufe höher, jedoch erhöht sich die minimale Ausgangsspannung und der Flächenverbrauch ist größer als in 2.8(a). Durch Hinzufügen eines Spannungsverschiebers (level shifter) kann die minimale Ausgangsspannung in 2.8(c) bei gleichem Ausgangswiderstand gegenüber 2.8(b) reduziert werden. Jedoch erhöht sich der Flächenverbrauch weiter und die Verlustleistung steigt.*

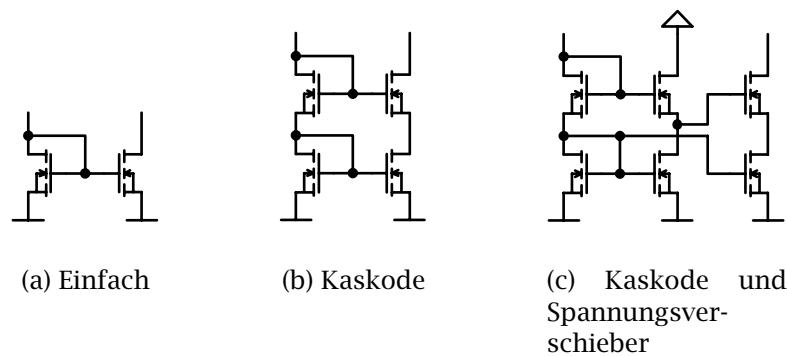


Abbildung 2.8: Stromspiegeltopologien

Automatische Topologieselektion

Bei gegebener Bibliothek bestehen die Schwierigkeiten der Topologieselektion in der Bewertung der Topologien hinsichtlich der in der Spezifikation geforderten Eigenschaften und in der Auswahl der besten Topologie. Dabei darf entsprechend dem Teile-und-herrsche-Ansatz der Dimensionierungsschritt nicht ausgeführt werden. Automatische Verfahren zur Selektion lassen sich in zwei Klassen einteilen: wissensbasierte und numerische Verfahren.

Wissensbasierte Verfahren sind zeitlich der ersten Generation automatischer Topologiesynthesewerkzeuge zuzuordnen. Es werden Entscheidungsbäume (z. B. Koh u. a. 1990) oder Regelsätze (z. B. El-Turky und Perry 1989; Swings u. a. 1991a) verwendet, um eine Topologie auszuwählen. Hierzu muss vorab Expertenwissen in Form von Heuristiken festgehalten und kodiert werden. Die Bäume bzw. Regeln werden der Topologiebibliothek hinzugefügt. Nach weitaus kürzerer Zeit als die Topologien selbst veralten auch diese Heuristiken bei Technologiewechsel und werden ungültig. Es hat sich in der Praxis gezeigt, dass der hohe Kodierungsaufwand den kurzzeitigen Nutzen weit übersteigt.

Beispiel 2.15 *Es wird für die Stromspiegeltopologien in Abbildung 2.8 ein Entscheidungsbaum zur Selektion entsprechend der in Beispiel 2.14 angeführten Eigenschaften aufgestellt. Der in Abbildung 2.9 gezeigte Baum führt bei Vorgabe der Eigenschaften Ausgangswiderstand und Ausgangsspannung zur Auswahl der Topologie mit dem geringsten Flächen- und Stromverbrauch.*

Dieses Beispiel zeigt ein typisches Problem bei der Aufstellung von Heuristiken auf: Für die praktische Anwendbarkeit müssen quantitative

2 Automatischer Entwurf analoger Schaltungen

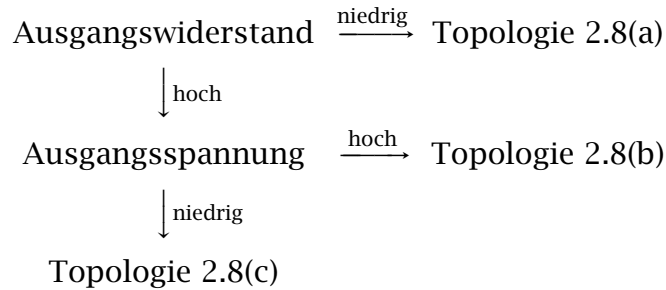


Abbildung 2.9: Entscheidungsbaum für Stromspiegeltopologien

Grenzen für qualitative Begriffe wie *groß* und *klein* festgelegt werden. Diese weisen besonders bei komplizierteren Topologien als in diesem Beispiel eine starke Abhängigkeit von technologischen Randbedingungen auf und müssen daher häufig angepasst werden.

Bei einer zweiten Generation von Werkzeugen zur Topologieselektion werden die wissensbasierten Entscheidungen durch numerische Berechnungen ersetzt. Der gemeinsame Ansatz dieser Verfahren ist, einmalig im voraus für jede Topologie der Bibliothek das Gebiet im Schaltungseigenchaftsraum E zu berechnen, das sich unter Ausnutzung aller Freiheitsgrade ergibt. Dieses Gebiet wird als Realisierbarkeitsgebiet bezeichnet. Formal entspricht der Vorgang einer Abbildung des vollständigen Entwurfparameterraums P_E mit der Analyseabbildung \mathbf{a} . Eine Näherung des Bildes kann durch einen mehrdimensionalen Parameterdurchlauf berechnet werden. Da der damit verbundene Aufwand (vgl. Abschnitt 2.2.6) die Lösung der Dimensionierungsaufgabe um ein Vielfaches übersteigen würde, werden Parameterschwankungen aufgrund von Toleranzen und Betriebsbereichen nicht berücksichtigt. Weiterhin werden nur grobe Einschließungen der Umrandungen der Gebiete bestimmt. Diese werden in der Bibliothek gespeichert. Sie werden bei der eigentlichen Selektion mit den spezifizierten Anforderungen verglichen, so dass die am besten geeignete Topologie ausgewählt werden kann. Dieser Ansatz ist in verschiedenen akademischen Werkzeugen verwirklicht worden (z. B. Harjani u. a. 1989; Veselinovic u. a. 1995). Da der hohe Aufwand für die Berechnung des Realisierbarkeitsgebietes durch Verzicht auf Genauigkeit reduziert werden muss, ist die Treffsicherheit entsprechend ungenügend.

Manuelle Topologieselektion

Wegen der o. a. Schwierigkeiten wird die Topologieselektion in der Praxis in den meisten Fällen nicht automatisch, sondern manuell durchgeführt. Der Schaltungsentwickler greift dabei ebenfalls auf bereits vorhandene Topologien und sein Wissen über die Eignung für die aktuelle Entwurfsaufgabe zurück, passt beides jedoch bei Bedarf graduell an den technologischen Fortschritt an. Die regelmäßige, aufwändige Überarbeitung ganzer Bibliotheken wird damit vermieden. Es entsteht jedoch eine starke Abhängigkeit vom Expertenwissen des Schaltungsentwicklers.

2.3.2 Konstruktive Topologiesynthese

Ein automatisches Topologiesyntheseverfahren muss in der Lage sein, dem technologischen Fortschritt angepasste, neuartige Topologien zu generieren. Dies kann eine Topologieselektion mit Bibliotheksansatz nicht leisten. Es ist statt dessen ein konstruktiver, d. h. kreativer Syntheseprozess erforderlich. Wie bei der Topologieselektion besteht auch hier das Problem, die erzeugten Topologien schnell und entsprechend dem Teile-und-herrsche-Prinzip ohne vollständige Dimensionierung bewerten zu müssen. Die Verwendung von Heuristiken ist hierbei im Gegensatz zur Topologieselektion jedoch unmöglich, da die erzeugten Topologien nicht im Voraus bekannt sind.

Algorithmen zur konstruktiven Topologiesynthese sind bisher ausschließlich Gegenstand der Forschung. Bereits veröffentlichte Ansätze stützen sich auf genetische Optimierung (Kruiskamp und Leenaerts 1995; Sripramong und Toumazou 2002). Diese führen zur Bewertung der erzeugten Topologien eine vollständige Dimensionierung durch. Sie werden in Abschnitt 2.3.10 beschrieben.

2.3.3 Dimensionierung

Bei der auf die Topologiesynthese folgenden Dimensionierung werden für die Entwurfsparameter der erzeugten Topologie Werte bestimmt, so dass das Resultat die Spezifikation erfüllt. Dies wird durch Worst-Case-Analyse und Vergleich überprüft. Ein nahe liegender Ansatz zur Dimensionierung ist daher die Umkehrung der Worst-Case-Analyse. Da die Worst-Case-Analyse-Abbildung a_{WC} nicht eindeutig umkehrbar ist, muss die Umkehrung formal durch eine Relation beschrieben werden:

Definition 2.20 *Die Dimensionierungsrelation D ist eine Relation über*

2 Automatischer Entwurf analoger Schaltungen

$P_E \times E^2$, die durch

$$D = \{(\mathbf{p}_E, \mathbf{e}_{WC,\min}, \mathbf{e}_{WC,\max}) : \mathbf{a}_{WC}(\mathbf{p}_E) = (\mathbf{e}_{WC,\min}, \mathbf{e}_{WC,\max})\} . \quad (2.31)$$

gegeben ist.

Die Dimensionierungsrelation D ist eine formale Vorschrift, mit der das Akzeptanzgebiet im Entwurfparameterraum A_{P_E} bestimmt werden kann:

Definition 2.21 Das Akzeptanzgebiet A_{P_E} ist die Lösungsmenge von $D(A_{P_E}, A_E^2)$ bei gegebenem A_E .

Das Gebiet A_{P_E} enthält also alle Punkte, für die die Schaltung die Nebenbedingungen der Spezifikation bei Betriebs- und Prozessparameterschwankungen einhält.

Beispiel 2.16 Abbildung 2.10 zeigt einen zweidimensionalen Entwurfparameterraum P_E sowie den zweidimensionalen Raum E^2 , der sich durch Zuordnen der Ergebnisse $e_{WC,\min}$ und $e_{WC,\max}$ der Worst-Case-Analyseabbildung \mathbf{a}_{WC} zu Abszisse bzw. Ordinate ergibt. Die Abbildung von P_E nach E^2 ist beispielhaft für einen Punkt B dargestellt. Dabei ergibt sich als Bildpunkt in E^2 der Punkt B' .

Durch $e > e_{Spez,\min}$ und $e < e_{Spez,\max}$ sei ein Akzeptanzgebiet A_E in E definiert. Um dieses mit Hilfe der Dimensionierungsrelation D nach P_E abzubilden, muss es zunächst in E^2 dargestellt werden. Das geschieht durch Bildung von A_E^2 , d. h. $e_{Spez,\min}$ und $e_{Spez,\max}$ werden beide jeweils auf der $e_{WC,\min}$ und der $e_{WC,\max}$ zugeordneten Achse angetragen, so dass sich das dargestellte Quadrat A_E^2 ergibt. Hiervon muss jedoch nur der Teil oberhalb der Winkelhalbierenden betrachtet werden, da $e_{WC,\max} > e_{WC,\min}$ gilt. Die Dimensionierungsrelation D ordnet dem Gebiet A_E^2 das gesuchte A_{P_E} zu.

Um eine eindeutige Lösung des Dimensionierungsproblems zu bestimmen, muss im Akzeptanzgebiet im Entwurfparameterraum A_{P_E} ein Punkt ausgewählt werden. Die Freiheitsgrade, die dadurch vorhanden sind, können entweder durch Hinzufügen von Gleichungsnebenbedingungen gebunden werden (s. Abschnitte 2.3.5 und 2.3.4), oder sie können verwendet werden, um eine oder mehrere in der Spezifikation gegebene Eigenschaften zu optimieren (s. Abschnitte 2.3.6 und 3.1.1).

An Stelle von \mathbf{a}_{WC} kann auch die Worst-Case-Analyse-Abbildung $\mathbf{a}_{WC,s}$, die sich von \mathbf{a}_{WC} durch den Parameter \mathbf{p}_s unterscheidet (vgl. Abschnitt 2.2.6) zur Dimensionierung auf Basis von Ausbeuteoptimierung

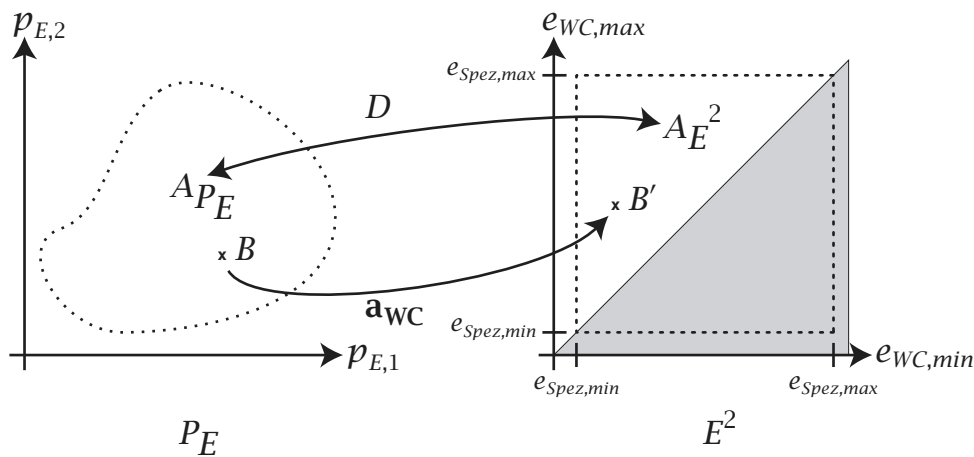


Abbildung 2.10: Beispiel einer Dimensionierungsrelation

(s. Abschnitt 2.3.8) oder Entwurfszentrierung (s. Abschnitt 2.3.9) verwendet werden.

Da das Dimensionierungsproblem zur Problemklasse \mathcal{NP} (vgl. Abschnitt 2.3) gehört, wird die Berechnung des Akzeptanzgebietes im Entwurfsparameterraum A_{P_E} von praktischen Dimensionierungsverfahren wegen des damit verbundenen Rechenaufwandes nicht vollständig durchgeführt.

Die bisher veröffentlichten Verfahren lassen sich in die Klassen: wissens-, gleichungslöser- und optimierungsbasierte Ansätze einteilen.

2.3.4 Wissensbasierte Verfahren zur Dimensionierung

Der direkteste Weg zur Definition der Dimensionierungsrelation D ist die Verwendung von Analysegleichungen (vgl. Abschnitt 2.2.3). In einfachen Fällen können diese explizit nach den Entwurfsparametern \mathbf{p}_E aufgelöst werden, so dass sich Dimensionierungsgleichungen ergeben. Voraussetzung ist, dass die durch die Analysegleichungen definierte Analyseabbildung \mathbf{a} eindeutig umkehrbar ist. Dafür ist eine notwendige Bedingung, dass keine Ungleichungs-, sondern nur Gleichungsnebenbedingungen spezifiziert sind und die Anzahl der Gleichungen der Anzahl der Entwurfsparameter entspricht. Parameterschwankungen können nicht berücksichtigt werden, d. h. es muss eine Dimensionierung im Nominalparameterpunkt $\mathbf{p}_{nom,BS}$ durchgeführt werden. Die Dimensionierungsgleichungen definieren dann eine Dimensionierungsabbildung, die die genaue Umkehrung der durch die Analysegleichungen definier-

2 Automatischer Entwurf analoger Schaltungen

ten Analyseabbildung ist. Sie liefert für jeden Punkt im Schaltungseigenenschaftsraum genau einen Punkt im Entwurfsparameterraum als Ergebnis der Dimensionierungsaufgabe.

In der Regel ist die Forderung nach der Entsprechung der Anzahl der Gleichungen und der Entwurfsparameter nicht erfüllt, sondern das Gleichungssystem ist unterbestimmt. Durch Hinzufügen von weiteren, meist heuristischen Gleichungen, z. B. für Bauelementegleichheit (matching), kann das unterbestimmte Gleichungssystem zu einem bestimmten erweitert werden. Oft ist danach trotzdem keine symbolische Umformung in Dimensionierungsgleichungen möglich, weil die Analysegleichungen im Allgemeinen nichtlinear sind. Durch Näherungen kann das Gleichungssystem vereinfacht werden, bis es schließlich auflösbar wird.

Beispiel 2.17 Für die in Abbildung 2.3 dargestellte Verstärkertopologie werden Dimensionierungsgleichungen für das Entwurfsparametertupel $\mathbf{p}_E = (R_1, R_2, R_C, R_E)$ entwickelt. In der Spezifikation sei dabei das Eigenschaftstupel aus Verstärkung und Ausgangswiderstand $\mathbf{e} = (V, r_{aus})$ durch Sollwerte vorgegeben. Das Problem ist unterbestimmt, da 4 Parametern nur 2 Gleichungen gegenüberstehen. Es werden daher 2 heuristische Gleichungen zur Arbeitspunktstabilisierung hinzugenommen:

$$\frac{U_{CC}}{R_1 + R_2} = 10I_B, \quad (2.32)$$

$$\frac{R_C}{R_E} = 10. \quad (2.33)$$

Mit den Näherungen $r_{ein} \gg R_G$, $r_{aus} \ll R_L$ und $r_{CE} \gg R_C$ lassen sich aus den Gleichungen 2.14, 2.18, 2.19 und 2.33 Vorschriften zur Bestimmung von R_C , R_E und I_C ableiten:

$$R_C = r_{aus}, \quad (2.34)$$

$$R_E = \frac{1}{10}r_{aus}, \quad (2.35)$$

$$I_C = \frac{|V|U_T}{r_{aus}}. \quad (2.36)$$

Mit $\beta \gg \frac{R_1 R_2}{R_E(R_1 + R_2)}$ und $\beta \gg 1$ folgt aus Gleichung 2.13

$$I_C = \frac{1}{R_E} \left(\frac{R_2}{R_1 + R_2} U_{CC} - U_{BE} \right) \quad (2.37)$$

und schließlich

$$R_1 = \left(10 \frac{U_{CC} - U_{BE}}{|V|U_T} - 1 \right) \frac{\beta}{100} r_{aus}, \quad (2.38)$$

$$R_2 = \left(1 + 10 \frac{U_{BE}}{|V|U_T} \right) \frac{\beta}{100} r_{aus}. \quad (2.39)$$

Für eine Dimensionierungsaufgabe, in der $|V| = 40$ dB und $r_{aus} = 1$ k Ω gefordert werden, erhält man mit diesen Dimensionierungsgleichungen $R_1 = 89,3$ k Ω , $R_2 = 9,48$ k Ω , $R_C = 1$ k Ω und $R_E = 100$ Ω . Eine Simulation mit dem Simulator PSpice ergibt mit diesen Parametern und $R_G = 0$ sowie unbelastetem Ausgang lediglich $|V| = 37,4$ dB und $r_{aus} = 976$ Ω . Die Abweichung ist auf die Näherungen sowohl bei der Aufstellung der Analysegleichungen als auch bei den daraus abgeleiteten Dimensionierungsgleichungen zurückzuführen.

Bei einem zweiten Dimensionierungsproblem wird die geforderte Verstärkung auf $|V| = 60$ dB erhöht. Die aus den Dimensionierungsgleichungen resultierende Dimensionierung $R_1 = 6,63$ k Ω , $R_2 = 3,25$ k Ω , $R_C = 1$ k Ω und $R_E = 100$ Ω zeigt in der Simulation in PSpice jedoch nur $|V| = 1,03$. Die Ursache dafür ist die bei der Ableitung der Analysegleichungen gemachte, aber hier nicht zutreffende Annahme, dass der Transistor sich im vorwärts-aktiven Betriebsbereich befindet. Er befindet sich in Wirklichkeit in der Sättigung. Daraus resultiert die geringe Verstärkung.

Dieses Beispiel zeigt, dass selbst bei einfachen Schaltungen infolge der bei der Ableitung der Dimensionierungsgleichungen erforderlichen Näherungen deutliche Abweichungen vom gewünschten Verhalten auftreten. Wenn getroffene Annahmen ungültig sind, kann es sogar zur völligen Fehlfunktion kommen. Aus diesen Gründen ist es erforderlich, das Ergebnis mit einem Verifikationswerkzeug auf Übereinstimmung mit der Spezifikation zu überprüfen.

Die Ableitung von Dimensionierungsgleichungen entspricht dem manuellen Vorgehen eines Schaltungsentwicklers. Aus den erhaltenen Gleichungen kann ein Entwurfsplan erstellt werden. Dies ist ein konstruktiver Algorithmus, der zur Lösung einer Dimensionierungsaufgabe automatisch abgearbeitet werden kann. Für eine vollständige Automatisierung der Schaltungssynthese ist es erforderlich, für jede mögliche Topologie einen passenden Entwurfsplan zu erstellen und in einer Bibliothek abzulegen. Diese enthält damit das gesammelte Expertenwissen, der Ansatz wird daher als wissensbasiert bezeichnet. Er weist durch das starre Bibliothekskonzept die bereits in Abschnitt 2.3.1 angeführten Nachteile

2 Automatischer Entwurf analoger Schaltungen

in Bezug auf Erstellungsaufwand und Lebensdauer auf. Untersuchungen haben ergeben, dass die Erstellung eines Entwurfsplanes etwa das Vierfache des Aufwandes eines entsprechenden manuellen Schaltungsentwurfes benötigt (Beenker u. a. 1993).

Wissensbasierte Dimensionierungswerkzeuge zeichnen sich durch einen sehr geringen Rechenaufwand aus, der der Länge des Entwurfsplanes entspricht. Unter der Voraussetzung, dass ein Experte in der Lage ist, Entwurfspläne aufzustellen, deren Länge polynomial mit der Problemgröße wächst, ist die Laufzeitkomplexität ebenfalls polynomial. Der hohe Aufwand zur Erstellung der Entwurfspläne, ihre kurze Lebensdauer, die Einschränkung auf eine Untermenge der allgemeinen Dimensionierungsprobleme und die Ungenauigkeiten, die sich bei der Simulation des Dimensionierungsergebnisses zeigen, haben jedoch zu einer geringen Akzeptanz dieser ersten Generation von Dimensionierungswerkzeugen in der Praxis geführt.

Nachfolgend werden zwei Beispiele für Synthesewerkzeuge beschrieben, die einen wissensbasierten Dimensionierungsansatz verwenden:

IDAC

Das Synthesewerkzeug IDAC (Degrauwe u. a. 1987) erfordert zunächst eine manuelle Topologieselektion aus einer Bibliothek. Anschließend wird der zugehörige handgeschriebene Entwurfsplan abgearbeitet. Dabei muss die Anzahl der Freiheitsgrade teilweise durch interaktives Festlegen von Parametern so weit reduziert werden, dass ein eindeutiges Ergebnis berechnet werden kann. Um das ungenaue Ergebnis des wissensbasierten Ansatzes zu verbessern, wird in einem zweiten Schritt eine gradientenbasierte Optimierung (s. Abschnitt 2.3.6) des Ergebnisses durchgeführt. IDAC ist zu einem kommerziellen Werkzeug weiterentwickelt worden.

OASYS

In OASYS (Harjani u. a. 1989) wird ein hierarchischer Ansatz verwendet, bei dem abwechselnd Topologieselektion und Dimensionierung mit Entwurfsplänen durchgeführt werden. Zur Topologieselektion werden auf jeder Hierarchieebene alle möglichen Realisierungen dimensioniert und bewertet. Für den Fall, dass sich Annahmen und Näherungen als falsch herausstellen, ist eine Rückverfolgung (backtracking) zur Behebung vorgesehen.

2.3.5 Gleichungslöserbasierte Dimensionierungsansätze

Bei gleichungslöserbasierten wird wie bei wissensbasierten Ansätzen eine Umkehrung der Analyseabbildung vorgenommen, jedoch wird die nachteilige Handableitung von Entwurfsplänen vermieden. Bezüglich der Menge der lösbaren Dimensionierungsprobleme gilt dieselbe Einschränkung.

Computeralgebrasysteme in Verbindung mit symbolischen Analysemethoden können verwendet werden, um die manuelle Aufstellung von Analysegleichungen und ihre Umformung in Dimensionierungsgleichungen automatisch durchzuführen (Hennig und Sommer 1995). Da das Ergebnis in symbolischer Form vorliegt, kann es als Entwurfsplan in einer Bibliothek abgelegt werden. Der Aufwand zur Erstellung einer umfassenden Bibliothek reduziert sich dadurch gegenüber der Handableitung erheblich. Über die Laufzeitkomplexität dieses Ansatzes gibt es bisher keine Untersuchungen, jedoch wachsen die durch symbolische Analyse gewonnenen Ausdrücke in ihrer Länge exponentiell mit der Schaltungsgröße an (Fernández u. a. 1998). Außerdem müssen diese zur Umformung in Dimensionierungsgleichungen symbolisch nach den Entwurfsparametern \mathbf{p}_E auflösbar sein. Dies ist auch bei linearen Systemen nicht allgemein möglich. Aus diesen Gründen sind automatisch durchgeführte Näherungen erforderlich. Besonders bei nichtlinearen Zusammenhängen liefern die zur Verfügung stehenden Verfahren jedoch im Vergleich zu von Hand aufgestellten Näherungen schlechtere Ergebnisse. Da sogar bei linearen Gleichungssystemen die Länge von Lösungsausdrücken, die in geschlossener symbolischer Form angegeben werden, exponentiell mit der Anzahl der Variablen wächst (vgl. Cramersche Regel), muss davon ausgegangen werden, dass auch die Umformung zu Dimensionierungsgleichungen exponentielle Laufzeitkomplexität aufweist.

Dieser Nachteil lässt sich bei Verzicht auf symbolische Ergebnisse vermeiden. Dazu wird ein numerischer Gleichungslöser, z. B. das Newton-Raphson-Verfahren, verwendet. Da nur ein numerisches Analyseergebnis erforderlich ist, kann statt symbolischer Analysegleichungen auch ein numerisches Analyseverfahren, etwa ein Simulator, eingesetzt werden. Die erzielbare Genauigkeit entspricht dann der des Simulators. Da die Gleichheit der Anzahl der Gleichungsnebenbedingungen und der Entwurfparameter notwendig, aber nicht hinreichend für die Umkehrbarkeit der Analyseabbildung ist, kann es jedoch zu Problemen bei der numerischen Lösung kommen. Meist ist daher ein Startpunkt in der Nähe der Lösung erforderlich. Ein Beispiel für ein kommerzielles Dimensionierungswerkzeug, das diesen Ansatz verwendet ist *Size!* (Berger und

Fischer-Binder 2000).

2.3.6 Optimierungsbasierte Ansätze

Die bisher beschriebenen Dimensionierungsansätze schränken die Klasse der Dimensionierungsprobleme stark ein: Es lassen sich nur bestimmte Probleme mit Gleichungsnebenbedingungen, ohne Betriebsbereiche und Prozessparametertoleranzen behandeln. Allgemeine Spezifikationen enthalten jedoch Ungleichungsnebenbedingungen, ein Optimierungsziel und Parameterschwankungen (vgl. Abschnitt 2.1). Bei der Lösung des resultierenden allgemeinen Dimensionierungsproblems muss eine Zielfunktion über einem durch die Ungleichungsnebenbedingungen definierten Akzeptanzgebiet optimiert werden (vgl. Abschnitt 2.3.3). Dieses Optimierungsproblem kann im Allgemeinen nur von einem numerischen Optimierungsverfahren in Verbindung mit Worst-Case-Analyse gelöst werden.

Ausschlaggebend für die Entwicklung einer zweiten Generation von Dimensionierungswerkzeugen auf der Grundlage numerischer Optimierungsverfahren war jedoch die Möglichkeit, als Analysewerkzeug einen Schaltungssimulator einsetzen zu können (simulator in the loop). Dadurch stehen alle Simulationsarten, Extraktionsmöglichkeiten und Bauelementemodelle automatisch auch für die Dimensionierung zur Verfügung, und die erreichbare Genauigkeit entspricht der des Simulators. Zur Kopplung an den Optimierer muss lediglich ein Skript für den Charakterisierungsvorgang erstellt werden. Neue Topologien lassen sich dadurch mit einem im Vergleich zu wissensbasierten Ansätzen erheblich geringeren erstmaligen manuellen Aufwand dimensionieren. Dieser Vorteil hat zu einer erheblich besseren Akzeptanz optimierungsbasierter Werkzeuge geführt.

Optimierungsbasierte Ansätze lassen sich nach dem ihnen zu Grunde liegenden Optimierungsverfahren klassifizieren.

Lokale Optimierungsverfahren

Ein Optimierungsproblem mit Ungleichungsnebenbedingungen ist durch eine Zielfunktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ und Nebenbedingungsfunktionen $\mathbf{c} : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_c}$ gegeben als

$$\mathbf{x}_{\min} = \underset{\mathbf{x} \in \Omega}{\operatorname{argmin}} f(\mathbf{x}) \quad (2.40)$$

mit $\Omega = \{\mathbf{x} \in \mathbb{R}^{n_x} : \mathbf{c}(\mathbf{x}) \geq 0\}$.

Soll statt des Minimums \mathbf{x}_{\min} das Maximum gesucht werden, so ist f durch $-f$ zu ersetzen. Falls mehrere Optimierungsziele gegeben sind, werden diese meist als gewichtete Summe zur Zielfunktion f zusammengefasst.

Viele numerische Verfahren zur Lösung von Problem 2.40 ersetzen dieses durch eine Folge von Optimierungsproblemen ohne Nebenbedingungen. Eine solche Folge ist so beschaffen, dass die Lösungen ihrer Elemente gegen die von Problem 2.40 konvergieren. Eine Möglichkeit dazu ist die Zusammensetzung von Zielfunktion und Nebenbedingungen mit Hilfe einer *Barriere-Funktion*, die das Verlassen von Ω verhindert. Bei Verwendung einer logarithmischen Barriere-Funktion ergibt sich

$$f_B(\mathbf{x}, \mu) = f(\mathbf{x}) - \mu \sum_{i=1}^{n_c} \log c_i(\mathbf{x}) . \quad (2.41)$$

Die Lösungen der Elemente der Folge werden als $\mathbf{x}_k = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^{n_x}} f_B(\mathbf{x}, \mu_k)$ definiert. Für μ_k ist eine fallende Folge mit $\lim_{k \rightarrow \infty} \mu_k = 0$ zu wählen und als Startwert für die Lösung des k -ten Problems ist \mathbf{x}_{k-1} zu verwenden. Der Startwert \mathbf{x}_0 muss in Ω liegen. Zur Lösung der resultierenden Optimierungsprobleme ohne Nebenbedingungen gibt es zwei verbreitete Ansätze:

Bei der *Liniensuche* (line search) wird zunächst eine Suchrichtung festgelegt. Daraufhin wird entlang dieser Richtung ein Minimum gesucht und dort iterativ erneut eine Suchrichtung bestimmt. Im einfachsten Fall ist die Suchrichtung durch den *steilsten Abstieg* $-\nabla f$ (steepest descent) gegeben. Diese Richtung folgt aus dem Ansatz einer Linearisierung von f . Es können auch Näherungen höherer Ordnung verwendet werden, z. B. ergibt sich aus einer quadratischen Näherung von f die *Newton-Richtung* (Nocedal und Wright 1999). Die Methode des steilsten Abstiegs wird u. a. in den Werkzeugen DELIGHT.SPICE (Nye u. a. 1988) und OPASYN (Koh u. a. 1990) benutzt.

Ein anderer Ansatz definiert einen *Vertrauensbereich* (trust region), in dem eine Näherung von f verwendet wird. Innerhalb des Vertrauensbereiches wird das Minimum berechnet und an seinem Ort eine neue Näherung mit Vertrauensbereich bestimmt. Auch hier können Näherungen verschiedener Ordnung angewendet werden, wobei die wichtigste die quadratische ist, die zum *Newton-Optimierungsverfahren* führt. Eine Variante des Vertrauensbereich-Ansatzes mit *Grenzkurven* wird im Werkzeug WiCkED (Schwencker u. a. 2002) eingesetzt.

Die beschriebene Methode mit Barriere-Funktionen ist zwar verbreitet, führt jedoch häufig zu numerischen Problemen für $\mu \rightarrow$

2 Automatischer Entwurf analoger Schaltungen

0. Eine Weiterentwicklung, die diese umgeht, arbeitet mit *Lagrange-Multiplikatoren* und wird z. B. im Optimierer des Simulators PSpice verwendet. Beim SQP-Verfahren (sequential quadratic programming) wird das Problem 2.40 durch quadratische Näherungen direkt auf eine Folge quadratischer Optimierungsprobleme *mit* Nebenbedingungen abgebildet. Es ist u. a. in das Synthesewerkzeug AMGIE (van der Plas u. a. 2001) integriert.

Alle in diesem Abschnitt beschriebenen Optimierungsverfahren finden bei Problemen mit mehreren lokalen Optima im Allgemeinen nicht das globale Optimum als Lösung des Dimensionierungsproblems. Sie terminieren abhängig vom vorzugebenden Startpunkt in einem meist benachbarten lokalen Optimum, erreichen dieses aber in wenigen Schritten und mit polynomialer Laufzeitkomplexität. Sie eignen sich daher zum Verbessern einer bereits vorliegenden Dimensionierung nahe dem globalen Optimum, die als Startpunkt vorgegeben wird. Typische Syntheseprobleme, bei denen noch keine hinreichend gute Dimensionierung vorliegt, können bestenfalls halbautomatisch gelöst werden. Ein weiterer Nachteil ist, dass unlösbare Probleme, bei denen das Akzeptanzgebiet im Entwurfparameterraum A_{p_E} leer ist, nicht als solche erkannt werden können. Für eine weitergehende Beschreibung lokaler Optimierungsverfahren siehe z. B. Nocedal und Wright (1999).

Simulated Annealing

Simulated Annealing gehört zu der Klasse der stochastischen Optimierungsverfahren. Es wird hier während seines dominierenden Anteils an praktisch eingesetzten stochastischen Verfahren stellvertretend für diese Klasse beschrieben.

Der Name des Optimierungsverfahren Simulated Annealing (Romeo und Sangiovanni-Vincentelli 1991) leitet sich aus der Analogie zwischen dem Ausglühen von Feststoffen und globaler Optimierung ab: Beim Ausglühen wird eine geschmolzene Substanz langsam abgekühlt, bis die Atome eine Kristallstruktur mit maximaler Regularität gebildet haben. Dieses entspricht einem globalen energetischen Minimum. Eine Simulation dieses Vorganges (Simulated Annealing) ist daher ein Ansatz für ein naturanaloges Optimierungsverfahren. Dabei entsprechen die Variablen des Optimierungsproblems \mathbf{x} den Atomen und die zu minimierende Zielfunktion f der Energie der Schmelze. Entscheidend für die Eigenschaften dieses Verfahrens ist die Berücksichtigung der Temperatur T der Schmelze. Zur Simulation der Vorgänge in der Schmelze wird iterativ aus dem aktuellen \mathbf{x}_k ein neues \mathbf{x}_{k+1} gebildet und mit einer Akzep-

tanzfunktion über Annahme oder Ablehnung dieses Schrittes entscheiden. Die Akzeptanzfunktion nimmt grundsätzlich alle Schritte an, für die $f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k)$ gilt. Bei Schritten mit $f(\mathbf{x}_{k+1}) > f(\mathbf{x}_k)$ wird die Entscheidung von einem Zufallsgenerator getroffen. Ein solcher Schritt wird mit der Wahrscheinlichkeit

$$p_A = e^{-\frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)}{T}}. \quad (2.42)$$

angenommen bzw. mit der Wahrscheinlichkeit $1 - p_A$ abgelehnt. Bei hohen Temperaturen T besteht also eine große Wahrscheinlichkeit, dass auch Schritte angenommen werden, die zu einer Erhöhung von f führen. Bei $T \rightarrow 0$ hingegen gilt $p_A \rightarrow 0$, so dass nur Schritte akzeptiert werden, die f nicht erhöhen. In Analogie zum natürlichen Ausglühen wird der Optimierungsprozess mit hoher Temperatur T begonnen (flüssige Schmelze), und diese dann langsam abgesenkt, bis $T = 0$ (erstarrte Schmelze). Der Verlauf von T in Abhängigkeit von k wird als Abkühlkurve bezeichnet (cooling schedule).

Die Eigenschaft, mit der Wahrscheinlichkeit p_A auch Schritte anzunehmen, die den Wert von f vergrößern statt zu verkleinern, ermöglicht dem Simulated-Annealing-Verfahren das Überwinden lokaler Minima. Dadurch ist das Verfahren prinzipiell in der Lage, zum globalen Minimum zu konvergieren. Dieses ist jedoch nur für $k \rightarrow \infty$ und unendlich langsame Abkühlung beweisbar (Romeo und Sangiovanni-Vincentelli 1991). Bei der Wahl einer endlichen Abkühlkurve liegt daher ein Zielkonflikt (tradeoff) zwischen Konvergenzwahrscheinlichkeit und Rechenaufwand vor. Die Optimierung von Abkühlkurven ist bis heute Forschungsgegenstand. Die erforderliche Iterationszahl zur Erlangung quasi-globaler Konvergenz liegt erheblich über der von Verfahren zur lokalen Optimierung. Eine Aussage über die Laufzeitkomplexität konnte bisher nicht bewiesen werden. Es gibt jedoch Hinweise darauf, dass sie exponentiell ist (Sasaki und Hajek 1988).

Das Simulated-Annealing-Verfahren ist nur zur Behandlung von Optimierungsproblemen ohne Nebenbedingungen geeignet. Die Nebenbedingungen des Dimensionierungsproblems müssen daher mit dem bei den Ausführungen zu lokalen Optimierungsverfahren (s. o.) beschriebenen Ansatz zur Zusammensetzung von Zielfunktion und Nebenbedingungen behandelt werden. Auch mit Simulated Annealing lassen sich unlösbare Dimensionierungsprobleme, bei denen $A_{p_E} = \emptyset$, nicht identifizieren.

Im Vergleich zu Werkzeugen, die lokale Optimierungsverfahren verwenden, sind Dimensionierungswerkzeuge, die Simulated Annealing einsetzen, wesentlich besser zur Behandlung von Problemen ohne gute Startlösung geeignet. Beispiele hierfür sind: ARIADNE (Swings u. a.

1991b) und FRIDGE (Medeiro Hidalgo u. a. 1994) sowie ASTRX/OBLX (Ochotta u. a. 1996), MAELSTROM (Krasnicki u. a. 1999) und ANACONDA (Phelps u. a. 1999), auf deren Basis ein kommerzielles Werkzeug entwickelt worden ist.

Geometrische Optimierung

Bei diesem Ansatz (del Mar Hershenson u. a. 2001) wird die Klasse der Dimensionierungsprobleme so eingeschränkt, dass sich ein konvexes Optimierungsproblem ergibt. Die Konvexität bietet eine Reihe von Vorteilen: Das Problem hat neben dem globalen keine lokalen Optima, Algorithmen zur Lösung konvexer Optimierungsprobleme garantieren unter Ausnutzung dieser Eigenschaft globale Konvergenz, und, falls keine Lösung existiert, wird dies vom Algorithmus erkannt. Lösungsalgorithmen, die auf der Innen-Punkt-Methode (interior point method) basieren, haben polynomiale Laufzeitkomplexität und können auch sehr große Probleme in kurzer Zeit lösen.

Um ein konvexes Optimierungsproblem zu erhalten, wird das Dimensionierungsproblem als geometrisches Optimierungsproblem formuliert, das sich in ein konvexes transformieren lässt. Ein geometrisches Optimierungsproblem ist gegeben durch

$$\mathbf{x}_{\min} = \underset{\mathbf{x} \in \Omega}{\operatorname{argmin}} f(\mathbf{x}) \quad (2.43)$$

mit $\Omega = \{\mathbf{x} \in \mathbb{R}^{n_x} : \mathbf{c}(\mathbf{x}) \geq 1 \wedge \mathbf{d}(\mathbf{x}) = 1 \wedge \mathbf{x} \geq 0\}$.

Dabei ist $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ die Zielfunktion, $\mathbf{c} : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_c}$ sind die Ungleichungsnebenbedingungsfunktionen, und $\mathbf{d} : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_c}$ die Gleichungsnebenbedingungsfunktionen. Die Funktionen f und \mathbf{c} müssen dabei komponentenweise *posynomiale* Ausdrücke der Form

$$\sum_i \left(k_i \prod_j x_j^{\alpha_{ij}} \right), \quad k_i > 0, \alpha_{ij} \in \mathbb{R} \quad (2.44)$$

sein. Die Funktion \mathbf{d} muss komponentenweise aus *monomialen* Ausdrücken der Form

$$k \prod_i x_i^{\alpha_i}, \quad k > 0, \alpha_i \in \mathbb{R} \quad (2.45)$$

bestehen. Das Problem 2.43 lässt sich durch komponentenweises Logarithmieren von \mathbf{x} , f sowie der Ungleichungs- und Gleichungsnebenbedingungen in ein konvexes Optimierungsproblem überführen.

Im Gegensatz zum allgemeinen Problem 2.40 enthält das geometrische Optimierungsproblem 2.43 explizit Gleichungsnebenbedingungen der Form $\mathbf{d}(\mathbf{x}) = 1$. Dies ist bei der kanonischen Darstellung 2.40 nicht erforderlich, da dort Gleichheitsnebenbedingungen durch Einsetzen eliminiert oder alternativ als Paare aus Ungleichungsnebenbedingungen ausgedrückt werden können. Dieses ist beim geometrischen Optimierungsproblem 2.43 aufgrund der o. a. Vorgaben der verwendbaren Ausdrücke nicht möglich.

Die Einschränkung auf geometrische Optimierungsprobleme führt dazu, dass im Gegensatz zu den anderen hier beschriebenen Optimierungsverfahren als Analysewerkzeug grundsätzlich kein Simulator, sondern nur Analysegleichungen posynomialer bzw. monomialer Form verwendet werden dürfen. Ebenso können Maschen- und Knotengleichungen nicht als Nebenbedingungen verwendet werden, da sie nicht der Form 2.45 entsprechen. Es wurde zwar gezeigt, dass typische Probleme wie die Dimensionierung von Operationsverstärkern in die benötigte Form gebracht werden können (del Mar Hershenson u. a. 2001). Die Aufstellung der Analysegleichungen erfordert jedoch Näherungen, die von Hand durchgeführt oder von einem empirischen Modellgenerator (Daems u. a. 2003) berechnet werden müssen.

Die Stärke von Dimensionierungsverfahren, die diesen Ansatz verwenden, liegt in der Fähigkeit, auch große Probleme mit vielen Freiheitsgraden, z. B. hierarchisch definierte Analog-Digital-Wandler mit variabler Stufenzahl (del Mar Hershenson 2002), optimal und schnell zu lösen. Ferner können durch die gute Skalierbarkeit des Verfahrens Parameter-Durchläufe durch Vervielfachen der Nebenbedingungen mit verschiedenen Parameterwerten realisiert werden. Nachteilig sind jedoch der Genauigkeitsverlust und der Aufwand der für jede Topologie erforderlichen Modellierung.

Auf Basis dieses Ansatzes ist ein kommerzielles Werkzeug entwickelt worden.

Das Verfahren dieser Arbeit garantiert ebenfalls globale Konvergenz, erfordert aber nicht die Verwendung posynomialer Gleichungen (s. Abschnitt 3.3.1).

Parameterschwankungen

Die Dimensionierungsergebnisse, die mit Optimierungsverfahren für Nominalparameterpunktdimensionierungsprobleme erzielt werden, sind oft so stark optimiert, dass schon geringe Abweichungen von den berechneten Entwurfparametern oder den spezifizierten

2 Automatischer Entwurf analoger Schaltungen

Betriebs- oder Herstellungsprozessparametern zu einer deutlichen Verletzung der Spezifikation führen, da z. B. MOS-Transistoren den Sättigungsbereich verlassen. Dies führt zu schlechter Robustheit und Ausbeute. Solche Ergebnisse lassen sich auch bei Nachoptimierung mit Ausbeuteoptimierungs- und Entwurfszentrierungswerkzeugen (s. Abschnitte 2.3.8 und 2.3.9) kaum verbessern (Ochotta u.a. 1998). Betriebsbereichs- und Herstellungsprozessparameterschwankungen müssen daher bei der Dimensionierung berücksichtigt werden, indem die Worst-Case-Analyse-Abbildung a_{WC} verwendet wird. Dies erhöht jedoch den Rechenaufwand bei der Optimierung zusätzlich. Im Werkzeug WiCkeD wird daher ein Ansatz mit Linearisierungen zur Beschleunigung verwendet (Schwencker u.a. 2002). Dieser, aber auch Ansätze mit Parameterdurchläufen oder Eckenanalysen, führen mit hoher Wahrscheinlichkeit zu einer Unterschätzung des tatsächlichen Worst-Case-Bereiches der untersuchten Eigenschaft, so dass Spezifikationsverletzungen entstehen können (vgl. Abschnitt 2.2.6). Der Ansatz dieser Arbeit garantiert die Korrektheit der berechneten Worst-Case-Schranken mit Hilfe formaler Methoden (s. Abschnitt 3.3.2).

2.3.7 Zielkonflikt von Dimensionierungsverfahren

Die Lösung einer Dimensionierungsaufgabe als Optimierungsproblem mit Hilfe eines iterativen numerischen Verfahrens erfordert erheblich mehr Rechenzeit als ein wissens- oder gleichungslöserbasierter Ansatz. Dem Rechenzeitbedarf lässt sich durch Verzicht auf globale Konvergenz, Genauigkeit der Analyse oder durch manuelle Vorarbeit entgegenwirken: Es liegt ein Zielkonflikt zwischen diesen Größen vor. Dies gilt nicht nur für optimierungsbasierte Ansätze, sondern allgemein für alle Dimensionierungsverfahren. Beispielsweise wird bei einem wissensbasierten Ansatz zu Gunsten kürzerer Rechenzeit und dem Wegfall von Konvergenzproblemen auf Genauigkeit verzichtet und außerdem manueller Aufwand in Kauf genommen. Bei einer Kombination von Simulated Annealing, Simulator und Worst-Case-Analyse wiederum werden durch einen sehr hohen Rechenaufwand Genauigkeit und quasi-globale Konvergenz gewonnen, wobei nur wenig manuelle Vorbereitung erforderlich ist. Um trotz der hohen Zahl an erforderlichen Iterationen mit Simulated Annealing zu akzeptablen Laufzeiten zu gelangen, sind verschiedene Ansätze veröffentlicht worden: Im Werkzeug ARIADNE (Swings u. a. 1991b) werden automatisch vereinfachte Analysegleichungen eingesetzt. Diese decken jedoch nur das Kleinsignalverhalten ab. Das Werk-

zeug ASTRX/OBLX (Ochotta u. a. 1996) verwendet ein schnelles Verfahren zur Kleinsignalanalyse (advanced waveform evaluation, AWE) und eine kompilierte Kostenfunktion. Im Gegensatz zu ARIADNE und ASTRX/OBLX wird in MAELSTROM (Krasnicki u. a. 1999) und ANACONDA (Phelps u. a. 1999) ein Simulator eingesetzt, wobei der Rechenaufwand durch massive Parallelisierung auf einen Rechnerverbund verteilt wird.

Die Eigenschaften der wissens-, gleichungslöser- und optimierungsbasierten Dimensionierungsansätze werden in Tabelle 2.5 abschließend verglichen. Dabei wird auch das Anwendungsgebiet aufgeführt. Unter *Synthese* ist dabei zu verstehen, dass das Verfahren in der Lage ist, im Rahmen der automatischen Schaltungssynthese ohne Startwert verlässlich eine brauchbare Lösung zu produzieren. *Nachoptimierung* hingegen bedeutet, dass ein z. B. durch Handentwurf gewonnener Startwert in der Nähe der gesuchten Lösung gegeben sein muss.

Es ist auffallend, dass einzig der Ansatz der geometrischen Optimierung Ergebnisse mit globaler Optimalität garantiert, dabei jedoch die Analyse auf posynomiale Analysegleichungen und die Form der Nebenbedingungen auf die in Problem 2.43 dargestellte einschränkt. Außerdem bietet keiner der Ansätze gleichzeitig Ergebnisse mit garantierter globaler Optimalität und die Möglichkeit, durch Verwendung eines geeigneten Worst-Case-Analysewerkzeugs ein Ergebnis mit garantierter Einhaltung der Nebenbedingungen bei Parameterschwankungen zu erhalten. Ein Verfahren, das durch die Verwendung formaler Methoden beides erreicht und gleichzeitig nicht der Einschränkung auf posynomiale Gleichungen unterliegt, wird in dieser Arbeit in Kapitel 3 vorgestellt.

2.3.8 Ausbeuteoptimierung

Die Ausbeuteoptimierung ist ein Sonderfall der optimierungsbasierten Dimensionierung mit der auf der Worst-Case-Analysefunktion $\mathbf{a}_{WC,S}$ basierenden Ausbeutefunktion Y (vgl. Gleichung 2.28) als zu maximierende Zielfunktion. In ihr sind die Nebenbedingungen durch die Verwendung der Akzeptanzfunktion 2.29 bereits enthalten. Diese werden daher bei der Optimierung nicht mehr gesondert berücksichtigt, so dass ein Optimierungsverfahren für Probleme ohne Nebenbedingungen zur Lösung eingesetzt werden kann. Statt mit dem Akzeptanzgebiet im Entwurfparameterraum A_{P_E} arbeitet dieser Ansatz mit einer Ausbeuteverteilung im Entwurfparameterraum P_E , die durch $Y(\mathbf{p}_E)$ gegeben ist.

Die Verwendung der aufwändigen Monte-Carlo-Simulation zur Ausbeuteanalyse führt in Verbindung mit einem iterativen Optimierungs-

2 Automatischer Entwurf analoger Schaltungen

	Wissensbasiert	Gleichungslöserbasiert		Optimierungsbasiert		
		Symbolisch	Numerisch	Lokal	Simulated Annealing	Geometrische Optimierung
Lösungsverfahren						
Methode	Konstruktiv	Konstruktiv	Iterativ	Iterativ	Iterativ	Iterativ
Optimierungsziel	Nein	Nein	Nein	Beliebig	Beliebig	Posynomiale Gleichung
Nebenbedingungen	Gleichungen	Gleichungen	Gleichungen	Beliebig	Beliebig	Geometrisches Programm
Unterbestimmte Probleme	Nein	Nein	Nein	Möglich	Möglich	Möglich
Konvergenz	—	—	Lokal	Lokal	Quasi-global	Global
Unlösbarkeits-erkennung	Ja	Ja	Nein	Nein	Nein	Ja
Anwendungsgebiet	Synthese	Synthese	Nach-optimierung	Nach-optimierung	Synthese	Synthese
Analyse						
Analysewerkzeug	Vereinfachte Gleichungen	Vereinfachte Gleichungen	Beliebig	Beliebig	Beliebig	Posynomiale Gleichungen
Maximale Genauigkeit	Niedrig	Niedrig	Hoch	Hoch	Hoch	Mittel
Worst-Case-Analyse	Nein	Nein	Nein	Möglich	Möglich	Parameter-durchlauf
Manueller Vorbereitungsaufwand	Sehr hoch	Mittel	Mittel	Niedrig	Niedrig	Hoch
Skalierbarkeit						
Laufzeitkomplexität	Polynomial	Vermutlich exponentiell	Polynomial	Polynomial	Vermutlich exponentiell	Polynomial
Maximale Problemgröße	Klein	Klein	Mittel	Mittel	Mittel	Groß
Dabei: Rechenzeitbedarf	Gering	Mittel	Mittel	Mittel	Hoch	Mittel
Formale Garantie für						
Globales Optimum	Nein	Nein	Nein	Nein	Nein	Ja
Einhaltung der Nebenbedingungen bei Parameterschwankungen	Nein	Nein	Nein	Möglich	Möglich	Nein

Tabelle 2.5: Vergleich der Dimensionierungsansätze

verfahren jedoch zu einem sehr hohen Rechenaufwand. Aus diesem Grund wird das Verfahren in der Praxis nicht im Rahmen der automatischen Synthese, sondern nur zur Nachoptimierung verwendet. Dadurch kann ein schnelles lokales Optimierungsverfahren eingesetzt werden, das nur wenige Iterationen benötigt.

2.3.9 Entwurfszentrierung

Ein der Ausbeuteoptimierung verwandtes Verfahren, das ebenfalls zu einem Optimierungsproblem ohne Nebenbedingungen führt, ist die Entwurfszentrierung. Bei ihr wird auf die explizite Berechnung der Ausbeute verzichtet und damit die aufwändige Monte-Carlo-Simulation vermieden. Es wird das Akzeptanzgebiet A_{P_S} im Herstellungsprozessparameterraum P_S betrachtet, das sich in Abhängigkeit des Entwurfspunktes \mathbf{p}_E und des Akzeptanzgebietes im Eigenschaftsraum A_E formal mit Hilfe einer Relation $S_{\mathbf{p}_E}$ definieren lässt:

Definition 2.22 Die Relation $S_{\mathbf{p}_E}$ ist eine Relation über $P_S \times E^2$ mit Parameter \mathbf{p}_E , die durch

$$S_{\mathbf{p}_E} = \{(\mathbf{p}_S, \mathbf{e}_{WC,\min}, \mathbf{e}_{WC,\max}) : \mathbf{a}_{WC,S}(\mathbf{p}_E, \mathbf{p}_S) = (\mathbf{e}_{WC,\min}, \mathbf{e}_{WC,\max})\}$$

gegeben ist.

Definition 2.23 Das Akzeptanzgebiet A_{P_S} ist die Lösungsmenge von $S_{\mathbf{p}_E}(A_{P_S}, A_E^2)$ bei gegebenem A_E und \mathbf{p}_E .

Das Akzeptanzgebiet A_{P_S} ist nur für ein gegebenes \mathbf{p}_E bestimmbar und enthält alle Punkte im Herstellungsprozessparameterraum, für die die Schaltung die Nebenbedingungen der Spezifikation bei Betriebsparameterschwankungen $P_{B,Spez}$ einhält.

Die Herstellungsprozessparameter \mathbf{p}_S sind jedoch nicht wählbar, sondern in ihrer statistischen Verteilung w (vgl. Abschnitt 2.2.8) durch den Herstellungsprozess vorgegeben. Ziel der Entwurfszentrierung ist es daher, die Entwurfparameter \mathbf{p}_E so zu bestimmen, dass das Gebiet A_{P_S} eine an der Wahrscheinlichkeitsverteilung w gemessen günstige Lage und Form hat. Dies führt zu einer Erhöhung der Ausbeute Y . Bei einer Gauß-Verteilung mit Kovarianzmatrix \mathbf{C} wird dazu das Abstandsmaß $\beta = \sqrt{\delta^T \mathbf{C}^{-1} \delta}$ maximiert. Dabei ist δ der Abstandsvektor zwischen dem nominalen Prozessparametersatz $\mathbf{p}_{nom,S}$ und dem Rand von A_{P_S} . Außenlage von $\mathbf{p}_{nom,S}$ ist als negativer Abstand zu werten (Antreich u. a. 1994).

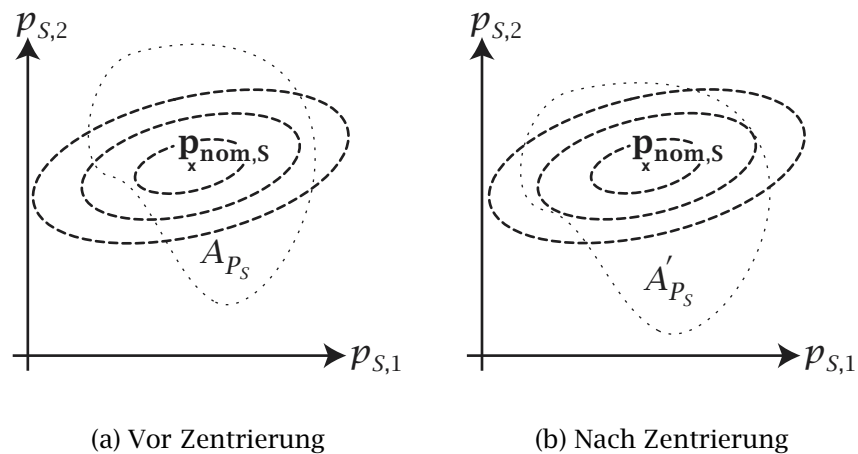


Abbildung 2.11: Entwurfszentrierung im Prozessparameterraum P_S

Beispiel 2.18 *Abbildung 2.11(a) zeigt ein Akzeptanzgebiet A_{P_S} , das den Nominalprozessparametersatz $\mathbf{p}_{\text{nom},S}$ einschließt. Das Akzeptanzgebiet A'_{P_S} nach Entwurfszentrierung ist in Abbildung 2.11(b) dargestellt. Zusätzlich sind für $\mathbf{p}_{\text{nom},S}$ jeweils drei Ellipsen gleicher Wahrscheinlichkeitsdichten eingezeichnet, die sich aus der angenommenen Gauß-Verteilung ergeben. Alle Punkte auf den einzelnen Ellipsen haben jeweils konstante Abstände β_i von $\mathbf{p}_{\text{nom},S}$. Während in Abbildung 2.11(a) die Punkte innerhalb der mittleren Ellipse teilweise außerhalb A_{P_S} liegen, sind diese Punkte in Abbildung 2.11(b) infolge der Entwurfszentrierung vollständig in A'_{P_S} enthalten.*

Die Berechnung von A_{P_S} ist jedoch für praktische Verfahren zu aufwändig. Im Dimensionierungswerkzeug WiCkeD wird der Rand von A_{P_S} daher durch simulatorbasierte Linearisierung der Nebenbedingungen angenähert und es werden die Abstände β_j für jede Nebenbedingung isoliert berechnet. Zur Entwurfszentrierung wird das Minimum dieser Abstände maximiert. Durch die Linearisierungen und die getrennte Behandlung der Nebenbedingungen ergibt sich eine in Abhängigkeit der Anzahl der Parameter polynomiale Laufzeitkomplexität, so dass auch Optimierungsverfahren, die viele Iterationen erfordern, eingesetzt werden können. Die Näherungen verursachen jedoch Abweichungen im Ergebnis.

Das Verfahren dieser Arbeit ermöglicht durch den Einsatz formaler Methoden eine Entwurfszentrierung für gleichförmige Wahrscheinlichkeitsverteilungen mit garantierter Einhaltung der Spezifikation durch-

geführt werden (s. Abschnitt 3.1.2).

2.3.10 Gleichzeitige Topologiesynthese und Dimensionierung

Zur Lösung des Schaltungssyntheseproblems ohne Aufteilung in Unterprobleme muss dieses als MINLP (vgl. Abschnitt 2.3) behandelt werden. Dadurch sind die Topologie und ihre kontinuierlichen Bauelementeparameter gleichzeitig Entwurfsvariable. Anzahl und Bedeutung der kontinuierlichen Parameter und damit auch die Analyseabbildung wechseln jedoch mit jeder Topologieänderung.

DARWIN

In diesem von Kruiskamp und Leenaerts (1995) vorgestellten ersten Werkzeug zur Topologiesynthese mit genetischer Optimierung (Schöneburg u. a. 1993) wird eine Bibliothek der 24 möglichen Operationsverstärkertopologien, die sich sinnvoll aus 11 Basisblöcken bilden lassen, verwendet. Für die einzelnen Blöcke werden mit Hilfe von heuristischen Randbedingungen und einem der geometrischen Optimierung (vgl. Abschnitt 2.3.6) ähnlichen Ansatz eine endliche Menge von zulässigen Entwurfsparametersätzen bestimmt. Die Startpopulation wird zufällig aus den 24 Bibliothekstopologien und den für die einzelnen Stufen berechneten Entwurfsparametersätzen ausgewählt. Danach werden die Individuen der Population iterativ mit Hilfe von Analysegleichungen bewertet und es werden die Operationen Selektion, Cross-Over und Mutation durchgeführt.

Durch die Diskretisierung der kontinuierlichen Parameter wird das MINLP in ein IP überführt. Dies und die Vorgabe von nur 24 Bibliothekstopologien schränken die Flexibilität dieses Werkzeuges in der Praxis stark ein. Ein kreativer Syntheseprozess, der an technologische Fortschritte angepasste Topologien erzeugen kann, ist nicht möglich.

SEAS

Das Synthesewerkzeug SEAS (Ning u. a. 1991) baut wie der von Sripramong und Toumazou (2002) vorgeschlagene Ansatz (s. u.) auf einer geschachtelten Optimierung auf: Es wird nach jeder Topologieveränderung eine vollständige Dimensionierung der Topologie durchgeführt und das Ergebnis zur Bewertung der Topologieveränderung verwendet.

2 Automatischer Entwurf analoger Schaltungen

In SEAS wird eine Netzliste aus Subblöcken vorgegeben. Die Aufgabe des Werkzeugs besteht darin, diesen dimensionierte Netzlisten auf Bauelementeebene zuzuweisen. Die Topologien der Subblöcke werden einer Bibliothek entnommen, die außerdem Analysegleichungen für ihr Verhalten enthält. Eine in einem Iterationsschritt vorliegende Zuordnung von Topologien zu den Subblöcken wird mit Hilfe von Simulated Annealing dimensioniert und anschließend in Hinblick auf die Spezifikation bewertet. Danach wird durch einen wissensbasierten Ansatz entschieden, welche der Subblocktopologien zur Verbesserung der Eigenschaften durch eine andere zu ersetzen ist.

Auch bei diesem Ansatz verhindert das bibliotheksbasierte Vorgehen die kreative Synthese von Topologien.

Sripramong

Die Beschränkung von DARWIN auf Bibliothekstopologien wird in diesem Ansatz, der ebenfalls genetische Optimierung verwendet, beseitigt. Es werden alle Möglichkeiten der Verschaltung von MOS-Transistoren zugelassen. Zur schnellen Eliminierung von Topologien, die gegen Entwurfsregeln verstoßen, z. B. durch Kurzschlüsse, wird eine *Stromflussanalyse* durchgeführt. Die Dimensionierung wird im Gegensatz zu DARWIN durch ein lokales Optimierungsverfahren durchgeführt.

Die große Zahl der Freiheitsgrade in Verbindung mit den aufwändigen Optimierungsverfahren Simulated Annealing bzw. genetische Optimierung führt sowohl bei SEAS als auch bei diesem Ansatz zu einer hohen Laufzeitkomplexität und in der Praxis zu sehr langen Rechenzeiten.

Keiner der in diesem Abschnitt beschriebenen Ansätze zur gleichzeitigen Topologiesynthese und Dimensionierung garantiert globale Optimalität oder die Korrektheit des Ergebnisses bei Parameterschwankungen.

3 Dimensionierung mit formalen Methoden

Im Folgenden wird das in dieser Arbeit entwickelte Dimensionierungsverfahren vorgestellt. Das Verfahren verwendet einen optimierungsbasierten Ansatz, da dieser die in Abschnitt 2.3.6 dargestellten Vorteile Genauigkeit, Flexibilität, Einbeziehung von Optimierungszielen, Handhabbarkeit unterbestimmter Probleme und den im Vergleich zu anderen Ansätzen geringsten manuellen Vorbereitungsaufwand bietet. Es berücksichtigt ferner den Einfluss von Parameterschwankungen, um Robustheit und Ausbeute zu verbessern. Durch den Einsatz von auf affiner Arithmetik basierenden formalen Methoden wird eine Garantie für die Einhaltung der Spezifikation und für die globale Optimalität der berechneten Ergebnisse erzielt. Falls die Spezifikation nicht erfüllbar ist, wird dies erkannt.

3.1 Problemdefinitionen

3.1.1 Definition des Dimensionierungsproblems als Optimierungsproblem

Die in dieser Arbeit verwendete Darstellung des Dimensionierungsproblems als Optimierungsproblem und seine formale Lösung werden in diesem Abschnitt definiert.

Bei der Dimensionierung von Analogschaltungen wird ein Entwurfsparametertupel $\mathbf{p}_E^* \in P_E$ gesucht, so dass die Spezifikation erfüllt wird. Schaltungstopologie und Herstellungsprozess sind gegeben.

Die Herstellungsprozessparameterschwankungen werden als vorzuziehendes Gebiet $P_{S,Spez}$ (vgl. Definition 2.7) berücksichtigt. Für statistische Verteilungen lässt sich für ein in der Spezifikation anzugebendes Y_{min} ein $P_{S,Spez}$ bestimmen (vgl. Abschnitt 2.2.8). Ein alternativer Ansatz, der eine Entwurfszentrierung durchführt, wird in Abschnitt 3.1.2 dargestellt.

3 Dimensionierung mit formalen Methoden

Betriebsparameterbereiche seien in der Spezifikation entsprechend Definition 2.5 durch $P_{B,Spez}$ gegeben. $P_{B,Spez}$ wird zur Abkürzung mit $P_{S,Spez}$ zusammengefasst.

Definition 3.1 *Das zusammengefasste Betriebs- und Herstellungsprozessparametergebiet $P_{BS,Spez}$ ist gegeben durch $P_{BS,Spez} = P_{B,Spez} \times P_{S,Spez}$.*

Der zugehörige Parameterraum ist P_{BS} .

Definition 3.2 *Der zusammengefasste Betriebs- und Herstellungsprozessparameterraum P_{BS} ist definiert als $P_{BS} = P_B \times P_S$ und hat $n_{p_{BS}} = n_{p_B} + n_{p_S}$ Dimensionen. Das zusammengefasste Betriebs- und Herstellungsprozessparametertupel ist $\mathbf{p}_{BS} \in P_{BS}$.*

Die Spezifikation enthält ferner Ungleichungsnebenbedingungen und Zielfunktionen (vgl. Kapitel 2.1). Die Ungleichungsnebenbedingungen seien in der kanonischen Form

$$\mathbf{c}(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0 \quad \forall \mathbf{p}_{BS} \in P_{BS,Spez} \quad (3.1)$$

mit dem Nebenbedingungsfunktionstupel $\mathbf{c} : P_E \times P_{BS} \rightarrow \mathbb{R}^{n_c}$ gegeben. Die Auswertung der Bedingung 3.1 beinhaltet bei praktischen Problemen die Durchführung der Worst-Case-Analyseabbildung \mathbf{a}_{WC} zur Bestimmung von Schranken der Eigenschaftsgrößen (vgl. Abschnitt 2.2.6). Das Akzeptanzgebiet A_{P_E} ist mit Bedingung 3.1

$$A_{P_E} = \{ \mathbf{p}_E \in P_E : \forall \mathbf{p}_{BS} \in P_{BS,Spez} : \mathbf{c}(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0 \} . \quad (3.2)$$

Die Zielfunktionen seien durch das Tupel

$$\mathbf{f} : P_E \times P_{BS} \rightarrow \mathbb{R}^{n_f} \quad (3.3)$$

gegeben. Ihre Berechnung erfordert in der Praxis auch die Durchführung der Analyseabbildung \mathbf{a} zur Berechnung der Eigenschaften (vgl. Abschnitt 2.2.2). Falls $n_f > 1$, handelt es sich um ein Optimierungsproblem mit mehreren Optimierungszielen. Es ist im Allgemeinen nicht möglich, mehrere Ziele unabhängig voneinander zu optimieren, da ein Zielkonflikt vorliegen kann. In dieser Arbeit werden die Ziele daher vereinigt, indem ihnen Gewichte \mathbf{g} zugewiesen werden und die Summe

$$\sum_{i=1}^{n_f} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \quad (3.4)$$

minimiert wird. Dabei ist das Gewicht g_i derjenigen f_i , die zu maximieren sind, negativ zu wählen. Die Summe 3.4 muss für alle $\mathbf{p}_{BS} \in P_{BS,Spez}$ minimiert werden. Dies kann ebenfalls zu einem Zielkonflikt führen. Es wird daher der schlechteste Fall (worst case) jedes Terms $g_i f_i$ betrachtet, d. h. das Maximum von $g_i f_i$ auf $P_{BS,Spez}$. In der Praxis muss hierfür die Worst-Case-Analyseabbildung \mathbf{a}_{WC} ausgewertet werden. Es lässt sich nun eine skalare Zielfunktion definieren.

Definition 3.3 Die skalare vereinigte Zielfunktion $f_u : P_E \rightarrow \mathbb{R}$ ist gegeben durch:

$$f_u(\mathbf{p}_E) = \sum_{i=1}^{n_f} \max_{\mathbf{p}_{BS} \in P_{BS,Spez}} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) . \quad (3.5)$$

Die zu bestimmende Lösung des Dimensionierungsproblems kann damit formal definiert werden.

Definition 3.4 Die Lösung des Dimensionierungsproblems mit Parameterschwankungen ist das Entwurfsparametertupel $\mathbf{p}_E^* \in P_E$, das durch

$$\mathbf{p}_E^* = \operatorname{argmin}_{\mathbf{p}_E \in A_{P_E}} \sum_{i=1}^{n_f} \max_{\mathbf{p}_{BS} \in P_{BS,Spez}} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \quad (3.6)$$

$$\text{mit: } A_{P_E} = \{\mathbf{p}_E \in P_E : \forall \mathbf{p}_{BS} \in P_{BS,Spez} : \mathbf{c}(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0\}$$

gegeben ist.

3.1.2 Definition des Entwurfszentrierungsproblems

Im Folgenden wird die in dieser Arbeit verwendete Form des Entwurfszentrierungsproblems durch Abwandlung des in Abschnitt 3.1.1 beschriebenen Dimensionierungsproblems definiert.

Angenommen wird eine gleichförmige Wahrscheinlichkeitsverteilung der Herstellungsprozessparameter \mathbf{p}_S innerhalb von Intervallen, deren Grenzen durch die maximalen Abweichungen vom Nominalpunkt $\mathbf{p}_{nom,S}$ definiert sind. Die maximalen Abweichungen seien durch ein Radientupel $\mathbf{p}_{S,Rad}$ gegeben:

Definition 3.5 Die Herstellungsprozessparameterradientupel $\mathbf{p}_{S,Rad}$ spannen den Herstellungsprozessparameterradienraum $P_{S,Rad}$ auf mit $\mathbf{p}_{S,Rad} \in P_{S,Rad}$ und $P_{S,Rad} \subseteq \mathbb{R}_+^{n_{ps}}$.

Mit $\mathbf{p}_{S,Rad}$ lässt sich ein Herstellungsprozessparametergebiet $P_{S,max}$ in P_S definieren.

3 Dimensionierung mit formalen Methoden

Definition 3.6 Das Herstellungsprozessparametergebiet $P_{S,max} \subseteq P_S$ ist das durch die maximale Abweichung vom Nominalpunkt $\mathbf{p}_{nom,S}$, die durch das Herstellungsprozessparametergradiententupel $\mathbf{p}_{S,Rad}$ gegeben ist, definierte Gebiet in P_S :

$$P_{S,max} = \{ \mathbf{p}_S \in P_S : | \mathbf{p}_S - \mathbf{p}_{nom,S} | \leq \mathbf{p}_{S,Rad} \} . \quad (3.7)$$

Auch wenn mit dieser Definition keine allgemeine Korrelation der Herstellungsprozessparameter ausgedrückt werden kann, können Korrelationskoeffizienten von ± 1 in jedem Fall durch Variablensubstitutionen umgesetzt werden.

Das Gebiet $P_{S,max}$ entspricht dem festen Gebiet $P_{S,Spez}$. Im Gegensatz zu diesem wird es aber als variabel behandelt. Der Entwurfszentrierungsansatz ist dann, die Entwurfparameter \mathbf{p}_E so zu bestimmen, dass $\mathbf{p}_{S,Rad}$ unter Einhaltung der Spezifikation maximiert werden kann (vgl. Abschnitt 2.3.9). Da $\mathbf{p}_{S,Rad}$ mehrere Komponenten besitzt, sind diese zur Optimierung mit Gewichten g_i zu versehen. Beispielsweise können sie als Vielfache der Standardabweichungen realer Gauß-Verteilungen der Parameter normiert werden, indem $g_i = \frac{1}{\sigma_i}$ gesetzt wird. Mit den g_i kann eine Zielfunktion formuliert werden:

Definition 3.7 Die zu maximierende Zielfunktion $f^Z : P_{S,Rad} \rightarrow \mathbb{R}$ des Entwurfszentrierungsproblems ist definiert als

$$f^Z(\mathbf{p}_{S,Rad}) = \min_{i=1}^{n_{P_S}} g_i p_{S,Rad,i} . \quad (3.8)$$

Das Gebiet $P_{S,Spez}$ ist in den Nebenbedingungen 3.1 durch das variable $P_{S,max}$ zu ersetzen, so dass sich

$$\mathbf{c}(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0 \quad \forall \mathbf{p}_{BS} \in P_{B,Spez} \times P_{S,max} \quad (3.9)$$

ergibt. Hieraus ergibt sich ein in Abhängigkeit von $P_{S,max}$ variables Akzeptanzgebiet

$$A_{P_E} = \{ \mathbf{p}_E \in P_E : \forall \mathbf{p}_{BS} \in P_{B,Spez} \times P_{S,max} : \mathbf{c}(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0 \} . \quad (3.10)$$

Beispiel 3.1 Abbildung 3.1 zeigt einen zweidimensionalen Herstellungsprozessparameterraum P_S mit dem Nominalprozessparameterpunkt $\mathbf{p}_{nom,S}$. Durch das Herstellungsprozessparametergradiententupel $\mathbf{p}_{S,Rad}$ wird

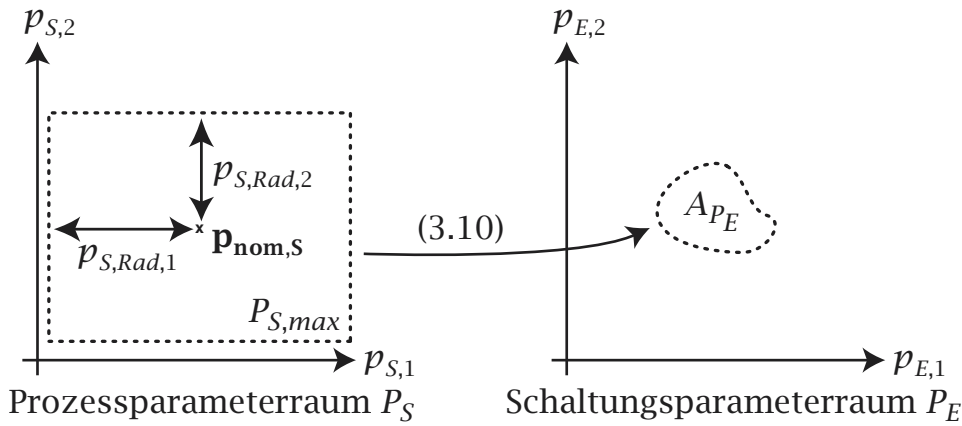


Abbildung 3.1: Herstellungprozessparametergebiet $P_{S,max}$ und Akzeptanzgebiet A_{P_E}

um $\mathbf{p}_{nom,S}$ das dargestellte Gebiet $P_{S,max}$ aufgespannt. Mit Hilfe von Gleichung 3.10 lässt sich in dem ebenfalls zweidimensional angenommenen Schaltungsparametererraum P_E das Akzeptanzgebiet A_{P_E} in Abhängigkeit von $P_{S,max}$ bestimmen.

Die Maximierung der Größe von $P_{S,max}$ entsprechend der Zielfunktion f^Z hat zur Folge, dass A_{P_E} gegen einen Punkt \mathbf{p}_E^{**} konvergiert, der Lösung des Entwurfszentrierungsproblems ist. Der Punkt \mathbf{p}_E^{**} ist wie folgt definiert:

Definition 3.8 Die Lösung des Entwurfszentrierungsproblems mit Parameterschwankungen ist ein Entwurfsparametertupel $\mathbf{p}_E^{**} \in A_{P_E}$, wobei

$$\max_{\mathbf{p}_{S,Rad} \in P_{S,Rad}} \min_{i=1}^{n_{P_S}} g_i p_{S,Rad,i} \quad (3.11)$$

unter der Randbedingung $A_{P_E} \neq \emptyset$ zu lösen ist mit

$$A_{P_E} = \{ \mathbf{p}_E \in P_E : \forall \mathbf{p}_{BS} \in P_{B,Spez} \times P_{S,max} : \mathbf{c}(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0 \} \quad (3.12)$$

und

$$P_{S,max} = \{ \mathbf{p}_S \in P_S : |\mathbf{p}_S - \mathbf{p}_{nom,S}| \leq \mathbf{p}_{S,Rad} \} . \quad (3.13)$$

Darstellung mit konstantem Akzeptanzgebiet

In Gleichung 3.12 hängt A_{P_E} über Gleichung 3.13 von $\mathbf{p}_{S,Rad}$ ab. Dies ist ein entscheidender Unterschied gegenüber dem Dimensionierungsproblem nach Definition 3.4, wo A_{P_E} konstant ist. Ein Algorithmus zur

3 Dimensionierung mit formalen Methoden

Lösung der Problems nach Definition 3.4, der die Konstanz von A_{P_E} ausnutzt, kann daher nicht verwendet werden, um das Entwurfszentrierungsproblem in der durch Definition 3.8 gegebenen Darstellung zu lösen. Um auch beim Entwurfszentrierungsproblem ein konstantes Akzeptanzgebiet zu erhalten, wird der vereinigte Raum $P_{S, Rad} \times P_E$ verwendet. Das hierin gelegene Akzeptanzgebiet $A_{P_{SE}}$ (s. Gleichung 3.15) hat die geforderte Konstanz. Die Lösung des Entwurfszentrierungsproblems \mathbf{p}_E^{**} lässt sich dann durch

$$(\mathbf{p}_{S, Rad}, \mathbf{p}_E^{**}) = \underset{(\mathbf{p}_{S, Rad}, \mathbf{p}_E^{**}) \in A_{P_{SE}}}{\operatorname{argmax}} \min_{i=1}^{n_{P_S}} g_i p_{S, Rad, i} \quad (3.14)$$

darstellen. Das Akzeptanzgebiet $A_{P_{SE}}$ ist dabei durch

$$A_{P_{SE}} = \{ (\mathbf{p}_{S, Rad}, \mathbf{p}_E) \in P_{S, Rad} \times P_E : \forall \mathbf{p}_{BS} \in P_{B, Spez} \times P_{S, max} : \mathbf{c}(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0 \} \quad (3.15)$$

gegeben. Für $P_{S, max}$ gilt weiterhin Gleichung 3.13.

Damit entspricht das nach dieser Darstellung zu lösende Optimierungsproblem in der Struktur dem Dimensionierungsproblem in Definition 3.4, so dass zur Lösung derselbe Algorithmus verwendet werden kann. Von dieser Möglichkeit wird in dieser Arbeit Gebrauch gemacht. Nachfolgend wird zur Vereinfachung der Darstellung ausschließlich die Formulierung in Definition 3.4 verwendet.

3.2 Gebietsarithmetiken

Die Berechnung von \mathbf{p}_E^* nach Definition 3.4 erfordert die Bestimmung des globalen Minimums der vereinigten Zielfunktion f_u unter der Randbedingung, dass \mathbf{p}_E^* in A_{P_E} enthalten ist. Um die Korrektheit der Lösung \mathbf{p}_E^* garantieren zu können, muss bewiesen sein, dass $\forall \mathbf{p}_{BS} \in P_{BS, Spez} : \mathbf{c}(\mathbf{p}_E^*, \mathbf{p}_{BS}) \geq 0$ (vgl. Gleichung 3.2) und $\forall \mathbf{p}_E \in P_E : \mathbf{p}_E \neq \mathbf{p}_E^* \Rightarrow f_u(\mathbf{p}_E) \geq f_u(\mathbf{p}_E^*)$. Ferner muss bei der Berechnung von $\max_{\mathbf{p}_{BS} \in P_{BS, Spez}} g_i f_i(\mathbf{p}_E^*, \mathbf{p}_{BS})$ jeweils ein $\mathbf{p}_{BS, i}^*$ bestimmt werden, für das $\forall \mathbf{p}_{BS} \in P_{BS, Spez} : g_i f_i(\mathbf{p}_E^*, \mathbf{p}_{BS}) \leq g_i f_i(\mathbf{p}_E^*, \mathbf{p}_{BS, i}^*)$ bewiesen ist.

In allen drei Fällen muss eine Aussage für alle Punkte in einem Gebiet bewiesen werden. Dieses ist mit herkömmlicher Gleitkomma-Arithmetik nicht in endlicher Zeit möglich. Die in den folgenden Abschnitten vorgestellten Werkzeuge der Intervallarithmetik und der affinen Arithmetik, in dieser Arbeit zusammenfassend als Gebietsarithmetiken bezeichnet,

ermöglichen den Beweis derartiger Aussagen, da mit ihrer Hilfe Ergebnisse nicht punkt-, sondern gebietsweise berechnet werden.

Die hier angeführten Überlegungen sind entsprechend auch auf die Berechnung von \mathbf{p}_E^{**} nach Gleichung 3.14 anzuwenden.

3.2.1 Intervallarithmetik

Die Intervallarithmetik ist ein von Moore (1966) entwickeltes Werkzeug zur Lösung von Problemen mit endlichen Gebieten, das mittlerweile in vielen Bereichen Anwendung gefunden hat (vgl. Abschnitt 1.3). Implementierungen sind auf akademischer und zunehmend auch auf kommerzieller Ebene verfügbar. Letztere finden sich nicht nur in Computer-Algebra-Systemen, sondern mittlerweile auch in Compiler-Bibliotheken für den produktiven Einsatz (SUN 2003).

Die für diese Arbeit relevanten Begriffe und Eigenschaften der Intervallarithmetik werden nachfolgend dargestellt.

Definition 3.9 Ein reelles Intervall $[X]$ ist eine Untermenge von \mathbb{R} mit

$$[X] \equiv [\underline{x}, \bar{x}] \equiv [\underline{x}; \bar{x}] = \{x \in \mathbb{R} : \underline{x} \leq x \leq \bar{x}\} . \quad (3.16)$$

Darin sind \underline{x} und \bar{x} die Unter- bzw. Obergrenze des Intervalls.

Ein Sonderfall ist das Punktintervall:

Definition 3.10 Ein Punktintervall $[x]$ ist gegeben durch:

$$[x] \equiv [x, x] = \{x\} . \quad (3.17)$$

Definition 3.11 Die Menge aller reellen Intervalle wird mit $\mathbb{I}_{\mathbb{R}}$ bezeichnet.

Entsprechende Definitionen gelten für Intervalltupel $[X] \in \mathbb{I}_{\mathbb{R}}^{n_x}$ und Punktintervalltupel $[y] \in \mathbb{I}_{\mathbb{R}}^{n_y}$.

Inklusionsisotonie

Definition 3.12 Die Eigenschaft der Inklusionsisotonie einer Funktion $[F] : \mathbb{I}_{\mathbb{R}}^n \rightarrow \mathbb{I}_{\mathbb{R}}$ besagt, dass für $[X] \in \mathbb{I}_{\mathbb{R}}^n$, $[Y] \in \mathbb{I}_{\mathbb{R}}^n$ die Folgerung:

$$[X] \subseteq [Y] \Rightarrow [F([X])] \subseteq [F([Y])] \quad (3.18)$$

gilt.

3 Dimensionierung mit formalen Methoden

Definition 3.13 Eine Intervallerweiterung für eine Funktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ ist eine inklusionsisotone Funktion $[F] : \mathbb{I}_{\mathbb{R}}^{n_x} \rightarrow \mathbb{I}_{\mathbb{R}}$ mit der Eigenschaft, dass $[F([\mathbf{x}])] = f(\mathbf{x})$. Hierin ist $[\mathbf{x}]$ ein Punktintervall.

In Zusammenhang mit der Eigenschaft, dass eine Intervallerweiterung $[F]$ für Punktintervalltupel $[\mathbf{x}]$ die gleichen Werte wie die Funktion f , die sie erweitert, liefert, lässt sich aus der Inklusionsisotonie eine zentrale Eigenschaft der Intervallarithmetik folgern: Eine Intervallerweiterung liefert mit $[F([\mathbf{X}])]$ ein Intervall, das garantiert alle Werte, die $f(\mathbf{x})$ für $\mathbf{x} \in [\mathbf{X}]$ einnehmen kann, enthält, d. h.

$$\forall \xi \in [\mathbf{X}] : f(\xi) \in [F([\mathbf{X}])] . \quad (3.19)$$

Das Intervall $[F([\mathbf{X}])]$ wird als Einschließung von f auf $[\mathbf{X}]$ bezeichnet. Um die Inklusionsisotonie in praktischen Anwendungen zu gewährleisten, müssen Rechenoperationen mit nach außen gerichtete Rundungsoperationen durchgeführt werden. Damit lassen sich mit einem numerischen Verfahren garantierte Schranken in Form der Intervallgrenzen berechnen.

Vereinigte Intervallerweiterung

Die Güte einer Einschließung wird anhand des Durchmessers des einschließenden Intervalls bewertet:

Definition 3.14 Der Durchmesser d eines Intervalltupels $[\mathbf{X}] \in \mathbb{I}_{\mathbb{R}}^{n_x}$ ist definiert als

$$d([\mathbf{X}]) = \|\bar{\mathbf{x}} - \underline{\mathbf{x}}\| . \quad (3.20)$$

Die schmalstmögliche Einschließung ist durch die *vereinigte Intervallerweiterung* gegeben:

Definition 3.15 Für eine Funktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ ist die *vereinigte Intervallerweiterung* $[F_v] : \mathbb{I}_{\mathbb{R}}^{n_x} \rightarrow \mathbb{I}_{\mathbb{R}}$:

$$[F_v([\mathbf{X}])] = \{f(\mathbf{x}) : \mathbf{x} \in [\mathbf{X}]\} . \quad (3.21)$$

Die vereinigte Intervallerweiterung liefert damit den exakten Wertebereich von f im Intervall $[\mathbf{X}]$. Für eine beliebige Intervallerweiterung $[F]$ und die vereinigte Intervallerweiterung $[F_v]$ von f gilt daher

$$[F([\mathbf{X}])] \supseteq [F_v([\mathbf{X}])] . \quad (3.22)$$

Dies ist äquivalent zu

$$d([F([\mathbf{X}]]) \geq d([F_v([\mathbf{X}]]) . \quad (3.23)$$

Die Berechnung der Intervallgrenzen von $[F_v]$ für beliebige f ist ein Extremwertproblem. Ein numerisches Verfahren auf der Grundlage herkömmlicher Gleitkomma-Arithmetik kann die erforderliche Inklusionsisotonie nicht gewährleisten, da Rundungsfehler auftreten und die globalen Extrema nicht mit Sicherheit gefunden werden können (vgl. Bemerkungen zur Worst-Case-Analyse in Abschnitt 2.2.6 und zur globalen Optimierung in Abschnitt 2.3.6). Lediglich für einfache Operationen, z. B. Addition, Multiplikation und transzendente Funktionen, bei denen sich die Intervallgrenzen analytisch bestimmen lassen, lassen sich geschlossene Ausdrücke für $[F_v]$ angeben. Die bei Dimensionierungsproblemen auftretenden Ausdrücke sind hierfür jedoch zu komplex.

Beispiel 3.2 Die Multiplikation zweier Zahlen a und b sei durch die Funktion $m(a, b) = a \cdot b$ gegeben. Die vereinigte Intervallerweiterung $[M_v]$ lässt sich durch

$$[M_v([A], [B])] = [\min \mathbf{p}, \max \mathbf{p}] \quad (3.24)$$

mit: $\mathbf{p} = (\underline{a} \cdot \underline{b}, \underline{a} \cdot \bar{b}, \bar{a} \cdot \underline{b}, \bar{a} \cdot \bar{b})$

berechnen.

Natürliche Intervallerweiterung

Aus den vereinigten Intervallerweiterungen für einfache Operationen lassen sich durch Verkettung Intervallerweiterungen für komplexe Ausdrücke bestimmen, deren vereinigte Intervallerweiterungen sich nicht analytisch bestimmen lassen. Die so erhaltenen Intervallerweiterungen werden als *natürliche Intervallerweiterungen* bezeichnet.

Definition 3.16 Die natürliche Intervallerweiterung $[F_n] : \mathbb{I}_{\mathbb{R}}^{n \times x} \rightarrow \mathbb{I}_{\mathbb{R}}$ einer durch einen arithmetischen Ausdruck definierten Funktion $f : \mathbb{R}^{n \times x} \rightarrow \mathbb{R}$ ergibt sich durch Ersetzen aller \mathbf{x} durch $[\mathbf{X}]$ und aller Operationen und Funktionen durch ihre vereinigten Intervallerweiterungen im arithmetischen Ausdruck, der $f(\mathbf{x})$ definiert.

Überschätzung

Nach Relation 3.22 kann das Ergebnisintervall von $[F]$ breiter sein als das schmalstmögliche Ergebnis, das sich aus $[F_v]$ ergibt. Dieser Effekt,

3 Dimensionierung mit formalen Methoden

der bei den meisten praktischen intervallarithmetischen Rechnungen auftritt, wird als Überschätzung bezeichnet.

Die Ursache der Überschätzung liegt, abgesehen vom nach außen gerichteten Runden, hauptsächlich darin, dass intervallarithmetische Operationen keine Korrelationsinformationen nutzen können, da Intervalle nach Definition 3.9 diese nicht enthalten. Dadurch wird bei einer Verknüpfungsoperation zwischen zwei oder mehr Intervallen grundsätzlich von unkorrelierten Größen ausgegangen. Dies führt zum breitestmöglichen Ergebnisintervall, da zur Erhaltung der Inklusionsisotonie der schlechteste Fall angenommen werden muss. Das folgende Beispiel verdeutlicht diesen Effekt.

Beispiel 3.3 Gegeben sind $-1 \leq a \leq 1$ und $-1 \leq b \leq 1$. Die Veränderlichen a und b sind unabhängig voneinander, also unkorreliert. Es sollen Minimum und Maximum des Produktes $a \cdot b$, welche analytisch zu -1 bzw. 1 bestimmt werden können, intervallarithmetisch berechnet werden. Dazu werden die Intervalle $[A] = [-1, 1]$ und $[B] = [-1, 1]$ definiert und die vereinigte Intervallerweiterung für die Multiplikation $[M_v]$ aus Beispiel 3.2 verwendet. Das Ergebnis ist:

$$[M_v([A], [B])] = [M_v([-1, 1], [-1, 1])] = [-1, 1]. \quad (3.25)$$

Dieses ist gleich dem analytisch berechneten.

In einem zweiten Fall sollen Minimum und Maximum von $a \cdot a$ berechnet werden, die sich bei analytischem Vorgehen zu 0 bzw. 1 ergeben. Die natürliche Intervallerweiterung nach Definition 3.16 führt beim Einsetzen der numerischen Intervalle in $[M_v([A], [A])]$ zu

$$[M_v([A], [A])] = [M_v([-1, 1], [-1, 1])] = [-1, 1]. \quad (3.26)$$

Durch das Einsetzen ist die Information, dass es sich bei beiden Argumenten um $[A]$ handelt, die Argumente also korreliert sind, verloren gegangen. Daraus ergibt sich ein überschätztes Ergebnis, das dem der obigen Rechnung mit unkorrelierten Intervallen $[A]$ und $[B]$ gleich ist.

Die natürliche Intervallerweiterung erzeugt für komplexe Ausdrücke einen Baum von Intervalloperationen mit zahlreichen Zwischenergebnissen, deren Korrelation nicht berücksichtigt wird. Dadurch kann die Überschätzung ohne Gegenmaßnahmen so stark anwachsen, dass die aus den berechneten Intervallgrenzen erhaltenen Schranken unbrauchbar werden.

Die Überschätzung des durch eine beliebige Intervallerweiterung berechneten Intervalls $[F([\mathbf{X}]])$ gegenüber dem mit der vereinigten Intervallerweiterung erzielten Ergebnis $[F_v([\mathbf{X}]])$ kann durch die Differenz Δ der Durchmesser bewertet werden. Für die Überschätzung der natürlichen Intervallerweiterung bei $d([\mathbf{X}]) \rightarrow 0$ gilt dabei allgemein

$$\Delta = d([F_n([\mathbf{X}]]) - d([F_v([\mathbf{X}]]) = \mathcal{O}(d([\mathbf{X}])). \quad (3.27)$$

Die Überschätzung der natürlichen Intervallerweiterung konvergiert also für schmaler werdende Argumente linear gegen 0. Eine Gegenmaßnahme gegen die Überschätzung bei Argumenten endlicher Breite ist daher ihre Aufteilung in Teilintervalle $[\mathbf{X}_i]$ mit anschließender einzelner Auswertung der Intervallerweiterung. Die Ergebnisse sind zu $\bigcup_i [F([\mathbf{X}_i])]$ zu vereinigen. Entsprechend der Konvergenzeigenschaft 3.27 verringert sich dabei die Überschätzung dann linear mit der Anzahl der Aufteilungen pro Dimension des Arguments. Jedoch wächst die Anzahl der zu berechnenden Teilintervalle und somit der Rechenaufwand exponentiell mit der Zahl der Dimensionen an.

Mittelwertsform

Eine weitere Intervallerweiterung ist die Mittelwertsform.

Definition 3.17 Die Mittelwertsform-Intervallerweiterung $[F_m] : \mathbb{I}_{\mathbb{R}}^{n_x} \rightarrow \mathbb{I}_{\mathbb{R}}$ einer Funktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ ist durch

$$[F_m([\mathbf{X}])] = f(\mathbf{x}_0) + [\nabla F([\mathbf{X}])] \cdot ([\mathbf{X}] - \mathbf{x}_0) \quad (3.28)$$

mit $\mathbf{x}_0 \in [\mathbf{X}]$ und $[\nabla F]$ als natürliche Intervallerweiterung von ∇f gegeben.

Für die Überschätzung von $[F_m([\mathbf{X}])]$ gilt für $d([\mathbf{X}]) \rightarrow 0$ eine quadratische Konvergenzeigenschaft:

$$d([F_m([\mathbf{X}])]) - d([F_v([\mathbf{X}])]) = \mathcal{O}(d([\mathbf{X}])^2). \quad (3.29)$$

Diese Verbesserung der Konvergenzordnung gegenüber der natürlichen Intervallerweiterung wird durch die zusätzlich erforderliche Auswertung von $[\nabla F]$ erkauft. Aufgrund der hohen Komplexität, die die Ausdrücke für ∇f bei praktischen Problemen haben, führt die Überschätzung von $[\nabla F]$ meist dazu, dass $[F_m([\mathbf{X}])]$ stärker überschätzt ist als $[F_n([\mathbf{X}])]$ (vgl. Abschnitt 4.2.4).

3.2.2 Affine Arithmetik

Von mehreren veröffentlichten Ansätzen, Korrelationsinformationen bei der Rechnung mit Intervallen zu berücksichtigen, ist der jüngste die von de Figueiredo und Stolfi (1997) entwickelte *affine Arithmetik*.

Affine Formen

In der affinen Arithmetik werden Größen durch *affine Formen* beschrieben:

Definition 3.18 Eine reelle affine Form \hat{x}_ε ist definiert als

$$\hat{x}_\varepsilon = x_0 + \sum_{i=1}^{n_\varepsilon} x_i \varepsilon_i . \quad (3.30)$$

Darin ist $x_0 \in \mathbb{R}$ der Mittelwert und $\mathbf{x} \in \mathbb{R}^{n_\varepsilon}$ das Koeffiziententupel von \hat{x}_ε sowie ε das globale Epsilontupel.

Definition 3.19 Das globale Epsilontupel ε der affinen Arithmetik ist ein veränderliches Tupel $\varepsilon \in \mathbb{U}^{n_\varepsilon}$, $\mathbb{U} = [-1, 1]$, dessen Wert in allen affinen Formen gleich ist.

Definition 3.20 Die Menge aller reellen affinen Formen mit dem globalen Epsilontupel ε wird als \mathbb{A}_ε bezeichnet.

Entsprechende Definitionen gelten für Tupel affiner Formen $\hat{\mathbf{x}}_\varepsilon \in \mathbb{A}_\varepsilon^{n_x}$.

Durch die Veränderlichkeit der Komponenten ε_i von ε drücken affine Formen ebenso wie Intervalle Zahlenbereiche aus. Wegen der Globalität von ε kann darüber hinaus zusätzlich die lineare Korrelation von affinen Formen ausgedrückt werden: Zwei affine Formen \hat{x}_ε und \hat{y}_ε sind genau dann linear korreliert, wenn $\exists i : x_i \neq 0 \wedge y_i \neq 0$.

Umwandlungsoperationen zwischen affinen Formen und Intervallen sind gegeben durch:

Definition 3.21 Einer affinen Form $\hat{x}_\varepsilon \in \mathbb{A}_\varepsilon$ wird mit dem Operator $[I] : \mathbb{A}_\varepsilon \rightarrow \mathbb{I}_\mathbb{R}$ nach folgender Vorschrift ein Intervall zugeordnet:

$$[I(\hat{x}_\varepsilon)] = x_0 + \sum_{i=1}^{n_\varepsilon} x_i \cdot [-1, 1] . \quad (3.31)$$

Definition 3.22 Ein Intervall $[X] \in \mathbb{I}_{\mathbb{R}}$ lässt sich mit dem Operator $\hat{\mathcal{A}}_{\varepsilon} : \mathbb{I}_{\mathbb{R}} \rightarrow \mathbb{A}_{\varepsilon}$ in eine affine Form konvertieren:

$$\hat{\mathcal{A}}_{\varepsilon}([X]) = \frac{\bar{x} + \underline{x}}{2} + \frac{\bar{x} - \underline{x}}{2} \varepsilon_k. \quad (3.32)$$

Die Wahl von k hat dabei so zu erfolgen, dass für unkorrelierte affine Formen \hat{x}_{ε} : $x_k = 0$ gilt. In praktischen Anwendungen mit sequentiellen Berechnungen wird dies durch Setzen von $k = n_{\varepsilon} + 1$ und nachfolgende Erhöhung von n_{ε} gewährleistet, wobei die Koeffiziententupel aller anderen affinen Formen mit einer Null ergänzt werden.

Beide Umwandlungsoperatoren sind auf Tupel komponentenweise anzuwenden.

Affine Operationen

Definition 3.23 Gegeben sind $\hat{x}_{\varepsilon}, \hat{y}_{\varepsilon} \in \mathbb{A}_{\varepsilon}$ und $a \in \mathbb{R}$. Die affinen Operationen sind $\hat{x}_{\varepsilon} \pm \hat{y}_{\varepsilon}$, $a \pm \hat{x}_{\varepsilon}$ und $a\hat{x}_{\varepsilon}$.

Die Definition gilt für Tupel affiner Formen entsprechend.

Die Ergebnisse der affinen Operationen sind wiederum affine Formen, die sich durch Anwenden von Definition 3.18 ergeben. Durch die Linearität der affinen Operationen wird die durch die Koeffizienten der ε_i ausgedrückte lineare Korrelation vollständig ausgenutzt, und das Ergebnis weist keine Überschätzung auf.

Beispiel 3.4 Bild 3.2 zeigt den Unterschied in der Korrelationserhaltung zwischen Intervallarithmetik und affiner Arithmetik graphisch: Es sind die Multiplikation eines Intervalls $[X] = [\underline{x}, \bar{x}]$ bzw. einer affinen Form $\hat{x}_{\varepsilon} = x_0 + x_1 \varepsilon_1$ mit einem reellen Skalierungsfaktor a dargestellt. Die Ergebnisse sind $a[X] = [a\underline{x}, a\bar{x}]$ bzw. $a\hat{x}_{\varepsilon} = ax_0 + ax_1 \varepsilon_1$. Die Argumente und die Ergebnisse sind jeweils als Achsenabschnitte auf den Abszissen bzw. Ordinaten dargestellt. Diese Abschnitte zeigen, dass die Ergebnisbereiche für sich genommen identisch sind. Die dargestellten Punktmengen verdeutlichen jedoch, dass bei Verwendung von Intervallarithmetik die Korrelation zwischen Argument und Ergebnis der Operation verloren ist, von der affinen Arithmetik jedoch erhalten wird, da die Werte von Argument \hat{x}_{ε} und Ergebnis $a\hat{x}_{\varepsilon}$ über die Variable ε_1 verbunden sind.

Wird in einem weiteren Rechenschritt das Argument $[X]$ bzw. \hat{x}_{ε} vom Ergebnis subtrahiert, so liefert die Intervallarithmetik das überschätzte Ergebnisintervall $[a\underline{x} - \bar{x}, a\bar{x} - \underline{x}] \neq (a - 1)[X]$, weil der Zusammenhang, also die Korrelation, zwischen $[X]$ und $a[X]$ verloren gegangen ist. Die affine Arithmetik nutzt die erhaltene Korrelation jedoch und liefert daher das exakte Ergebnis $(a - 1)x_0 + (a - 1)x_1 \varepsilon_1 = (a - 1)\hat{x}_{\varepsilon}$.

3 Dimensionierung mit formalen Methoden

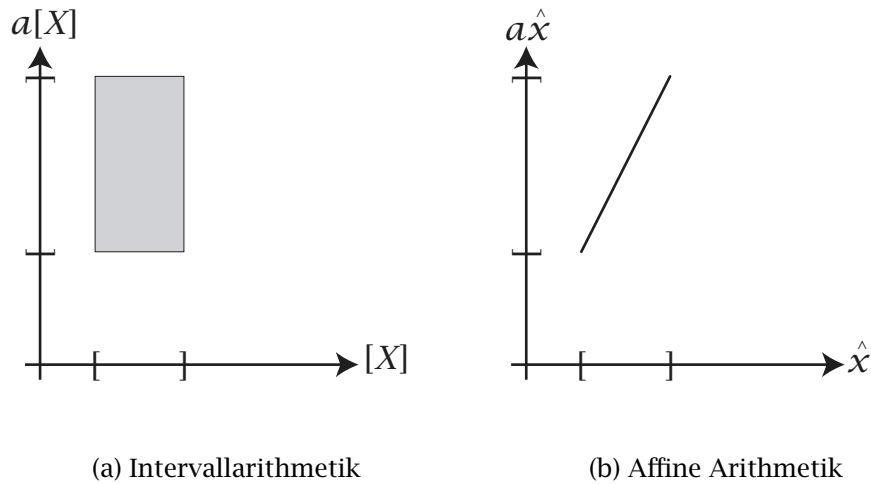


Abbildung 3.2: Korrelation bei Multiplikation mit reeller Zahl a

Affine Erweiterungen

Alle nicht in Definition 3.2.2 aufgeführten Operationen ergeben eine nichtlineare Korrelation zwischen Argumenten und Ergebnis. Diese kann jedoch bei Beschränkung auf Ausdrücke der Form 3.30 nicht dargestellt werden. Es wird daher als *affine Erweiterung* eine Einschließung mit einer die Nichtlinearität annähernden linearen Korrelation und einem unkorrelierten Fehlerintervall verwendet.

Definition 3.24 Für eine Funktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ ist eine affine Erweiterung $\hat{f}_\varepsilon : \mathbb{A}_\varepsilon^{n_x} \rightarrow \mathbb{A}_\varepsilon$ gegeben durch

$$\hat{f}_\varepsilon(\hat{\mathbf{x}}_\varepsilon) = a(\hat{\mathbf{x}}_\varepsilon) \cdot \hat{\mathbf{x}}_\varepsilon + b(\hat{\mathbf{x}}_\varepsilon) + \hat{\mathcal{A}}_\varepsilon([C(\hat{\mathbf{x}}_\varepsilon)]) . \quad (3.33)$$

Dabei ist $[C] : \mathbb{A}_\varepsilon^{n_x} \rightarrow \mathbb{I}_\mathbb{R}$ definiert als

$$[C(\hat{\mathbf{x}}_\varepsilon)] = [C_v([I(\hat{\mathbf{x}}_\varepsilon)])] \quad (3.34)$$

mit C_v als vereinigter Intervallerweiterung von $c(x) = f(x) - ax - b$. Die Funktionen $a, b : \mathbb{A}_\varepsilon^{n_x} \rightarrow \mathbb{R}$ sind so zu wählen, dass

$$\exists \xi, \zeta \in I(\hat{\mathbf{x}}_\varepsilon) : \nabla f(\xi) = a(\hat{\mathbf{x}}_\varepsilon) \wedge f(\zeta) = a(\hat{\mathbf{x}}_\varepsilon) \cdot \zeta + b(\hat{\mathbf{x}}_\varepsilon) . \quad (3.35)$$

Durch die Vorschrift 3.34 wird gewährleistet, dass die Inklusionsisotonie nach Definition 3.12 im Sinne von

$$[I(\hat{\mathbf{x}}_\varepsilon)] \subseteq [I(\hat{\mathbf{y}}_\varepsilon)] \Rightarrow [I(\hat{f}_\varepsilon(\hat{\mathbf{x}}_\varepsilon))] \subseteq [I(\hat{f}_\varepsilon(\hat{\mathbf{y}}_\varepsilon))] \quad (3.36)$$

erhalten bleibt.

Infolge der Linearisierung von Nichtlinearitäten kommt es durch die unkorrelierten Fehlerintervalle $[C(\hat{\mathbf{x}}_\varepsilon)]$ auch bei Rechnungen mit affiner Arithmetik zu Überschätzung. Die Bedingung 3.35 führt dabei zu einer quadratischen Konvergenzordnung für $d([\mathcal{I}(\hat{\mathbf{x}}_\varepsilon)]) \rightarrow 0$:

$$d([\mathcal{I}(\hat{f}_\varepsilon(\hat{\mathbf{x}}_\varepsilon))]) - d([F_v([\mathcal{I}(\hat{\mathbf{x}}_\varepsilon)])]) = \mathcal{O}(d([\mathcal{I}(\hat{\mathbf{x}}_\varepsilon)])^2). \quad (3.37)$$

Wie bei der vereinigten Intervallerweiterung lassen sich für einfache Operationen, z. B. Addition, Multiplikation und transzendente Funktionen, affine Erweiterungen nach Definition 3.24 analytisch bestimmen. Ebenfalls ist es möglich, durch Verkettung affine Erweiterungen für komplexe Ausdrücke zu berechnen, für die sich keine affinen Erweiterungen auf analytischem Weg bestimmen lassen. Die quadratische Konvergenzordnung 3.37 bleibt dabei erhalten.

Vergleich mit der Mittelwertsform der Intervallarithmetik

Die mit Intervallarithmetik durch die Mittelwertsform erreichbare Konvergenzordnung entspricht der der affinen Arithmetik (vgl. Gleichungen 3.29 und 3.37). Für Argumente endlicher Breite liefert die affine Arithmetik schmalere Einschließungen, da keine Intervallauswertung von Ableitungen durchgeführt wird. Ein weiterer Vorteil der affinen Arithmetik gegenüber der Mittelwertsform-Intervallerweiterung betrifft die Berechnung von Einschließungen mehrdimensionaler Funktionen. Die mit affiner Arithmetik berechneten Einschließungen drücken durch die affinen Formen die Korrelation der einzelnen Komponenten untereinander aus. Die mit der Mittelwertsform erzielten Ergebnisse sind jedoch Intervalltupel, bei denen die komponentenweise Korrelationsinformation verworfen ist. Die geometrische Darstellung des Tupels aus affinen Formen ist ein punktsymmetrisches, konvexes Polytop. Das Intervalltupel ist dagegen ein achsenparalleler Hyperquader, der das Polytop einschließt. Die affine Arithmetik ist damit für mehrdimensionale Einschließungen besser geeignet.

Wie bei der Intervallarithmetik können auch bei der affinen Arithmetik Aufteilungen erforderlich sein, um die Überschätzung zu verringern, so dass die Laufzeitkomplexität exponentiell von der Zahl der aufzuteilenden Dimensionen abhängt. Wegen der besseren Konvergenzeigenschaften ist der exponentielle Anstieg jedoch flacher als bei der Intervallarithmetik.

Minimalbereichserweiterung

Die Freiheitsgrade, die Definition 3.24 hinsichtlich der Wahl von a und b lässt, ermöglichen es, die affine Erweiterung hinsichtlich gewünschter Eigenschaften zu optimieren:

Definition 3.25 Die Minimalbereichserweiterung ist diejenige affine Erweiterung \hat{f}_ε , für die $[1(\hat{f}_\varepsilon(\hat{\mathbf{x}}_\varepsilon))]$ minimal wird.

Neben der in dieser Arbeit verwendeten Minimalbereichserweiterung gibt es weitere spezielle affine Erweiterungen, die hinsichtlich anderer Kriterien optimal sind (de Figueiredo und Stolfi 1997).

Beispiel 3.5 Für $f(x) = \sqrt{x}$ wird eine Minimalbereichserweiterung $\hat{f}_\varepsilon(\hat{x}_\varepsilon) = a\hat{x}_\varepsilon + b + \hat{\mathcal{A}}_\varepsilon([C])$ bestimmt. Als Abkürzung wird $[X] = [\underline{x}, \bar{x}] = [1(\hat{x}_\varepsilon)]$ verwendet.

Zur Erzielung eines minimalen $[1(\hat{f}_\varepsilon(\hat{\mathbf{x}}_\varepsilon))]$ werden a und b der Geradengleichung $y = ax + b$ der Tangente an $f(x)$ im Punkt $(\bar{x}, \sqrt{\bar{x}})$ entnommen. Dies entspricht der Wahl von $\xi = \zeta = \bar{x}$ in Bedingung 3.35. Dann ist

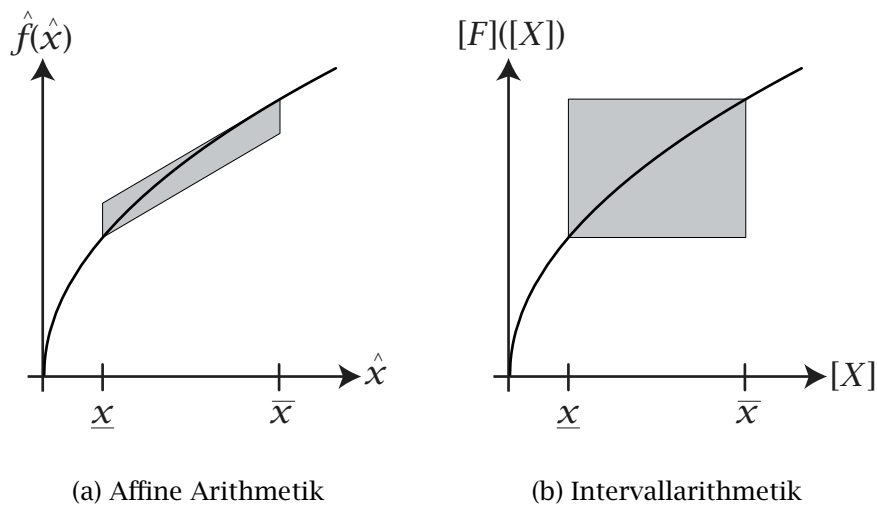
$$a = \frac{1}{2\sqrt{\bar{x}}}, \quad (3.38)$$

$$b = \frac{\sqrt{\bar{x}}}{2}, \quad (3.39)$$

$$[C] = \left[\frac{\underline{x}}{2\sqrt{\bar{x}}} + \frac{\sqrt{\bar{x}}}{2} - \sqrt{\underline{x}}, \quad 0 \right]. \quad (3.40)$$

Dies ist graphisch in Abbildung 3.3(a) dargestellt. Das Parallelogramm enthält dabei die Punktmenge, die sich aus der Korrelation von \hat{x}_ε und $\hat{f}_\varepsilon(\hat{x}_\varepsilon)$ ergibt. Zur Verdeutlichung ist ferner der Verlauf von f abgebildet. Der endliche Flächeninhalt des Parallelogramms repräsentiert den teilweisen Verlust von Korrelationsinformation zwischen \hat{x}_ε und $\hat{f}_\varepsilon(\hat{x}_\varepsilon)$ durch die Linearisierung und Einführung eines Fehlerintervalls. Zum Vergleich ist die entsprechende Punktmenge, die sich bei Verwendung von Intervallarithmetik ergibt, in Abbildung 3.3(b) gezeigt. Das einschließende achsenparallele Rechteck zeigt, dass die Korrelation hierbei vollständig verworfen wird. Der Flächeninhalt ist dadurch erheblich größer als der des Parallelogramms, das die affine Arithmetik ergibt.

An diesem Beispiel wird deutlich, dass sich die unter Anwendung von Definition 3.24 berechneten Einschließungen in Gestalt von affinen Formen als Linearisierungen mit garantierten Fehlerschranken auffassen lassen.

Abbildung 3.3: Korrelation bei $f(x) = \sqrt{x}$

Begrenzung der Laufzeitkomplexität

Die Länge des globalen Epsilontupels n_ε wächst wie unter Definition 3.22 beschrieben bei jeder Verwendung des Operators $\hat{\mathcal{A}}_\varepsilon$ und damit auch bei jeder Auswertung einer affinen Erweiterung an. Dies führt dazu, dass die als Ergebnis einzelner Rechenschritte entstehenden affinen Formen linear mit der Anzahl der Schritte anwachsen. Da dies eine quadratische Rechenzeitkomplexität in Abhängigkeit der Schrittzahl verursacht, muss dieser Prozess begrenzt werden. Dazu sind regelmäßig eine Anzahl n der betragsmäßig kleinsten von Null verschiedenen Korrelationskoeffizienten in den Koeffiziententupeln (vgl. Definition 3.18) der affinen Formen zu einem unkorrelierten Fehlerintervall zusammenzufassen. Dadurch verringert sich die Anzahl der von Null verschiedenen Koeffizienten um $n - 1$, so dass bei nachfolgenden Rechenschritten entsprechend weniger Gleitkomma-Operationen erfolgen müssen. Auf diese Weise kann eine lineare Laufzeitkomplexität in Abhängigkeit der Schrittzahl erzielt werden.

Alternativ kann die Kette der Berechnungsschritte an geeigneter Stelle dadurch unterbrochen werden, dass alle affinen Formen in Intervalle umgewandelt werden. Dieses entspricht der Wahl von $n = n_\varepsilon$ bei einem Vorgehen nach obigem Muster.

3 Dimensionierung mit formalen Methoden

	Intervallarithmetik		Affine Arithmetik
	Natürliche Intervall-erweiterung	Mittelwertsform	
Überschätzung	$\mathcal{O}(d([\mathbf{X}_i]))$ $= \mathcal{O}\left(\frac{1}{n}\right)$	$\mathcal{O}(d([\mathbf{X}_i])^2)$ $= \mathcal{O}\left(\frac{1}{n^2}\right)$	$\mathcal{O}(d([\mathbf{X}_i])^2)$ $= \mathcal{O}\left(\frac{1}{n^2}\right)$
Überschätzung bei großen Gebieten	Mittel	Hoch	Mittel
Korrelationsinformation im Ergebnis	Nein	Nein	Ja
Eignung für mehrdimensionale Einschließungen	Schlecht	Schlecht	Gut
Rechenaufwand pro Operation	Gering	Mittel	Mittel

Tabelle 3.1: Vergleich von Gebietsarithmetiken ($[\mathbf{X}_i]$ ist ein Teilintervall, n die Anzahl der Unterteilungen)

3.2.3 Vergleich und Bewertung

Die Eigenschaften der Gebietsarithmetiken Intervallarithmetik und affine Arithmetik werden abschließend in Tabelle 3.1 gegenübergestellt. Für die Intervallarithmetik sind die Varianten natürliche Intervallerweiterung und Mittelwertsform getrennt aufgeführt. Die Tabelle zeigt, dass die affine Arithmetik Ergebnisse mit der geringsten Überschätzung liefert und am besten für mehrdimensionale Einschließungen geeignet ist. Die Intervallarithmetik hingegen bietet den geringsten Rechenaufwand pro Operation. Im hier vorgestellten Verfahren wird daher bei der Berechnung von Einschließungen zunächst eine schnelle, grobe Einschließung mit Intervallarithmetik berechnet. Wenn die damit erzielte Genauigkeit nicht ausreicht, wird eine exaktere Einschließung mit affiner Arithmetik ermittelt.

3.3 Dimensionierungsalgorithmus mit formalen Methoden

Dieser Abschnitt beschreibt die Anwendung der Gebietsarithmetik auf die in den Abschnitten 3.1.1 und 3.1.2 definierten Optimierungsprobleme. Die Behandlung der Teilaufgaben globale Optimierung, Worst-Case-Analyse und Lösung nichtlinearer Gleichungssysteme mit Parameterschwankungen werden in den folgenden Abschnitten im Einzelnen erläutert.

3.3.1 Globale Optimierung

Globale Optimierung ohne Nebenbedingungen

Zunächst wird das globale Optimierungsproblem $\min_{\mathbf{x} \in \mathbb{R}^{n_x}} f(\mathbf{x})$ ohne Nebenbedingungen mit einer Zielfunktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ betrachtet. Falls statt des Minimums ein Maximum gesucht wird, ist f durch $-f$ zu ersetzen. Ein Lösungsverfahren mit garantierter Konvergenz auf der Basis von Intervallarithmetik ist u. a. von Hansen (1980) vorgeschlagen worden. Das Prinzip dieses Verfahrens wird in dieser Arbeit zu einem Ansatz für globale Optimierung mit Nebenbedingungen, Parameterschwankungen und nichtlinearen Gleichungen weiterentwickelt. Der Ansatz basiert auf dem im Folgenden dargestellten Branch-and-Bound-Verfahren.

Branch-and-Bound-Verfahren

Eine Methode zur deterministischen globalen Optimierung ist die Suche nach dem Branch-and-Bound-Prinzip, das im Folgenden für die Minimierung definiert wird:

Definition 3.26 *Bei einer Minimierung nach dem Branch-and-Bound-Prinzip wird der Suchraum rekursiv in Teilräume, so genannte Äste (branches), zerlegt, so dass sich ein Suchbaum ergibt. Dem Wurzelknoten jedes Teilbaums wird eine untere Schranke (bound) für den Wert der Zielfunktion zugeordnet. Die Suche verzweigt ausgehend von der Wurzel in die Teilbäume mit der kleinsten Schranke. Da sich die Güte der Schranken mit der Verkleinerung der betrachteten Räume verbessert, kann ein nachträgliches Ändern von Verzweigungsentscheidungen (backtracking) erforderlich werden. Die Suche endet bei Erreichen eines Blattes des Suchbaums, wenn der dort exakt berechenbare Wert der Zielfunktion kleiner*

3 Dimensionierung mit formalen Methoden

oder gleich dem Wert der Schranken aller anderen Teilbäume ist. Anderenfalls muss ein Backtracking durchgeführt werden.

Bei der Suche nach dem Branch-and-Bound-Prinzip entfällt bei ausreichender Güte der berechneten Schranken die Durchsuchung von Teilen des Suchraumes. Das globale Optimum wird damit gefunden, ohne dass eine vollständige Suche durchgeführt werden muss. Wenn die berechneten Schranken garantiert korrekt sind, so ist auch die Optimalität der Lösung garantiert.

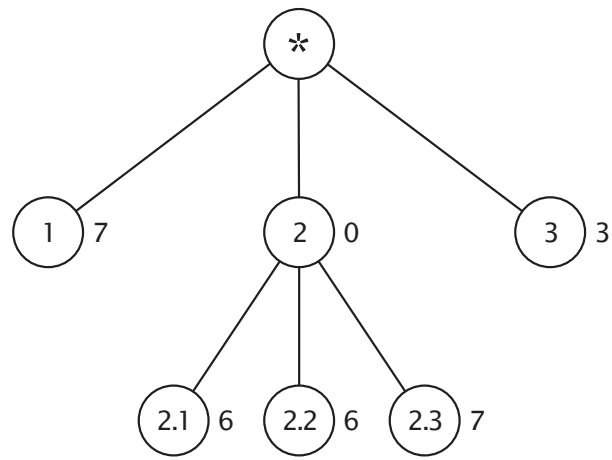
Beispiel 3.6 *Abbildung 3.4 gibt ein Beispiel für das Vorgehen bei der Minimierung mit dem Branch-and-Bound-Verfahren. Der Suchraum wird zunächst in drei Teile zerlegt, so dass sich unterhalb des Wurzelknotens \star die drei Knoten 1, 2 und 3 ergeben. Für die zugehörigen Teilsuchräume werden die unteren Schranken der Zielfunktion zu 7, 0 und 3 bestimmt und den Knoten zugeordnet (s. Abbildung 3.4(a)). Da Knoten 2 die niedrigste untere Schranke aufweist, wird die Suche in seinem Teilsuchraum fortgesetzt. Bei der rekursiven Zerlegung dieses Teilsuchraumes werden für die Teile mit den Knoten 2.1, 2.2 und 2.3 untere Schranken von 6, 6 und 7 berechnet (vgl. Abbildung 3.4(a)). Die untere Schranke von Knoten 2 ist damit von 0 auf 6 zu erhöhen. Dieser und die nachfolgenden Schritte sind in Abbildung 3.4(b) gezeigt.*

Infolge der Erhöhung der unteren Schranke für Knoten 2 ist die im Wurzelknoten getroffene Entscheidung, das Minimum im Teilsuchraum von Knoten 2 zu suchen, rückgängig zu machen (backtracking). Statt dessen ist jetzt in den Teilsuchraum unter Knoten 3 zu verzweigen, da dieser die niedrigste untere Schranke aufweist. Eine weitere Aufteilung in 3.1, 3.2 und 3.3 mit unteren Schranken 5, 4 und 7 führt zur Fortsetzung der Suche in 3.2. Hier ergeben sich die Blätter 3.2.1, 3.2.2 und 3.2.3, d. h. der Suchraum kann nicht weiter aufgeteilt werden. Die Zielfunktion lässt sich in den Blättern exakt zu 5, 6 und 6 auswerten. Der Vergleich mit den unteren Schranken der Knoten 1, 2, 3.1 und 3.3 zeigt, dass damit das globale Minimum gefunden ist.

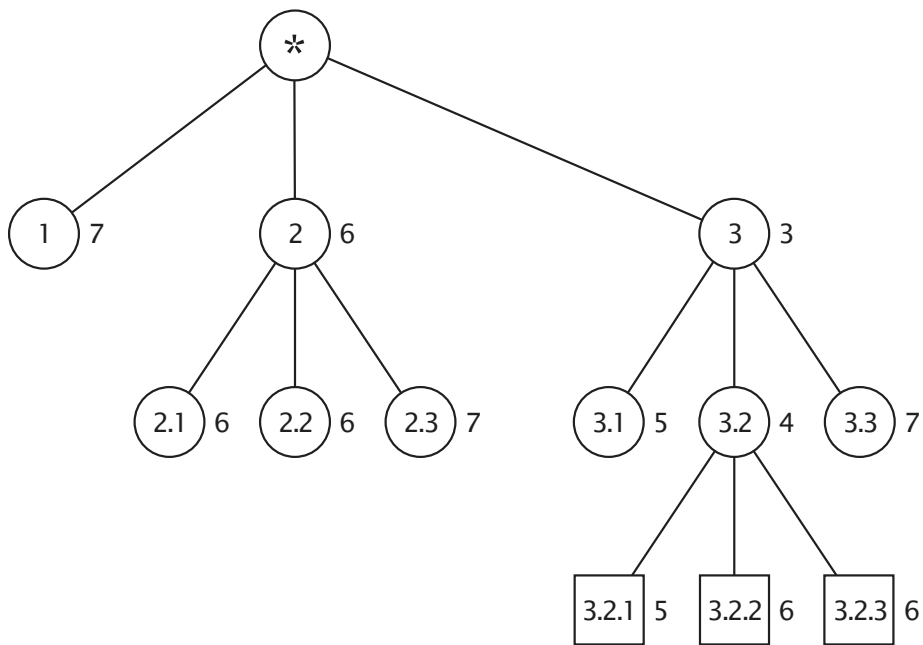
In diesem Beispiel hätte eine vollständige Suche des Minimums bei rekursivem Aufteilen des Suchraumes in drei Teile und Erreichen der Blätter mit der dritten Aufteilung die Berechnung der Zielfunktion in 27 Blättern erfordert. Durch das Branch-and-Bound-Verfahren mussten dagegen nur für 10 Knoten Schranken oder Zielfunktionswerte bestimmt werden.

Das Branch-and-Bound-Verfahren funktioniert in der beschriebenen Form jedoch nur, wenn eine Möglichkeit besteht, Schranken für Teile

3.3 Dimensionierungsalgorithmus mit formalen Methoden



(a)



(b)

Abbildung 3.4: Minimierung mit dem Branch-and-Bound-Verfahren

3 Dimensionierung mit formalen Methoden

des Suchraumes zu berechnen, und der Algorithmus nach endlich vielen Schritten zu einem Blatt gelangt, wo der Suchraum unteilbar ist. Letzteres ist nur bei kombinatorischen Optimierungsproblemen der Fall.

Branch-and-Bound-Verfahren mit Gebietsarithmetik

Bei der Anwendung der Branch-and-Bound-Methode auf Probleme mit kontinuierlichen Parametern ergeben sich zwei Probleme: die Berechnung garantierter Schranken und die Formulierung eines Abbruchkriteriums.

Das Branch-and-Bound-Verfahren findet nur dann garantiert das globale Optimum, wenn auch die verwendeten Schranken garantiert sind (vgl. voriger Abschnitt). Für beliebige Funktionen lassen sich keine allgemeingültigen Berechnungsvorschriften für die Ermittlung garantierter Schranken auf der Grundlage herkömmlicher Gleitkomma-Arithmetik angeben (vgl. Abschnitt 3.2). Dieses Problem wird in dieser Arbeit durch Gebietsarithmetik gelöst, indem die damit berechneten, durch Inklusionsisotonie garantierten Einschließungen als Schranken verwendet werden.

Bei einem kontinuierlichen Problem ist es nicht möglich, durch eine endliche Zahl von Aufteilungen zu einem einzelnen Punkt im Suchraum zu gelangen, also zu einem Blatt im Suchbaum. Dies ist jedoch für das in Definition 3.26 aufgeführte Abbruchkriterium nötig, da nur für Blätter exakte Zielfunktionswerte berechnet werden können.

Ein Ansatz, der die direkte Anwendung des Branch-and-Bound-Verfahrens ermöglicht, ist die Überführung des kontinuierlichen in ein kombinatorisches Problem. Dazu wird der Suchraum in Teilsuchräume aufgeteilt, deren Vereinigung den Suchraum ergibt und deren Schnittmenge leer ist. Jedem Teilsuchraum wird die mit Gebietsarithmetik berechnete untere Schranke der Zielfunktion des kontinuierlichen Problems als Zielfunktionswert des kombinatorischen Problems zugeordnet. Dadurch kann das überführte Problem mit dem Branch-and-Bound-Verfahren gelöst werden. Es wird jedoch nicht mehr das Ausgangsproblem gelöst. Deshalb weicht das berechnete Optimum von dem des Ausgangsproblems ab. Diese Abweichung lässt sich durch Verfeinerung der Aufteilung verringern, bis eine obere Schranke der Abweichung unter einer vorgegebenen Grenze liegt. Nach diesem Ansatz wurde ein angepasstes Branch-and-Bound-Verfahren für kontinuierliche Optimierungsprobleme entwickelt, dessen Abbruchkriterium eine untere *und* eine obere Schranke der Zielfunktion verwendet:

3.3 Dimensionierungsalgorithmus mit formalen Methoden

Definition 3.27 Das Branch-and-Bound-Verfahren mit Gebietsarithmetik beendet die Suche nach dem globalen Minimum, wenn in einem Teilsuchraum X^* der relative Abstand ϵ der oberen Schranke f_{oben} der Zielfunktion f von der unteren Schranke f_{unten}^* von f in allen Teilsuchräumen X_i des Suchraumes X_0 maximal ϵ_{Tol} beträgt:

$$\epsilon \leq \epsilon_{Tol} \quad \text{mit} \quad \epsilon = \frac{f_{oben} - f_{unten}^*}{|f_{unten}^*|}. \quad (3.41)$$

Dieses Abbruchkriterium ermöglicht die Folgerung

$$\begin{aligned} & \forall \xi \in X^* : f_{oben} \geq f(\xi) \\ \wedge & \quad \forall i : \forall \zeta \in X_i : f_{unten}^* \leq f(\zeta) \\ \wedge & \quad \frac{f_{oben} - f_{unten}^*}{|f_{unten}^*|} \leq \epsilon_{Tol} \\ \Rightarrow & \quad \forall i : \forall \xi \in X^*, \zeta \in X_i : \frac{f(\xi) - f(\zeta)}{|f(\zeta)|} \leq \epsilon_{Tol}. \end{aligned} \quad (3.42)$$

Aus $\bigcup_i X_i = X_0$ folgt

$$\forall \xi \in X^*, \zeta \in X_0 : \frac{f(\xi) - f(\zeta)}{|f(\zeta)|} \leq \epsilon_{Tol}. \quad (3.43)$$

Damit ist bei Beendigung der Suche ein Gebiet X^* gefunden, für das garantiert ist, dass die Zielfunktionswerte aller enthaltenen Punkte maximal die relative Abweichung ϵ_{Tol} vom globalen Minimum haben. Das globale Minimum lässt sich durch Wahl von ϵ_{Tol} beliebig genau finden.

Der Suchraum X_0 ist im Folgenden als Intervalltupel $[X_0]$ gegeben. Zur Aufteilung in Teilsuchräume wird ein Bisektionsoperator definiert:

Definition 3.28 Für ein Intervalltupel $[X] \in \mathbb{I}_{\mathbb{R}}^{n_x}$ ist der Bisektionsoperator *bisect* definiert als:

$$\begin{aligned} \text{bisect } [X] = \{ & ([X_1], [X_2], \dots, [\underline{x}_i, \xi], \dots, [X_n]), \\ & ([X_1], [X_2], \dots, [\xi, \bar{x}_i], \dots, [X_n]) \} \end{aligned} \quad (3.44)$$

mit $\xi \in [X_i]$ und $1 \leq i \leq n_x$.

Die durch den Operator *bisect* erzeugten Teilräume sind damit ebenfalls Intervalltupel.

Zur Bestimmung der Schranken f_{unten}^* und f_{oben} werden Schrankenfunktionen entsprechend den folgenden Definitionen verwendet.

3 Dimensionierung mit formalen Methoden

Definition 3.29 Für eine Funktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ ist eine untere Schrankenfunktion f_{unten} eine Funktion $f_{\text{unten}} : \mathbb{I}_{\mathbb{R}}^{n_x} \rightarrow \mathbb{R}$ mit der Eigenschaft

$$\forall \xi \in [\mathbf{X}] : f_{\text{unten}}([\mathbf{X}]) \leq f(\xi) . \quad (3.45)$$

Definition 3.30 Für eine Funktion $f : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ ist eine obere Schrankenfunktion f_{oben} eine Funktion $f_{\text{oben}} : \mathbb{I}_{\mathbb{R}}^{n_x} \rightarrow \mathbb{R}$ mit der Eigenschaft

$$\forall \xi \in [\mathbf{X}] : f_{\text{oben}}([\mathbf{X}]) \geq f(\xi) . \quad (3.46)$$

Die Implementierung dieser Schrankenfunktionen kann sowohl in Intervallarithmetik als auch zur Verringerung der Überschätzung in affiner Arithmetik erfolgen. Im letzteren Fall wird das Intervalltupel $[\mathbf{X}]$ mit dem Operator \hat{A}_ε in eine affine Form umgewandelt. Die Berechnung der Zielfunktion f erfordert bei Dimensionierungsproblemen die Durchführung der Analyseabbildung \mathbf{a} zur Berechnung der Eigenschaften (vgl. Abschnitt 3.1.1). Die Auswertung der Schrankenfunktionen ist daher sehr aufwändig und wird in den Abschnitten 3.3.2 und 3.3.3 ausführlich erläutert.

Es ist nicht erforderlich, den gesamten Suchbaum in einer Datenstruktur zu halten, da nur die Blätter des bereits besuchten Teilbaums zur Optimierung benötigt werden. Damit ergibt sich Algorithmus 3.1. Dieser

Algorithmus 3.1 Globale Optimierung ohne Nebenbedingungen

```

P = {[X0]}
repeat
    funten* = min[X] ∈ P funten([X])
    [X*] = argmin[X] ∈ P funten([X])
    P = P \ {[X*]}
    P = P ∪ bisect([X*])
until ∃ [X*] ∈ P :  $\frac{f_{\text{oben}}([\mathbf{X}^*]) - f_{\text{unten}}([\mathbf{X}^*])}{|f_{\text{unten}}([\mathbf{X}^*])|} \leq \epsilon_{\text{Tol}}$ 

```

berechnet für einen Suchraum $[\mathbf{X}_0]$ ein Ergebnis $[\mathbf{X}^*]$, das garantiert nur Punkte enthält, deren Funktionswerte von f maximal ϵ_{Tol} relative Abweichung vom globalen Minimum von f auf $[\mathbf{X}_0]$ haben.

In Algorithmus 3.1 wird eine Menge P zur Speicherung der Blätter des bereits besuchten Teilbaums verwendet. In der praktischen Implementierung werden aus Effizienzgründen sortierte Datenstrukturen (Prioritätsschlange, Heap) eingesetzt und bereits berechnete Funktionswerte gespeichert. Dann werden alle argmin- und min-Operationen mit Laufzeitkomplexität $\mathcal{O}(1)$ und die Einfüge- und Löschoptionen mit $\mathcal{O}(\log n)$ ausgeführt, wenn n die Anzahl der Teilräume ist.

3.3 Dimensionierungsalgorithmus mit formalen Methoden

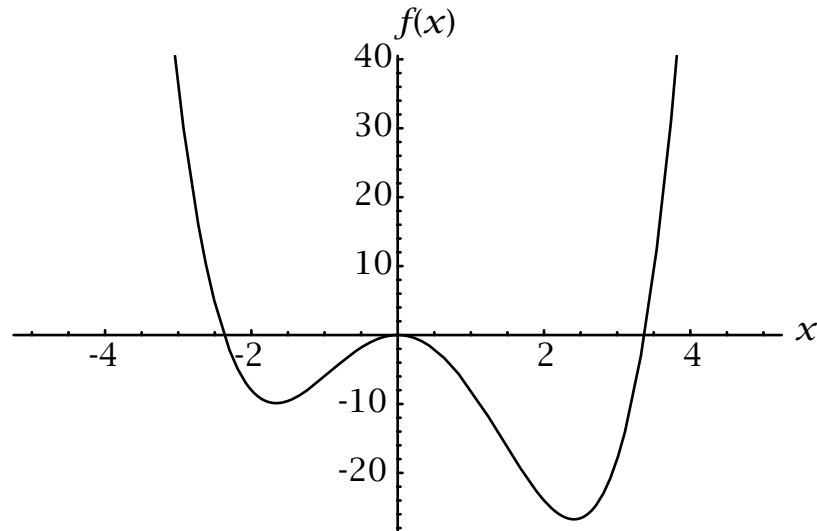


Abbildung 3.5: Funktion mit zwei lokalen Minima

Beispiel 3.7 Die in Abbildung 3.5 dargestellte Funktion $f(x) = x^4 - x^3 - 8x^2$ hat zwei lokale Minima, von denen das bei $(2,41; -26,73)$ gelegene gleichzeitig das globale Minimum ist. Dieses soll mit Hilfe von Algorithmus 3.1 durch Suche im Intervall $[-4; 4]$ mit $\epsilon_{Tol} = 1\%$ bestimmt werden.

Bei Verwendung von affiner Arithmetik zur Bestimmung der Schranken von f erreicht der Algorithmus das Ziel nach 39 Auswertungen von f und liefert das Ergebnisintervall $[X^*] = [2,406; 2,438]$. Wird statt dessen eine natürliche Intervallerweiterung von f eingesetzt, so führt der Algorithmus 297 Auswertungen durch und liefert $[X^*] = [2,406; 2,409]$ ¹ als Ergebnis.

Die größere Anzahl an Iterationen, die in Beispiel 3.7 bei Verwendung der natürlichen Intervallerweiterung erforderlich ist, ist auf die größere Überschätzung der damit berechneten Schranken zurückzuführen. Diese ist erst bei einem sehr schmalen Intervall $[X^*]$ so klein, dass die Abbruchbedingung von Algorithmus 3.1 erfüllt wird. Trotz des schmalen Ergebnisintervalls wird wie bei der Rechnung mit affiner Arithmetik auch hier nur $\epsilon_{Tol} = 1\%$ garantiert.

¹Das Ergebnisintervall muss das Minimum nicht einschließen, da lediglich die Einhaltung der Toleranz ϵ_{Tol} gefordert wird.

Globale Optimierung mit Nebenbedingungen

Im Folgenden wird die Lösung des in Definition 3.4 gegebenen Dimensionierungsproblems betrachtet. Das Vorgehen ist, wie in Abschnitt 3.1.2 beschrieben, direkt auf das Entwurfszentrierungsproblem übertragbar.

Zur Lösung des Dimensionierungsproblems ist die globale Optimierung der Zielfunktion f_u in einem durch die Nebenbedingungen 3.1 definierten Akzeptanzgebiet A_{P_E} durchzuführen. Die vollständige Bestimmung von A_{P_E} ist sehr aufwändig und zur Lösung des Optimierungsproblems auch nicht erforderlich. In dem in dieser Arbeit entwickelten Verfahren wird statt dessen zunächst davon ausgegangen, dass der gesamte Suchraum zum Akzeptanzgebiet gehört. Diese Annahme wird nur bei der Untersuchung von Lösungskandidaten bewiesen bzw. widerlegt. Dazu werden zwei Prädikate: falsifiziert und verifiziert für als Intervalltupel beschriebene Teilräume von P_E eingeführt.

Definition 3.31 Das Prädikat *falsifiziert* ist eine boolesche Funktion *falsifiziert* : $\mathbb{P}_E \rightarrow \{\text{falsch, wahr}\}$, für die die Folgerung

$$\text{falsifiziert}([\mathbf{P}_E]) \Rightarrow [\mathbf{P}_E] \notin A_{P_E} \quad (3.47)$$

gilt.

Definition 3.32 Das Prädikat *verifiziert* ist eine boolesche Funktion *verifiziert* : $\mathbb{P}_E \rightarrow \{\text{falsch, wahr}\}$ mit

$$\text{verifiziert}([\mathbf{P}_E]) \Rightarrow [\mathbf{P}_E] \in A_{P_E} . \quad (3.48)$$

Aufgrund dieser Definitionen gilt nicht $\text{falsifiziert} \Leftrightarrow \neg \text{verifiziert}$, sondern nur, dass *falsifiziert* und *verifiziert* sich gegenseitig ausschließen:

$$\text{falsifiziert} \Rightarrow \neg \text{verifiziert} \quad \wedge \quad \text{verifiziert} \Rightarrow \neg \text{falsifiziert} \quad (3.49)$$

Ein Teilgebiet $[\mathbf{P}_E]$ ist in Bezug auf die Nebenbedingungen der Spezifikation *verifiziert* oder *falsifiziert* (vgl. Definitionen 2.13 und 2.14), wenn die Prädikate *verifiziert* bzw. *falsifiziert* für $[\mathbf{P}_E]$ wahr sind.

Algorithmus 3.1 wird mit Hilfe dieser Prädikate zur Behandlung des Optimierungsproblems mit Nebenbedingungen erweitert. Dazu wird die Menge der Teilsuchräume P durch eine Menge P_{NF} ersetzt, die alle bis zum jeweiligen Punkt im Verlauf der Lösungssuche nicht falsifizierten Teilräume von P_E enthält. Es gilt daher

$$\bigcup_{[\mathbf{P}_E] \in P_{NF}} \supseteq A_{P_E} . \quad (3.50)$$

3.3 Dimensionierungsalgorithmus mit formalen Methoden

P_{NF} beschreibt eine Einschließung von A_{P_E} . Nach jedem Bisektionsschritt werden diejenigen Teilräume, für die falsifiziert = wahr gilt, aus P_{NF} entfernt, so dass P_{NF} gegen A_{P_E} konvergiert. Ferner wird ein Rückschritt (backtracking) durchgeführt, wenn der Suchbaum durch diese Beschneidung keine Verzweigung vom aktuellen Knoten mehr zulässt. Sobald die Toleranzbedingung 3.41 für mindestens einen Teilraum in P_{NF} erfüllt ist, wird dort eine Breitensuche nach einem Teilraum, für den verifiziert = wahr gilt, durchgeführt. Wird ein solcher gefunden, so ist das Optimierungsproblem mit Nebenbedingungen gelöst.

Dieses Vorgehen ist in Algorithmus 3.2 formalisiert. Der Algorithmus

Algorithmus 3.2 Globale Optimierung mit Nebenbedingungen

```

 $P_{NF} = \{ [\mathbf{P}_E^0] \}$ 
repeat
   $f_{u,unten}^* = \min_{[\mathbf{P}_E] \in P_{NF}} f_{u,unten}([\mathbf{P}_E])$ 
   $P'_{NF} = P_{NF}$ 
  for all  $[\mathbf{P}_E] \in P_{NF}$  do
    if  $\frac{f_{u,oben}([\mathbf{P}_E^*]) - f_{u,unten}^*}{|f_{u,unten}^*|} \leq \epsilon_{Tol} \vee f_{u,unten}([\mathbf{P}_E]) = f_{u,unten}^*$  then
       $P'_{NF} = P'_{NF} \setminus \{ [\mathbf{P}_E] \}$ 
      if  $\neg$  falsifiziert( $[\mathbf{P}_E]$ ) then
         $P'_{NF} = P'_{NF} \cup \text{bisect } [\mathbf{P}_E]$ 
      end if
    end if
  end for
   $P_{NF} = P'_{NF}$ 
until  $\exists [\mathbf{P}_E^*] \in P_{NF} : \text{verifiziert } ([\mathbf{P}_E^*]) \wedge \frac{f_{u,oben}([\mathbf{P}_E^*]) - f_{u,unten}^*}{|f_{u,unten}^*|} \leq \epsilon_{Tol}$ 

```

geht von einem als Intervalltupel $[\mathbf{P}_E^0]$ gegebenen Suchraum aus und berechnet ein Ergebnis $[\mathbf{P}_E^*]$, das nur Punkte enthält, die die Nebenbedingungen der Spezifikation erfüllen und deren Funktionswerte von f_u maximal ϵ_{Tol} relative Abweichung vom globalen Minimum von f_u auf A_{P_E} haben.

Die Anmerkungen zu Algorithmus 3.1 bezüglich der Verwendung affiner Arithmetik und sortierter Datenstrukturen gelten entsprechend auch für Algorithmus 3.2.

Beispiel 3.8 Ein möglicher Ablauf von Algorithmus 3.2 ist in Abbildung 3.6 dargestellt. Die zu minimierende Zielfunktion ist durch

$$f_u(\mathbf{p}_E) = p_{E1} + p_{E2} \quad (3.51)$$

3 Dimensionierung mit formalen Methoden

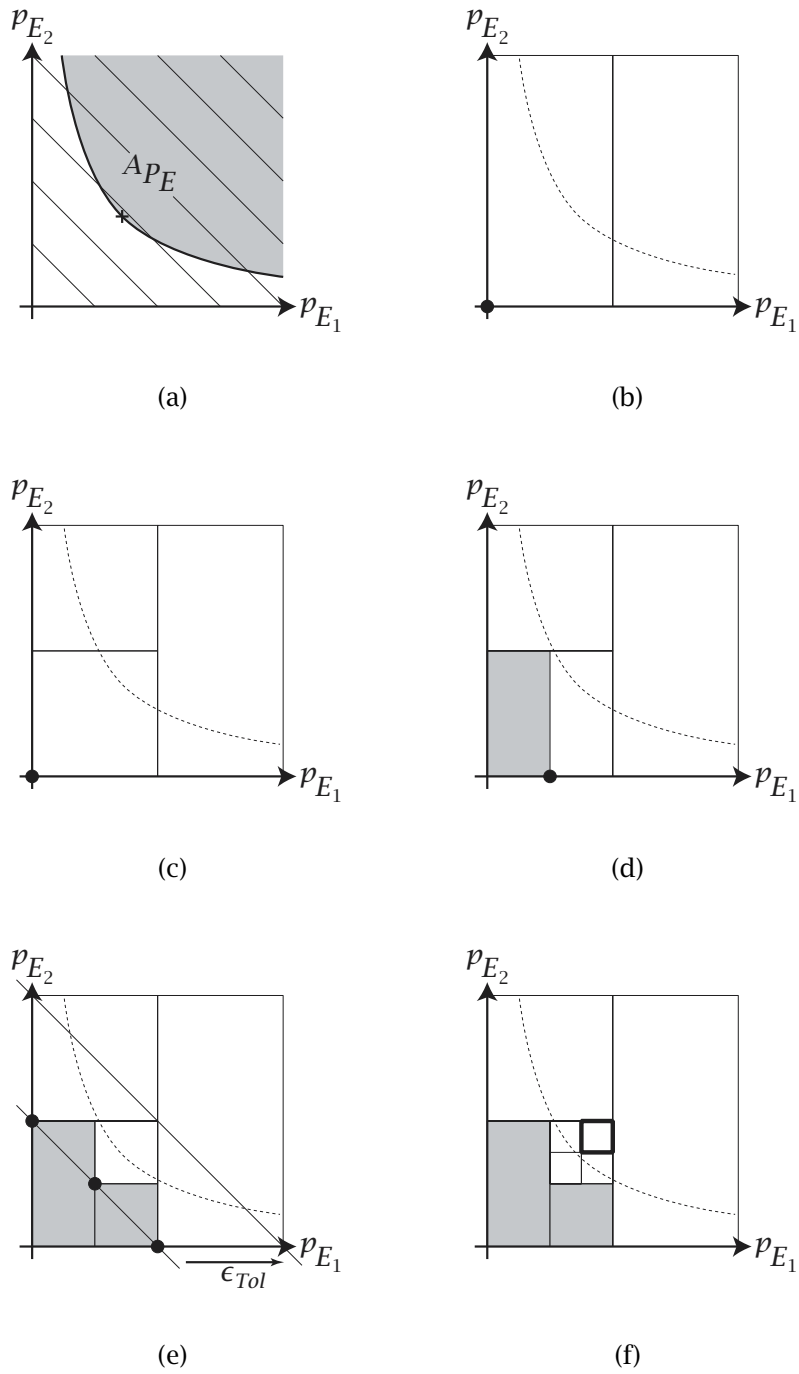


Abbildung 3.6: Minimierung mit dem gebietsarithmetischen Branch-and-Bound-Verfahren

3.3 Dimensionierungsalgorithmus mit formalen Methoden

gegeben. Zur graphischen Darstellung sind in Abbildung 3.6(a) Höhenlinien von f eingetragen. Ferner ist eine Nebenbedingung

$$p_{E_1} p_{E_2} \leq a \quad \text{mit: } a > 0 \quad (3.52)$$

zu berücksichtigen. Sie definiert das Akzeptanzgebiet A_{P_E} . Seine durch $p_{E_1} p_{E_2} = a$ gegebene Begrenzung ist in Abbildung 3.6(a) ebenfalls dargestellt. Das auf der Begrenzung liegende Kreuz kennzeichnet das globale Optimum des Problems. Für das Abbruchkriterium wird ein vergleichsweise großes $\epsilon_{Tol} = 1$ gewählt, um in diesem Beispiel zu gut darstellbaren Ergebnissen zu gelangen.

Die Bisektionen der Unterräume von P_E werden in diesem Beispiel durch Halbieren durchgeführt. Dabei werden abwechselnd die Komponenten p_{E_1} und p_{E_2} halbiert. Als Suchraum $[P_E^0]$ wird der gesamte in Abbildung 3.6 dargestellte Ausschnitt von P_E verwendet.

Abbildung 3.6(b) zeigt das Ergebnis der ersten Bisektion an p_{E_1} . Der im Ursprung eingetragene Punkt markiert diejenige Ecke der Teilräume in P_{NF} , deren Zielfunktionswert gleich $f_{u,unten}^*$ ist. Im zugehörigen linken Teilraum wird die Lösungssuche fortgesetzt, so dass die folgende Bisektion an p_{E_2} zu Abbildung 3.6(c) führt.

Da alle Elemente der bisher erzeugten Teilraummenge P_{NF} den Rand des Akzeptanzgebietes A_{P_E} schneiden, ergibt das falsifiziert-Prädikat jeweils falsch, und die Teilräume verbleiben in P_{NF} . Ebenso befindet sich $f_{u,unten}^*$ weiterhin an der linken, unteren Ecke des linken, unteren Teilraumes. In diesem wird die Suche daher fortgesetzt.

Eine weitere Bisektion dieses Teilraumes an p_{E_1} ergibt einen linken Teilraum, für den das falsifiziert-Prädikat wahr ist, und einen rechten für den es falsch ergibt. Der falsifizierte linke Teilraum ist in Abbildung 3.6(d) schattiert dargestellt. Er wird aus P_{NF} entfernt. Der Markierungspunkt zeigt nun auf die linke untere Ecke des rechten Teilraumes, wo der Zielfunktionswert $f_{u,unten}^*$ vorliegt. Durch Bisektion dieses Teilraumes an p_{E_2} kann wiederum die untere Hälfte aus P_{NF} entfernt werden (vgl. Abbildung 3.6(e)). Das neue $f_{u,unten}^*$ liegt jetzt gleichzeitig an den drei markierten Ecken vor.

Die Suche wird zuerst im kleinsten der drei Teilräume fortgesetzt, da eine baldige Erfüllung des Abbruchkriteriums oder des falsifiziert-Prädikats dort am wahrscheinlichsten ist (in Algorithmus 3.2 nicht dargestellt). In diesem Teilraum ist die Bedingung

$$\frac{f_{u,oben}([P_E^*]) - f_{u,unten}^*}{|f_{u,unten}^*|} \leq \epsilon_{Tol}, \quad (3.53)$$

3 Dimensionierung mit formalen Methoden

die zur Breitensuche in $[\mathbf{P}_E]$ führt, erfüllt. Zur Verdeutlichung sind die zwei Höhenlinien von f , die sich aus $f_u(\mathbf{p}_E) = f_{u,unten}^*$ und $f_u(\mathbf{p}_E) = (1 + \epsilon_{Tol}) f_{u,unten}^*$ ergeben, eingezeichnet. Die Einhaltung von Bedingung 3.53 ist damit sichtbar.

Die Breitensuche liefert nach zwei Bisektionsschritten das Ergebnis in Abbildung 3.6(f). Das verifiziert-Prädikat ist für den hervorgehobenen Teilraum wahr, so dass der Algorithmus terminiert und diesen als Ergebnis $[\mathbf{P}_E^*]$ zurückliefert.

3.3.2 Worst-Case-Analyse

Wie in Abschnitt 3.1.1 beschrieben, beinhaltet sowohl die Auswertung der Nebenbedingungen 3.1 als auch die Berechnung der Zielfunktion f_u bei praktischen Dimensionierungsproblemen die Durchführung der Worst-Case-Analyseabbildung \mathbf{a}_{WC} , um Schranken für die Schaltungseigenschaftsgrößen \mathbf{e} zu berechnen. Die Durchführung der Worst-Case-Analyse mit Hilfe formaler Methoden wird nachfolgend beschrieben.

Die in den Definitionen 3.31 und 3.32 gegebenen Prädikate falsifiziert und verifiziert müssen garantiert korrekte Aussagen über die Zugehörigkeit eines Teilraumes von P_E zum Akzeptanzgebiet A_{P_E} für das gesamte in der Spezifikation gegebene Betriebs- und Prozessparametergebiet $P_{BS,Spez}$ ermöglichen. Nur so wird auch die Korrektheit der mit Algorithmus 3.2 berechneten Ergebnisse gewährleistet. Ebenso müssen die Schranken der Zielfunktion f_u , die mit den Schrankenfunktionen $f_{u,unten}$ und $f_{u,oben}$ berechnet werden, garantiert in $P_{BS,Spez}$ richtig sein, damit globale Konvergenz gesichert ist. Beides erfordert eine Worst-Case-Analyse in $P_{BS,Spez}$.

Berechnung der Prädikate

Die Korrektheit der Prädikate falsifiziert und verifiziert lässt sich durch Verwendung von Gebietsarithmetik unter Ausnutzung der Inklusionsisotonie (vgl. Definition 3.12 und Relation 3.36) sicherstellen. Dazu wird eine Intervallerweiterung $[\mathbf{C}]$ bzw. eine affine Erweiterung $\hat{\mathbf{c}}_\epsilon$ des Nebenbedingungsfunktionstupels \mathbf{c} (vgl. Bedingung 3.1) verwendet. Die nachfolgende Darstellung basiert auf der Intervallerweiterung. Die affine Erweiterung kann mit Hilfe der Umwandlungsoperatoren $[I]$ und $\hat{\mathcal{A}}_\epsilon$ zur Verbesserung des Ergebnisses gleichermaßen genutzt werden.

Mit Hilfe der Intervallerweiterung $[\mathbf{C}]$ werden garantierte Schranken von \mathbf{c} auf $P_{BS,Spez}$ in als Intervalltupeln gegebenen Teilräumen von P_E berechnet. Da $[\mathbf{C}]$ intervallwertige Argumente erfordert, muss statt $P_{BS,Spez}$

3.3 Dimensionierungsalgorithmus mit formalen Methoden

eine Intervalleinschließung $[\mathbf{P}_{BS,Spez}] \supseteq P_{BS,Spez}$ verwendet werden. Auf diese Weise bleibt die Inklusionsisotonie erhalten.

Zunächst werden Hilfsprädikate *falsifiziert'* und *verifiziert'* eingeführt.

Definition 3.33 Für eine gegebene Intervallerweiterung $[C]$ von c ist das Prädikat *falsifiziert'* eine boolesche Funktion $\text{falsifiziert}' : \mathbb{I}_{P_E} \times P_{BS} \rightarrow \{\text{falsch}, \text{wahr}\}$, die durch

$$\text{falsifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS}]) \Leftrightarrow \max([C([\mathbf{P}_E], [\mathbf{P}_{BS}])]) < 0 \quad (3.54)$$

definiert ist.

Definition 3.34 Für eine gegebene Intervallerweiterung $[C]$ von c ist das Prädikat *verifiziert'* eine boolesche Funktion $\text{verifiziert}' : \mathbb{I}_{P_E} \times P_{BS} \rightarrow \{\text{falsch}, \text{wahr}\}$ mit

$$\text{verifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS}]) \Leftrightarrow \min([C([\mathbf{P}_E], [\mathbf{P}_{BS}])]) \geq 0. \quad (3.55)$$

Die Hilfsprädikate *falsifiziert'* und *verifiziert'* erfüllen bei Einsetzen von $[\mathbf{P}_{BS,Spez}]$ für $[\mathbf{P}_{BS}]$ die in den Definitionen 3.31 und 3.32 für die Prädikate *falsifiziert* und *verifiziert* geforderten Eigenschaften. Dies lässt sich mit Hilfe der Inklusionsisotonie zeigen:

$$\begin{aligned} & \text{falsifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS}]) \\ \Leftrightarrow & \max([C([\mathbf{P}_E], [\mathbf{P}_{BS}])]) < 0 \\ \Rightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E], \mathbf{p}_{BS} \in [\mathbf{P}_{BS}] : c(\mathbf{p}_E, \mathbf{p}_{BS}) < 0 \\ \Rightarrow & [\mathbf{P}_E] \notin A_{P_E} \end{aligned} \quad (3.56)$$

$$\begin{aligned} & \text{verifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS}]) \\ \Leftrightarrow & \min([C([\mathbf{P}_E], [\mathbf{P}_{BS}])]) \geq 0 \\ \Rightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E], \mathbf{p}_{BS} \in [\mathbf{P}_{BS}] : c(\mathbf{p}_E, \mathbf{p}_{BS}) \geq 0 \\ \Leftrightarrow & [\mathbf{P}_E] \in A_{P_E} \end{aligned} \quad (3.57)$$

Die Umformungen 3.56 und 3.57 zeigen einen wesentlichen Unterschied zwischen den Prädikaten *falsifiziert'* und *verifiziert'* auf: Die Folgerung, dass $[\mathbf{P}_E] \notin A_{P_E}$, führt über zwei Implikationen, während $[\mathbf{P}_E] \in A_{P_E}$ in Umformung 3.56 nur eine Implikation benötigt. Die Verwendung einer zusätzlichen Implikation anstelle einer Äquivalenz in Umformung 3.56 ist erforderlich, da

$$\forall \mathbf{p}_E \in [\mathbf{P}_E] : \underline{\exists \mathbf{p}_{BS} \in [\mathbf{P}_{BS}] : c(\mathbf{p}_E, \mathbf{p}_{BS}) < 0} \Leftrightarrow [\mathbf{P}_E] \notin A_{P_E} \quad (3.58)$$

3 Dimensionierung mit formalen Methoden

gilt, hier aber nur

$$\forall \mathbf{p}_E \in [\mathbf{P}_E], \mathbf{p}_{BS} \in [\mathbf{P}_{BS}] : c(\mathbf{p}_E, \mathbf{p}_{BS}) < 0 \quad \Rightarrow \quad [\mathbf{P}_E] \notin A_{P_E} \quad (3.59)$$

verwendet werden kann, da sich aus der Inklusionsisotonie nicht die linke Seite der Äquivalenz 3.58, sondern die linke Seite der Implikation 3.59 ergibt.

Die anderen beiden in den Umformungen 3.56 und 3.57 verwendeten Implikationen ergeben sich aus der Inklusionsisotonie (vgl. Definition 3.12) und sind auf den Überschätzungseffekt zurückzuführen. Wegen der Verwendung der Implikation 3.59 und der Inklusionsisotonie sind die Bedingungen, die sich durch Anwendung der Definitionen 3.31 und 3.32 in Verbindung mit den Prädikaten *falsifiziert'* und *verifiziert'* für die Zugehörigkeit bzw. Nichtzugehörigkeit zum Akzeptanzgebiet verfügbar sind, lediglich hinreichend, aber nicht notwendig. Sie lassen bei Intervalltupeln $[\mathbf{P}_{BS,Spez}]$ großer Breite oftmals keine Aussage zu, da der Fall $\neg \text{falsifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS}]) \wedge \neg \text{verifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS}])$ häufig eintritt. Dies würde dazu führen, dass Algorithmus 3.2 keine Lösung findet, obwohl eine existiert. Deshalb ist eine Aufteilung von $[\mathbf{P}_{BS,Spez}]$ in n Teilintervalle $[\mathbf{P}_{BS,Spez,i}]$ durchzuführen und die Bedingungen sind auf die schmalen Teilintervalle anzuwenden. Damit lassen sich die Prädikate *falsifiziert* und *verifiziert* wie folgt berechnen:

$$\text{falsifiziert}([\mathbf{P}_E]) = \bigvee_{i=1}^n \text{falsifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS,Spez,i}]) \quad (3.60)$$

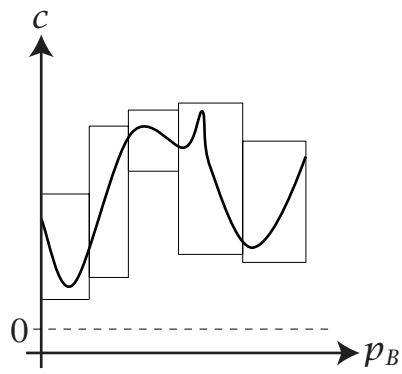
$$\text{verifiziert}([\mathbf{P}_E]) = \bigwedge_{i=1}^n \text{verifiziert}'([\mathbf{P}_E], [\mathbf{P}_{BS,Spez,i}]) \quad (3.61)$$

Durch die Intervallaufteilung wird die Überschätzung reduziert (vgl. Abschnitte 3.2.1 und 3.2.2). Gleichzeitig wird durch die Disjunktion in Gleichung 3.60 der \exists -Quantor, der in der Äquivalenz 3.58 verwendet werden muss, auf der Teilintervallebene nachgebildet. Der Fall $\text{falsifiziert}([\mathbf{P}_E]) \vee \text{verifiziert}([\mathbf{P}_E])$ kann damit für hinreichend große n und hinreichend schmale Intervalltupel $[\mathbf{P}_E]$ immer erreicht werden. Die Aufteilung von $[\mathbf{P}_E]$ findet dabei in Algorithmus 3.2 als Teil der Lösungssuche statt.

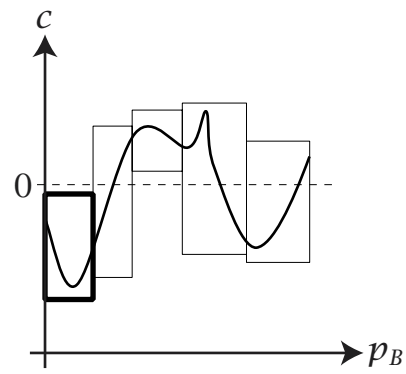
Die Auswertung der Gleichungen 3.60 und 3.61 lässt sich in der Praxis durch Ausnutzung der Dominanz des Wertes wahr in der Disjunktion bzw. des Wertes falsch in der Konjunktion beschleunigen.

Beispiel 3.9 *Abbildung 3.7 zeigt die Anwendungen der Prädikate falsifiziert und verifiziert. In den Unterabbildungen ist jeweils der Ver-*

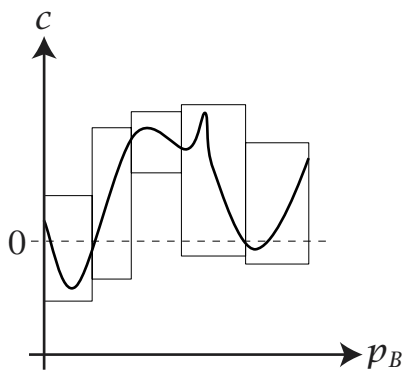
3.3 Dimensionierungsalgorithmus mit formalen Methoden



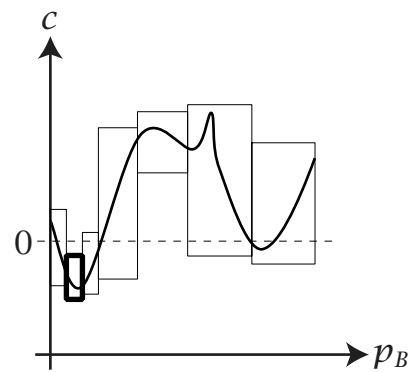
(a) verifiziert = wahr



(b) falsifiziert = wahr



(c) Aufteilung erforderlich



(d) nach Aufteilung

Abbildung 3.7: Anwendung der Prädikate falsifiziert und verifiziert

3 Dimensionierung mit formalen Methoden

lauf einer Nebenbedingungsfunktion c in Abhängigkeit eines Betriebsparameters $p_B \in [P_{B,Spez}]$ dargestellt. Die Nebenbedingungsfunktionen unterscheiden sich dabei durch einen konstanten Anteil. Die zugehörigen Nebenbedingungen lauten jeweils:

$$\forall p_B \in [P_{B,Spez}] : c(p_B) \geq 0 . \quad (3.62)$$

Das Intervall $[P_{B,Spez}]$ wird in Teilintervalle aufgeteilt, für die mit Hilfe einer Intervallerweiterung $[C]$ Intervalleinschließungen der Werte von c bestimmt werden. Daraus ergibt sich eine abschnittsweise Einschließung des Verlaufes von c , die durch die Rechtecke in Abbildung 3.7 dargestellt sind.

Die Anwendung des verifiziert-Prädikats auf den in Abbildung 3.7(a) dargestellten Fall ergibt den Wert wahr, da alle Teile der Einschließung oberhalb der Nulllinie liegen. Damit ist die Nebenbedingung garantiert erfüllt.

Abbildung 3.7(b) zeigt einen Fall, in dem das falsifiziert-Prädikat den Wert wahr ergibt. Dafür ist wegen der Dominanz des Wertes wahr in der Disjunktion in Gleichung 3.60 ausreichend, dass der hervorgehobene, linke Teil der Einschließung unterhalb der Nulllinie liegt.

Ein dritter Fall ist in Abbildung 3.7(c) dargestellt. Hier ergeben die Prädikate falsifiziert und verifiziert auf Grundlage der gezeigten Einschließung beide den Wert falsch, da diese weder gänzlich über noch mit einem ihrer Teile unter der Nulllinie liegt. Dem Verlauf von c lässt sich entnehmen, dass die Nebenbedingung nicht erfüllt ist. Um falsifiziert = wahr zu erhalten, ist eine feinere Aufteilung von $[P_{B,Spez}]$ erforderlich. Dies ist in Abbildung 3.7(d) gezeigt. Hier liegt der hervorgehobene, unterste Teil unterhalb der Nulllinie, so dass falsifiziert = wahr ist.

Berechnung der Schrankenfunktionen

Zur Berechnung garantiert korrekter Worst-Case-Schranken der Zielfunktion f_u auf $P_{BS,Spez}$ wird ebenfalls die Inklusionsisotonie der Gebietsarithmetik herangezogen. Die nachfolgende, auf der Intervallerweiterung $[F_u]$ basierende Darstellung lässt sich wieder durch Verwendung der Umwandlungsoperatoren \mathcal{I} und $\hat{\mathcal{A}}_\epsilon$ auch auf affine Arithmetik übertragen. Anstelle von $P_{BS,Spez}$ ist eine Intervalleinschließung $[P_{BS,Spez}] \supseteq P_{BS,Spez}$ zu verwenden, da $[F_u]$ intervallwertige Argumente erfordert.

Eine Möglichkeit, Schranken von f_u zu berechnen, ist bei intervall-

3.3 Dimensionierungsalgorithmus mit formalen Methoden

arithmetischer Auswertung von Gleichung 3.5 durch

$$\tilde{f}_{u,unten}([\mathbf{P}_E]) = \sum_{i=1}^{n_f} \min (g_i[F_i([\mathbf{P}_E], [\mathbf{P}_{BS,Spez}])]), \quad (3.63)$$

$$\tilde{f}_{u,oben}([\mathbf{P}_E]) = \sum_{i=1}^{n_f} \max (g_i[F_i([\mathbf{P}_E], [\mathbf{P}_{BS,Spez}])]), \quad (3.64)$$

gegeben. Ausgehend von der Inklusionsisotonie lässt sich die Korrektheit dieser Schranken im Sinne der Definitionen 3.29 und 3.30 zeigen. Dabei gilt für die untere Schranke

$$\begin{aligned} & \tilde{f}_{u,unten}([\mathbf{P}_E]) = \min ([F_u([\mathbf{P}_E]]) \\ \Leftrightarrow & \tilde{f}_{u,unten}([\mathbf{P}_E]) = \sum_{i=1}^{n_f} \min (g_i[F_i([\mathbf{P}_E], [\mathbf{P}_{BS,Spez}])] \\ \Rightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E], \mathbf{p}_{BS} \in [\mathbf{P}_{BS}] : \tilde{f}_{u,unten}([\mathbf{P}_E]) \leq \sum_{i=1}^{n_f} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \\ \Leftrightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E] : \tilde{f}_{u,unten}([\mathbf{P}_E]) \leq \sum_{i=1}^{n_f} \min_{\mathbf{p}_{BS} \in [\mathbf{P}_{BS,Spez}]} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \\ \Rightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E] : \tilde{f}_{u,unten}([\mathbf{P}_E]) \leq \sum_{i=1}^{n_f} \max_{\mathbf{p}_{BS} \in [\mathbf{P}_{BS,Spez}]} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \\ \Leftrightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E] : \tilde{f}_{u,unten}([\mathbf{P}_E]) \leq f_u(\mathbf{p}_E). \end{aligned} \quad (3.65)$$

Für die obere Schranke ergibt sich

$$\begin{aligned} & \tilde{f}_{u,oben}([\mathbf{P}_E]) = \max ([F_u([\mathbf{P}_E]]) \\ \Leftrightarrow & \tilde{f}_{u,oben}([\mathbf{P}_E]) = \sum_{i=1}^{n_f} \max (g_i[F_i([\mathbf{P}_E], [\mathbf{P}_{BS,Spez}])] \\ \Rightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E], \mathbf{p}_{BS} \in [\mathbf{P}_{BS}] : \tilde{f}_{u,oben}([\mathbf{P}_E]) \geq \sum_{i=1}^{n_f} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \\ \Leftrightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E] : \tilde{f}_{u,oben}([\mathbf{P}_E]) \geq \sum_{i=1}^{n_f} \max_{\mathbf{p}_{BS} \in [\mathbf{P}_{BS,Spez}]} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \\ \Leftrightarrow & \forall \mathbf{p}_E \in [\mathbf{P}_E] : \tilde{f}_{u,oben}([\mathbf{P}_E]) \geq f_u(\mathbf{p}_E). \end{aligned} \quad (3.66)$$

Die Umformung 3.65 enthält im Vergleich zu Umformung 3.66 eine Im-

3 Dimensionierung mit formalen Methoden

plikation mehr. Diese muss verwendet werden, da die Inklusionsisotonie

$$\tilde{f}_{u,unten}([\mathbf{P}_E]) \leq \sum_{i=1}^{n_f} \min_{\mathbf{p}_{BS} \in [\mathbf{P}_{BS,Spez}]} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \quad (3.67)$$

liefert, für Äquivalenz aber

$$\tilde{f}_{u,unten}([\mathbf{P}_E]) \leq \sum_{i=1}^{n_f} \max_{\mathbf{p}_{BS} \in [\mathbf{P}_{BS,Spez}]} g_i f_i(\mathbf{p}_E, \mathbf{p}_{BS}) \quad (3.68)$$

benötigt wird. Die Relationen 3.67 und 3.68 sind jedoch nicht äquivalent, da aus Relation 3.68 nicht Relation 3.67 folgt.

Die anderen zwei in den Umformungen 3.65 und 3.66 verwendeten Implikationen ergeben sich aus der Inklusionsisotonie und sind auf den Überschätzungseffekt zurückzuführen. Die Bedingungen 3.65 und 3.66 für die Korrektheit der Schranken $\tilde{f}_{u,unten}([\mathbf{P}_E])$ und $\tilde{f}_{u,oben}([\mathbf{P}_E])$ sind wegen der zusätzlichen Implikation in Umformung 3.65 und der Inklusionsisotonie lediglich hinreichend, aber nicht notwendig. Das kann dazu führen, dass bei breiten Intervalltupeln $[\mathbf{P}_{BS,Spez}]$ die berechneten Schranken so stark überschätzt werden, dass das Abbruchkriterium von Algorithmus 3.2 nie erfüllt werden kann. Wie bei der Berechnung der Prädikate muss daher eine Aufteilung von $[\mathbf{P}_{BS,Spez}]$ in n Teilintervalle $[\mathbf{P}_{BS,Spez,j}]$ durchgeführt werden. Die Schranken sind damit wie folgt zu berechnen:

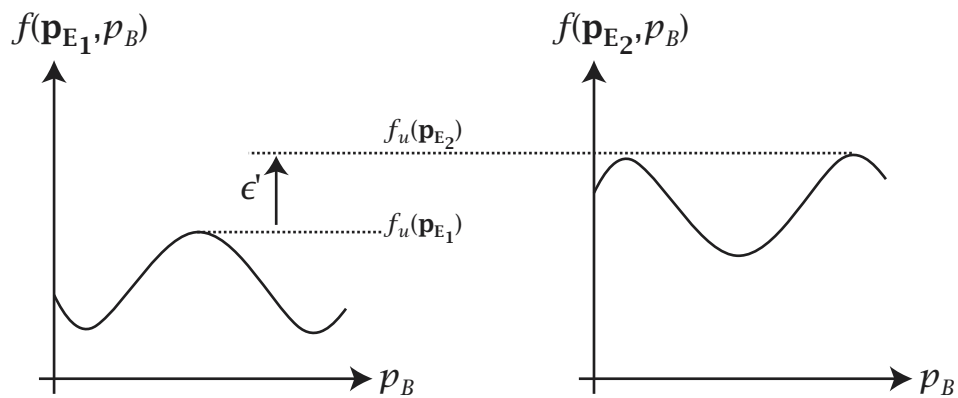
$$f_{u,unten}([\mathbf{P}_E]) = \sum_{i=1}^{n_f} \max_{j=1}^n \min (g_i [F_i([\mathbf{P}_E], [\mathbf{P}_{BS,Spez,j}])]), \quad (3.69)$$

$$f_{u,oben}([\mathbf{P}_E]) = \sum_{i=1}^{n_f} \max_{j=1}^n \max (g_i [F_i([\mathbf{P}_E], [\mathbf{P}_{BS,Spez,j}])]). \quad (3.70)$$

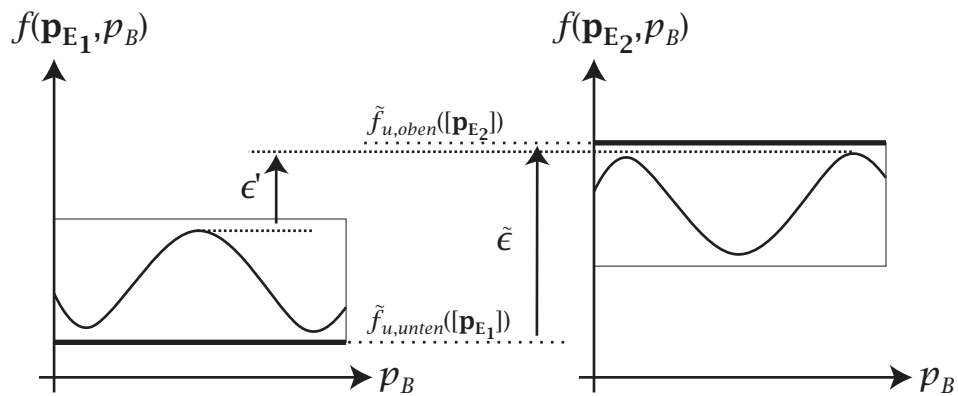
Die Intervallaufteilung reduziert die Überschätzung. Zusätzlich wird in Gleichung 3.69 durch die Auswahl des Maximums aller Teilintervallergebnisse der max-Operator in Relation 3.68 auf Teilintervallebene nachgebildet. Das Abbruchkriterium von Algorithmus 3.2 kann damit für hinreichend große n und hinreichend schmale Intervalltupel $[\mathbf{P}_E]$ immer erfüllt werden. Die Aufteilung von $[\mathbf{P}_E]$ findet dabei in Algorithmus 3.2 als Teil der Lösungssuche statt.

Beispiel 3.10 *Abbildung 3.8 zeigt die Berechnung einer unteren Schranke $f_{u,unten}([\mathbf{p}_{E1}])$ für die vereinigte Zielfunktion f_u an der Stelle \mathbf{p}_{E1} und*

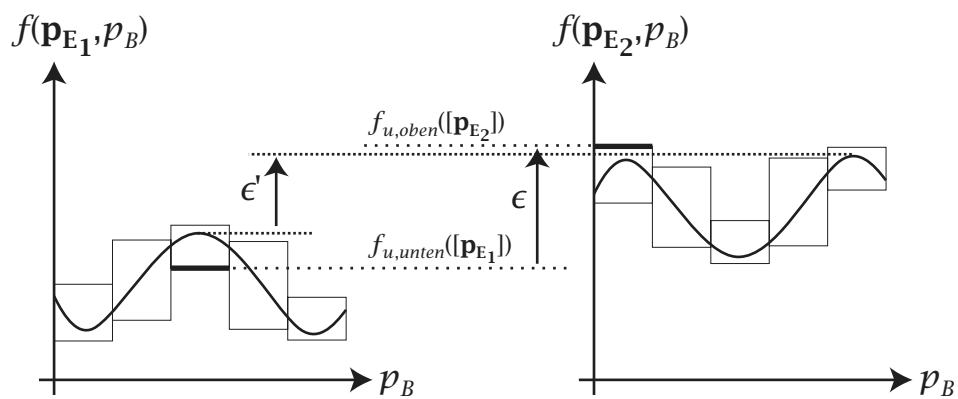
3.3 Dimensionierungsalgorithmus mit formalen Methoden



(a) Exakter relativer Abstand ϵ'



(b) Relativer Abstand $\tilde{\epsilon}$ von $\tilde{f}_{u,oben}([p_{E2}])$ zu $\tilde{f}_{u,unten}([p_{E1}])$



(c) Relativer Abstand ϵ von $f_{u,oben}([p_{E2}])$ zu $f_{u,unten}([p_{E1}])$ für $n = 5$

Abbildung 3.8: Schranken der Zielfunktion f_u und relativer Abstand ϵ

3 Dimensionierung mit formalen Methoden

einer oberen Schranke $f_{u,oben}([\mathbf{p}_{E2}])$ an der Stelle \mathbf{p}_{E2} . Hieraus wird der in Gleichung 3.41 verwendete relative Abstand

$$\epsilon = \frac{f_{u,oben}([\mathbf{p}_{E2}]) - f_{u,unten}([\mathbf{p}_{E1}])}{|f_{u,unten}([\mathbf{p}_{E1}])|} \quad (3.71)$$

berechnet. Die Zielfunktion f_u ist dabei mit $n_f = 1$, $n_{p_B} = 1$ und $n_{p_S} = 0$ durch

$$f_u(\mathbf{p}_E) = \max_{p_B \in P_{B,Spez}} f(\mathbf{p}_E, p_B) \quad (3.72)$$

gegeben (vgl. Gleichung 3.5).

Die Abbildungen 3.8(a) bis 3.8(c) zeigen jeweils den Verlauf von f in Abhängigkeit des Betriebsparameters $p_B \in [P_{B,Spez}]$ an den Stellen \mathbf{p}_{E1} und \mathbf{p}_{E2} . Aus den Verläufen lässt sich der eingetragene exakte relative Abstand

$$\epsilon' = \frac{f_u(\mathbf{p}_{E2}) - f_u(\mathbf{p}_{E1})}{|f_u(\mathbf{p}_{E1})|} \quad (3.73)$$

anhand der Maxima von f direkt ablesen (s. Abbildung 3.8(a)). Bei allgemeinen Problemen ist dies jedoch nicht möglich (vgl. Abschnitte 2.2.6 und 3.2).

Mit Hilfe der in den Gleichungen 3.63 und 3.64 definierten Schrankenfunktionen werden Schranken $\tilde{f}_{u,unten}([\mathbf{p}_{E1}])$ und $\tilde{f}_{u,oben}([\mathbf{p}_{E2}])$ bestimmt. Die dazu benötigten gebietsarithmetischen Einschließungen $[F_u([\mathbf{p}_{E1}], [P_{B,Spez}])]$ und $[F_u([\mathbf{p}_{E2}], [P_{B,Spez}])]$ von f_u sind in Abbildung 3.8(b) als Rechtecke dargestellt. Die sich hieraus ergebenden Schranken sind durch die Verlängerungen der hervorgehobenen Seiten eingezeichnet. Der relative Abstand $\tilde{\epsilon}$ entsprechend Gleichung 3.71 ist dabei erheblich größer als der exakte relative Abstand ϵ' . Der Grund liegt offensichtlich darin, dass eine untere Schranke $\min([F([\mathbf{p}_{E1}], [P_{B,Spez}])])$ des Minimums von f verwendet wird, obwohl der Abstand der Maxima gesucht ist. Dieses spiegelt die zusätzliche Implikation in der Bedingung für die Korrektheit der unteren Schranke 3.65 gegenüber Bedingung 3.66 wider.

Abbildung 3.8(c) zeigt die auf einer Intervallaufteilung von $[P_{B,Spez}]$ in $n = 5$ Teilintervalle beruhende Berechnung der Schrankenfunktionen $f_{u,unten}([\mathbf{p}_{E1}])$ und $f_{u,oben}([\mathbf{p}_{E2}])$ (vgl. Gleichungen 3.69 bzw. 3.70). Es ergibt sich eine abschnittsweise Einschließung von f_u , die durch die dargestellten Rechtecke gegeben ist. Die mit $f_{u,oben}([\mathbf{p}_{E2}])$ berechnete obere Schranke reduziert sich infolge der Verminderung der Überschätzung geringfügig gegenüber $\tilde{f}_{u,oben}([\mathbf{p}_{E2}])$. Die untere Schranke, die sich aus $f_{u,unten}([\mathbf{p}_{E1}])$ ergibt, liegt jedoch deutlich über der mit

$\tilde{f}_{u,unten}([\mathbf{p}_{E1}])$ ermittelten. Dies ist auf die Auswahl der größten unteren Schranke $\min([F([\mathbf{p}_{E1}], [P_{B,Spez,j}])])$ aller Teilintervalle entsprechend Gleichung 3.69 zurückzuführen. Der damit berechnete Abstand ϵ ist deutlich kleiner als der mit $\tilde{f}_{u,unten}([\mathbf{p}_{E1}])$ und $\tilde{f}_{u,oben}([\mathbf{p}_{E2}])$ berechnete $\tilde{\epsilon}$ (vgl. Abbildung 3.8(b)).

3.3.3 Lösung der nichtlinearen Systemgleichungen

Die im vorigen Abschnitt vorgestellten formalen Methoden zur Worst-Case-Analyse erfordern die gebietsarithmetische Auswertung des Nebenbedingungsfunktionstupels \mathbf{c} und des Zielfunktionstupels \mathbf{f} in Abhängigkeit vom intervallwertigen Parametertupel $[\mathbf{P}]$. Dieses umfasst in der Praxis die Auswertung der Analyseabbildung \mathbf{a} mit Gebietsarithmetik.

In einem Dimensionierungsproblem für Schaltungen auf Bauelementeebene betreffen die meisten spezifizierten Schaltungseigenschaften das elektrische Verhalten. Sie werden auf Basis einer Analyse der Knotenspannungen und Zweigströme der Netzliste berechnet (vgl. Abschnitte 2.2.7). Es ist für das Verfahren dieser Arbeit entscheidend, dass bei den gebietsarithmetisch durchzuführenden Berechnungen die Inklusionsisotonie gewährleistet wird, da die in den vorangegangenen Abschnitten vorgestellten formalen Methoden auf ihr aufbauen. Das am häufigsten eingesetzte Werkzeug Schaltungssimulator leistet dies jedoch nicht (vgl. Abschnitt 2.2.4). Für diese Arbeit wurde daher ein Analyseverfahren zur Bestimmung von Knotenspannungen und Zweigströme in Netzlisten mit nichtlinearen Bauelementen entwickelt. Dieses wird in den folgenden Abschnitten für die Analyse nichtlinearen statischen Verhaltens und dynamischen Kleinsignalverhaltens dargestellt.

Nichtlineares statisches Verhalten

Zunächst wird die als Netzliste auf Bauelementeebene in der strukturellen Sicht gegebene Schaltungsbeschreibung mit Hilfe eines Standardnetzwerkanalyseverfahrens (vgl. Abschnitt 2.2.4) in ein Gleichungssystem in der Funktionssicht überführt. Das entstehende System ist ein nichtlineares differential-algebraisches Gleichungssystem (differential algebraic equations, DAE):

Definition 3.35 *Eine Schaltungsbeschreibung durch ein nichtlineares differential-algebraisches Gleichungssystem ist in allgemeiner Form*

3 Dimensionierung mit formalen Methoden

durch

$$\mathbf{f}_{\text{DAE}}(\mathbf{u}, \dot{\mathbf{u}}, \mathbf{p}, t) = \mathbf{0} \quad (3.74)$$

gegeben. Hierin ist $\mathbf{u} \in \mathbb{R}^{n_u}$ das Tupel der unbekanntenen Spannungen und Ströme, $\mathbf{p} \in P$ das Parametertupel, t die Zeit und $\mathbf{f}_{\text{DAE}} : \mathbb{R}^{n_u} \times \mathbb{R}^{n_u} \times P \times \mathbb{R} \rightarrow \mathbb{R}^{n_u}$ das Tupel der Systemgleichungsfunktionen.

Zur Analyse des nichtlinearen statischen Verhaltens wird dieses in ein nichtlineares algebraisches Gleichungssystem (algebraic equations, AE) überführt (vgl. Abschnitt 2.2.4).

Definition 3.36 Das nichtlineare algebraische Gleichungssystem

$$\mathbf{f}_{\text{AE}}(\mathbf{u}, \mathbf{p}) = \mathbf{0} \quad (3.75)$$

mit $\mathbf{u} \in \mathbb{R}^{n_u}$, $\mathbf{p} \in P$ und $\mathbf{f}_{\text{AE}} : \mathbb{R}^{n_u} \times P \rightarrow \mathbb{R}^{n_u}$ ergibt sich aus Gleichungssystem 3.74 durch Nullsetzen aller zeitlichen Ableitungen und Konstantsetzen aller Zeitveränderlichen in \mathbf{f}_{DAE} .

Eine Möglichkeit zur Bestimmung von \mathbf{f}_{AE} ist $\mathbf{f}_{\text{AE}}(\mathbf{u}, \mathbf{p}) = \mathbf{f}_{\text{DAE}}(\mathbf{u}, \mathbf{0}, \mathbf{p}, 0)$.

Formal lässt sich eine exakte Lösungsmenge U bei vorgegebenem intervallwertigen Parametertupel $[P]$ definieren.

Definition 3.37 Für $[P] \in \mathbb{I}_P$ und $\mathbf{f}_{\text{AE}} : \mathbb{R}^{n_u} \times P \rightarrow \mathbb{R}^{n_u}$ ist die exakte Lösungsmenge $U \subseteq \mathbb{R}^{n_u}$ gegeben durch

$$U = \{\mathbf{u} \in \mathbb{R}^{n_u} : \exists \mathbf{p} \in [P] : \mathbf{f}_{\text{AE}}(\mathbf{u}, \mathbf{p}) = \mathbf{0}\} . \quad (3.76)$$

Ziel des gebietsarithmetischen Analyseverfahrens ist es, eine möglichst enge Intervalleinschließung $[U_e] \supseteq U$ zu berechnen. Diese wird zur Berechnung von gebietsarithmetischen Erweiterungen des Nebenbedingungsfunktionstupels \mathbf{c} und des Zielfunktionstupels \mathbf{f} verwendet, so dass die geforderte Inklusionsisotonie gewährleistet ist.

Einschließungstest

Die Intervallerweiterung von \mathbf{f}_{AE} ist durch $[F_{\text{AE}}]$ gegeben. Für $[U] \in \mathbb{I}_{\mathbb{R}^{n_u}}$ und $[P] \in \mathbb{I}_P$ liefert die Inklusionsisotonie von $[F_{\text{AE}}]$:

$$\forall \mathbf{u} \in [U], \mathbf{p} \in [P] : \mathbf{f}_{\text{AE}}(\mathbf{u}, \mathbf{p}) \in [F_{\text{AE}}([U], [P])] . \quad (3.77)$$

Hieraus folgt die Implikation

$$\exists \mathbf{u} \in [U], \mathbf{p} \in [P] : \mathbf{f}_{\text{AE}}(\mathbf{u}, \mathbf{p}) = \mathbf{0} \quad \Rightarrow \quad \mathbf{0} \in [F_{\text{AE}}([U], [P])] . \quad (3.78)$$

3.3 Dimensionierungsalgorithmus mit formalen Methoden

Wegen $(A \Rightarrow B) \Leftrightarrow (\neg A \Leftarrow \neg B)$ für allgemeine Aussagen A und B ist Implikation 3.78 äquivalent zu

$$\neg \exists \mathbf{u} \in [\mathbf{U}], \mathbf{p} \in [\mathbf{P}] : \mathbf{f}_{\text{AE}}(\mathbf{u}, \mathbf{p}) = \mathbf{0} \quad \Leftarrow \quad \mathbf{0} \notin [\mathbf{F}_{\text{AE}}([\mathbf{U}], [\mathbf{P}])] . \quad (3.79)$$

Bei Verwendung von Gleichung 3.76 folgt, dass dies äquivalent zu

$$\mathbf{0} \notin [\mathbf{F}_{\text{AE}}([\mathbf{U}], [\mathbf{P}])] \quad \Rightarrow \quad [\mathbf{U}] \cap U = \emptyset \quad (3.80)$$

ist.

Die Implikation 3.80 ist ein einfacher Einschließungstest, mit dem sich $[\mathbf{U}_e]$ durch Entfernen von Teilintervallen, die sich nicht mit U schneiden, verkleinern und damit der exakten Lösung U annähern lässt. Wegen des Überschätzungseffektes ist Implikation 3.80 nur eine hinreichende, aber keine notwendige Bedingung. Es ist daher oft erforderlich, $[\mathbf{U}]$ wiederholt in Teilintervalle $[\mathbf{U}_i]$ aufzuteilen, bis die Überschätzung so weit reduziert ist, dass $\mathbf{0} \notin [\mathbf{F}_{\text{AE}}([\mathbf{U}_i], [\mathbf{P}])]$ gilt und sich damit $[\mathbf{U}_i] \cap U = \emptyset$ folgern lässt.

Beispiel 3.11 *Es wird Gleichung 3.75 für eine eindimensionale Funktion*

$$f_{\text{AE}}(u) = u^2 - u \quad (3.81)$$

betrachtet. Zur Vereinfachung wird angenommen, dass f_{AE} keine Abhängigkeit von \mathbf{p} hat. Ferner wird nur die Nullstelle bei 1 betrachtet.

Mit Hilfe des Einschließungstests 3.80 wird eine Verbesserung von $[U_e^0] = [0,9; 1,1]$ berechnet. Abbildung 3.9 zeigt jeweils den Verlauf von f_{AE} im Intervall $[U_e^0]$ sowie in Abbildung 3.9(a) eine abschnittsweise Einschließung bei Aufteilung von $[U_e^0]$ in 3 Teilintervalle und in Abbildung 3.9(b) in 4 Teilintervalle. In beiden Fällen wurden die Einschließungen mit der natürlichen Intervallerweiterung $[F_{\text{AE},n}]$ berechnet. Beide weisen eine deutliche Überschätzung des Wertebereiches von f_{AE} auf. Dies spiegelt sich formal darin wider, dass Bedingung 3.80 nur hinreichend, aber nicht notwendig ist. Es ist wegen der Überschätzung nicht möglich, in Abbildung 3.9(a) für das linke oder das rechte Teilintervall mit Hilfe der Bedingung 3.80 zum Schluss $[U_i] \cap U = \emptyset$ zu gelangen. Dieses ist erst mit der in Abbildung 3.9(b) gezeigten Aufteilung in 4 Teile für das hervorgehobene Teilintervall $[1,05; 1,1]$ erfolgreich. Das verbesserte $[U_e^1]$ ist demnach durch

$$[U_e^1] = [U_e^0] \setminus [1,05; 1,1] = [0,9; 1,05] \quad (3.82)$$

gegeben.

3 Dimensionierung mit formalen Methoden

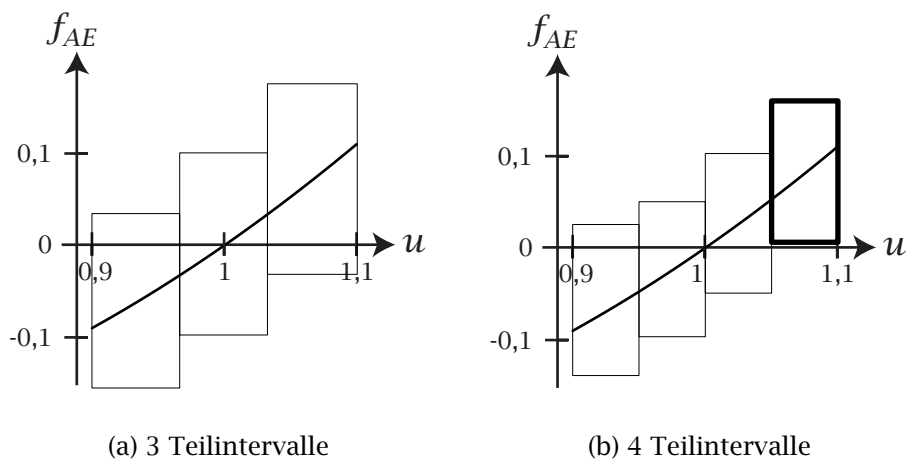


Abbildung 3.9: Verbesserung von $[U_e]$ mit dem Einschließungstest

Das Beispiel zeigt den Nachteil der lediglich linearen Konvergenzordnung (vgl. Gleichung 3.27) der mit Intervallarithmetik berechneten Einschließungen: Auch bei nahezu linearem Verlauf der betrachteten Funktion (vgl. Abbildung 3.9) und beliebiger Verringerung der Breite der Teilintervalle $[U_i]$ bleibt die relative Überschätzung

$$\frac{d([F_{AE,n}([U_i]))}{d([F_{AE,v}([U_i]))} - 1 \quad (3.83)$$

konstant und endlich. Die Anzahl der erforderlichen Intervallaufteilungen von $[U_e]$ pro Verbesserungsschritt ist daher ebenfalls konstant.

In den Abbildungen 3.9(a) und 3.9(b) entspricht die durch den Ausdruck 3.83 gegebene relative Überschätzung dem relativen Anteil eines mit $[F_{AE,n}]$ berechneten einschließenden Rechtecks, der nicht vom Verlauf von f_{AE} überstrichen wird.

Newton-Operator mit affiner Arithmetik

Der Einschließungstest 3.80 basiert auf Intervallarithmetik. Leistungsfähigere Bedingungen lassen sich mit Hilfe der affinen Arithmetik aufstellen, da diese gegenüber der Intervallarithmetik bessere Konvergenzeigenschaften und Vorteile bei mehrdimensionalen Problemen aufweist (vgl. Abschnitt 3.2.3). Im Folgenden wird ein auf affiner Arithmetik aufbauender Newton-Operator für die Verbesserung von $[U_e]$ hergeleitet, der die Forderung der Inklusionsisotonie erfüllt. Hierzu wurde der von

3.3 Dimensionierungsalgorithmus mit formalen Methoden

de Figueiredo und Stolfi (1997) für den eindimensionalen Fall vorgeschlagene Operator im Rahmen dieser Arbeit zu einem mehrdimensionalen Operator erweitert.

Die Schnittmenge eines beliebigen Intervalltupels $[U] \in \mathbb{I}_{\mathbb{R}}^{n_u}$ mit U ergibt sich aus Gleichung 3.76 zu

$$[U] \cap U = \{\mathbf{u} \in [U] : \exists \mathbf{p} \in [P] : \mathbf{f}_{AE}(\mathbf{u}, \mathbf{p}) = \mathbf{0}\} . \quad (3.84)$$

Sie lässt sich mit den affinen Formen $\hat{\mathbf{u}}_{\varepsilon} = \hat{\mathcal{A}}_{\varepsilon}([U])$ und $\hat{\mathbf{p}}_{\varepsilon} = \hat{\mathcal{A}}_{\varepsilon}([P])$ als

$$[U] \cap U = \{\mathbf{u} \in [U] : \exists \varepsilon \in \mathbb{U}^{n_{\varepsilon}} : \mathbf{u} = \hat{\mathbf{u}}_{\varepsilon} \wedge \mathbf{f}_{AE}(\hat{\mathbf{u}}_{\varepsilon}, \hat{\mathbf{p}}_{\varepsilon}) = \mathbf{0}\} \quad (3.85)$$

ausdrücken. Wird \mathbf{f}_{AE} darin durch eine affine Erweiterung $\hat{\mathbf{f}}_{AE,\varepsilon}$ ersetzt, ergibt dies eine Einschließung $U^* \supseteq [U] \cap U$ mit

$$U^* = \{\mathbf{u} \in [U] : \exists \varepsilon \in \mathbb{U}^{n_{\varepsilon}} : \mathbf{u} = \hat{\mathbf{u}}_{\varepsilon} \wedge \hat{\mathbf{f}}_{AE,\varepsilon}(\hat{\mathbf{u}}_{\varepsilon}, \hat{\mathbf{p}}_{\varepsilon}) = \mathbf{0}\} . \quad (3.86)$$

Die schmalste Intervalleinschließung $[U^{**}] \supseteq U^*$ lässt sich durch

$$[U^{**}] = \mathbf{u}_0 + \sum_{i=1}^{n_{\varepsilon}} [\varepsilon_{i,min}, \varepsilon_{i,max}] |\mathbf{u}_i| \quad (3.87)$$

berechnen, wobei die \mathbf{u}_i die Tupel der Mittelwerte bzw. der Koeffizienten des Tupels affiner Formen $\hat{\mathbf{u}}_{\varepsilon}$ sind. Die Grenzen $\varepsilon_{i,min}$ und $\varepsilon_{i,max}$ in Gleichung 3.87 sind

$$\varepsilon_{i,min} = \min \varepsilon_i , \quad (3.88)$$

$$\varepsilon_{i,max} = \max \varepsilon_i \quad (3.89)$$

unter den Randbedingungen

$$\varepsilon \in \mathbb{U}^{n_{\varepsilon}} \wedge \hat{\mathbf{f}}_{AE,\varepsilon}(\hat{\mathbf{u}}_{\varepsilon}, \hat{\mathbf{p}}_{\varepsilon}) = \mathbf{0} . \quad (3.90)$$

Gleichungen 3.88 bis 3.90 bilden ein Satz von $2n_u$ linearen Optimierungsproblemen, die gelöst werden müssen, um $[U^{**}]$ zu bestimmen. Der benötigte Rechenaufwand lässt sich bei üblichen Lösungsverfahren durch Ausnutzung der Gleichheit der Nebenbedingungen 3.90 in allen $2n_u$ Optimierungsproblemen erheblich reduzieren, da die Bestimmung eines Punktes, der die Nebenbedingungen erfüllt, nur einmal erforderlich ist (Nocedal und Wright 1999).

3 Dimensionierung mit formalen Methoden

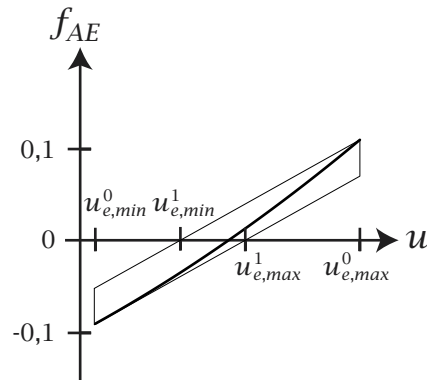


Abbildung 3.10: Newton-Operator

Definition 3.38 Für Gleichung 3.75 und ein beliebiges Intervalltupel $[U] \in \mathbb{R}^{n_u}$ ist der Newton-Operator $[U_N]: \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_u}$ durch

$$[U_N([U])] = [U^{**}] \quad (3.91)$$

definiert, wobei $[U^{**}]$ entsprechend den Gleichungen 3.87 bis 3.90 zu bestimmen ist.

Die Anwendung dieses Newton-Operators führt einen Newton-Iterationsschritt durch. Wie das klassische Newton-Verfahren arbeitet auch dieser Operator $[U_N]$ mit Ableitungen von f_{AE} , um eine Kontraktion zur gesuchten Lösung zu erreichen: Die Ableitungen sind Teil der affinen Erweiterung $\hat{f}_{AE,\varepsilon}$. Für hinreichend schmale Intervalltupel $[U]$ und $[P]$ liefert eine iterative Anwendung daher ebenfalls quadratische Konvergenz. Die Folge von Newton-Iterationsschritten wird durch $[U_N]$ in eine Folge von linearen Optimierungsproblemen überführt.

Beispiel 3.12 Es wird erneut die Aufgabe aus Beispiel 3.11 betrachtet, wobei hier der Newton-Operator verwendet wird. Abbildung 3.10 zeigt den Verlauf von f_{AE} im Intervall $[U_e^0] = [u_{e,min}^0, u_{e,max}^0] = [0,9; 1,1]$. Mit Hilfe der affinen Erweiterung $\hat{f}_{AE,\varepsilon}$ wird eine Einschließung berechnet, die in Abbildung 3.10 als Parallelogramm dargestellt ist. Das lineare Optimierungsproblem, das durch die Gleichungen 3.88 bis 3.90 gegeben ist, kann in diesem Beispiel graphisch gelöst werden, da sowohl u als auch f_{AE} eindimensional sind: Innerhalb des Parallelogramms sind unter der Nebenbedingung $f_{AE}(u) = 0$ das Minimum $u_{e,min}^1$ und das Maximum $u_{e,max}^1$ zu bestimmen. Damit ist die durch den Newton-Operator berechnete Lösung

$$[U_N([U_e^0])] = [U_e^1] = [u_{e,min}^1, u_{e,max}^1] = [0,9625; 1,0125]. \quad (3.92)$$

3.3 Dimensionierungsalgorithmus mit formalen Methoden

Weitere, in Abbildung 3.10 nicht dargestellte Iterationen liefern

$$[U_e^2] = [U_N([U_e^1])] = [0,9988; 1,0016] \quad (3.93)$$

und

$$[U_e^3] = [U_N([U_e^2])] = [0,9999; 1,00001] . \quad (3.94)$$

An diesem Beispiel wird sichtbar, dass das Verfahren die quadratische Konvergenz des klassischen Newton-Verfahrens ebenfalls aufweist: In hinreichender Nähe zur Lösung verdoppelt sich durch jede Iteration die Anzahl der richtigen Stellen im Ergebnis.

Vergleich des Newton-Operators mit dem Einschließungstest

Der Einschließungstest 3.80 liefert für ein Testintervalltupel $[U]$ einen Wahrheitswert für die Aussage, dass sich dieses mit der exakten Lösung U überlappt. Auf dieser Grundlage lässt sich eine Verbesserung einer Einschließung $[U_e] \supseteq U$ durch ein Bisektionsverfahren erreichen. Ein solches Verfahren hätte lineare Konvergenz, wenn der Einschließungstest 3.80 nicht nur eine hinreichende, sondern auch eine notwendige Bedingung wäre. Aufgrund von Überschätzungseffekten ist dies jedoch nicht gegeben. Lineare Konvergenz wird in der Praxis daher nicht erreicht.

Ein Bedingung 3.80 entsprechender Einschließungstest lässt sich auch mit Hilfe des Newton-Operators $[U_N]$ aufstellen:

$$[U_N([U])] = \emptyset \quad \Rightarrow \quad [U] \cap U = \emptyset \quad (3.95)$$

Diese Variante ist der mit Intervallarithmetik 3.80 aus den in Abschnitt 3.2.3 genannten Gründen überlegen. Ein auf Test 3.95 beruhendes Bisektionsverfahren kann bestenfalls lineare Konvergenz haben. Der Newton-Operator $[U_N]$ liefert bei direkter Anwendung zusätzlich eine Kontraktion des Testintervalltupels $[U]$ mit quadratischer Konvergenz zur Schnittmenge $[U] \cap U$ hin. Die quadratische Konvergenz tritt in der Praxis jedoch nur für hinreichend schmale Intervalltupel $[U]$ und $[P]$ ein.

Der Newton-Operator erfordert mit der Lösung von linearen Optimierungsproblemen einen deutlich höheren Rechenaufwand als der Einschließungstest 3.80 mit Intervallarithmetik. In dem Verfahren dieser Arbeit wird daher zunächst der Test 3.80 ausgewertet und nur bei Misserfolg zusätzlich der Newton-Operator.

Einbindung in den Optimierungsalgorithmus

Ausgehend von einer Starteinschließung $[U_e^0] \supseteq U$, die aus trivialen Schranken für die Knotenspannungen und Zweigströme gewonnen wird, werden iterativ nacheinander der Einschließungstest 3.80 und der Newton-Operator angewendet sowie Intervallaufteilungen von $[U_e]$ vorgenommen.

Gleichstrom-Transfer-Simulation

Die beschriebene Analyseverfahren liefert garantierte Schranken für die Schwankungen des Arbeitspunktes der Schaltung in $[P_E]$ und $[P_{BS,Spez}]$. Damit kann nicht nur die Lage des Arbeitspunktes selbst analysiert werden, sondern auch sein Verlauf in Abhängigkeit einer Eingangsgröße oder eines Betriebsparameters als Teil von $[P_{B,Spez}]$. Dies entspricht einer Gleichstrom-Transfer-Simulation ohne den Nachteil simulatorbasierter Analysen, dass nur einzelne Punkte im zu betrachtenden Intervall untersucht werden können und das Ergebnis daher systematisch unterschätzt wird (vgl. Abschnitt 2.2.4). Durch entsprechend formulierte Nebenbedingungen lässt sich eine Schaltung so dimensionieren, dass eine vorgegebene statische Kennlinie eingehalten wird (s. Abschnitt 4.2.2).

Dynamisches Kleinsignalverhalten

Zur Analyse des dynamischen Kleinsignalverhaltens im Frequenzbereich wird das sich aus der Netzwerkanalyse ergebende nichtlineare differential-algebraische Gleichungssystem zunächst linearisiert. Im Gegensatz zu auf diskreten Arbeitspunkten basierenden Dimensionierungsverfahren darf bei dem Ansatz dieser Arbeit jedoch keine Linearisierung in einzelnen Arbeitspunkten vorgenommen werden. Es muss statt dessen eine Linearisierung erzeugt werden, die garantiert auf ganz $[P]$ Gültigkeit hat, damit die Inklusionsisotonie erhalten wird. Dazu sind die bei der Analyse des nichtlinearen statischen Verhaltens berechneten Schranken für die Schwankungen des Arbeitspunktes heranzuziehen. Die so erzielte Linearisierung wird durch Fourier-Transformation in den Frequenzbereich übertragen. Dadurch ergibt sich ein lineares Gleichungssystem in komplexen Intervallen.

Definition 3.39 *Das durch Linearisierung und Fourier-Transformation von f_{DAE} erzeugte lineare komplexe Intervallgleichungssystem ist gege-*

3.3 Dimensionierungsalgorithmus mit formalen Methoden

ben als

$$([\mathcal{A}_{Re}] + j[\mathcal{A}_{Im}]) ([\mathbf{u}_{Re}] + j[\mathbf{u}_{Im}]) = ([\mathbf{b}_{Re}] + j[\mathbf{b}_{Im}]) . \quad (3.96)$$

Darin ist $[\mathcal{A}_{Re}] \in \mathbb{R}^{n_u \times n_u}$ der Realteil der intervallwertigen, komplexen Systemmatrix $[\mathcal{A}]$, und $[\mathbf{u}_{Re}], [\mathbf{b}_{Re}] \in \mathbb{R}^{n_u}$ sind die Realteile des Lösungsvektors $[\mathbf{u}]$ bzw. der rechten Seite $[\mathbf{b}]$. Entsprechendes gilt für die Imaginärteile.

Das komplexe Gleichungssystem 3.96 lässt sich in das nachfolgende reelle Gleichungssystem überführen (Golub und Loan 1996):

$$\begin{pmatrix} [\mathcal{A}_{Re}] & -[\mathcal{A}_{Im}] \\ [\mathcal{A}_{Im}] & [\mathcal{A}_{Re}] \end{pmatrix} \begin{pmatrix} [\mathbf{u}_{Re}] \\ [\mathbf{u}_{Im}] \end{pmatrix} = \begin{pmatrix} [\mathbf{b}_{Re}] \\ [\mathbf{b}_{Im}] \end{pmatrix} . \quad (3.97)$$

Das in dieser Darstellung vorliegende System wird mit Standardmethoden der linearen Algebra für reelle Gleichungssysteme gelöst. Dabei sind sämtliche Rechenoperationen durch gebietsarithmetische Erweiterungen zu ersetzen, damit die Inklusionsisotonie erhalten wird.

In der Spezifikation werden Kleinsignaleigenschaften meist für Frequenzintervalle gefordert. Diese Intervalle werden als Teil der Betriebsbereichsparameter $[\mathbf{P}_{B,Spez}]$ behandelt. Ferner werden oft Betrag und Phase an Stelle von Real- und Imaginärteil spezifiziert. Zu ihrer Bestimmung werden gebietsarithmetische Erweiterungen der Operationen $|\cdot|$ und \arg verwendet.

Dynamisches Großsignalverhalten

Das in den vorigen Abschnitten beschriebene Verfahren umfasst nicht die Analyse des dynamischen Großsignalverhaltens im Zeitbereich. Falls hierfür in der Spezifikation Bedingungen gegeben sind, z. B. für Anstiegsgeschwindigkeiten, müssen Analysegleichungen vorgegeben werden (vgl. Abschnitt 2.2.3), die im Rahmen des Verfahrens gebietsarithmetisch ausgewertet werden.

Der dargestellte Ansatz kann grundsätzlich auch auf die Analyse dynamischen Großsignalverhaltens im Zeitbereich erweitert werden. Dies trifft ebenso auf andere, hier nicht aufgeführte Analysearten, z. B. Rauschanalyse, zu.

3.3.4 Laufzeitkomplexität

Nachfolgend wird die Laufzeitkomplexität des vorgestellten Dimensionierungsalgorithmus abgeschätzt. Dabei wird zunächst eine Betrachtung

3 Dimensionierung mit formalen Methoden

für Intervallararithmetik durchgeführt. Darauf aufbauend werden die Verbesserungen durch den Einsatz der affinen Arithmetik dargestellt. Die in den vorhergehenden Abschnitten durchgeführte Aufteilung des Algorithmus in die Komponenten: globale Optimierung, Worst-Case-Analyse und Lösung der Systemgleichungen wird hier übernommen.

Globale Optimierung

Zur globalen Optimierung mit Nebenbedingungen wird das in Algorithmus 3.2 beschriebene Bisektionsverfahren nach der Branch-and-Bound-Methode verwendet. Es ist in der Lage, durch iteratives Halbieren eines Ausgangsgebietes im Entwurfparameterraum P_E ohne Rückschritte (backtracking) zur Lösung $[\mathbf{P}_E^*]$ zu gelangen, wenn die berechneten Schranken der Zielfunktion f_u immer hinreichend genau sind und eines der Prädikate falsifiziert und verifiziert in jedem Teilgebiet den Wert wahr ergibt. Unter dieser Voraussetzung benötigt das Verfahren eine Laufzeit von

$$L_{ges} = \mathcal{O}\left(n_{p_E} \log_2 \frac{1}{\epsilon_{Tol}}\right) \cdot N_{P_E} \cdot L_{WC} . \quad (3.98)$$

Hierin ist L_{WC} die Laufzeit der untergeordneten Worst-Case-Analyse. Der Faktor N_{P_E} berücksichtigt den Aufwand der Intervallaufteilungen des betrachteten Teilgebietes von P_E . Durch die Aufteilungen muss die durch die Intervallararithmetik verursachte Überschätzung so weit reduziert werden, dass die Schranken der Zielfunktion f_u und die Prädikate falsifiziert und verifiziert die o. a. Voraussetzungen erfüllen. Wegen der linearen Konvergenzordnung der Intervallararithmetik bleibt die relative Überschätzung unabhängig von der Größe des Teilgebietes gleich und die Anzahl der bei jedem Bisektionsschritt benötigten Intervallaufteilungen je Entwurfparameter ist konstant (vgl. Erläuterungen zu Beispiel 3.11). Es ergeben sich somit $b_{P_E}^{n_{p_E}}$ Teilintervalle, wenn $b_{P_E} - 1$ die Anzahl der Aufteilungen je Entwurfparameter ist (vgl. Erläuterungen zu Gleichung 3.27). Für den Faktor N_{P_E} gilt daher

$$N_{P_E} = b_{P_E}^{n_{p_E}} . \quad (3.99)$$

Worst-Case-Analyse

Bei der Worst-Case-Analyse muss die Überschätzung der Intervallararithmetik durch komponentenweise Intervallaufteilung von $P_{BS,Spez}$ hinreichend reduziert werden, damit die Schranken der Zielfunktion f_u und

3.3 Dimensionierungsalgorithmus mit formalen Methoden

die Prädikate falsifiziert und verifiziert bei der übergeordneten Branch-and-Bound-Optimierung die richtigen Entscheidungen bzgl. der Verzweigungen im Suchbaum ermöglichen. Es ergeben sich $b_{P_{BS}}^{n_{P_{BS}}}$ Teilintervalle, wenn $b_{P_{BS}} - 1$ die Anzahl der Aufteilungen je Dimension von P_{BS} ist. Für jedes dieser Teilintervalle muss die Lösung der Systemgleichungen durchgeführt werden, so dass für den Laufzeitbedarf der Worst-Case-Analyse

$$L_{WC} = \mathcal{O}(b_{P_{BS}}^{n_{P_{BS}}}) \cdot L_{L\ddot{u}ser} \quad (3.100)$$

gilt.

Lösung der Systemgleichungen

Zur Lösung der nichtlinearen Systemgleichungen wird ein Bisektionsverfahren eingesetzt, das die Überschätzung vorgegebener Schranken des Arbeitsbereiches durch Verbesserung auf Δ_{soll} reduziert. Δ_{soll} muss so klein sein, dass auch die Überschätzung der Worst-Case-Analyse hinreichend klein (s. o.) werden kann. Damit ist der Rechenaufwand

$$L_{L\ddot{u}ser} = \mathcal{O}\left(n_u \log_2 \frac{1}{\Delta_{soll}}\right) \cdot L_{ET} N_U + L_{AC}. \quad (3.101)$$

Der Faktor L_{ET} erfasst den Aufwand zur einmaligen Auswertung des Einschließungstests 3.80. Dieser ist $L_{ET} = \mathcal{O}(n_u)$. N_U berücksichtigt die in jedem Bisektionsschritt erforderlichen Intervallaufteilungen (vgl. Gleichung 3.99) und ist entsprechend durch $N_U = b_U^{n_u}$ gegeben. Der Summand L_{AC} enthält den Laufzeitbedarf der Wechselstromanalyse. Dieser ist bei Verwendung des Gaußschen Eliminationsverfahrens zur Lösung des Gleichungssystems: $L_{AC} = \mathcal{O}(n_u^3)$. Der erste Summand ist wegen des exponentiellen Anwachsens von N_U gegenüber L_{AC} bezüglich der Komplexitätsordnung dominant. Daher gilt

$$L_{L\ddot{u}ser} = \mathcal{O}\left(L_{ET} N_U n_u \log_2 \frac{1}{\Delta_{soll}}\right). \quad (3.102)$$

Gesamtkomplexität

Durch Einsetzen der o. a. Gleichungen kann folgende Abschätzung der Gesamtlaufzeit angegeben werden:

$$L_{ges} = \mathcal{O}\left(n_{p_E} n_u^2 \log_2 \left(\frac{1}{\epsilon_{Tol}}\right) \log_2 \left(\frac{1}{\Delta_{soll}}\right) b_{P_E}^{n_{P_E}} b_{P_{BS}}^{n_{P_{BS}}} b_U^{n_u}\right). \quad (3.103)$$

3 Dimensionierung mit formalen Methoden

Es besteht eine exponentielle Laufzeitkomplexität in Abhängigkeit der Zahl der Entwurfparameter n_{p_E} , der Herstellungsprozess- und Betriebsbereichsparameter $n_{p_{BS}}$ und mit n_u auch der Schaltungsgröße. Im Vergleich dominiert die Abhängigkeit von $n_{p_{BS}}$, da jede Worst-Case-Analyse, die von dem Dimensionierungsverfahren durchgeführt wird, das ganze Gebiet $P_{BS,Spez}$ berücksichtigen muss. Die Überschätzung geht dabei direkt ins Ergebnis der Worst-Case-Analyse ein. Sie muss durch eine feine Aufteilung von $P_{BS,Spez}$ reduziert werden. Dadurch ergibt sich ein großes $b_{p_{BS}}$. Im Verhältnis dazu sind b_{p_E} und b_U kleiner, denn durch die Bisektionsschritte müssen nur immer kleiner werdende Teilgebiete von $[P_e^0]$ bzw. $[U_e^0]$ betrachtet werden und die Überschätzung muss nur so weit verringert werden, dass eine Verzweigungsentscheidung im Bisektionsbaum möglich ist.

Einsatz affiner Arithmetik

Für Gebietseinschließungen mit affiner Arithmetik gilt im Gegensatz zur linearen Konvergenzordnung der Intervallarithmetik (vgl. Gleichung 3.27) eine quadratische Konvergenzordnung (vgl. Gleichung 3.37). Dadurch nimmt die relative Überschätzung einer affinen Erweiterung mit der Breite des Argumentes linear ab, so dass bei einem Bisektionsverfahren keine Aufteilungen in Teilintervalle erforderlich sind, wenn die Argumente hinreichend schmal sind. Die Faktoren N_{p_E} und N_U fallen dadurch nach $\mathcal{O}(n_{p_E})$ bzw. $\mathcal{O}(n_u)$ Bisektionsschritten auf den Wert 1. Diese Anlaufrechnung ist wegen der exponentiellen Abhängigkeit der Faktoren N_{p_E} und N_U von der Problemgröße für die Laufzeitkomplexität dominant. Sie beträgt daher

$$L_{ges,AA} = \mathcal{O}\left(n_{p_E} b_{p_E}^{\frac{n_{p_E}}{2}}\right) \cdot L_{WC,AA} . \quad (3.104)$$

Gegenüber Gleichung 3.99 ist der Exponent von b_{p_E} hier wegen der quadratischen Konvergenz der affinen Arithmetik zu halbieren. Aus demselben Grund muss der Exponent von $b_{p_{BS}}$ gegenüber Gleichung 3.100 halbiert werden:

$$L_{WC,AA} = \mathcal{O}\left(b_{p_{BS}}^{\frac{n_{p_{BS}}}{2}}\right) \cdot L_{Löser,AA} . \quad (3.105)$$

Die bei der Lösung der Systemgleichungen ebenfalls dominante Anlaufrechnung verursacht einen Laufzeitbedarf von

$$L_{Löser,AA} = \mathcal{O}\left(n_u b_U^{\frac{n_u}{2}}\right) \cdot L_{NO} . \quad (3.106)$$

3.3 Dimensionierungsalgorithmus mit formalen Methoden

Die Auswertung des Newton-Operators $[U_N]$ erfordert die Lösung von $2n_u$ linearen Optimierungsproblemen. In dieser Arbeit wird der Simplex-Algorithmus verwendet, für den hier von einer Laufzeitkomplexität von $\mathcal{O}(n_u^2)$ ausgegangen wird, auch wenn er für einzelne, entartete Probleme exponentielle Komplexität aufweist. Grundsätzlich lassen sich lineare Optimierungsprobleme immer in polynomialer Laufzeit lösen, z. B. mit der Innen-Punkt-Methode (interior point method) (Nocedal und Wright 1999). Der Laufzeitbedarf des Newton-Operators beträgt damit $L_{NO} = \mathcal{O}(n_u^3)$.

Damit lässt sich die Gesamtlaufzeitkomplexität angeben:

$$L_{ges,AA} = \mathcal{O}\left(n_{p_E} n_u^4 b_{p_E}^{\frac{n_{p_E}}{2}} b_{p_{BS}}^{\frac{n_{p_{BS}}}{2}} b_U^{\frac{n_u}{2}}\right). \quad (3.107)$$

Gegenüber L_{ges} (vgl. Gleichung 3.103) hat sich der Einfluss der exponentiellen Faktoren durch die Verwendung affiner Arithmetik verringert. Dennoch besteht weiterhin eine exponentielle Laufzeitkomplexität in Abhängigkeit von n_{p_E} , $n_{p_{BS}}$ und n_u . Der Einfluss der quadratischen Konvergenz des Newton-Operators wird in Gleichung 3.107 nicht sichtbar, da sie nur für hinreichend schmale Intervalle, also nach der Anlaufrechnung, gilt.

Eine Vergleich des Rechenaufwandes bei Verwendung von Intervallarithmetik und affiner Arithmetik bei einem praktischen Beispiel ist in Abschnitt 4.2.4 dargestellt.

3 Dimensionierung mit formalen Methoden

4 Implementierung und Ergebnisse

4.1 Implementiertes Programmsystem

Das in Kapitel 3 beschriebene Dimensionierungsverfahren ist als Prototyp in der interpretierten Programmierumgebung des Computeralgebrasystems Mathematica (Wolfram Research 2002) implementiert worden. Die affine Arithmetik wurde ebenfalls mit ihrer Hilfe realisiert. Zur Lösung der linearen Optimierungsprobleme, die bei der Auswertung des Newton-Operators auftreten, wird ein externes Lösungsverfahren (LINDO Systems 2001) verwendet.

Die Netzwerk- und Modellgleichungen der im Folgenden beschriebenen Beispiele werden mit dem Symbolische-Analyse-Paket Analog Insydes (ITWM 2002) aus ihren Netzlisten in Verbindung mit den im Paket enthaltenen Modellen erzeugt. Die Gleichungen werden so weit wie möglich durch algebraische Umformungen vereinfacht. Falls Näherungen durchgeführt werden, werden diese bei der Beschreibung der Beispiele erläutert.

Die angegebenen Rechenzeiten wurden auf einem Doppel-Intel-Xeon-PC mit 2,4 GHz Taktfrequenz unter Nutzung eines Prozessors ermittelt.

4.2 Testschaltungen und Ergebnisse

Tabelle 4.1 gibt eine Übersicht über die in den nachfolgenden Abschnitten dargestellten Dimensionierungsbeispiele: Operationsverstärker, Differenzverstärker und Inverter. Die Übersicht zeigt, dass mit den Beispielen sowohl die CMOS- als auch die Bipolartechnologie abgedeckt werden. Alle drei Beispiele erfordern in der Eigenschaftsanalyse die Berechnung von Arbeitspunkten. Darüber hinaus werden bei der Dimensionierung des Operationsverstärkers das dynamische Klein- und Großsignalverhalten analysiert, letzteres unter Verwendung einer Analysegleichung. Eine

4 Implementierung und Ergebnisse

	Operations- verstärker	Differenz- verstärker	Inverter	
			I	II
MOS-Transistoren	×			
Bipolartransistoren		×	×	×
Arbeitspunkt- simulation	×	×	×	×
Statisches Großsignalverhalten		×		
Dynamisches Kleinsignalverhalten	×			
Dynamisches Großsignalverhalten	× ^a			
Betriebsbereich		×		
Prozesstoleranzen			×	×
Globales Flächenminimum	×	×		×
Worst-Case-Analyse		×	×	×
Entwurfzentrierung			×	

^a Verwendung einer Analysegleichung

Tabelle 4.1: Beispielübersicht

Dimensionierung mit Worst-Case-Analyse des statischen Großsignalverhaltens wird am Beispiel des Differenzverstärkers gezeigt.

Bei den Beispielen Operationsverstärker, Differenzverstärker und Inverter II wird eine flächenminimale Dimensionierung durchgeführt. Am Beispiel Inverter I wird eine Entwurfszentrierung unter Berücksichtigung von Herstellungsprozess toleranzen demonstriert.

Das Beispiel Inverter II dient dem Vergleich des Rechenaufwandes zwischen einer intervallarithmetischen Implementierung des Dimensionierungsverfahrens und der mit affiner Arithmetik. Die ersten drei Beispiele werden ausschließlich mit affiner Arithmetik berechnet.

Als Transistormodelle werden die im Werkzeug Analog Insydes implementierten Spice-Modelle wie im Folgenden angegeben verwendet. Die berechneten Ergebnisse werden jeweils durch Simulationen mit dem Simulator PSpice (Cadence 2003a) graphisch dargestellt und, soweit möglich, überprüft.

4.2.1 Operationsverstärker

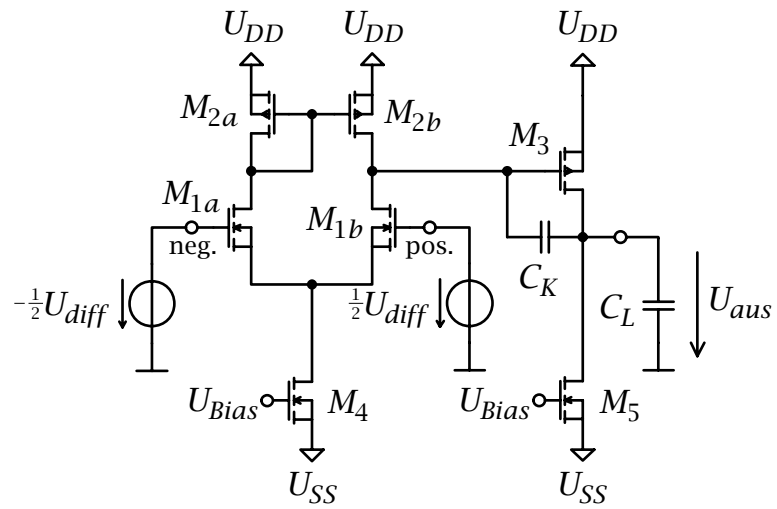
Die Dimensionierung eines Operationsverstärkers ist eine in der Praxis häufig durchzuführende Entwurfsaufgabe. Aus Kostengründen wird soweit möglich eine CMOS-Technologie verwendet. Die Topologie des in diesem Beispiel betrachteten CMOS-Operationsverstärkers ist in Abbildung 4.1(a) dargestellt. Die zweite Stufe enthält zur Sicherstellung der Stabilität bei Rückkopplung eine Kompensationskapazität C_K , deren Wirkung durch den Miller-Effekt verstärkt wird (Laker und Sansen 1994). Die Belastung des Ausganges wird durch die Lastkapazität C_L berücksichtigt.

Die Transistoren M_4 und M_5 dienen als Biasstromquellen. Zur Vereinfachung werden sie in diesem Beispiel durch ideale Stromquellen mit festen Werten ersetzt. Abbildung 4.1(b) zeigt die Topologie mit idealen Biasstromquellen.

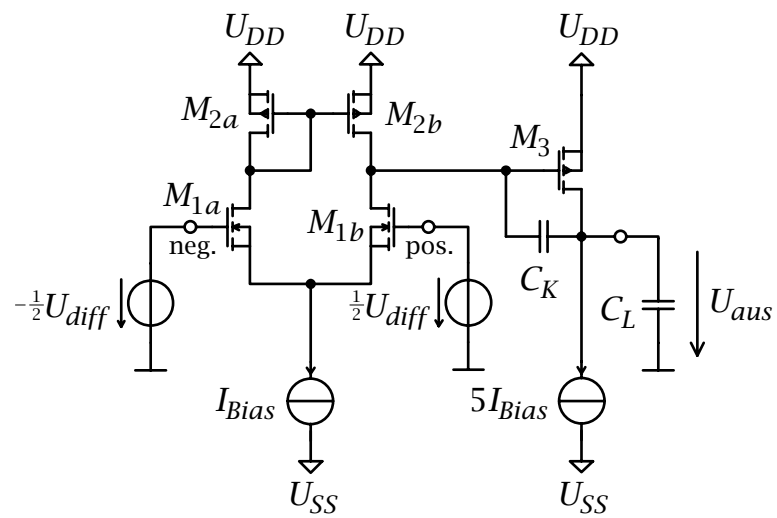
Tabelle 4.2 enthält die Spezifikation, nach der die Dimensionierung durchgeführt werden soll. Es werden sowohl das dynamische Kleinsignalverhalten mit Verstärkung, Phasenreserve und Transitfrequenz als auch das dynamische Großsignalverhalten mit Anstiegsgeschwindigkeit (slew rate) und die Layoutfläche spezifiziert. Das Schaltungseigenschaftstupel ist damit

$$\mathbf{e} = (\text{Verstärkung, Phasenreserve, Transitfrequenz, Anstiegsgeschwindigkeit, Layoutfläche}) . \quad (4.1)$$

4 Implementierung und Ergebnisse



(a) Unvereinfacht



(b) Bias-Stromquellen vereinfacht

Abbildung 4.1: CMOS-Operationsverstärker

4.2 Testschaltungen und Ergebnisse

Eigenschaft	Spezifikation
Nebenbedingungen	
Verstärkung	> 80 dB
Phasenreserve	> 60°
Transitfrequenz	> 1 MHz
Anstiegsgeschwindigkeit	> 5 V/μs
Optimierungsziel	
Fläche	Minimum (5 % Toleranz)
Prozessparameter	
Schwellenspannung VTO	1 V
Kanallängenmodulationsparameter LAMBDA	0,03 V ⁻¹
Steilheitsparameter KP	50 μA/V ² (NMOS) bzw. 17 μA/V ² (PMOS)
Gateoxiddicke TOX	42,5 nm
Betriebsparameter	
Versorgungsspannung	±2,5 V
Biasstrom I_{Bias}	20 μA
Lastkapazität C_L	10 pF
Entwurfparameter	
Kanalweiten	$W_{1a} = W_{1b} = W_1, W_{2a} = W_{2b} = \frac{1}{10} W_3 = W_2,$ $10 \mu\text{m} \leq W_1, W_2 \leq 100 \mu\text{m}$
Kanallängen	10 μm
Kompensationskapazität	$0,1 \text{ pF} \leq C_K \leq 100 \text{ pF}$

Tabelle 4.2: Spezifikation des Operationsverstärkers

4 Implementierung und Ergebnisse

Da das Dimensionierungsverfahren dieser Arbeit kein Analyseverfahren für dynamisches Großsignalverhalten beinhaltet, muss für die Anstiegsgeschwindigkeit eine Analysegleichung verwendet werden (Laker und Sansen 1994):

$$\text{Anstiegsgeschwindigkeit} = \frac{I_{Bias}}{C_K}. \quad (4.2)$$

Als Transistormodell wird das Spice-MOS-Level-1-Modell eingesetzt. Da die Spezifikation einen flächenminimalen Entwurf fordert, ergeben sich bei Verwendung dieses Modells für alle Transistoren minimale Kanallängen, die in diesem Beispiel als $10\ \mu\text{m}$ gegeben sind. Die verbleibenden Freiheitsgrade der Dimensionierung sind die Kanalweiten und die Kompensationskapazität C_K . Die Symmetrie der Differenzstufe erfordert die Gleichheit (matching) der Transistorpaare M_{1a} und M_{1b} sowie M_{2a} und M_{2b} . Die Kanalweiten dieser Transistoren sind daher ebenfalls paarweise gleich. Das Verhältnis der Kanalweiten von M_{2a} und M_{2b} zu der von M_3 muss dem der Biasströme durch diese Transistoren (vgl. Abbildung 4.1(b)) entsprechen, damit keine systematischer Offsetfehler entsteht. Es verbleiben daher als Freiheitsgrade die Kanalweiten W_1 und W_2 sowie die Kompensationskapazität C_K , so dass das Entwurfsparametertupel durch $\mathbf{p}_E = (W_1, W_2, C_K)$ gegeben ist.

Zur Beschleunigung der Wechselstromanalyse wird eine mit Hilfe des Werkzeuges Analog Insydes erzeugte Näherung der Übertragungsfunktion verwendet. Diese weist im Frequenzbereich bis 10 MHz einen gemittelten Amplitudenfehler von unter 1 % sowie einen gemittelten Phasenfehler von $0,3^\circ$ auf. Um sicherzustellen, dass alle Transistoren im Sättigungsbereich betrieben werden, werden dem Optimierungsproblem für jeden Transistor Nebenbedingungen der Form

$$0 \leq U_{GS} - V_{TO} \leq U_{DS} \quad (4.3)$$

hinzugefügt.

Als Suchraum werden die in Tabelle 4.2 aufgeführten Grenzen der Entwurfparameter verwendet. Die Ergebnisse der Dimensionierung sind in Tabelle 4.3 aufgeführt. Der Algorithmus garantiert im Rahmen der oben beschriebenen Modellierung die Einhaltung der Nebenbedingungen der Spezifikation innerhalb der berechneten Intervalle für die Entwurfparameter W_1 , W_2 und C_K . Außerdem wird gewährleistet, dass die Layoutfläche der berechneten Dimensionierung maximal um die spezifizierte Toleranz von 5 % größer als das globale Minimum ist. Die Berechnung benötigte die in Tabelle 4.3 genannte Anzahl an Newton-Operationen und Rechenzeit.

Größe	Ergebnis
$W_1 / \mu\text{m}$	[13,345; 13,355]
$W_2 / \mu\text{m}$	[52,22; 52,23]
C_K / pF	[3,9972; 3,9981]
Newton-Operationen	$3,3 \cdot 10^6$
Rechenzeit	4,5 d

Tabelle 4.3: Dimensionierungsergebnisse des Operationsverstärkers

Zur Veranschaulichung des Vorgehens des Dimensionierungsalgorithmus wird eine erneute Berechnung mit $C_K = 4 \text{ pF}$ durchgeführt, damit eine zweidimensionale Darstellung des Entwurfsparameterraumes P_E möglich ist. Die Ergebnisse für W_1 und W_2 entsprechen denen in Tabelle 4.3. Abbildung 4.2 zeigt den Entwurfsraum P_E mit der Menge P_{NF} der nicht falsifizierten Teilräume von P_E nach Ablauf des Dimensionierungsalgorithmus. Zusätzlich sind eine durch punktweise Berechnung gewonnene Abschätzung des Akzeptanzgebietes A_{P_E} und vier Höhenlinien der Zielfunktion f_u dargestellt. Das durch die Ergebnisintervalle für W_1 und W_2 gegebene Gebiet in P_E befindet sich im Zentrum des eingekreisten Bereiches. Die Ausdehnung dieses Gebietes selbst ist im zur Darstellung des Suchraumes erforderlichen Maßstab nicht erkennbar, wird hier aber auch nicht benötigt. Der Algorithmus hat die Suche entlang der Höhenlinien von f_u in Richtung größerer Werte durchgeführt. Dabei wurden so lange Teile von P_{NF} falsifiziert und entfernt, bis der Rand des Akzeptanzgebietes A_{P_E} erreicht wurde. Ein an der Innenseite der Berandung liegendes Teilgebiet, das die Abbruchbedingung 3.41 erfüllt, wurde verifiziert und als Lösung zurückgegeben. Für eine genauere Darstellung der Funktionsweise des Algorithmus vgl. Abschnitt 3.3.1.

Das Dimensionierungsergebnis in Tabelle 4.3 wird durch Simulationen mit dem Simulator PSpice auf Einhaltung der Nebenbedingungen überprüft. Als Entwurfparameter werden die Mittelpunkte der berechneten Intervalle verwendet. Die Simulationen werden auf Grundlage der sich aus Abbildung 4.1(b) ergebenden Netzliste ohne weitere Näherungen durchgeführt. Abbildung 4.3 zeigt das Ergebnis einer Wechselstromsimulation als Bodediagramm. Hieraus lassen sich die Verstärkung, die Phasenreserve und die Transitfrequenz durch Eigenschaftsextraktion bestimmen. Die Zahlenwerte sind in Tabelle 4.4 angegeben. Der Vergleich mit der Spezifikation zeigt, dass die Nebenbedingungen hinsichtlich dieser Eigenschaften eingehalten werden und die Verwendung der genäherten Übertragungsfunktion für die Dimensionierung zulässig ist.

4 Implementierung und Ergebnisse

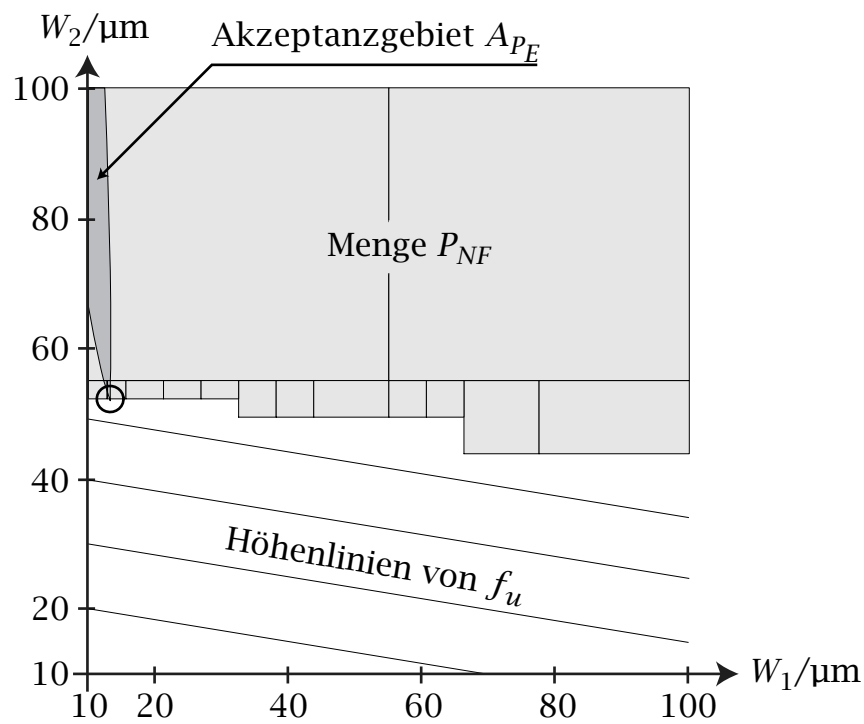


Abbildung 4.2: Entwurfparameterraum P_E nach Ablauf des Dimensionierungsalgorithmus für $C_K = 4 \text{ pF}$

Eigenschaft	Simulation	Spezifikation
Verstärkung	80,003 dB	> 80 dB
Phasenreserve	60,7°	> 60°
Transitfrequenz	1,39 MHz	> 1 MHz
Anstiegsgeschwindigkeit	4,75 V/μs	> 5 V/μs

Tabelle 4.4: Simulationsergebnisse des Operationsverstärkers

4.2 Testschaltungen und Ergebnisse

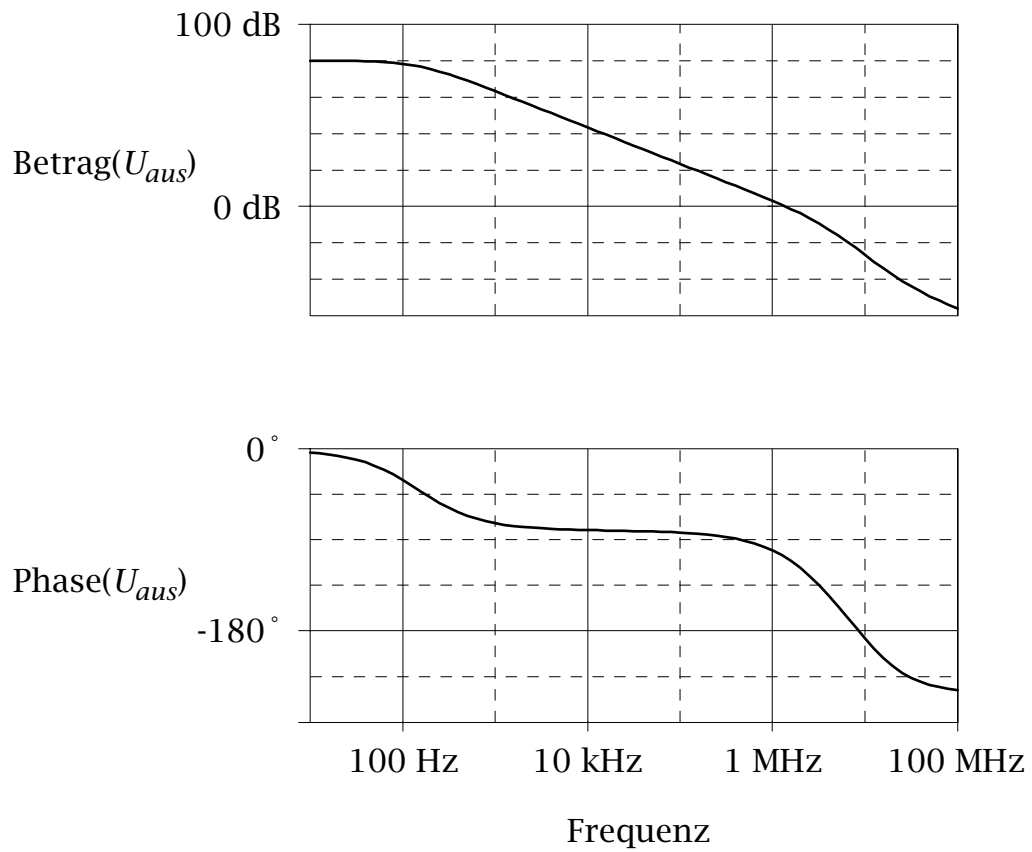


Abbildung 4.3: Wechselstromsimulation des Operationsverstärkers

4 Implementierung und Ergebnisse

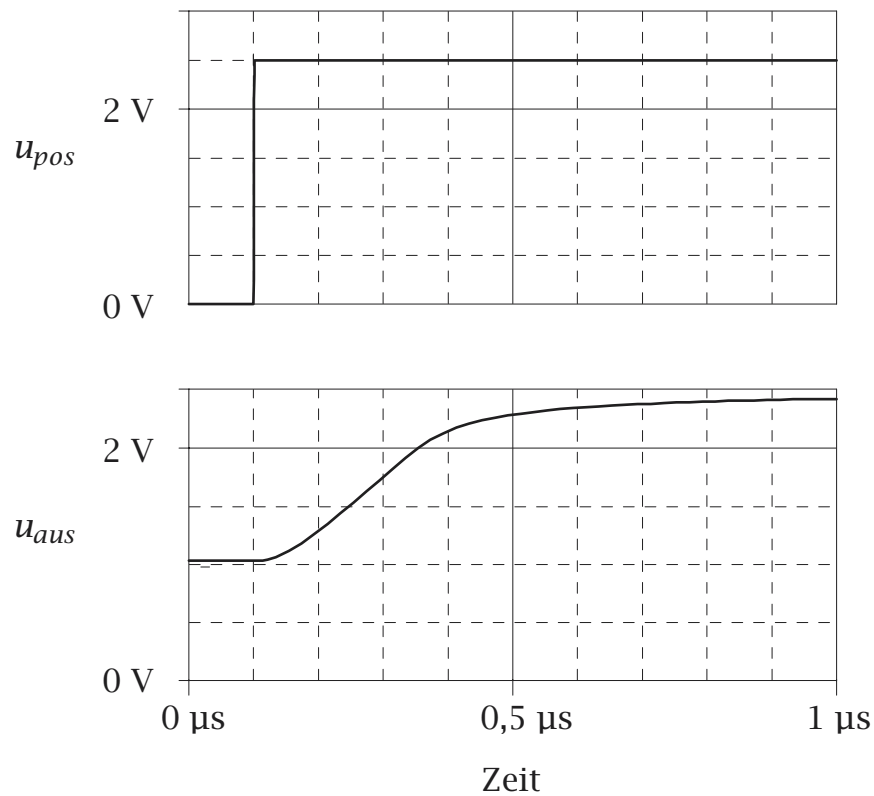


Abbildung 4.4: Transientensimulation des Operationsverstärkers

Zur Überprüfung der Anstiegsgeschwindigkeit wird eine Transientensimulation mit Erregung des nichtinvertierenden (positiven) Einganges des Operationsverstärkers mit einem Sprung von 0 V auf 2,5 V durchgeführt. Abbildung 4.4 zeigt den Zeitverlauf der Ausgangsspannung u_{aus} . Hieraus kann die Steigung der Flanke von u_{aus} als Anstiegsgeschwindigkeit zu $4,75 \text{ V}/\mu\text{s}$ bestimmt werden. Die Nebenbedingung der Spezifikation, die mindestens $5 \text{ V}/\mu\text{s}$ für die Anstiegsgeschwindigkeit verlangt, ist damit verletzt. Die Auswertung der Analysegleichung 4.2 für das in Tabelle 4.3 aufgeführte Intervall ergibt im Gegensatz zur Simulation eine Anstiegsgeschwindigkeit von $[5,002; 5,004] \text{ V}/\mu\text{s}$, die die Spezifikation in Übereinstimmung mit der durch den Algorithmus gelieferten Garantie einhält. Die Verletzung der Spezifikation in der Simulation ist damit auf die ungenaue Modellierung der Anstiegsgeschwindigkeit durch Gleichung 4.2 zurückzuführen, nicht jedoch auf einen Fehler des Dimensionierungsalgorithmus.

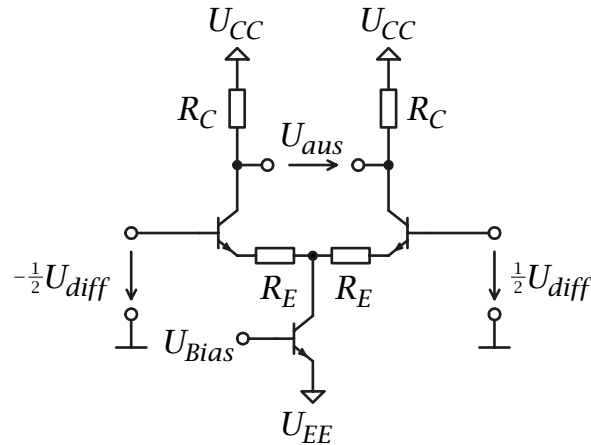


Abbildung 4.5: Differenzverstärker

4.2.2 Differenzverstärker

Als Beispiel einer Dimensionierung mit Worst-Case-Analyse wird die in Abbildung 4.5 dargestellte Differenzverstärkertopologie betrachtet. Durch die Emitttergegenkopplung soll die statische Großsignalübertragungskennlinie des Verstärkers linearisiert werden, so dass der Betrag des relativen Linearitätsfehlers

$$\Delta U_{rel} = \frac{U_{aus}}{V \cdot U_{diff}} - 1 \quad (4.4)$$

im gesamten spezifizierten Eingangsspannungsbereich unterhalb einer vorgegebenen Grenze bleibt. Hierbei ist V die Sollverstärkung. Die Überprüfung dieser Bedingung erfordert eine Worst-Case-Analyse der Eigenschaft ΔU_{rel} mit der Eingangsspannung als Betriebsparameter $p_B = U_{diff}$. Um die Singularität von Gleichung 4.4 bei $U_{diff} = 0$ zu vermeiden, muss ein endliches Intervall um 0 vom betrachteten Eingangsspannungsbereich ausgeklammert werden. Unter Einhaltung der o. a. Nebenbedingung für den Linearitätsfehler wird eine flächenminimale Dimensionierung verlangt. Die vollständige Spezifikation dieses Beispiels ist in Tabelle 4.5 dargestellt.

Das Schaltungseigenschaftstupel ist in diesem Beispiel

$$\mathbf{e} = (\Delta U_{rel}, \text{Layoutfläche}) . \quad (4.5)$$

Die Großsignalverstärkung ist keine explizite Komponente von \mathbf{e} , da sie bereits im Linearitätsfehler ΔU_{rel} enthalten ist (vgl. Gleichung 4.4). Die

4 Implementierung und Ergebnisse

Eigenschaft	Spezifikation
Nebenbedingungen	
Großsignalverstärkung	$V = 10$
Linearitätsfehler	$ \Delta U_{rel} < 5\%$
Optimierungsziel	
Fläche	Minimum (5 % Toleranz)
Prozessparameter	
Sättigungsstrom	$I_S = 10 \text{ fA}$
Vorwärtsstromverstärkung	$BF = 100$
Rückwärtsstromverstärkung	$BR = 1$
Betriebsparameter	
Eingangsspannung	$-0,4 \text{ V} \leq U_{diff} \leq -0,1 \text{ V}$ $\vee 0,1 \text{ V} \leq U_{diff} \leq 0,4 \text{ V}$
Versorgungsspannung	$\pm 5 \text{ V}$
Biasspannung	$U_{Bias} = 0,735 \text{ V}$
Entwurfsparameter	
Kollektorwiderstand	$100 \Omega \leq R_C \leq 100 \text{ k}\Omega$
Emitterwiderstand	$100 \Omega \leq R_E \leq 100 \text{ k}\Omega$

Tabelle 4.5: Spezifikation des Differenzverstärkers

Größe	Ergebnis
$R_C / \text{k}\Omega$	[23,91; 24,29]
$R_E / \text{k}\Omega$	[2,10; 2,12]
Newton-Operationen	$7,7 \cdot 10^5$
Rechenzeit	19,7 h

Tabelle 4.6: Dimensionierungsergebnisse des Differenzverstärkers

zu dimensionierenden Elemente sind die Widerstände R_C und R_E , so dass sich das Entwurfparametertupel $\mathbf{p}_E = (R_C, R_E)$ ergibt.

Die Berechnung des Ergebnisses erfolgte unter Verwendung des Spice-Bipolartransistormodells. Um zu gewährleisten, dass sich die Transistoren im vorwärts-aktiven Betriebsbereich befinden, wurde die Nebenbedingung

$$0 \leq U_{BE} \leq U_{CE} \quad (4.6)$$

für jeden Transistor hinzugefügt. Der Suchraum ist durch die in Tabelle 4.5 gegebenen Grenzen der Entwurfparameter gegeben. Tabelle 4.6 stellt die Ergebnisse der Dimensionierung dar. Der Algorithmus garantiert, dass die Nebenbedingungen für die ermittelten Intervalle der Entwurfparameter eingehalten werden und dass die Layoutfläche der berechneten Dimensionierung maximal 5% größer als die des globalen Minimums ist. Der Rechenaufwand in Newton-Operationen und Zeit ist ebenfalls in Tabelle 4.6 gegeben.

Zur Überprüfung der Einhaltung der Nebenbedingung wird eine Simulation mit dem Simulator PSpice durchgeführt. Als Entwurfparameter werden die Mittelpunkte der berechneten Intervalle in Tabelle 4.6 verwendet. Da mit dem Simulator keine garantierte Einschließung der Worst-Case-Eigenschaften berechnet werden kann (vgl. Abschnitt 2.2.4), wird statt dessen eine Gleichstrom-Transfer-Simulation durchgeführt. Als Ergebnis der Gleichstrom-Transfer-Simulation ist die Eigenschaft ΔU_{rel} in Abhängigkeit des Betriebsparameters U_{diff} in Abbildung 4.6 dargestellt. Alle vom Simulator berechneten Punkte, die in Abbildung 4.6 durch Markierungen hervorgehoben sind, halten die Nebenbedingung $|\Delta U_{rel}| < 5\%$ ein. Eine Garantie für den gesamten spezifizierten Bereich von U_{diff} ist dadurch jedoch nicht gegeben. Diese Garantie gibt aber der Dimensionierungsalgorithmus dieser Arbeit.

4 Implementierung und Ergebnisse

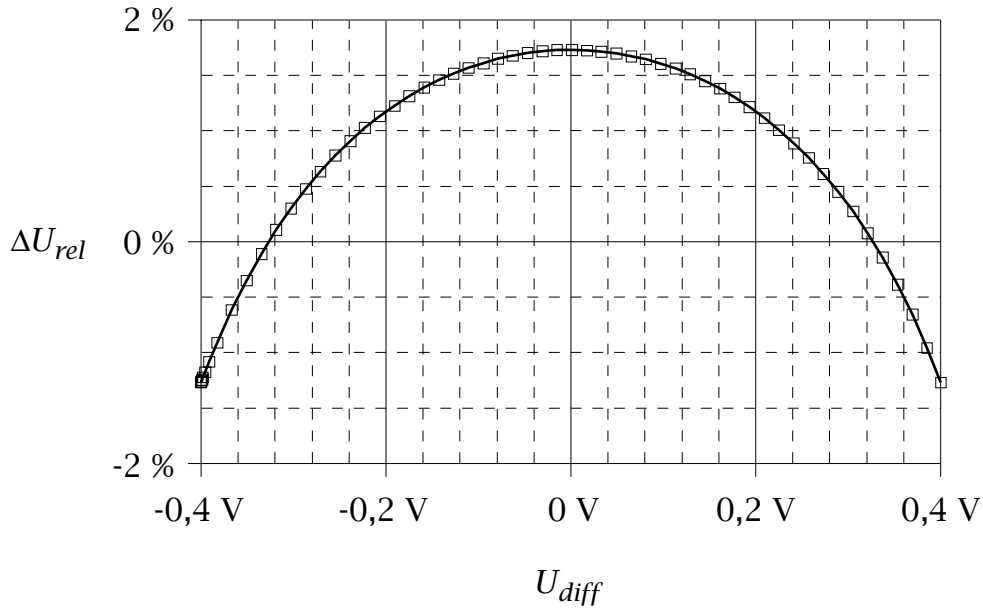


Abbildung 4.6: Gleichstrom-Transfer-Simulation des Differenzverstärkers

4.2.3 Inverter I (Entwurfszentrierung)

Dieses Beispiel dient zur Demonstration der Entwurfszentrierung mit dem in Abschnitt 3.1.2 beschriebenen Ansatz. Für den in Abbildung 4.7 dargestellten Inverter soll der Entwurfparameter Basiswiderstand $p_E = R_B$ so bestimmt werden, dass bei einer Eingangsspannung $U_{ein} = U_{Schwelle}$ die Schaltungseigenschaft Ausgangsspannung $e = U_{aus}$ um nicht mehr als ΔU von $U_{Schwelle}$ abweicht. Die unter dieser Bedingung zulässige Schwankung $p_{S,Rad} = BF_{Rad}$ des Prozessparameters $p_S = BF$ des Transistors um seinen Nominalwert $p_{nom,S} = BF_{nom}$ soll durch die Wahl von

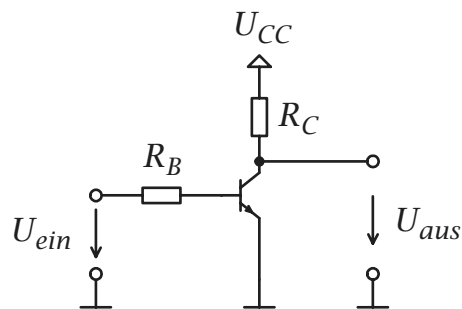


Abbildung 4.7: Inverter

Eigenschaft	Spezifikation
Nebenbedingungen	
Ausgangsspannung	$ U_{aus} - U_{Schwelle} \leq \Delta U$
Schwellspannung	$U_{Schwelle} = 2,5 \text{ V}$
Maximale Abweichung	$\Delta U = 0,1 \text{ V}$
Optimierungsziel	
Entwurfszentrierung: Toleranzradius der Vorwärtsverstärkung BF_{Rad}	Maximum (5 % Toleranz)
Prozessparameter	
Sättigungsstrom	$IS = 10 \text{ fA}$
Vorwärtsstromverstärkung	$BF_{nom} = 100$
Rückwärtsstromverstärkung	$BR = 1$
Betriebsparameter	
Eingangsspannung	$U_{ein} = U_{Schwelle}$
Versorgungsspannung	$U_{CC} = 5 \text{ V}$
Entwurfsparameter	
Kollektorwiderstand	$R_C = 1 \text{ k}\Omega$
Basiswiderstand	$100 \Omega \leq R_B \leq 100 \text{ k}\Omega$

Tabelle 4.7: Spezifikation des Inverters (Entwurfszentrierung)

R_B maximiert werden. Tabelle 4.7 fasst die Spezifikation zusammen und enthält Zahlenwerte für die Parameter und den Entwurfsraum.

Die Ergebnisse der Entwurfszentrierung sind in Tabelle 4.8 aufgeführt. Der Algorithmus garantiert, dass der mit dem berechneten R_B erzielte Wert für BF_{Rad} höchstens 5 % unter dem globalen Maximum liegt und dass die Nebenbedingungen der Spezifikation im berechneten Intervall für R_B bei Schwankungen von BF von maximal BF_{Rad} um den Nominalwert BF_{nom} eingehalten werden.

Die Einhaltung der Nebenbedingungen wurde für den Mittelpunkt des in Tabelle 4.8 aufgeführten Intervalls für R_B durch eine Simulation mit dem Simulator PSpice überprüft. Da mit dem Simulator keine garantierte Einschließung der Worst-Case-Eigenschaften berechnet werden kann, wird statt dessen eine Gleichstrom-Transfer-Simulation mit variablem BF durchgeführt. Als Ergebnis ist U_{aus} in Abhängigkeit von BF in Abbildung 4.8 dargestellt. Alle vom Simulator berechneten Punkte, die in Abbildung 4.8 durch Markierungen hervorgehoben sind, halten die Ne-

4 Implementierung und Ergebnisse

Größe	Ergebnis
$R_B / \text{k}\Omega$	[68,05; 68,09]
BF_{Rad}	4,0
Newton-Operationen	$5,7 \cdot 10^3$
Rechenzeit	9 min

Tabelle 4.8: Entwurfszentrierungsergebnisse

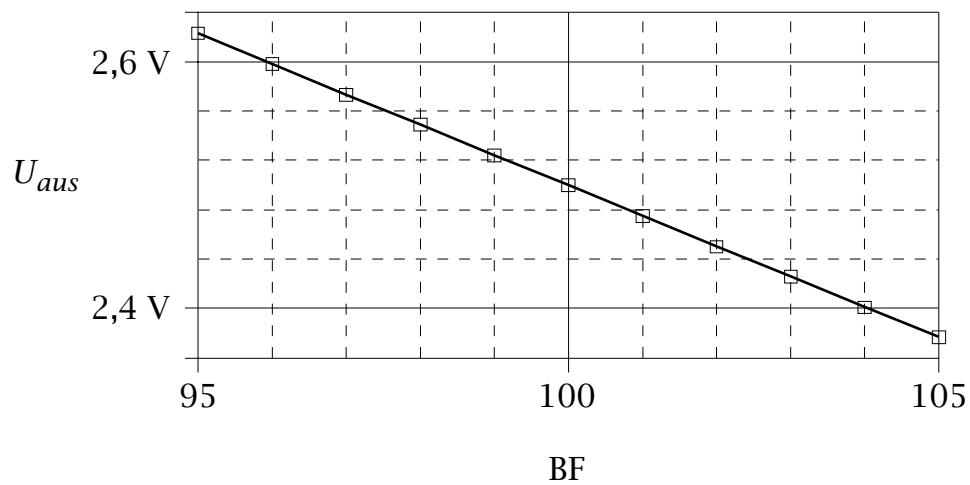


Abbildung 4.8: Simulation der Ausgangsspannung des Inverters (Entwurfszentrierung)

Eigenschaft	Spezifikation
Nebenbedingungen	
Ausgangsspannung	$ U_{aus} - U_{Schwelle} \leq \Delta U$
Schwellspannung	$U_{Schwelle} = 2,5 \text{ V}$
Maximale Abweichung	$\Delta U = 1 \text{ V}$
Optimierungsziel	
Fläche	Minimum (5% Toleranz)
Prozessparameter	
Sättigungsstrom	$I_S = 10 \text{ fA}$
Vorwärtsstromverstärkung	$90 \leq BF \leq 110$
Rückwärtsstromverstärkung	$BR = 1$
Betriebsparameter	
Eingangsspannung	$U_{ein} = U_{Schwelle}$
Versorgungsspannung	$U_{CC} = 5 \text{ V}$
Entwurfsparameter	
Kollektorwiderstand	$R_C = 1 \text{ k}\Omega$
Basiswiderstand	$100 \Omega \leq R_B \leq 100 \text{ k}\Omega$

Tabelle 4.9: Spezifikation des Inverters (flächenminimal)

benbedingung $|U_{aus} - 2,5 \text{ V}| \leq 0,1 \text{ V}$ für $|BF - 100| \leq 4$ ein. Eine Garantie für den gesamten Bereich von BF ist damit jedoch nicht gegeben. Diese Garantie gibt aber der Dimensionierungsalgorithmus dieser Arbeit.

4.2.4 Inverter II (flächenminimal)

Das vorige Beispiel wird in diesem Abschnitt so weit vereinfacht, dass es mit einer auf Intervallarithmetik eingeschränkten Variante des Dimensionierungsalgorithmus in praktikabler Laufzeit berechnet werden kann. Der benötigte Rechenaufwand wird anschließend mit dem einer Lösung mit affiner Arithmetik verglichen.

Statt der Entwurfszentrierung wird eine Flächenminimierung durchgeführt, so dass der Freiheitsgrad $p_{S, Rad} = BF_{Rad}$ entfällt. Für den Bereich des Prozessparameters $p_S = BF$ des Transistors sind feste Grenzen gegeben. Als Entwurfsparameter bleibt $p_E = R_B$ und als betrachtete Schaltungseigenschaft $e = U_{aus}$. Tabelle 4.9 fasst die Dimensionierungsaufgabe zusammen.

Die Ergebnisse der Dimensionierung mit Intervallarithmetik und mit

4 Implementierung und Ergebnisse

Größe	Ergebnis	
	IA ^a	AA ^b
$R_B / k\Omega$	[62,35; 62,53]	[62,35; 62,53]
Einschließungstests (3.80)	7761	—
Newton-Operationen	—	617
Rechenzeit	52 s	56 s

^aIntervallarithmetik (natürliche Erweiterung und Mittelwertsform)

^bAffine Arithmetik

Tabelle 4.10: Dimensionierungsergebnisse des Inverters

affiner Arithmetik sind in Tabelle 4.10 dargestellt. Es zeigt sich, dass die Anzahl der Auswertungen des Einschließungstests 3.80 bei Verwendung von Intervallarithmetik mit natürlicher Intervallerweiterung und Mittelwertsform deutlich höher als die Anzahl der Newton-Operationen bei affiner Arithmetik ist.¹ Dies ist eine Bestätigung der Ergebnisse des in Abschnitt 3.2.3 dargestellten Vergleiches von Intervallarithmetik und affiner Arithmetik.

Entsprechend der in Abschnitt 3.3.4 hergeleiteten Abschätzungen der Laufzeitkomplexität des Dimensionierungsalgorithmus wächst die Laufzeit des Dimensionierungsalgorithmus mit Intervallarithmetik bei Verwendung der natürlichen Intervallarithmetik exponentiell im Verhältnis zur Laufzeit des Algorithmus mit affiner Arithmetik. Die Praxis zeigt, dass dies auch bei Verwendung der Mittelwertsform trotz ihrer quadratischen Konvergenzordnung gilt. Der Versuch, das komplexere Beispiel in Abschnitt 4.2.3 mit Intervallarithmetik, Mittelwertsform und natürlicher Intervallerweiterung zu berechnen, wurde nach 2,5 Wochen Rechenzeit ohne Ergebnis abgebrochen. Aus den in Abschnitt 3.2.1 genannten Gründen stellt sich die quadratische Konvergenz der Mittelwertsform bei praktischen Problemen nicht ein.

¹Dass sich in diesem Beispiel trotzdem nahezu gleiche Rechenzeiten ergeben, ist darauf zurückzuführen, dass die Intervallarithmetik im Gegensatz zur affinen Arithmetik Bestandteil des Mathematica-Systems ist. Intervallarithmetische Operationen müssen daher nicht durch interpretierte Programmteile ausgeführt werden, sondern liegen in kompilierter Form vor und sind dadurch um einen konstanten Faktor schneller. Infolge der höheren Laufzeitkomplexität wächst die Rechenzeit bei Intervallarithmetik aber deutlich stärker als bei affiner Arithmetik.

5 Einordnung und Bewertung

Das in dieser Arbeit vorgestellte Dimensionierungsverfahren wird abschließend den in Abschnitt 2.3.6 beschriebenen optimierungsbasierten Verfahren gegenübergestellt. Dazu werden die Kriterien aus Tabelle 2.5 verwendet. Das Ergebnis ist in Tabelle 5.1 dargestellt. Alle aufgeführten Verfahren behandeln durch den optimierungsbasierten Ansatz das Dimensionierungsproblem in der allgemeinen Form, die sich aus der Spezifikation ergibt: als (unterbestimmtes) Optimierungsproblem mit Nebenbedingungen. Falls mehrere Optimierungsziele vorliegen, müssen diese mit Gewichten versehen werden. Die Lösung des Problems ist das globale Optimum. Nur das Verfahren dieser Arbeit und das auf geometrischer Optimierung aufbauende garantieren, dass sie das globale Optimum finden oder andernfalls die Lösbarkeit des Problems widerlegen. Ansätze mit lokalen Optimierungsverfahren terminieren abhängig vom vorgegebenden Startpunkt in einem lokalen Optimum, das meist nicht mit dem globalen Optimum identisch ist. Das Verfahren mit Simulated-Annealing-Optimierer ist im Gegensatz zu den bisher genannten nicht-deterministisch. Es findet das globale Optimum mit einer von der Anzahl der durchgeführten Iterationen abhängigen Wahrscheinlichkeit, die bei endlicher Laufzeit immer unter 1 bleibt (vgl. Abschnitt 2.3.6), und bietet daher keine Garantie. Für die automatische Synthese von Analogschaltungen ist es entscheidend, dass kein Startpunkt benötigt wird, der von einem Schaltungsentwickler vorgegeben werden muss. Daher eignen sich unter diesem Aspekt nur das Verfahren dieser Arbeit sowie die auf Simulated Annealing und geometrischer Optimierung aufbauenden Verfahren für die automatische Synthese.

Der entscheidende Vorteil des Ansatzes dieser Arbeit gegenüber dem auf geometrischer Optimierung aufbauenden liegt darin, dass er ebenso wie dieser globale Konvergenz garantiert (s. o.), die Struktur des Optimierungsproblems und der Modellgleichungen aber nicht einschränkt. Die geometrische Optimierung erfordert u. a., dass Modellgleichungen, Nebenbedingungen und Zielfunktionen aus posynomialen Ausdrücken gebildet werden müssen. Es ist nicht möglich, Maschen- oder Knotengleichungen aufzustellen, weil dieses gegen die Struktur eines geometrischen Optimierungsproblems verstoßen würde. Die Lösung von Di-

5 Einordnung und Bewertung

	Lokale Optimierung ^a	Simulated Annealing ^a	Geometrische Optimierung ^a	Formale Methoden
Lösungsverfahren				
Methode	Iterativ	Iterativ	Iterativ	Iterativ
Optimierungsziel	Beliebig	Beliebig	Posynomiale Gleichung	Nichtlineare Gleichung
Nebenbedingungen	Beliebig	Beliebig	Geometrisches Programm	Nichtlineare Ungleichungen
Unterbestimmte Probleme	Möglich	Möglich	Möglich	Möglich
Konvergenz	Lokal	Quasi-global	Global	Global
Unlösbarkeitserkennung	Nein	Nein	Ja	Ja
Anwendungsgebiet	Nach-optimierung	Synthese	Synthese	Synthese
Analyse				
Analysewerkzeug	Beliebig	Beliebig	Posynomiale Gleichungen	Nichtlineares Gleichungssystem
Maximale Genauigkeit	Hoch	Hoch	Mittel	Hoch
Worst-Case-Analyse	Möglich	Möglich	Parameterdurchlauf	Ja
Manueller Vorbereitungs-aufwand	Niedrig	Niedrig	Hoch	Mittel
Skalierbarkeit				
Laufzeitkomplexität	Polynomial	Vermutlich exponentiell	Polynomial	Exponentiell
Maximale Problemgröße	Mittel	Mittel	Groß	Klein
Dabei: Rechenzeitbedarf	Mittel	Hoch	Mittel	Hoch
Formale Garantie für				
Globales Optimum	Nein	Nein	Ja	Ja
Einhaltung der Nebenbedingungen bei Parameterschwankungen	Möglich	Möglich	Nein	Ja

^aZu Beschreibungen der Verfahren und Literaturverweisen zu Implementierungen vgl. Abschnitt 2.3.6.

Tabelle 5.1: Vergleich der optimierungsbasierten Dimensionierungsverfahren

mensionierungsproblemen mit diesem Ansatz erfordert daher immer eine aufwändige Modellierung des Schaltungsverhaltens durch Analysegleichungen. Wegen der Einschränkung auf posynomiale Ausdrücke sind dabei Näherungen erforderlich (vgl. Abschnitt 2.3.6). Der Vorbereitungsaufwand zur Lösung einer Dimensionierungsaufgabe mit diesem Verfahren ist daher als hoch einzustufen. Im Gegensatz dazu lässt das Verfahren dieser Arbeit nichtlineare Gleichungssysteme in allgemeiner Form zu, so dass beliebige Bauelementemodelle und Netzwerkgleichungen verwendet werden können und eine entsprechende Genauigkeit erreicht wird. Gemeinsamer Nachteil des Verfahrens dieser Arbeit und des auf geometrischer Optimierung aufbauenden ist jedoch, dass zur Analyse kein beliebiger Simulator eingesetzt werden kann. Obwohl das Verfahren dieser Arbeit keine Einschränkung der Struktur der verwendbaren Gleichungen erfordert, bedeutet dies dennoch, dass alle Analyseverfahren in Gebietsarithmetik implementiert worden sein müssen. In dieser Arbeit sind bisher nur Analyseverfahren für das statische Großsignal- und das dynamische Kleinsignalverhalten entwickelt worden. Für andere Analysearten müssen Analysegleichungen verwendet werden. Daher ergibt sich ein als mittel einzustufender Vorbereitungsaufwand. Der Ansatz ist aber grundsätzlich für alle Analysearten geeignet. Diese sollten im Rahmen einer Weiterentwicklung hinzugefügt werden, um die Anwendungsmöglichkeiten des Verfahrens zu erweitern.

Ein zweiter wesentlicher Vorteil des Verfahrens dieser Arbeit ist, dass es im Gegensatz zu den anderen aufgeführten Verfahren Betriebs- und Prozessparameterschwankungen durch eine garantierte Worst-Case-Analyse berücksichtigt. Es berechnet garantierte obere und untere Schranken des Schaltungsverhaltens bei Parameterschwankungen und verwendet diese bei der Suche nach dem Dimensionierungsergebnis. Dadurch wird gewährleistet, dass die Spezifikation auch bei Abweichungen vom Nominalpunkt eingehalten wird. Während der Ansatz mit geometrischer Optimierung überhaupt keine Möglichkeit bietet, Parameterschwankungen zu berücksichtigen, müsste bei den Verfahren mit lokaler Optimierung oder Simulated Annealing ein zusätzliches Worst-Case-Analyseverfahren eingesetzt werden, das die Eigenschaften des Verfahrens dieser Arbeit aufweist. Ein Simulator nach Spice-Muster bietet keine Garantie bei der Worst-Case-Analyse (vgl. Abschnitt 2.2.6) und sollte daher nicht verwendet werden. Statt dessen müsste zur Analyse ein Verifikationswerkzeug eingesetzt werden (vgl. Abschnitt 2.2.9). Dadurch würde ein Vorteil dieser Verfahren, die Einsetzbarkeit eines Simulators, entfallen. Ferner würde sich die Laufzeitkomplexität stark erhöhen, da bisher vorgestellte Verifikationswerkzeuge für Analogschaltungen expo-

5 Einordnung und Bewertung

nentielles Laufzeitverhalten zeigen (Hedrich 1997).

Das Verfahren dieser Arbeit weist im Vergleich mit den anderen in Tabelle 5.1 genannten zwei Nachteile auf: die Verwendung gebietsarithmetischer Analyseverfahren (s. o.) und die exponentielle Laufzeitkomplexität. Aus letzterer folgt für Dimensionierungsprobleme mit vielen Freiheitsgraden ein hoher Rechenaufwand. Dies führt in Verbindung mit der für die Prototypentwicklung gewählten interpretierenden und damit sehr langsamen Programmierumgebung zu einer starken Einschränkung der handhabbaren Problemgröße durch die resultierenden Rechenzeiten (vgl. Kapitel 4). Jedoch erfordern die anderen in Tabelle 5.1 aufgeführten Ansätze entweder ebenfalls einen hohen Rechenaufwand, bieten dabei aber keine garantierte Konvergenz zum globalen Optimum (Simulated Annealing), oder sie gewährleisten zwar das Finden des globalen Optimums bei kurzen Rechenzeiten, erfordern aber einen hohen Modellierungsaufwand zur Vorbereitung (geometrische Optimierung). Zudem ist das Verfahren dieser Arbeit das einzige, das eine Worst-Case-Analyse mit garantierten Ergebnissen vorsieht. Die Implementierung in einer leistungsfähigeren Umgebung mit einer nicht-interpretierten Sprache in Verbindung mit einer Parallelisierung würde die Behandlung größerer Probleme ermöglichen.

Ein zusammenfassender Vergleich ergibt, dass der Ansatz dieser Arbeit durch den Einsatz formaler Methoden als einziger Ergebnisse mit globaler Optimalität und gleichzeitiger Einhaltung der Spezifikation bei Parameterschwankungen garantiert. Er schränkt die Struktur der zur Analyse verwendbaren Gleichungen und die der Nebenbedingungen nicht ein. Da der Algorithmus keinen Startwert benötigt, ist das Verfahren für den Einsatz im Rahmen einer vollautomatischen Synthese von Analogschaltungen geeignet. Diesen Vorteilen stehen die exponentielle Laufzeitkomplexität des Ansatzes und die Verwendung gebietsarithmetischer Analyseverfahren als Nachteil gegenüber.

6 Zusammenfassung

In dieser Arbeit ist ein neuartiges Dimensionierungsverfahren mit formalen Methoden vorgestellt worden. Zunächst wurde im Anschluss an eine Einführung in die Verfahren des Analogschaltungsentwurfs der Stand der Technik auf dem Gebiet der automatischen Dimensionierung dargestellt und bewertet. Auf der Grundlage des heute vorherrschenden optimierungsbasierten Ansatzes wurde ein Algorithmus mit formalen Methoden auf der Basis affiner Arithmetik entwickelt. Es wurde gezeigt, dass dieser sowohl für Dimensionierungsprobleme mit beliebigen Zielfunktionen als auch zur Entwurfszentrierung geeignet ist. Der Algorithmus wurde hinsichtlich seiner Laufzeitkomplexität untersucht. Mit Hilfe einer Implementierung als Prototyp wurde die Funktion des Algorithmus anhand von Beispielp Problemen demonstriert. Abschließend wurde er bezüglich seiner Eigenschaften verschiedenen optimierungsbasierten Dimensionierungsverfahren, die dem Stand der Technik entsprechen, gegenübergestellt und bewertet.

Mit dem vorgestellten Verfahren steht erstmals ein Werkzeug zur Verfügung, das automatisch eine auch bei Parameterschwankungen garantiert korrekte, global optimale Lösung des Dimensionierungsproblems liefert und damit für die vollautomatische Synthese von Analogschaltungen geeignet ist.

6 Zusammenfassung

Literaturverzeichnis

Accellera 2002 ACCELLERA: *Verilog Analog/Mixed Signal version 2.0*.
URL <http://www.accellera.org/poform.pdf>, 2002

Antreich u. a. 1994 ANTREICH, Kurt J. ; GRÄB, Helmut E. ; WIESNER, Claudia U.: Circuit Analysis and Optimization Driven by Worst-Case Distances. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 13 (1994), Januar, Nr. 1, S. 57-71

Basaran u. a. 1993 BASARAN, Bulent ; RUTENBAR, Rob A. ; CARLEY, L. R.: Latchup-Aware Placement and Parasitic-Bounded Routing of Custom Analog Cells. In: *IEEE/ACM International Conference on Computer-Aided Design*, November 1993, S. 415-421. - ISBN 0-8186-4490-7

Beenker u. a. 1993 BEENKER, G. ; CONWAY, J. ; SCHROOTEN, G. ; SLENTER, A.: *Analog Circuit Design*. Kap. 15, S. 347-367, Kluwer Academic Publishers, 1993. - ISBN 0-7923-8400-8

Berger und Fischer-Binder 2000 BERGER, Michael ; FISCHER-BINDER, Jörg-Oliver: Software Tools for Analogue Circuit Simulation. In: *Elektronik* 49 (2000), Dezember, Nr. 26, S. 58-62

Box u. a. 1978 BOX, George E. P. ; HUNTER, William G. ; HUNTER, J. S. ; HUNTER, William G.: *Statistics for Experimenters: An Introduction to Design, Analysis, and Model Building*. John Wiley & Sons, 1978. - ISBN 0471093157

Cadence 2003a CADENCE DESIGN SYSTEMS, INC.: *PSpice*. 2003. - URL <http://www.cadencepcb.com/products/pspice>

Cadence 2003b CADENCE DESIGN SYSTEMS, INC.: *PSpice A/D User's Guide*. URL http://www.cadencepcb.com/pdf/pspicestudentver/psp_pdf.zip, 2003

- Cadence 2003c** CADENCE DESIGN SYSTEMS, INC.: *Spectre Circuit Simulator*. 2003. - URL <http://www.cadence.com/products/spectre.html>
- Cohn u. a. 1991** COHN, John M. ; GARROD, David J. ; RUTENBAR, Rob A. ; CARLEY, L. R.: Techniques for Simultaneous Placement and Routing of Custom Analog Cells in KOAN/ANAGRAM II. In: *IEEE/ACM International Conference on Computer-Aided Design*, November 1991, S. 394-397. - ISBN 0-8186-2157-5
- Daems u. a. 2003** DAEMS, Walter ; GIELEN, Georges G. E. ; SANSEN, Willy: Simulation-Based Generation of Posynomial Performance Models for the Sizing of Analog Integrated Circuits. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 22 (2003), Mai, Nr. 5, S. 517-534. - ISSN 0278-0070
- Daems u. a. 2001** DAEMS, Walter ; GIELEN, Georges G. E. ; SANSEN, Willy M. C.: Simulation-based Automatic Generation of Signomial and Posynomial Performance Models for Analog Integrated Circuit Sizing. In: *IEEE/ACM International Conference on Computer-Aided Design*, November 2001, S. 70-74. - ISBN 0-7803-7249-2
- Degrauwe u. a. 1987** DEGRAUWE, Marc G. ; NYS, Olivier ; DIJKSTRA, Evert u. a.: IDAC: An Interactive Design Tool for Analog CMOS Circuits. In: *IEEE Journal on Solid-State Circuits* 22 (1987), Dezember, Nr. 6, S. 1106-1116. - ISSN 0018-9200
- Doboli u. a. 1999** DOBOLI, Alex ; NUNEZ-ALDANA, Adrian ; DHANWADA, Nagu ; GANESAN, Sree ; VEMURI, Ranga: Behavioral Synthesis of Analog Systems Using Two-Layered Design Space Exploration. In: *Proceedings of the 36th ACM/IEEE Design Automation Conference*, Juni 1999, S. 951-957. - ISBN 0-7803-5560-1
- El-Turky und Perry 1989** EL-TURKY, Fatehy ; PERRY, Elizabeth E.: BLADES: An Artificial Intelligence Approach to Analog Circuit Design. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 8 (1989), Juni, Nr. 6, S. 680-692. - ISSN 0278-0070
- Femia und Spagnuolo 2000** FEMIA, Nicola ; SPAGNUOLO, Giovanni: True Worst-Case Circuit Tolerance Analysis Using Genetic Algorithms and Affine Arithmetic. In: *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications* 47 (2000), September, Nr. 9, S. 1285-1296. - ISSN 1057-7122

- Fernández u.a. 1998** FERNÁNDEZ, Francisco V. (Hrsg.) ; RODRIGUEZ-VÁZQUEZ, Angel (Hrsg.) ; HUERTAS, José L. (Hrsg.) ; GIELEN, Georges G. E. (Hrsg.): *Symbolic Analysis Techniques*. IEEE Press, 1998
- de Figueiredo und Stolfi 1997** FIGUEIREDO, Luiz H. de ; STOLFI, Jorge: *Self-Validated Numerical Methods and Applications*. IMPA/CNPq, Rio de Janeiro, Brazil, 1997 (Brazilian Mathematics Colloquium monographs). - URL <http://www.dcc.unicamp.br/~stolfi/EXPORT/papers/by-tag/fig-sto-97-iaaa.ps.gz>
- Gajski und Kuhn 1983** GAJSKI, Daniel ; KUHN, Robert H.: Guest Editors' Introduction-New VLSI Tools. In: *IEEE Computer* 16 (1983), Dezember, Nr. 12, S. 11-14. - ISSN 0018-9162
- Garey und Johnson 1979** GAREY, Michael R. ; JOHNSON, David S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979. - ISBN 0-7167-1045-5
- Gielen und Sansen 1991** GIELEN, Georges G. E. ; SANSEN, Willy M. C.: *Symbolic Analysis for Automated Design of Analog Integrated Circuits*. Kluwer Academic Publishers, 1991. - ISBN 0-7923-9161-6
- Golub und Loan 1996** GOLUB, Gene H. ; LOAN, Charles F. V.: *Matrix Computations*. 3rd Edition. Johns Hopkins University Press, 1996
- Hansen 1980** HANSEN, E.: Global Optimization Using Interval Analysis-The Multi-Dimensional Case. In: *Numerische Mathematik* 34 (1980), Nr. 1, S. 247-270. - ISSN 0029-599X
- Hansen 1975** HANSEN, Eldon R.: A Generalized Interval Arithmetic. In: *Interval Mathematics* Bd. 29. Springer Verlag, 1975, S. 7-18
- Harjani u.a. 1989** HARJANI, Ramesh ; RUTENBAR, Rob. A. ; CARLEY, L. R.: OASYS: A Framework for Analog Circuit Synthesis. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 8 (1989), Dezember, Nr. 12, S. 1247-1266. - ISSN 0278-0070
- Hartong u.a. 2002** HARTONG, Walter ; HEDRICH, Lars ; BARKE, Erich: Model Checking Algorithms for Analog Verification. In: *Proceedings of the 39th IEEE/ACM Design Automation Conference*, Juni 2002, S. 542-547. - ISBN 1-58113-461-4

- Hedrich 1997** HEDRICH, Lars: *Ansätze zur formalen Verifikation analoger Schaltungen*. VDI Verlag, 1997. - ISBN 3-18-325720-3
- Hedrich und Barke 1998** HEDRICH, Lars ; BARKE, Erich: A Formal Approach to Verification of Linear Analog Circuits with Parameter Tolerances. In: *Design, Automation and Test in Europe*, Februar 1998, S. 649-654. - ISBN 0-8186-8359-7
- Hennig und Sommer 1995** HENNIG, Eckhard ; SOMMER, Ralf: Application of Computer Algebra Methods to Analog Circuit Sizing. In: *European Conference on Circuit Theory and Design*, August 1995
- Hosticka u. a. 1994** HOSTICKA, Bedrich J. ; BROCKHERDE, Werner ; KLINKE, Roland ; KOKOZINSKI, Rainer: Design Methodology for Analog Monolithic Circuits. In: *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications* 41 (1994), Mai, Nr. 5, S. 387-394. - ISSN 1057-7122
- IEEE 1999** IEEE: *1076.1-1999 IEEE Standard VHDL Analog and Mixed-Signal Extensions*. 1999. - ISBN 0-7381-1641-6
- ITWM 2002** FRAUENHOFER INSTITUT FÜR TECHNO- UND WIRTSCHAFTSMATHEMATIK: *Analog Insydes Release 2.0*. 2002. - URL <http://www.analog-insydes.de>
- Kampe und Scarbata 1993** KAMPE, Jürgen ; SCARBATA, Gerd: Automatisierbare Synthesemethoden für analoge Systemkomponenten. In: *6. E.I.S.-Workshop*, November 1993
- Koh u. a. 1990** KOH, Han Y. ; SÉQUIN, Carlo H. ; GRAY, Paul R.: OPASYN: A Compiler for CMOS Operational Amplifiers. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 9 (1990), Februar, Nr. 2, S. 113-125. - ISSN 0278-0070
- Krasnicki u. a. 1999** KRASNICKI, Michael ; PHELPS, Rodney ; RUTENBAR, Rob. A. ; CARLEY, L. R.: MAELSTROM: Efficient Simulation-Based Synthesis for Custom Analog Cells. In: *Proceedings of the 36th ACM/IEEE Design Automation Conference*, Juni 1999, S. 945-950. - ISBN 0-7803-5560-1
- Kruiskamp und Leenaerts 1995** KRUISKAMP, Wim ; LEENAERTS, Domine: DARWIN: CMOS Opamp Synthesis by Means of a Genetic Algorithm. In: *Proceedings of the 23rd ACM/IEEE Design Automation Conference*, 1995, S. 433-438. - ISBN 0-8186-0702-5

- Laker und Sansen 1994** LAKER, Kenneth R. ; SANSEN, Willy M. C.: *Design of Analog Integrated Circuits and Systems*. McGraw-Hill Inc., 1994. - ISBN 0-07-113458-1
- Leenaerts 1990** LEENAERTS, D. M. W.: Application of Interval Analysis for Circuit Design. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 37 (1990), Juni, Nr. 6, S. 803-807. - ISSN 0278-0070
- Lemke u. a. 2002** LEMKE, Andreas ; HEDRICH, Lars ; BARKE, Erich: Analog Circuit Sizing Based on Formal Methods Using Affine Arithmetic. In: *IEEE/ACM International Conference on Computer-Aided Design*, November 2002, S. 486-489. - ISBN 0-7803-7607-2
- Lemke u. a. 2003** LEMKE, Andreas ; HEDRICH, Lars ; BARKE, Erich: Dimensionierung analoger Schaltungen mit formalen Methoden. In: *7. ITG/GMM-Diskussionssitzung Entwicklung von Anlogschaltungen mit CAE-Methoden*, September 2003, S. 135-140. - ISBN 3-8007-2778-1
- LINDO Systems 2001** LINDO SYSTEMS, INC.: *LINDO API 2.0*. 2001. - URL <http://www.lindo.com>
- del Mar Hershenson 2002** MAR HERSHENSON, Maria del: Design of Pipeline Analog-to-Digital Converters via Geometric Programming. In: *IEEE/ACM International Conference on Computer-Aided Design*, November 2002, S. 317-324. - ISBN 0-7803-7607-2
- del Mar Hershenson u. a. 2001** MAR HERSHENSON, Maria del ; BOYD, Stephen P. ; LEE, Thomas H.: Optimal Design of a CMOS Op-Amp via Geometric Programming. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 20 (2001), Januar, Nr. 1, S. 1-21. - ISSN 0278-0070
- McCalla 1988** MCCALLA, William J.: *Fundamentals of Computer-Aided Circuit Simulation*. Kluwer Academic Publishers, 1988. - ISBN 0-89838-248-3
- Medeiro Hidalgo u. a. 1994** MEDEIRO HIDALGO, Fernando ; FERNÁNDEZ FERNÁNDEZ, Francisco V. ; DOMÍNGUEZ CASTRO, Rafael ; RODRÍGUEZ VÁZQUEZ, Angel: A Statistical Optimization-Based Approach for Automated Sizing of Analog Cells. In: *IEEE/ACM International Conference on Computer-Aided Design*, November 1994, S. 594-597

- Mentor 2003** MENTOR GRAPHICS CORPORATION: *ADVance MS (ADMS)*. 2003. - URL <http://www.mentor.com/ams/adms.html>
- Moore 1966** MOORE, Ramon E.: *Interval Analysis*. Prentice-Hall, 1966
- Muller und Kamins 1977** MULLER, Richard S. ; KAMINS, Theodore I.: *Device Electronics for Integrated Circuits*. John Wiley & Sons, 1977. - ISBN 0-4718-8758-7
- Nagel 1975** NAGEL, Laurence W.: SPICE2: A Computer Program to Simulate Semiconductor Circuits. In: *Memorandum ERL-M250*. University of California, Berkeley, Mai 1975
- Neumaier 1990** NEUMAIER, A.: *Interval Methods for Systems of Equations*. Cambridge University Press, 1990. - ISBN 052133196X
- Ning u. a. 1991** NING, Zhen-Qiu ; MOUTHAN, Ton ; WALLINGA, Hans: SEAS: A Simulated Evolution Approach for Analog Circuit Synthesis. In: *IEEE Custom Integrated Circuits Conference*, Mai 1991, S. 5.2-1-5.2-4
- Nocedal und Wright 1999** NOCEDAL, Jorge ; WRIGHT, Stephen J.: *Numerical Optimization*. Springer, 1999. - ISBN 0-378-98793-2
- Nye u. a. 1988** NYE, William ; RILEY, David C. ; SANGIOVANNI-VINCENTELLI, Alberto ; TITS, André L.: DELIGHT.SPICE: An Optimization-Based System for the Design of Integrated Circuits. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 7 (1988), April, Nr. 4, S. 501-519. - ISSN 0278-0070
- Ochotta u. a. 1996** OCHOTTA, Emil S. ; MUKHERJEE, Tamal ; RUTENBAR, Rob. A. ; CARLEY, L. R.: ASTRX/OBLX: Tools for Rapid Synthesis of High-Performance Analog Circuits. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 15 (1996), März, Nr. 3, S. 273-294. - ISSN 0278-0070
- Ochotta u. a. 1998** OCHOTTA, Emil S. ; MUKHERJEE, Tamal ; RUTENBAR, Rob. A. ; CARLEY, L. R.: *Practical Synthesis of High-Performance Analog Circuits*. Kluwer Academic Publishers, 1998. - ISBN 0-7923-8237-4
- Phelps u. a. 1999** PHELPS, Rodney ; KRASNICKI, Michael ; RUTENBAR, Rob. A. ; CARLEY, L. R. ; HELLUMS, James R.: ANACONDA: Robust Synthesis of Analog Circuits via Stochastic Pattern Search. In: *IEEE Custom Integrated Circuits Conference*, Mai 1999, S. 567-570. - ISBN 0-7803-5443-5

- van der Plas u. a. 2001** PLAS, G. van der ; DEBYSER, G. ; LEYN, F. ; LAMPAERT, K. ; VANDENBUSSCHE, J. ; GIELEN, G.G.E. ; SANSEN, W. ; VESELINOVIC, P. ; LEENARTS, D.: AMGIE-A Synthesis Environment for CMOS Analog Integrated Circuits. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 20 (2001), Dezember, Nr. 9, S. 1037-1058. - ISSN 0278-0070
- Romeo und Sangiovanni-Vincentelli 1991** ROMEO, Fabio ; SANGIOVANNI-VINCENTELLI, Alberto F.: A Theoretical Framework for Simulated Annealing. In: *Algorithmica* 6 (1991), Nr. 3, S. 302-345. - ISSN 0178-4617
- Saleh und Newton 1990** SALEH, Resve A. ; NEWTON, A. R.: *Mixed-Mode Simulation*. Kluwer Academic Publishers, 1990. - ISBN 0-7923-9107-1
- Sasaki und Hajek 1988** SASAKI, Galen H. ; HAJEK, Bruce: The Time Complexity of Maximum Matching by Simulated Annealing. In: *Journal of the ACM* 35 (1988), April, Nr. 2, S. 387-403. - ISSN 0004-5411
- Schwencker u. a. 2002** SCHWENCKER, Robert ; SCHENKEL, Frank ; PRONATH, Michael ; GRÄB, Helmut: Analog Circuit Sizing using Adaptive Worst-Case Parameter Sets. In: *Design, Automation and Test in Europe*, März 2002, S. 581-585. - ISBN 0-7695-1471-5
- Schöneburg u. a. 1993** SCHÖNEBURG, Eberhard ; HEINZMANN, Frank ; FEDDERSEN, Sven: *Genetische Algorithmen und Evolutionsstrategien*. Addison-Wesley, 1993. - ISBN 3-8931-9493-2
- Spence und Soin 1988** SPENCE, Robert ; SOIN, Randeep S.: *Tolerance Design of Electronic Circuits*. Addison-Wesley, 1988. - ISBN 0-201-18242-4
- Spiro 1990** SPIRO, Hans: *Simulation integrierter Schaltungen*. Oldenbourg, 1990. - ISBN 3-486-21660-0
- Sripramong und Toumazou 2002** SRIPRAMONG, Thanwa ; TOUMAZOU, Christofer: The Invention of CMOS Amplifiers Using Genetic Programming and Current-Flow Analysis. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 21 (2002), November, Nr. 11, S. 1237-1252. - ISSN 0278-0070
- SUN 2003** MICROSYSTEMS, SUN: *Forte C++*. 2003. - URL <http://www.sun.com/software/sundev/previous/cplusplus/interval/>

Literaturverzeichnis

- Swings u. a. 1991a** SWINGS, Koen ; DONNAY, Stéphane ; SANSEN, Willy M. C.: HECTOR: A Hierarchical Topology-Construction Program for Analog Circuits Based on a Declarative Approach to Circuit Modeling. In: *IEEE Custom Integrated Circuits Conference*, Mai 1991, S. 5.3.1-5.3.4. - ISBN 0-7803-0015-7
- Swings u. a. 1991b** SWINGS, Koen ; GIELEN, Georges G. E. ; SANSEN, Willy M. C.: An Intelligent Analog IC Design System Based on Manipulation of Design Equations. In: *IEEE Custom Integrated Circuits Conference*, Mai 1991, S. 8.6.1-8.6.5. - ISBN 0-7803-0015-7
- Veselinovic u. a. 1995** VESELINOVIC, P. ; LEENAERTS, D. M. W. ; BOKHOVEN, W. van ; LEYN, F. ; PROESMANS, F. ; GIELEN, Georges G. E. ; SANSEN, Willy M. C.: A Flexible Topology Selection Program as Part of an Analog Synthesis System. In: *IEEE European Design and Test Conference*, März 1995, S. 119-123. - ISBN 0-8186-7039-8
- Vladimirescu 1994** VLADIMIRESCU, Andrei: *The Spice Book*. John Wiley & Sons, 1994. - ISBN 0-471-60926-9
- Wolfram Research 2002** WOLFRAM RESEARCH, INC.: *Mathematica 4*. 2002. - URL <http://www.wolfram.com/products/mathematica>