



# **Entwurf und Optimierung der zweidimensionalen Gitterströmung axialer Turbomaschinenbeschaufelungen mit Neuronalen Netzen und Genetischen Algorithmen**

Inaugural-Dissertation zur Erlangung des akademischen Grades eines  
Doktor-Ingenieurs (Dr.-Ing.) im Fachbereich Maschinenbau  
der Universität Gesamthochschule Kassel

vorgelegt von Dipl.-Inform. Michael Uelschen aus Hannover

Kassel, August 2000

Als Dissertation vom Fachbereich Maschinenbau angenommen am 14. Oktober 1999

Erster Gutachter: Univ.-Prof. Dr.-Ing. M. Lawerenz  
Zweiter Gutachter: Univ.-Prof. Dr.-Ing. R. Niehuis

Tag der mündlichen Prüfung:

28. April 2000

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

**Uelschen, Michael**

Entwurf und Optimierung der zweidimensionalen Gitterströmung axialer Turbomaschinenbeschaufelungen mit Neuronalen Netzen und Genetischen Algorithmen / Michael Uelschen. - Kassel : kassel univ. press, 2000. - XVI, 176 S. : III.

Zugl.: Kassel, Univ., Diss. 2000

ISBN 3-933146-38-0

© 2000, kassel university press GmbH, Kassel

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsschutzgesetzes ist ohne Zustimmung des Verlags unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Umschlaggestaltung: 5 Büro für Gestaltung, Kassel

Druck und Verarbeitung: Zentraldruckerei der Universität Gesamthochschule Kassel  
Printed in Germany

## Zusammenfassung

Durch die ständig wachsende Anzahl von geometrischen und aerodynamischen Parametern ist es für den Menschen zunehmend schwieriger, Abhängigkeiten und Zusammenhänge im Auslegungsprozeß axialer Turbomaschinen vorteilhaft einzubeziehen. In dieser Arbeit wird ein zweidimensionales Entwurfsverfahren auf der Basis periodischer B-Spline-Funktionen entwickelt. Vorhandene Profile lassen sich dabei mit Hilfe einer Approximation in diese mathematische Darstellung überführen. Durch eine Simulated Annealing Optimierung kann die Anzahl der zur Beschreibung notwendigen Parameter minimiert werden. Ein C-Netz bildet die Basis der aerodynamischen Analyse. Die Berechnung der Strömung erfolgt mit einem Stromfunktionsansatz auf  $S_1$ -Stromflächen und ist iterativ mit einem Grenzschichtverfahren gekoppelt. Künstliche neuronale Netze werden im Profilentwurf eingesetzt, um Zusammenhänge zwischen Geometrie und Aerodynamik abzubilden. Dieses führt zu einem schnellen Werkzeug zum Entwurf und zur Bewertung von Profilen. Um speziellen Anforderungen angepaßte Beschaufelungen zu erhalten, ist eine weitergehende aerodynamische Optimierung notwendig. Hierzu wird ein genetischer Algorithmus eingesetzt, der sich am biologischen Vorbild der Evolution orientiert und auf Selektion, Mutation und Rekombination basiert.

**Schlüsselwörter:** Approximation, B-Spline-Funktion, Genetischer Algorithmus, Gittergenerierung, Grenzschicht, Neuronales Netz, Optimierung,  $S_1$ -Stromfläche, Simulated Annealing, Stromfunktion

## Abstract

Due to the steady growing number of geometric and aerodynamic parameters, it is increasingly difficult for the human being to include dependencies and correlations in the design process of modern axial turbomachines. In this work a two-dimensional design method is developed, based on the geometrical definition of profiles by periodic b-spline functions. By means of approximation of existing profiles it is possible to get a uniform mathematical description. In a simulated annealing algorithm the number of parameters which is necessary to describe a profile is being minimized. The aerodynamic analysis starts with the generation of a C-grid. The flow is computed on  $S_1$ -surfaces using a stream function approach and is combined iteratively with a boundary layer calculation. An artificial neural network is being used in the profile design in order to model the connection between geometry and aerodynamic. This leads to a fast tool to design and to evaluate profiles. A further aerodynamic optimization is necessary to deal with the predefined constraints of the blade. In the optimization we use a genetic algorithm which tries to adapt the biological principle of evolution and is based on selection, mutation and crossover.

**Keywords:** approximation, boundary layer, b-spline function, genetic algorithm, grid generation, neural network, optimization,  $S_1$ -surface, simulated annealing, stream function



# Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Fachgebiet Strömungsmaschinen des Instituts für Thermische Energietechnik an der Universität Gesamthochschule in Kassel. An dieser Stelle möchte ich allen danken, die zum Erfolg der vorliegenden Dissertation beigetragen haben.

Mein besonderer Dank gilt Herrn Univ.-Prof. Dr.-Ing. Martin Lawerenz, der mir die einmalige Möglichkeit bot, beim Aufbau seines neu gegründeten Fachgebiets mitzuwirken. Sein in mich gesetztes Vertrauen und die Bereitschaft, mir jederzeit meine Fragen zur Auslegung von Turbomaschinen zu beantworten, waren entscheidene Beiträge zum Gelingen dieser Arbeit.

Bei Herrn Univ.-Prof. Dr.-Ing. Reinhard Niehuis bedanke ich mich herzlich für die Übernahme des Korreferats.

Bei meinen Kollegen und Kolleginnen am Fachgebiet Strömungsmaschinen bedanke ich mich für die gute Zusammenarbeit und das angenehme Arbeitsklima. Die gemeinsamen Diskussionen und Anregungen waren oftmals eine wertvolle Hilfe.

Meinen Eltern danke ich dafür, daß sie mir meine Ausbildung ermöglichten.



# Inhaltsverzeichnis

<b>1 Einleitung</b>	<b>1</b>
1.1 Turbomaschinenentwurf und Strömungsberechnung . . . . .	1
1.2 Künstliche und berechenbare Intelligenz . . . . .	3
1.3 Inhalt dieser Arbeit . . . . .	4
<b>2 Geometrischer Entwurf mit B–Spline–Funktionen</b>	<b>7</b>
2.1 Entwurfsebene axialer Profile . . . . .	7
2.2 Freiformprofile . . . . .	9
2.2.1 Bézier–Funktionen . . . . .	9
2.2.2 B–Spline–Funktionen . . . . .	11
2.2.3 Approximation und Interpolation . . . . .	15
2.2.4 Erweiterungen . . . . .	21
2.3 3D–Geometrie . . . . .	23
2.3.1 Tensorprodukt–B–Spline–Flächen . . . . .	23
2.3.2 Skinning . . . . .	24
2.4 Implementierte Verfahren . . . . .	25
2.5 Andere Ansätze und Ausblick . . . . .	27
<b>3 Numerische Strömungsberechnung</b>	<b>29</b>
3.1 Überblick . . . . .	29
3.2 Elliptische Gittergenerierung . . . . .	30
3.2.1 Transformation . . . . .	31
3.2.2 Der Ansatz von Steger und Sorensen . . . . .	32
3.2.3 Diskretisierung und iterative Lösung . . . . .	33
3.2.4 Konturpunktverteilung . . . . .	34
3.2.5 Mehrstufige Gitter . . . . .	39
3.2.6 Implementierte Verfahren . . . . .	40
3.3 Stromfunktionsverfahren . . . . .	43
3.3.1 Einführung und Grundgleichungen . . . . .	43
3.3.2 Geometrie der $S_1$ –Stromfläche . . . . .	44
3.3.3 Hauptgleichung des Stromfunktionsverfahrens . . . . .	44
3.3.4 Dichtebestimmung . . . . .	46
3.3.5 Randbedingungen . . . . .	48

---

3.3.6	Kuttasche Abflußbedingung . . . . .	49
3.3.7	Diskretisierung . . . . .	50
3.3.8	Iterative Lösung . . . . .	55
3.4	Grenzschichtverfahren . . . . .	58
3.4.1	Grundgleichungen . . . . .	59
3.4.2	Algorithmus . . . . .	60
3.4.3	Beispiel einer ebenen Plattengrenzschicht . . . . .	69
3.4.4	Koppelung mit Stromfunktionsverfahren . . . . .	69
3.4.5	Profilverluste . . . . .	71
3.5	Nachrechnung . . . . .	77
3.5.1	Verdichterprofil FVV–V1 . . . . .	77
3.5.2	Verdichterprofil MAN–GHH–1–S1 . . . . .	80
<b>4</b>	<b>Approximation mit neuronalen Netzen</b>	<b>83</b>
4.1	Einführung . . . . .	83
4.2	Künstliche neuronale Netze . . . . .	84
4.2.1	Abbildung neuronaler Netze . . . . .	84
4.2.2	Lernverfahren . . . . .	86
4.3	Stuttgarter Neuronale Netze Simulator (SNNS) . . . . .	89
4.4	Profilentwurf . . . . .	91
4.4.1	Netzarchitektur . . . . .	92
4.4.2	Aufbau der Datenbasis und Training . . . . .	96
4.4.3	Approximationsgüte und Generalisierungseigenschaften . . . . .	97
4.5	Ausblick . . . . .	105
<b>5</b>	<b>Optimierung</b>	<b>107</b>
5.1	Optimierungsverfahren . . . . .	107
5.2	Mutations–Selektions–Algorithmen . . . . .	108
5.2.1	Der Mutationsoperator . . . . .	108
5.2.2	Simulated Annealing . . . . .	110
5.2.3	Beispiel: Geometrieoptimierung . . . . .	113
5.2.4	Weitere Ansätze . . . . .	118
5.3	Genetische Algorithmen . . . . .	118
5.3.1	Einführung . . . . .	119
5.3.2	Die C++ Klassenbibliothek GAlib (MIT) . . . . .	123
5.3.3	Approximation mit B–Spline–Funktionen . . . . .	124
5.4	Optimierung von Profilen . . . . .	127
5.4.1	Zielfunktion und Genom . . . . .	128
5.4.2	Inverser Entwurf . . . . .	128
5.4.3	Direkter Entwurf . . . . .	130
5.5	Ausblick . . . . .	137
<b>6</b>	<b>Resümee</b>	<b>139</b>

<b>A Algorithmen und Bildanhang</b>	<b>141</b>
A.1 Algorithmus von de Boor . . . . .	141
A.2 Ausgewählte Wölbungs- und Dickenverteilungen . . . . .	143
A.3 Konturpunktverteilung . . . . .	145
A.4 Profilentwurf mit neuronalen Netzen . . . . .	148
A.5 Approximation mit Simulated Annealing . . . . .	159
<b>Literaturverzeichnis</b>	<b>165</b>



# Verzeichnis der Formelzeichen

## Lateinische Buchstaben

Zeichen	Bedeutung
$a_d, a_w$	Geometrieparameter der Dicken– bzw. Wölbungsverteilung
$a_j$	Aktivierungszustand eines Neurons $j$
$A$	Fläche
$b$	Schichtdicke der Stromfläche
$B_{i,n}$	$i$ -tes Bernsteinpolynom vom Grade $n$ für $i = 0, 1, \dots, n$
$c_f$	Reibungsbeiwert
$c_f$	Abkühlungsfaktor im Simulated Annealing Verfahren
$d$	Dicke
$\mathbf{D}_k, \mathbf{Q}_k$	Ableitung, Punkte zur Approximation/Interpolation mit $k = 0, 1, \dots, r$
$\vec{e}_r, \vec{e}_u, \vec{e}_z$	zylindrische Einheitsvektoren
$E$	Entrainmentkoeffizient
$E$	Approximationsfehler
$f, f^*$	Zielfunktion, Fitneß
$f_{act}$	Aktivierungsfunktion
$h, h_{rot}$	Enthalpie, Rothalpie
$h_1, h_{12}$	Grenzschichtformfaktor
$\vec{i}, \vec{j}, \vec{k}$	kartesische Einheitsvektoren
$l$	Länge, Sehnenlänge
<b>LSF</b>	Approximation mit der Methode der kleinsten Quadrate
$Ma$	Machzahl
$\mathcal{N}(\mu, \sigma)$	Gaußverteilung mit Erwartungswert $\mu$ und Standardabweichung $\sigma$
$net_j$	Propagierungsfunktion eines Neurons $j$
$N_{i,p}$	$i$ -te Basisfunktion vom Grade $p$ für $i = 0, 1, \dots, n$
$N_{eq}$	Anzahl Iterationen bis Gleichgewichtszustand erreicht
$o_j$	Ausgabewert eines Neurons $j$
$p$	Druck
$p, p_r, p_m$	Wahrscheinlichkeit (Rekombination, Mutation)
$\mathbf{P}_i, \mathbf{P}_{i,j}$	Kontrollpunkt für $i = 0, 1, \dots, n$ und $j = 0, 1, \dots, m$

<b>P</b>	Population
$P, Q$	Kontrollfunktionen
$r$	Radius
$r$	diskretisierter Integrand des Volumenintegrals
$r, \varphi, z$	Zylinderkoordinate
$r_{\text{lam}}, r_{\text{turb}}$	laminarer, turbulenter Rückgewinnfaktor
$R_{i,p}$	$i$ -te rationale Basisfunktion vom Grade $p$ für $i = 0, 1, \dots, n$
$Re, Re_2$	Reynolds-Zahl, mit $\delta_2$ gebildet
$S_1, S_2$	Stromfläche erster und zweiter Art
$t$	Teilung
$t$	Kurvenparameter
$t_i$	$i$ -ter Zeitpunkt
$t_i$	Knoten für $i = 0, 1, \dots, m$
$\mathbf{t}$	Knotenvektor mit $\mathbf{t} = \{t_0, t_1, \dots, t_m\}$
$\bar{\mathbf{t}}$	Parametervektor mit $\bar{\mathbf{t}} = \{\bar{t}_0, \bar{t}_1, \dots, \bar{t}_r\}$ (Parametrisierung)
$\hat{\mathbf{t}}$	Parametervektor mit $\hat{\mathbf{t}} = \{\hat{t}_0, \hat{t}_1, \dots, \hat{t}_r\}$ (Punktinversion)
$T$	Temperatur
$Tu$	Turbulenzgrad
$u$	Umfangsgeschwindigkeit
$\mathcal{U}(a, b)$	Gleichverteilung im Intervall $[a, b]$
$u, v$	Geschwindigkeitskomponente in kartesischen Koordinaten
$w$	Geschwindigkeit im Relativsystem
$w_r, w_u, w_z$	Geschwindigkeitskomponenten im Relativsystem
$w_{ij}$	Gewichtung zwischen zwei Neuronen $i$ und $j$
$x, x_i$	Chromosom/Individuum, $i$ -tes Chromosom
$x_i$	$i$ -te Stützstelle
$x_{\text{lam}}, x_{\text{tr}}$	laminare Ablöseposition, Position laminar-turbulenter Umschlag
$x, y, z$	kartesische Koordinate
$z, u$	transformierte Zylinderkoordinaten mit $u = r\varphi$

### Griechische Buchstaben

Zeichen	Bedeutung
$\alpha$	Strömungswinkel
$\beta$	Staffelungswinkel
$\delta$	Grenzschichtdicke
$\delta_1$	Verdrängungsdicke
$\delta_2$	Impulsverlustdicke
$\delta_E$	Entrainmentdicke
$\alpha, \beta, \gamma, \delta$	Seiten eines Kontrollvolumens
$\zeta_{i,j}$	rechte Seite im Stromfunktionsverfahren
$\eta, \eta^-, \eta^+$	Lernparameter Backpropagation-Verfahren

---

$\theta_j$	Schwellenwert eines Neurons $j$
$\Phi, \Sigma$	Dichtefunktionen
$\kappa$	Isentropenkoeffizient
$\lambda_1, \lambda_2$	Stromflächensteigung
$\mu$	Viskosität
$\nu$	kinematische Viskosität
$\xi, \eta$	konturangepaßte Koordinaten
$\rho$	Dichte
$\tau$	Schubspannung
$\bar{\varphi}, \bar{z}$	Koordinaten der Stromschicht
$\psi, \psi_{i,j}$	Stromfunktion, Stromfunktionswert an der Stelle $i, j$
$\psi_{i,j}^k$	Stromfunktionswert des Nachbarknotens $k$ für den Punkt $i, j$
$\omega$	Winkelgeschwindigkeit, Kreisfrequenz
$v_{i,j}^k$	Gewichtungsfaktor des Nachbarknotens $k$ für den Punkt $i, j$

## Indices

Zeichen	Bedeutung
1, 2	Eintritt, Austritt
1/3, 2/3	Größe auf 0.34 bzw. 0.67 einer Strecke
$ax$	axial
$i$	inkompressibel
min, max	Minimum, Maximum
$t$	Totalgröße
$w$	Größe an der Wand
$\delta$	Größe am Grenzschichtrand
ind	Indifferenzpunkt
$\infty$	ungestört vom Körper
start, end	Größe zu Beginn bzw. am Ende der Iteration
krit	kritisch
left, right	linke bzw. rechte Begrenzung einer Größe
Platte	ebene Plattengrenzschicht

## Sonstige Zeichen

Zeichen	Bedeutung
$\rightarrow$	vektorielle Größe
$'$	veränderte Größe, Schwankungsgröße, Ableitung
$*$	normierte Größe, dimensionslos
$-$	Mittelwert, zeitlich gemittelt
$ _i$	Größe am Gitterknoten $i$

## Pseudocode

Zur Beschreibung der in dieser Arbeit wichtigen Algorithmen wird auf eine formalisierte Darstellung sowie auf Elemente der Mengenalgebra zurückgegriffen (siehe SMITH (1989) und BÖHME (1990)). Zur Erhaltung der Lesbarkeit wird allerdings an einigen Stellen auf eine vollständige formale Angabe verzichtet.

Die Beschreibung eines Algorithmus wird durch die Schlüsselworte

**algorithm** NAME **begin** Operationen **end**.

begrenzt. Teile von Algorithmen können in sogenannten Routinen, das sind Funktionen oder Prozeduren, wie folgt zusammengefaßt werden:

**routine** Name (Parameter) Operationen **end**

Ablaufverzweigungen sind durch die Sequenz

**if** (Boolesche Bedingung) —> Operationen **else** Operationen **fi**

möglich. Hierbei werden die ersten Operationen dann ausgeführt, wenn die Boolesche Bedingung erfüllt (wahr) ist. Ansonsten wird der zweite Zweig bearbeitet.

Wiederholungen, d.h. Schleifen, werden durch die Schreibweise

**do** (Boolesche Bedingung) —> Operationen **od**

ermöglicht. Die angegebenen Operationen werden solange ausgeführt, bis die Boolesche Bedingung nicht erfüllt (falsch) wird.

## Operatoren

Zeichen	Bedeutung
$\leftarrow$	Zuweisungsoperator
$\neg, \vee, \wedge$	Boolescher Nicht-, Oder-, Und-Operator
div, mod	Ganzzahldivision, ganzzahliger Rest (Modulo)
$\lfloor a \rfloor$	größter ganzzahliger Wert $\leq a$
<b>return</b> <i>a</i>	liefert den Wert <i>a</i> zurück
terminate	Abbruchbedingung, z.B. maximale Anzahl von Iterationen erreicht
<i>(Kommentar)</i>	Kommentarzeile

## Mengenalgebra

Zeichen	Bedeutung
<b>A × B</b>	$\{(x, y)   x \in \mathbf{A} \wedge y \in \mathbf{B}\}$ (Produktmenge)
<b>A ∪ B</b>	$\{x   x \in \mathbf{A} \vee x \in \mathbf{B}\}$ (Vereinigungsmenge)
<b>A ⊂ B</b>	$\{x   \forall x \in \mathbf{A} \Rightarrow x \in \mathbf{B}\}$ (Teilmenge)
$ \mathbf{A} $	Anzahl der Elemente in <b>A</b> (Kardinalität)
$x \in [a, b]$	$a \leq x \leq b$ (geschlossenes Intervall)
$x \in [a, b)$	$a \leq x < b$ (halbseitig geöffnetes Intervall)

Tu erst das Notwendige, dann das Mögliche,  
und plötzlich schaffst du das Unmögliche.

FRANZ VON ASSISI



# Kapitel 1

## Einleitung

Diese Arbeit behandelt den Entwurf und die Optimierung axialer Turbomaschinenbeschauflungen. Bevor in den nachfolgenden Kapiteln auf die verwendeten Methoden, die durchgeführten Untersuchungen und die erzielten Ergebnisse eingegangen wird, erfolgt eine Einordnung in die betreffenden Wissensgebiete.

### 1.1 Turbomaschinentwurf und Strömungsberechnung

Der Entwurf von Turbomaschinen ist durch eine große Anzahl, sowohl geometrischer als auch aerodynamischer, Freiheitsgrade gekennzeichnet. Ziel einer jeden Auslegung ist es nun, eine optimale, zuvor definierter Randbedingungen genügende, Konfiguration zu finden. Moderne Turbomaschinen zeichnen sich bereits heutzutage durch einen hohen Wirkungsgrad aus. Für das Erreichen einer weiteren Leistungssteigerung ist daher ein zunehmender technischer Aufwand nötig.

Die in Turbomaschinen zu betrachtende Strömung ist dreidimensional, kompressibel, reibungsbehaftet und insbesondere auch instationär. Die numerische Behandlung erfordert es, eine Reihe von Vereinfachungen zu treffen. HILDEBRANDT (1998) gibt einen aktuellen Überblick über eine Reihe von Verfahren, die die Navier–Stokes Gleichungen in ihrer dreidimensionalen Formulierung für stationäre Vorgänge lösen. Trotz der schnell fortschreitenden Entwicklung der Rechnerhardware sind nur Hoch– und Höchstleistungsrechner in der Lage, detaillierte Ergebnisse in akzeptablen Zeiträumen zu liefern. Vor allem die Handhabung turbulenter Eigenschaften erfordert einen hohen rechentechnischen Aufwand (vgl. MOIN UND KIM (1997)).

Da die verschiedenen Optimierungsverfahren eine Vielzahl, i.allg. hunderte oder tausende, von Iterationszyklen durchlaufen, sind solche dreidimensionalen Verfahren bis auf weiteres für Optimierungsaufgaben unpraktikabel. Der von WU (1952) beschriebene Ansatz reduziert den Aufwand der dreidimensionalen Strömungsberechnung, in dem diese durch zwei miteinander gekoppelte zweidimensionale Berechnungsverfahren ersetzt wird (*quasidreidimensionale Berechnung*). Hierzu werden zwei Scharen von Stromflächen eingeführt, die von Schaufel zu Schaufel (*blade-to-blade*) bzw. von Nabe zum Gehäuse (*hub-to-tip*) sich erstrecken und als  $S_1$ – bzw.  $S_2$ – Stromflächen bezeichnet werden. Bei den von CASEY

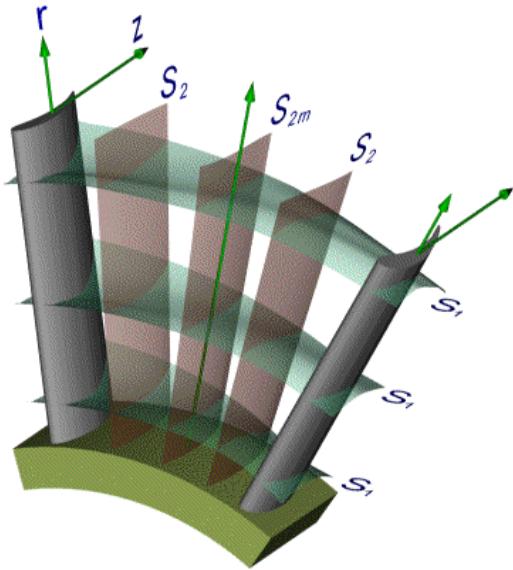


Abbildung 1.1: Der Aufwand zur Bestimmung der dreidimensionalen Strömung lässt sich durch eine kombinierte, zweidimensionale Berechnung auf Stromflächen erster und zweiter Art nach WU (1952) reduzieren.

(1994) oder auch COFER IV (1995) beschriebenen Methoden zeigt sich, daß die zweidimensionalen Verfahren trotz der inhärenten Vereinfachungen weiterhin die wichtigsten Bausteine zur rechnergestützten aerodynamischen Auslegung von Strömungsmaschinen sind. Dreidimensionale Verfahren werden demnach als letzter Schritt bei der Entwicklung zur Untersuchung von einzelnen Strömungsphänomenen eingesetzt.

Neben den Stromflächen ist die Einführung einer sogenannten Stromfunktion  $\psi$  das im folgenden verwendete grundlegende Konzept. Die Bewegung im stationären Zustand eines infinitesimalen Fluidteilchens erfolgt hierbei entlang von Stromlinien. Der Zusammenhang zwischen Geschwindigkeit und der Stromlinie wird durch die Funktion  $\psi$  modelliert. Stromfunktionsverfahren haben sich als robuste Verfahren erwiesen, wie zum Beispiel die Arbeiten von LÜCKING (1982, 1989) und RÖPER (1994) zeigen. Diese bilden die Basis der in dieser Arbeit durchgeführten zweidimensionalen Strömungsberechnung.

Da das Stromfunktionsverfahren nur reibungsfreie Phänomene wiedergeben kann, lässt sich die Abbildung von Reibungseffekten mit einem Grenzschichtverfahren modellieren. Hierbei fließen u.a. (halb-) empirische Korrelationen mit ein.

In OLIVER ET AL. (1994) oder auch MILLER IV ET AL. (1997) wird die gängige Vorgehensweise zur geometrischen Konstruktion von Schaufeln in Strömungsmaschinen dargestellt. Der Definition der einzelnen Stromflächen, dieses können einfache Zylinder- und Kegel-

flächen bis hin zu beliebigen Freiformflächen sein, folgt der zweidimensionale Entwurf auf den transformierten Rotationsflächen. Zur Beschreibung der Freiformprofile werden i.allg. Bézier- oder B-Spline-Funktionen verwendet. Somit ist eine Anpassung an geometrische und aerodynamische Bedingungen möglich. Durch die Angabe einer Raumkurve ist es so dann möglich, die einzelnen Profilschnitte radial aufzufädeln. Nach der Berechnung der einhüllenden Oberfläche ergibt sich abschließend die dreidimensionale Schaufel.

Zunehmend stellt das menschliche Ingenium einen limitierenden Faktor bei der Handhabung der Komplexität dar. So sind zum Beispiel im Entwurfsverfahren von TONG UND GREGORY (1990) über 1800 freie Parameter bei der Auslegung einer mehrstufigen Turbine in Beziehung zueinander zu setzen und dabei Abhängigkeiten zu beachten. Daher liegt es nahe, Verfahren und Methoden zu entwickeln, die menschliche Intelligenz in den Rechner abbilden und somit zur Unterstützung im Auslegungsprozeß von Strömungsmaschinen herangezogen werden können.

## 1.2 Künstliche und berechenbare Intelligenz

Seit der Entwicklung des ersten programmgesteuerten Rechners ist auch die Frage einer maschinellen Intelligenz von zentralem Interesse. Es haben sich hierbei zwei konkurrierende Paradigmen zur Erfassung von künstlicher Intelligenz (*Artifical Intelligence, AI*) gebildet:

**Symbolische Informationsverarbeitung** versucht, Wissen in Form von Regeln und Produktionssystemen in einen rechnerkonformen Formalismus abzubilden. Die Symbolmanipulation leitet sich aus der Philosophie ab, wobei ausgehend von einzelnen Fakten und einem Regelwerk neues Wissen folgt.

**Konnektionismus** wird als der Ansatz verstanden, das (menschliche) Gehirn als Interaktion von vernetzten und parallel arbeitenden Neuronen nachzuvollziehen. Diese Simulation von Neuronen entspringt eher den Erkenntnissen der Neurowissenschaft aus Medizin und Biologie.

DREYFUS UND DREYFUS (1996) sprechen in diesem Zusammenhang von den unterschiedlichen Strategien „einen Geist bauen gegen ein Gehirn modellieren“. Durch neue Erkenntnisse seit Mitte der achtziger Jahre erleben die künstlichen neuronalen Netze zunehmend Aufmerksamkeit und stellen ein breites Forschungsgebiet unterschiedlicher Disziplinen dar, welches an dieser Stelle auch nicht nur annähernd wiedergegeben werden kann. Für eine Diskussion, was (künstliche) Intelligenz ist und was diese zu leisten vermag, sowie der auftretenden Probleme, sei auf PENROSE (1991) und GRAUBARD (1996) hingewiesen.

Ungeachtet der theoretischen und psychologischen Fragestellungen zur künstlichen Intelligenz hat sich seit Mitte der neunziger Jahre ein Forschungsfeld etabliert, welches unter dem Begriff der berechenbaren Intelligenz (*Computational Intelligence, CI*) die folgenden drei Wissenszweige zusammenfaßt und miteinander verbindet:

**Künstliche neuronale Netze** sind Abbildungen von Verbänden biologischer Nervenzellen und bilden informationsverarbeitende Einheiten, die untereinander vernetzt sind.

Ein wesentliches Merkmal ist ihre Lernfähigkeit, d.h. die Fähigkeit, eine Aufgabe anhand zuvor trainierter Beispiele zu lösen.

**Genetische Algorithmen und Evolutionsstrategien** sind stochastische, d.h. dem Zufall unterliegende Optimierungsverfahren, die sich an den Prinzipien der natürlichen Evolution orientieren. Im Gegensatz zu klassischen Gradientenverfahren ist eine Bestimmung der Ableitungen der Zielfunktion nicht nötig.

**Fuzzy–Control** Ansätze stellen den Versuch dar, Informationen auf einer symbolischen Ebene zu verarbeiten. Erfahrungen werden in Form von Wenn–Dann–Regeln festgelegt. Die mathematischen Grundlagen werden durch die Theorie der unscharfen Mengen bereitgestellt.

Methoden der berechenbaren Intelligenz orientieren sich insbesondere an praktischen Problemen. Diese stellen somit ein Betätigungsfeld für Informatiker und Ingenieure gleichermaßen dar, wie u.a. die Konferenz in Dortmund von REUSCH (1997) und die vom VDI (1998) durchföhrte Tagung in Berlin zeigen.

In der vorliegenden Arbeit werden Verfahren künstlicher neuronaler Netze sowie genetischer Algorithmen eingesetzt. Ergebnisse aktueller Forschung von evolutionären Verfahren aus dem Bereich der Ingenieurwissenschaften sind in WINTER ET AL. (1995) sowie DAS-GUPTA UND MICHALEWICZ (1997a) zu finden. Fuzzy–Control Methoden werden hier nicht betrachtet und bleiben somit Forschungsgegenstand weiterer, zukünftiger Untersuchungen. Für eine Einföhrung in die Thematik von Fuzzy–Logik und Fuzzy–Control sei auf JAANI-NEH UND MAIJOHANN (1996) oder KOCH ET AL. (1996) hingewiesen. BOTHE (1998) gibt einen Überblick von sogenannten Fuzzy–Neuro Methoden.

Fragestellungen zur Optimierung von Strömungsmaschinen, vornehmlich auch mit evolutionären Techniken, werden in den letzten Jahren im Rahmen größerer Projekte auf nationaler und internationaler Ebene betrachtet. Hierzu sind u.a. die europäischen Projekte EUROOPT und ECARP zu nennen (siehe PÉRIAUX ET AL. (1997, 1998)).

Das Verbundprojekt EVOTECH des BMBF<sup>1</sup> untersucht Evolutionsstrategien in unterschiedlichen technischen Anwendungsgebieten (vgl. RECHENBERG (1998); BAIER (1998)). Im Rahmen des TURBOTECH Forschungsvorhabens der Arbeitsgemeinschaft Hochtemperatur–Gasturbine (AG TURBO, vgl. hierzu SERVATY UND KEPPEL (1998)) untersuchen SCHWARZ (1992), LAWERENZ (1995) und MÜLLER-TÖWS (2000) verschiedene, klassische und evolutionäre, Optimierungsverfahren. DREYER ET AL. (1995) stellen einen neuen Ansatz vor, der die Optimierungsaufgabe und die Strömungsberechnung simultan löst. KÖLLER (1999) entwickelt und optimiert eine neue Profilsystematik für Gasturbinen.

### 1.3 Inhalt dieser Arbeit

Wie sich bereits erkennen läßt, befaßt sich diese interdisziplinäre Arbeit mit Verfahren zweier Wissensgebiete: zum einen mit Verfahren zur Strömungsberechnung und zum anderen mit Methoden der berechenbaren Intelligenz.

---

<sup>1</sup>Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie

Das Ziel dieser Arbeit ist es, ein zweidimensionales Entwurfswerkzeug, Verfahren zur Analyse und zur Optimierung von axialen Turbomaschinenbeschaufelungen zu entwickeln. Insbesondere werden hier zwei Methoden, künstliche neuronale Netze und genetische Algorithmen, untersucht, die sich an biologischen Vorbildern orientieren.

### Geometrischer Entwurf

Ausgangsbasis für den Entwurf axialer Beschaufelungen ist die Geometriedefinition zweidimensionaler Schaufelprofile. In dieser Arbeit wird ein neuer Ansatz mit periodischen B-Spline-Funktionen entwickelt. Im Gegensatz zu zusammengesetzten Bézier-Kurven lassen sich Stetigkeitsbedingungen einfacher garantieren. Durch die Approximation und Interpolation vorhandener Profile, z.B. aus der NACA-Familie, ist es möglich, diese in eine einheitliche mathematische Form zu überführen. Mit einem Simulated Annealing Ansatz wird die Anzahl der die Geometrie beschreibenden Parameter minimiert. Durch die Auffädelung und die Berechnung einer einhüllenden Oberfläche (*Skinning*) ergibt sich anschließend das dreidimensionale Schaufelblatt.

### Zweidimensionale Strömungsberechnung

Voraussetzung zur numerischen Strömungsberechnung ist die Diskretisierung des Strömungsraums im Bereich des zweidimensionalen Schaufelprofils. In der vorliegenden Arbeit wird ein Verfahren zur Gittergenerierung verwendet, welches auf der Lösung einer elliptischen partiellen Differentialgleichung beruht. Der implementierte Ansatz ermöglicht eine Kontraktion der Gitterlinien sowie eine Orthogonalität zwischen diesen. Durch ein neues Verfahren zur Konturpunktverteilung ist es möglich, insbesondere an der Vorderkante eine Verdichtung der Punkte auf der Kontur zu berechnen.

Um die innerhalb des Optimierungsprozesses erforderliche große Zahl von Konfigurationen zu untersuchen, ist ein schnelles Verfahren zur Strömungsberechnung notwendig. Hierzu ist zur aerodynamischen Bewertung von Profilen ein Stromfunktionsverfahren entwickelt worden, welches die kompressible Strömung auf rotationssymmetrischen  $S_1$ -Flächen bestimmt. Dazu wird die Hauptgleichung der  $S_1$ -Stromfläche in der konservativen Formulierung für einzelne Kontrollelemente diskretisiert und numerisch gelöst. Reibungseffekte an der Profiloberfläche werden mit einem Grenzschichtverfahren aufgelöst. Die von der Grenzschicht ausgeübte Verdrängungswirkung wird in einem iterativen Prozeß mit dem Stromfunktionsansatz gekoppelt.

### Approximation mit neuronalen Netzen

Künstliche neuronale Netze stellen den Versuch dar, die Struktur und die Eigenschaften des biologischen Vorbilds in eine formale, berechenbare Form abzubilden. Abstrakt lässt sich ein neuronales Netz als *Black Box* betrachten, welche eine  $n$ -stellige Eingabe in eine  $m$ -stellige Ausgabe überführt.

Neuronale Netze werden hier im Profilentwurf eingesetzt, um Zusammenhänge zwischen Geometrie und Aerodynamik abzubilden. Zum Aufbau des neuronalen Netzes wird eine

Trainingsmenge von mehreren hundert geometrischen und aerodynamischen Konfigurationen mit dem Stromfunktionsverfahren berechnet. Erweiterte *Backpropagation*-Strategien ermöglichen ein schnelles Lernen der Zusammenhänge aus den Trainingsdaten. Ein trainiertes neuronales Netz ist in der Lage, ohne rechenzeitaufwendige Strömungsberechnungen aerodynamische Aussagen zu tätigen. Durch die Definition bestimmter Zielvorgaben (z.B. axiale Breite oder Machzahlverlauf) generiert das neuronale Netz eine Profilgeometrie als B-Spline-Kurve. Zusätzlich werden Aussagen über zu erwartende aerodynamische Eigenschaften (z.B. Grenzschichtbelastung) getroffen. Das trainierte neuronale Netz lässt sich somit als schnelles Entwurfswerkzeug bei der Profilauslegung einsetzen.

### **Optimierung mit genetischen Algorithmen**

Um speziellen Anforderungen angepaßte Beschaufelungen zu erhalten, ist eine weitergehende Optimierung unerlässlich. Durch das neuronale Netz generierte Profile dienen als Startkonfiguration und reduzieren somit die benötigte Rechenzeit entscheidend. Zur Optimierung werden in dieser Arbeit genetische Algorithmen eingesetzt. Diese lösen komplexe Optimierungsaufgaben nicht nach herkömmlichen Methoden (z.B. Gradientenverfahren), sondern orientieren sich an dem biologischen Vorbild der Evolution. Durch das sukzessive Anwenden von Mutation, Rekombination und Selektion auf eine Population von Profilkonfigurationen ist es möglich, eine Zielfunktion zu minimieren. Diese setzt sich aus einer gewichteten Summation geometrischer (z.B. Profildicke) und aerodynamischer Zielvorgaben (z.B. Machzahlverlauf) zusammen.

Nicht unerwähnt bleiben soll an dieser Stelle die durchweg objektorientierte Modellierung und Implementierung der beschriebenen Verfahren und Algorithmen. Die Programmierung erfolgt in C++ (vgl. STROUSTRUP (1991)); zur Gestaltung der graphischen Bedienschnittstellen wurde auf Tcl/Tk von OUSTERHOUT (1995) zurückgegriffen.

## Kapitel 2

# Geometrischer Entwurf mit B–Spline–Funktionen

In den folgenden Abschnitten wird auf den Entwurf axialer Leit– und Laufschaufeln von Turbomaschinen eingegangen, wobei der Schwerpunkt in dieser Arbeit auf der zweidimensionalen Beschreibung liegt. Moderne Turbomaschinenschaufeln sind komplexe, i.allg. verwundene, dreidimensionale Körper, die von einem Fluid umströmt werden. Ihre mathematische Beschreibung erfolgt üblicherweise über radiale Schnitte, die übereinander geschichtet und ausgerichtet werden. Durch ein geeignetes Verfahren ist die Schaufeloberfläche zwischen jeweils zwei Schnitten zu berechnen.

### 2.1 Entwurfsebene axialer Profile

In Hinblick auf eine aerodynamische Berechnung, gegebenenfalls mit anschließender Optimierung, erscheint es sinnvoll, die konstruktive Entwurfsebene von Profilen mit dem Berechnungsverfahren zu koppeln. Wird zur aerodynamischen Auslegung zum Beispiel ein zweidimensionales Stromfunktionsverfahren verwendet, welches die Strömung auf rotationssymmetrischen Stromflächen erster Art (den sogenannten  $S_1$ -Flächen) berechnet, bietet es sich an, diese Stromflächen als die Ebenen zu betrachten, in denen die Profile entworfen werden. Alternativ wäre auch eine Beschreibung auf kartesischen Ebenen denkbar, allerdings ist die Koppelung mit dem Stromfunktionsverfahren auch im Hinblick auf eine Optimierung schwieriger.

Abbildung 2.1 zeigt das im folgenden verwendete Koordinatensystem. Zur Verdeutlichung ist das Zylinderkoordinatensystem in einem kartesischen System angelegt. Es wird angenommen, daß die radiale Komponente  $r$  in der  $x, y$ -Ebene liegt. Die  $z$ -Achse bildet die Maschinenachse, um die sich die Schaufel dreht (Rotationsachse).

Es sei  $f(t) = (f_x(t), 0, f_z(t))$  eine geeignete in der  $z, x$ -Ebene (Meridianebene) definierte parametrische Kurve. Die durch Rotation entstehende parametrische Rotationsfläche ergibt sich zu

$$\vec{s}_f(u, v) = f_x(u) \cos(v) \vec{i} + f_x(u) \sin(v) \vec{j} + f_z(u) \vec{k} \quad (2.1)$$

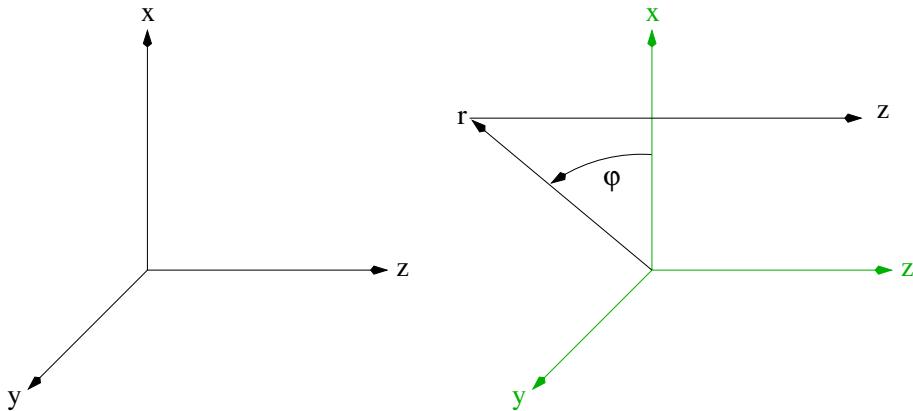


Abbildung 2.1: Kartesisches und Zylinderkoordinatensystem

oder in Zylinderkoordinatendarstellung

$$\vec{s}_f(u, v) = f_x(u)\vec{e}_r + v\vec{e}_u + f_z(u)\vec{e}_z \quad (2.2)$$

für  $u \in [u_0, u_1]$  und  $v \in [-\pi, +\pi]$ . Die Funktion  $f$  wird auch als Spur oder erzeugende Funktion der Rotationsfläche bezeichnet. Zur weiteren Vereinfachung gelte  $f_z(t) = t$ . Eine rotationssymmetrische  $S_1$ -Stromfläche wird demnach durch

$$\vec{s}_1(z, \varphi) = r(z)\vec{e}_r + \varphi\vec{e}_u + z\vec{e}_z \quad (2.3)$$

beschrieben (vgl. Abbildung 2.2).

Beim interaktiven Entwurf von Profilen ist es vorteilhaft, die Rotationsfläche in eine kartesische zweidimensionale Ebene abzubilden. Die Transformation  $(r, \varphi, z) \rightarrow (z, u)$  mit  $u = r\varphi$  bildet einen Punkt (in Zylinderkoordinaten) auf der Rotationsfläche in einen Punkt der kartesischen Ebene ab. Hierbei ist zu beachten, daß i.allg. nur für Zylinder- bzw. Kegelflächen eine längen- und winkeltreue Abbildung (Abwicklung) möglich ist. In den anderen Fällen findet eine Verzerrung statt.

Ein in der kartesischen Ebene definiertes Profil  $\vec{c}(t) = (c_x(t), c_y(t))$  kann durch Verwendung einer Spur  $r(z) > 0$  in eine dreidimensionale, in der Rotationsfläche liegende Kurve

$$\vec{p} = \vec{s}_1(c_x, c_y/r(c_y)) \quad (2.4)$$

in Zylinderkoordinaten

$$= r(c_x)\vec{e}_r + c_y/r(c_x)\vec{e}_u + c_x\vec{e}_z \quad (2.5)$$

bzw. kartesischen Koordinaten

$$= r(c_x)\cos(c_y/r(c_x))\vec{i} + r(c_x)\sin(c_y/r(c_x))\vec{j} + c_x\vec{k} \quad (2.6)$$

zurücktransformiert werden.

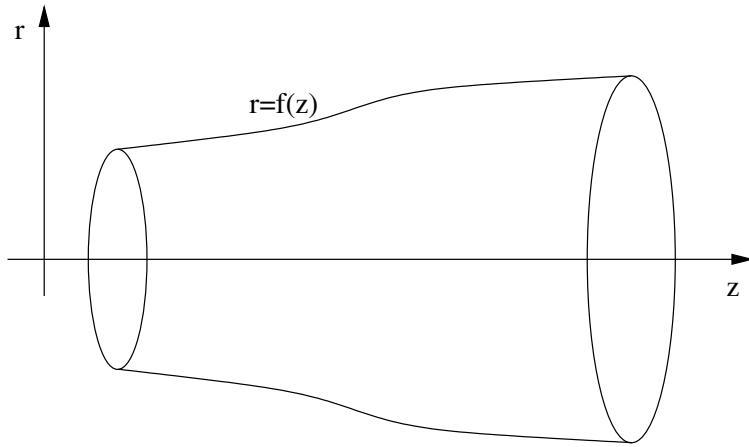


Abbildung 2.2: Rotationsfläche in der Meridianebene

## 2.2 Freiformprofile

Ein Ansatz zur mathematischen Beschreibung von Profilen ist die Verknüpfung einer Skettlinie mit einer Dickenverteilung (vergleiche hierzu z.B. KÄPPELI (1987) und Anhang A.2). Typische Vertreter dieser Vorgehensweise sind die NACA-Profilfamilien. Andere modernere Verfahren setzen Profile aus zwei Kreis- oder Ellipsensegmenten und mehreren interpolierenden Polynomen zusammen.

Zur weiteren Verbesserung des aerodynamischen Verhaltens bedarf es jedoch der Definition von Schaufeln durch Freiformprofile, da nur so dem Konstrukteur der notwendige Spielraum zur Formgebung, im Gegensatz zum Beispiel zu den NACA-Profilen, geboten wird. In den nächsten Abschnitten werden zuerst die mathematischen Voraussetzungen dargestellt, die dann nachfolgend die Grundlage für die in dieser Arbeit verwendeten Freiformprofile bilden.

### 2.2.1 Bézier–Funktionen

Zum Beschreiben von Konturen sind solche Funktionen wünschenswert, die zum einen eine umfangreiche Menge von Kurven abdecken, zum anderen jedoch effizient durch den Rechner zu behandeln sind. Im allgemeinen bieten Polynome diese Eigenschaften, zumal Funktionswerte und Ableitungen einfach (z.B. mit dem Hornerschema) ermittelt werden können. Polynome vom Grade  $n$  in der Monomdarstellung

$$C(t) = \sum_{i=0}^n \mathbf{a}_i t^i \quad (2.7)$$

sind für den (interaktiven) Entwurf gänzlich ungeeignet, da die Koeffizienten  $\mathbf{a}_i \in \mathbb{R}^2$  bzw.  $\mathbb{R}^3$  keinen direkten Rückschluß auf die geometrische Form erlauben. Einen wesentlich besseren Ansatz bietet die mathematisch äquivalente Bézier–Darstellung.

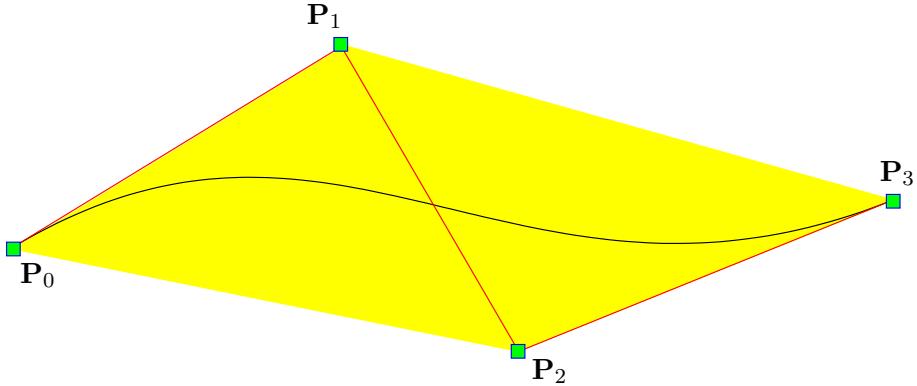


Abbildung 2.3: Die kubische Bézier-Kurve liegt in der konvexen Hülle (grau unterlegt) des zugehörigen Kontrollpolygons.

#### DEFINITION 1 (BERNSTEINPOLYNOM)

Das  $i$ -te Bernsteinpolynom vom Grade  $n$  über dem Intervall  $t \in [a, b]$  ist definiert als

$$B_{i,n}(t) = \frac{1}{(b-a)^n} \binom{n}{i} (t-a)^i (b-t)^{n-i} \quad \text{für } i = 0, \dots, n \quad . \quad (2.8)$$

Eine Vereinfachung ergibt sich durch die Beschränkung auf das Intervall  $t \in [0, 1]$ . Wichtige Eigenschaften der Bernsteinpolynome sind Nichtnegativität, Symmetrie und die Zerlegung der Eins. Für weitere Ausführungen sei an dieser Stelle auf DEUFLARD UND HOHMANN (1993) verwiesen. Jedes Polynom lässt sich als Linearkombination der Bernsteinbasis beschreiben.

#### DEFINITION 2 (BÉZIER-KURVE)

Es seien  $\{B_{i,n}(t)\}$  nach Definition 1 gebildete Basisfunktionen und  $\{\mathbf{P}_i\}$  Punkte im  $\mathbb{R}^2$  mit  $i = 0, \dots, n$ . Eine parametrische Bézier-Kurve  $B(t)$  vom Grade  $n$  wird dann durch

$$B(t) = \sum_{i=0}^n B_{i,n}(t) \mathbf{P}_i \quad (2.9)$$

definiert.

Die Koeffizienten  $\{\mathbf{P}_i\}$  heißen Kontroll- oder Bézier-Punkte, die lineare Verbindung durch diese dementsprechend Kontroll- oder Bézier-Polygon. Bevor auf die wichtigsten Eigenchaften eingegangen wird, bedarf es zuerst einer weiteren Definition.

#### DEFINITION 3 (KONVEXE HÜLLE)

Eine Menge  $\mathbf{M} \subset \mathbb{R}^2$  heißt konvex, wenn für je zwei Punkte  $x, y \in \mathbf{M}$  auch deren Verbindungsstrecke  $\overline{xy}$  ganz innerhalb der Menge  $\mathbf{M}$  liegt, d.h.

$$\forall x, y \in \mathbf{M} \implies \{\lambda x + (1 - \lambda)y \mid \lambda \in [0, 1]\} \subset \mathbf{M} \quad . \quad (2.10)$$

Die konvexe Hülle von  $\mathbf{M}$  ist die kleinste konvexe Teilmenge von  $\mathbb{R}^2$ , die  $\mathbf{M}$  enthält.

Der Graph einer Bézier–Funktion liegt in der konvexen Hülle des durch die Kontrollpunkte aufgespannten Polygons und steht daher unmittelbar mit diesem in einem geometrisch interpretierbaren Zusammenhang (vgl. Abbildung 2.3). Wird ein Kontrollpunkt (interaktiv) verändert, folgt die Kurve dem modifizierten Polygon.

Eine weitere Eigenschaft (*variation diminishing*) begrenzt die Anzahl der Schnittpunkte einer Geraden  $g$  mit der Kurve durch die Anzahl der Schnittpunkte der Geraden  $g$  mit dem Kontrollpolygon. Daraus ergibt sich insbesondere, daß ein konkaves Kontrollpolygon eine konvexe Kurve zur Folge hat.

Eine effiziente Berechnung von Kurvenpunkten bietet der sogenannte de Casteljau–Algorithmus, auf den aber an dieser Stelle nicht weiter eingegangen wird.

Zur Erhöhung des Gestaltungsspielraumes lassen sich mehrere Polynome stückweise zu Splines zusammensetzen<sup>1</sup>. Ein Vorteil solcher Funktionen ist die einfache Handhabung von Polynomen in Verbindung mit der Möglichkeit, glatt, d.h. ohne starke Schwingungen zwischen den Stützstellen, zu interpolieren. Spline–Funktionen bilden inzwischen ein übliches und häufig verwendetes mathematisches Werkzeug, zum Beispiel bei Interpolationsproblemen, so daß an dieser Stelle auf eine umfassende Darstellung verzichtet und auf die hinreichend vorhandenen Monographien (z.B. HÄMMERLIN UND HOFFMAN (1989) oder DEUFLARD UND HOHMANN (1993)) verwiesen werden kann.

Sollen mehrere Bézier–Kurven segmentweise zu einem Spline zusammengesetzt werden, so sind an den gemeinsamen Berührungs punkten bestimmte Stetigkeitsbedingungen zu erfüllen. Hierdurch wird die Interaktion eingeschränkt, da die Bézier–Punkte benachbarter Segmente bestimmte Positionen zueinander haben. So müssen zum Beispiel die Endpunkte identisch sein und die beiden letzten bzw. die beiden ersten Bézier–Punkte der zwei Segmente kollinear zueinander liegen (vgl. auch KUPPE (1997)).

Diese Schwierigkeiten werden durch die Verwendung von Basis–Spline–Funktionen, im folgenden verkürzt mit B–Spline bezeichnet, vermieden, da die gewünschten Stetigkeitsanforderungen einfach zu garantieren und keine eingeschränkten Freiheitsgrade zu berücksichtigen sind.

### 2.2.2 B–Spline–Funktionen

Zur mathematischen Einführung wird der Ansatz (vgl. PIEGL UND TILLER (1995b)) über die rekursiv definierten Basisfunktionen gewählt, da eine algorithmische Handhabung unmittelbar aus diesem folgt.

#### DEFINITION 4 (BASISFUNKTION)

Es sei  $\mathbf{t} = \{t_0, t_1, \dots, t_m\}$  eine Folge von nicht fallenden reellen Zahlen, d.h.  $t_i \leq t_{i+1}$  für

---

<sup>1</sup>Es werden hier nur sogenannte Polynom–Splines betrachtet. Eine Verallgemeinerung auf andere Basisfunktionen ist jedoch möglich.

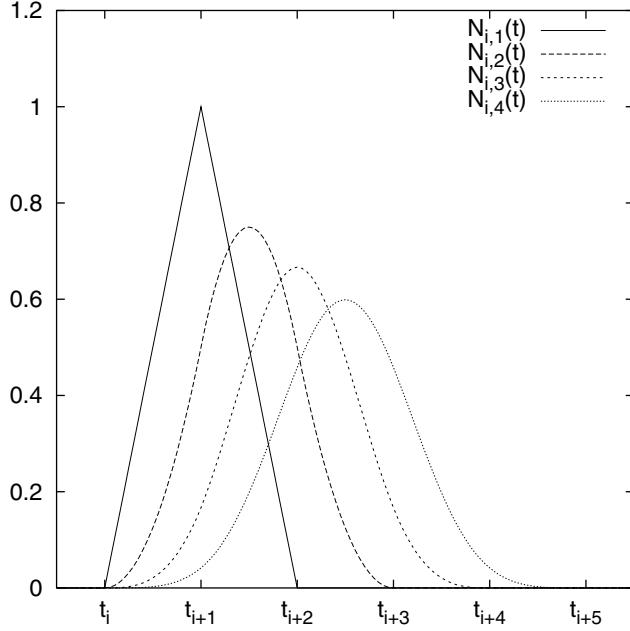


Abbildung 2.4: Für B-Spline-Basisfunktionen gilt die lokale Trägereigenschaft, d.h. für ein  $t \notin [t_i, t_{i+p+1}]$  folgt  $N_{i,p}(t) = 0$ .

$i = 0, \dots, m - 1$ . Die  $i$ -te Basisfunktion vom Grade  $p$  (Ordnung  $p + 1$ ) ergibt sich zu

$$N_{i,0}(t) = \begin{cases} 1 & \text{für } t_i \leq t < t_{i+1} \\ 0 & \text{ansonsten} \end{cases} \quad (2.11)$$

$$N_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t) . \quad (2.12)$$

Die einzelnen  $t_i$  werden als Träger, Knotenpunkt oder kurz als Knoten bezeichnet. Die Multiplizität gibt die Vielfachheit des Auftretens eines Knotens in der Folge  $\mathbf{t}$  an. Die folgenden Eigenschaften lassen sich aus der o.g. Definition der Basisfunktionen (vgl. auch Abbildung 2.4) ableiten.

1. Für ein  $t \notin [t_i, t_{i+p+1}]$  gilt  $N_{i,p}(t) = 0$ . Dieses wird als lokale Stütz- oder Trägereigenschaft bezeichnet.
2. In jedem Knotenintervall  $[t_j, t_{j+1})$  sind maximal  $p + 1$  der Basisfunktionen  $N_{i,p}(t)$  nicht null.
3. Die Eigenschaft für alle  $i, p, t : N_{i,p}(t) \geq 0$  wird als Nichtnegativität bezeichnet.
4. Die Basisfunktionen  $N_{i,p}(t)$  bilden für alle  $t \in [t_j, t_{j+1}]$  eine Zerlegung der Eins,

genauer gilt:

$$\sum_{i=j-p}^j N_{i,p}(t) = 1 .$$

5. Alle Ableitungen von  $N_{i,p}(t)$  existieren innerhalb eines Knotenintervalls. An den Knotenpunkten ist  $N_{i,p}(t)$  ( $p-k$ ) mal stetig differenzierbar, wobei  $k$  die Multiplizität des Knotenpunktes ist.
6. Für  $p > 0$  hat  $N_{i,p}(t)$  genau ein Maximum.

Die Ableitung einer Basisfunktion ist durch

$$N'_{i,p}(t) = \frac{p}{t_{i+p} - t_i} N_{i,p-1}(t) - \frac{p}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t) \quad (2.13)$$

gegeben. Mit Hilfe der soeben eingeführten Basisfunktionen folgt die Definition einer B–Spline–Kurve.

#### DEFINITION 5 (B–SPLINE–KURVE)

Es seien  $\{N_{i,p}(t)\}$  nach Definition 4 gebildete Basisfunktionen und  $\{\mathbf{P}_i\}$  Punkte im  $\mathbb{R}^2$  (bzw.  $\mathbb{R}^3$ ) mit  $i = 0, \dots, n$ . Eine parametrische B–Spline–Kurve  $B(t)$  vom Grade  $p$  (Ordnung  $p+1$ ) wird dann durch

$$B(t) = \sum_{i=0}^n N_{i,p}(t) \mathbf{P}_i \quad (2.14)$$

definiert.

Bei der Wahl der Knoten– und Kontrollpunkte sind die folgenden beiden Fälle zu unterscheiden:

#### 1. NICHT PERIODISCHE KURVE

Der Knotenvektor einer nicht periodischen Kurve ergibt sich zu

$$\mathbf{t} = \underbrace{\{t_0, t_1, \dots, t_p, t_{p+1}, \dots, t_{m-p-1},}_{p+1 \text{ Knoten}} \underbrace{t_{m-p}, \dots, t_m\}}_{p+1 \text{ Knoten}}, \quad (2.15)$$

wobei die ersten bzw. letzten  $p+1$  Knoten identisch sind, d.h.  $t_i = a$  und  $t_{m-p+i} = b$  für  $i = 0, \dots, p$ .

#### 2. PERIODISCHE KURVE

Der Knotenvektor einer periodischen Kurve ergibt sich zu

$$\mathbf{t} = \underbrace{\{t_0, t_1, \dots, t_{p-1}, t_p, \dots, t_{m-p},}_{p \text{ Knoten}} \underbrace{t_{m-p+1}, \dots, t_m\}}_{p \text{ Knoten}}, \quad (2.16)$$

wobei die ersten bzw. letzten  $p$  Knoten eine periodische Fortsetzung vorangehender Knoten sind, d.h.  $t_{p-i-1} = t_{p-i} - (t_{n-i+1} - t_{n-i})$  und  $t_{n+i+2} = t_{n+i+1} + (t_{p+i+1} - t_{p+i})$  für  $i = 0, \dots, p-1$ . Für die Kontrollpunkte gilt dementsprechend die Beziehung  $\mathbf{P}_{n+1-p+i} = \mathbf{P}_i$  für  $i = 0, \dots, p-1$ . Die Periodenlänge ergibt sich zu  $n+1-p$ .

Die Kurve ist im nicht periodischen Fall im Intervall  $[a, b]$  und im periodischen Fall auf  $[t_p, t_{m-p}]$  definiert. Oftmals werden die Kontrollpunkte auch als *de Boor-Punkte* bezeichnet. Die lineare Verbindung der Kontrollpunkte  $\{\mathbf{P}_i\}$  wird analog zur Bézier-Funktion als Kontrollpolygon bezeichnet. Nachfolgend sind kurz die Eigenschaften von B-Spline-Kurven aufgeführt, die aus der obigen Definition und den Eigenschaften der Basisfunktionen unmittelbar folgen.

1. Der Grad  $p$ , die Anzahl der Kontrollpunkte  $n + 1$  und die Anzahl der Knoten  $m + 1$  sind durch die Beziehung  $m = n + p + 1$  gegeben.
2. Für die nicht periodische Kurve fallen der Anfang bzw. das Ende der Kurve mit den entsprechenden Kontrollpunkten zusammen, d.h.  $B(a) = \mathbf{P}_0$  und  $B(b) = \mathbf{P}_n$ .
3. Eine affine Transformation, wie z.B. Translation, Rotation und Skalierung, ist invariant.
4. Für  $t \in [t_i, t_{i+1}]$  und  $p \leq i < m - p - 1$  liegt  $B(t)$  in der durch die Kontrollpunkte  $\mathbf{P}_{i-p}, \dots, \mathbf{P}_i$  erzeugten konvexen Hülle (vgl. Definition 3). Zusätzlich gilt die *variation diminishing* Eigenschaft.
5. Wird der Kontrollpunkt  $\mathbf{P}_i$  verändert, so wirkt sich dieses nur auf den vom Parameterintervall  $[t_i, t_{i+p+1}]$  bestimmten Teil der Kurve  $B(t)$  aus. Der Kurvenpunkt  $B(t')$  mit dem Parameter  $t' \in [t_i, t_{i+1}]$  wird nur von den Kontrollpunkten  $\mathbf{P}_i, \dots, \mathbf{P}_{i-p}$  beeinflusst.
6. Das Kontrollpolygon stellt eine stückweise lineare Approximation der Kurve dar. Je geringer der Grad  $p$ , umso näher liegt die Kurve am Kontrollpolygon. Im Fall  $p = 1$  ist die Kurve und das Kontrollpolygon identisch.

Aus der Differenzierbarkeit der Basisfunktionen folgt unmittelbar die Differenzierbarkeit der gesamten B-Spline-Kurve. Die  $k$ -te Ableitung einer B-Spline-Funktion ist unter Beachtung von Gleichung (2.13) durch

$$B^{(k)}(t) = \sum_{i=0}^n N_{i,p}^{(k)}(t) \mathbf{P}_i \quad (2.17)$$

gegeben.

In Abbildung 2.5 ist der Graph einer periodischen B-Spline-Funktion und des Parameterraums mit dem Knotenvektor  $\mathbf{t}$  dargestellt.

B-Splines teilen die folgenden Vorteile mit den Bézier-Kurven. Die Kontrollpunkte  $\{\mathbf{P}_i\}$  haben eine geometrische Bedeutung und stehen mit dem Graphen visuell in direktem Zusammenhang. Im Gegensatz zur Bézier-Kurve hat eine Modifikation aufgrund der Trägereigenschaft jedoch nur eine lokale Auswirkung, wodurch detailliertes Arbeiten möglich ist. Bézier-Splines haben eine Vielzahl von gegenseitig abhängigen Parametern, die eine sinnvolle Benutzerinteraktion erschweren. B-Splines haben diese Beschränkungen nicht.

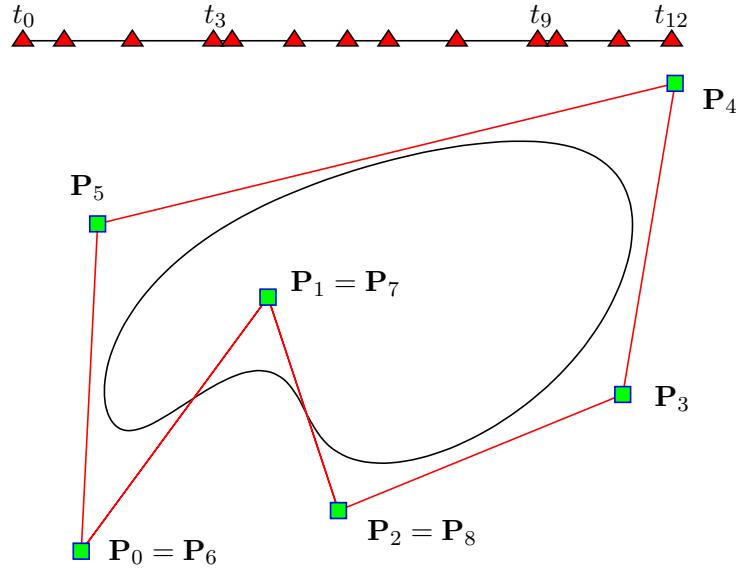


Abbildung 2.5: Die B–Spline–Funktion  $B(t)$  mit  $p = 3, m = 12$  und  $n = 8$  für ein  $t \in [t_3, t_9]$  liegt in der konvexen Hülle des durch  $\{\mathbf{P}_i\}$  aufgespannten Polygons. Oberhalb des Graphen ist der Knotenvektor  $\mathbf{t}$  mit den periodischen Rändern dargestellt.

Die Kontrollpunkte  $\{\mathbf{P}_i\}$ , die Knoten  $\mathbf{t}$  und der Grad  $p$  bilden die sogenannten Designparameter, die verändert werden können, um eine Kurve mit bestimmten Eigenschaften zu formulieren. Für die Konstruktion von Konturen sind insbesondere die periodischen B–Spline–Kurven wichtig, da diese per Definition geschlossen sind.

Bei der effizienten Berechnung von Kurvenpunkten braucht nicht direkt auf die rekursive Definition 4 zurückgegriffen werden, da ein erheblicher Teil der Basisfunktionen null ist. Der sogenannte *Algorithmus von de Boor* nutzt die lokale Trägereigenschaft aus. Im Anhang A.1 ist eine Variante von PIEGL UND TILLER (1995b) angegeben.

### 2.2.3 Approximation und Interpolation

Zwei fundamentale Algorithmen für B–Spline–Funktionen sind die Approximation und Interpolation vorgegebener Punkte  $\mathbf{Q}_k$  mit  $k = 0, \dots, r$ . Dieses ist insbesondere dann wichtig, wenn vorhandene Schaufelprofile, zum Beispiel aus den NACA oder C4 Profilfamilien, in die mathematische B–Spline–Form überführt und somit in nachfolgenden Schritten, z.B. Strömungsberechnung, weiterverwendet werden sollen. Diskrete Approximationsverfahren haben gegenüber den Methoden der Interpolation den Vorteil, daß die Anzahl der B–Spline Parameter (Knotenvektor und Kontrollpunkte) unabhängig von der Größe der vorliegenden Punktmenge ist.

### Parametrisierung

Ein nicht zu unterschätzendes Problem ist die Parametrisierung eines B-Splines. Der Knotenvektor  $\mathbf{t}$  stellt einen Freiheitsgrad bei der Interpolations- bzw. Approximationsaufgabe dar. Ungünstig gewählte Parametrisierungen führen zu Schleifen oder spitzen Ecken im Graphen der B-Spline-Funktion, oder die im weiteren auftretenden Gleichungssysteme werden dadurch singulär.

Die zugrundeliegende Idee ist es, den zu approximierenden bzw. interpolierenden Punkten<sup>2</sup> die Parameterwerte  $\bar{t}_k$  durch  $\mathbf{Q}_k = B(\bar{t}_k)$  so zuzuordnen, daß diese der Verteilung der  $\{\mathbf{Q}_k\}$  entsprechen. Die sich anschließende Bestimmung des Knotenvektors  $\mathbf{t}$  aus der Parametrisierung  $\bar{\mathbf{t}} = \{\bar{t}_0, \dots, \bar{t}_r\}$  ist von den einzelnen Approximations- bzw. Interpolationsverfahren abhängig und wird in den entsprechenden Abschnitten vorgestellt.

**Äquidistante Parametrisierung** Der naheliegende Ansatz einer Parametrisierung mit äquidistanten Abständen, d.h.

$$\bar{t}_k = \frac{k}{r+1} \quad \text{für } k = 1, \dots, r \quad \text{mit } \bar{t}_0 = 0 \quad \text{und } \bar{t}_{r+1} = 1 \quad , \quad (2.18)$$

liefert für die meisten Fälle (wenn die Punkte nicht ungefähr gleichen Abstand haben) unbefriedigende Kurven. Um ein möglichst gutes Approximations- bzw. Interpolationsergebnis zu erhalten, bietet sich daher u.a. eines der beiden folgenden Parametrisierungsverfahren an.

**Chordale Parametrisierung** Für die  $r + 1$  Punkte  $\{\mathbf{Q}_k\}$  läßt sich die Bogenlänge  $d$  mit

$$d = \sum_{k=0}^r |\mathbf{Q}_k - \mathbf{Q}_{k-1}| \quad \text{mit } \mathbf{Q}_{-1} = \mathbf{Q}_r \quad (2.19)$$

berechnen. Mit  $\bar{t}_0 = 0$  und  $\bar{t}_{r+1} = 1$  ergeben sich dann die inneren Parameterwerte zu

$$\bar{t}_k = \bar{t}_{k-1} + \frac{|\mathbf{Q}_k - \mathbf{Q}_{k-1}|}{d} \quad \text{für } k = 1, \dots, r. \quad (2.20)$$

Dieses ist die am häufigsten eingesetzte Methode der Parametrisierung.

**Zentripetale Parametrisierung** Die zentripetale Parametrisierung zeigt bei Punktverläufen mit großen Krümmungsänderungen, wie sie zum Beispiel bei dünnen Verdichterprofilen mit scharfen Vorder- und Hinterkanten vorkommen, ein besseres Approximations- bzw. Interpolationsverhalten. Sie berechnet sich für  $k = 1, \dots, r$  mit  $\bar{t}_0 = 0$  und  $\bar{t}_{r+1} = 1$  zu

$$\bar{t}_k = \bar{t}_{k-1} + \frac{\sqrt{|\mathbf{Q}_k - \mathbf{Q}_{k-1}|}}{d} \quad \text{mit } d = \sum_{k=0}^r \sqrt{|\mathbf{Q}_k - \mathbf{Q}_{k-1}|} \quad \text{und } \mathbf{Q}_{-1} = \mathbf{Q}_r. \quad (2.21)$$

---

<sup>2</sup>Die  $r + 1$  Punkte  $\{\mathbf{Q}_k\}$  sind nach Gleichung (2.16) periodisch fortzusetzen.

### Methode der kleinsten Fehlerquadrate

Das im folgenden beschriebene Verfahren approximiert vorgegebene Punkte mit der Methode der kleinsten Fehlerquadrate. Hierbei handelt es sich um eine Adaption des von PIEGL UND TILLER (1995b) vorgestellten Verfahrens an periodische B-Spline-Kurven.

Das Approximationsproblem besteht aus der Bestimmung einer nach Definition 5 beschriebenen periodischen B-Spline-Kurve  $B(t)$ , die eine vorgegebene Anzahl  $r + 1$  von zweidimensionalen<sup>3</sup> Punkten  $\mathbf{Q}_0, \dots, \mathbf{Q}_r$  hinreichend genau approximiert. Die Methode der kleinsten Fehlerquadrate fordert:

$$\text{Minimiere } \sum_{k=0}^r (\mathbf{Q}_k - B(\bar{t}_k))^2 . \quad (2.22)$$

Der Grad  $p$  und die Anzahl der Kontrollpunkte  $n + 1$  werden vorgegeben. Die Parameterwerte  $\bar{\mathbf{t}} = \{\bar{t}_0, \bar{t}_1, \dots, \bar{t}_r\}$  lassen sich nach der chordalen oder zentripetalen Methode berechnen. Anschließend sind nun  $m + 1$  Knoten  $t_i$  zu bestimmen, wobei  $2p$  Knoten periodische Fortsetzungen sind. Mit  $t_p = 0$  und  $t_{m-p} = 1$  lässt sich nun aus den berechneten Parameterwerten durch

$$\begin{aligned} d &= \frac{m+1}{m-2p} & i &= \lfloor jd \rfloor & \alpha &= jd - i \\ t_{p+j} &= (1-\alpha)\bar{t}_{i-1} + \alpha\bar{t}_i & \text{für } j &= 1, \dots, r \end{aligned} \quad (2.23)$$

der Knotenvektor  $\mathbf{t}$  der B-Spline-Kurve ermitteln. Diese Berechnung stellt sicher, daß sich in jedem Knotenintervall  $[t_i, t_{i+1}]$  mindestens ein Parameterwert  $\bar{t}_j$  befindet. Das nun folgende Gleichungssystem wird dadurch positiv definit und lösbar.

Im letzten Schritt sind jetzt die  $n+1-p$  Kontrollpunkte<sup>4</sup> zu berechnen. Aus der Forderung (2.22) ergibt sich nun nach Differentiation bezüglich  $\mathbf{P}_i = (x_i, y_i)$  und zu null setzen das folgende Gleichungssystem:

$$\sum_{i=0}^{n-p} \left( \sum_{k=0}^m N_{i,p}(\bar{t}_k) N_{l,p}(\bar{t}_k) \right) \mathbf{P}_i = \sum_{k=0}^m N_{l,p}(\bar{t}_k) \mathbf{Q}_k \quad \text{für } l = 0, \dots, n-p . \quad (2.24)$$

In die Matrizenschreibweise umgeformt:

$$(NN^T)\mathbf{P} = \mathbf{Q} \quad (2.25)$$

mit

$$N = \begin{bmatrix} N_{0,p}(\bar{t}_0) & \cdots & N_{0,p}(\bar{t}_m) \\ \vdots & \ddots & \vdots \\ N_{n-p,p}(\bar{t}_0) & \cdots & N_{n-p,p}(\bar{t}_m) \end{bmatrix} \quad \text{und} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_0 \\ \vdots \\ \mathbf{P}_{n-p} \end{bmatrix} \quad (2.26)$$

<sup>3</sup>Eine Übertragung auf drei- oder höherdimensionale B-Spline-Funktionen ergibt sich analog.

<sup>4</sup>Es reicht an dieser Stelle aus, nur die Kontrollpunkte ohne periodische Fortsetzung zu betrachten.

sowie

$$\mathbf{Q} = \begin{bmatrix} N_{0,p}(\bar{t}_0)\mathbf{Q}_0 + \cdots + N_{0,p}(\bar{t}_m)\mathbf{Q}_m \\ \vdots \\ N_{n-p,p}(\bar{t}_0)\mathbf{Q}_0 + \cdots + N_{n-p,p}(\bar{t}_m)\mathbf{Q}_m \end{bmatrix}. \quad (2.27)$$

Die Matrix  $NN^T$  hat Bandstruktur mit der Bandbreite  $p+1$ . Das Gleichungssystem lässt sich nach dem Gauß–Verfahren ohne Pivotstrategie lösen.

Zur Berechnung des Approximationsfehlers ist eine Punktinversion nötig, d.h. zu einem gegebenen Punkt  $\mathbf{Q}_k$  ist der Kurvenpunkt  $B(\hat{t}_k)$  zu bestimmen, der den Abstand minimiert. Das Verfahren wird von PIEGL UND TILLER (1995b) vorgestellt. Sind die Parameterwerte  $\hat{t}_k$  mit  $k = 0, \dots, r$  berechnet, lässt sich der mittlere Approximationsfehler durch

$$\bar{E} = \frac{1}{r+1} \sum_{k=0}^r |B(\hat{t}_k) - \mathbf{Q}_k| \quad (2.28)$$

und der maximale Approximationsfehler durch

$$E_{\max} = \max_{k=0, \dots, r} |B(\hat{t}_k) - \mathbf{Q}_k| \quad (2.29)$$

angeben. Soweit nicht anders beschrieben, wird der Fehler zusätzlich auf die Sehnenlänge des Profils normiert ( $\bar{E}^*, E_{\max}^*$ ). In Abbildung 2.6 ist die Approximation mit  $\bar{E}^* = 0.001$  und  $E_{\max}^* = 0.005$  eines VKI–1 Profils (vgl. WATZLAWICK (1991)) mit einem periodischen B-Spline aufgeführt.

Zur Rechenzeitreduzierung der sich der Approximation des Profils anschließenden aerodynamischen Optimierung ist eine geringe Anzahl von freien Parametern zur Beschreibung der Geometrie wünschenswert. In Abschnitt 5.2.3 wird das hier vorgestellte Approximationsverfahren um einen Ansatz zur Minimierung der Geometrieparameter erweitert.

Eine ähnliche Vorgehensweise vergleichbar dem Entwurf von Turbomaschinenschaufeln ergibt sich beim Design von Tragflächen von Flugzeugen. Insofern wird an dieser Stelle kurz auf die Arbeit von BOEHM (1987) eingegangen, der die Approximation von NACA Flügelprofilen mit Bézier–Kurven beschreibt. Die aus einer Dicken– und Wölbungsverteilung definierten NACA Profile können durch Bézier–Kurven siebten Grades exakt beschrieben werden. Um den hohen Grad zu vermeiden, wurden die Profile mit Bézier–Kurven dritten Grades an 5 äquidistanten Punkten Hermite interpoliert. Eine wesentlich geringere Abweichung von den Originalprofilen wurde durch die Approximation mit Hilfe der Methode der kleinsten Fehlerquadrate erzielt.

## Interpolation

Bei der Interpolation der Punkte  $\{\mathbf{Q}_k\}$  ergeben sich  $n+1$  Gleichungen<sup>5</sup> der Form

$$\mathbf{Q}_k = B(\bar{t}_k) = \sum_{i=0}^n N_{i,k}(\bar{t}_k) \mathbf{P}_i \quad \text{mit } k = 0, \dots, n. \quad (2.30)$$

---

<sup>5</sup> $n+1 = r+p+1$  ist die Anzahl der zu interpolierenden Punkte inklusive der periodischen Fortsetzung.

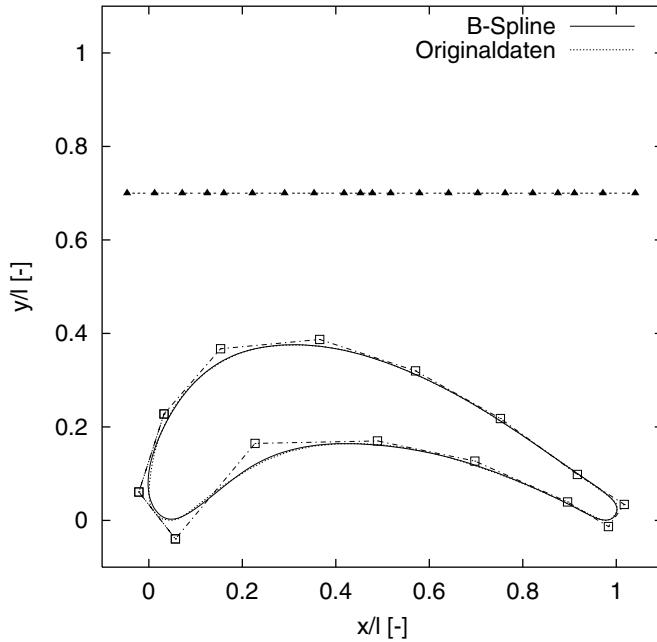


Abbildung 2.6: Die Qualität der Approximation des VKI-1 (vgl. WATZLAWICK (1991)) Profils ist von der Anzahl der Kontrollpunkte abhängig. Für dieses Beispiel ist eine Konfiguration mit  $p = 3$ ,  $m = 20$  und  $n = 16$  gewählt worden.

Analog zur Approximation ist in einem ersten Schritt eine, beispielsweise chordale oder zentripetale, Parametrisierung von  $\bar{\mathbf{t}}$  und daraus der Knotenvektor  $\mathbf{t}$  zu berechnen. Zu beachten ist, daß die Anzahl der Knoten  $t_i$  von der Anzahl der zu interpolierenden Punkte im Gegensatz zur Methode der kleinsten Fehlerquadrate abhängig ist. Für einen ungeraden Grad  $p$  werden die Knoten  $t_i$  direkt berechnet, d.h.

$$t_{p+i} = \bar{t}_i \quad \text{für } i = 0, \dots, r+1 \quad , \quad (2.31)$$

ansonsten für ein gerades  $p$  ergibt sich

$$t_p = \bar{t}_0 - (\bar{t}_r - \bar{t}_{r-1}) \quad \text{und} \quad t_{p+i} = t_{p+i-1} + (\bar{t}_i - \bar{t}_{i-1})/2 \quad \text{für } i = 1, \dots, r+1 \quad . \quad (2.32)$$

In beiden Fällen ist der Knotenvektor an den Rändern periodisch fortzusetzen. Die aus Gleichung (2.30) folgende  $(n+1) \times (n+1)$  Matrix läßt sich nun mit einem Gauß–Verfahren lösen, wobei zu beachten ist, daß nur die  $n+1-p$  nicht periodischen Punkte betrachtet werden müssen.

In der Abbildung 2.7 ist links zur Verdeutlichung ein einfaches Interpolationsproblem dargestellt. Die 4 Punkte  $\mathbf{Q}_0, \dots, \mathbf{Q}_3$  sind durch einen geschlossenen B–Spline mit  $p = 3$  zu interpolieren. Aus der Parametrisierung  $\bar{\mathbf{t}} = \{0, 0.25, 0.5, 0.75, 1\}$  läßt sich dann der

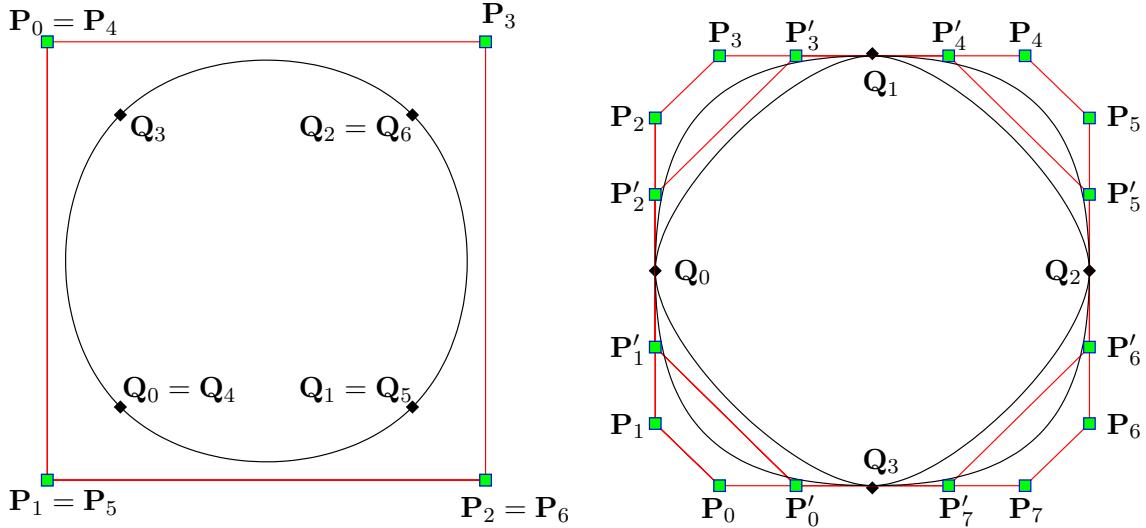


Abbildung 2.7: Zwei Beispiele verdeutlichen die Interpolation von Punkten  $\{\mathbf{Q}_k\}$  mit periodischen B-Spline-Funktionen. Im rechten Bild ist zusätzlich die erste Ableitung  $\mathbf{D}_k = \alpha_k \mathbf{T}_k$  zu berücksichtigen.  $\alpha_k$  ist ein zusätzlicher Formfaktor bei der Interpolation, der sich auf die gesamte Kurve auswirkt. Zur besseren Übersicht ist bei der rechten Abbildung auf die Beschriftung der periodischen Punkte verzichtet worden.

Knotenvektor  $\mathbf{t} = \{-0.75, -0.5, -0.25, 0, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75\}$  unter Verwendung von Gleichung (2.31) unmittelbar ableiten. Anschließend ist das Gleichungssystem nach Formel (2.30) aufzustellen und für die Punkte  $\mathbf{P}_0, \dots, \mathbf{P}_3$  zu lösen.

### Interpolation mit ersten Ableitungen

Die Interpolationsaufgabe kann dadurch erweitert werden, daß an den  $n + 1$  Punkten  $\{\mathbf{Q}_k\}$  auch die jeweils erste Ableitung  $\{\mathbf{D}_k\}$  interpoliert werden soll, d.h. für die  $2(n + 1)$  Gleichungen

$$\mathbf{Q}_k = B(\bar{t}_k) = \sum_{i=0}^{2n+1} N_{i,p}(\bar{t}_k) \mathbf{P}_i \quad \text{und} \quad \mathbf{D}_k = B'(\bar{t}_k) = \sum_{i=0}^{2n+1} N'_{i,p}(\bar{t}_k) \mathbf{P}_i \quad (2.33)$$

mit  $k = 0, \dots, n$  sind die unbekannten Kontrollpunkte  $\mathbf{P}_i$  zu bestimmen. Aus der Parametrisierung ist nun der Knotenvektor  $\mathbf{t}$  zu berechnen, wobei für jedes  $\bar{t}_i$  zwei Knoten notwendig sind. Für ein ungerades  $p$  ist

$$t_{p+2i} = \bar{t}_i \quad \text{und} \quad t_{p+2i+1} = (\bar{t}_i + \bar{t}_{i+1})/2 \quad \text{für } i = 0, \dots, r+1 \quad (2.34)$$

möglich. Ist  $p$  gerade, dann ist diese Methode nicht möglich. Statt dessen können doppelte Knoten verwendet werden.

Oftmals liegen nicht die Ableitungen, sondern nur die Richtungen in Form eines Einheitstangentenvektors  $\{\mathbf{T}_k\}$  vor. Hierzu lassen sich die Ableitungen durch  $\mathbf{D}_k = \alpha_k \mathbf{T}_k$  abschätzen. Eine Möglichkeit ist es,  $\alpha_k = d$  anzusetzen, wobei  $d$  die Länge des von  $\{\mathbf{Q}_k\}$  aufgespannten Polygons (Bogenlänge) ist. Für interaktive Designwerkzeuge bietet sich dieses als zusätzliche Operation zur Veränderung der Kontur an.

In Abbildung 2.7 ist rechts der Einfluß des Formfaktors  $\alpha_k$  anhand zweier interpolierter B-Spline Funktionen mit den Kontrollpunkten  $\{\mathbf{P}_i\}$  und  $\{\mathbf{P}'_i\}$  dargestellt.

### Direkte Berechnung des Kontrollpolygons

Eine wesentlich einfachere Methode zur schnellen Geometrieerzeugung ist es, die Kontrollpunkte mit den vorgegebenen Punkten gleichzusetzen, d.h.  $\mathbf{P}_k = \mathbf{Q}_k$  für  $k = 0, \dots, n$ . Ein so erzeugtes Profil kann anschließend entweder interaktiv weiterbearbeitet oder zum Beispiel als Startgeometrie für einen Optimierungsprozeß verwendet werden.

Dieses Verfahren ist keine Approximation bzw. Interpolation im herkömmlichen Sinn, sondern eine schnelle Methode, aus analytischen, durch Wölbungs- und Dickenverteilungen zusammengesetzten, Profilen eine angenäherte B-Spline-Form zu erhalten. Hierbei wird die Eigenschaft der B-Spline-Funktionen ausgenutzt, die das Kontrollpolygon als stückweise Approximation des Graphen beschreibt. In Abbildung 2.8 ist an einem einfachen Beispiel die Vorgehensweise verdeutlicht.

#### 2.2.4 Erweiterungen

Die hier beschriebenen B-Spline-Funktionen sind ein mächtiges mathematisches Werkzeug zur Beschreibung von zwei- und dreidimensionalen geometrischen Objekten. Sie bilden die Grundlage der in dieser Arbeit ausgeführten Strömungsberechnung und Optimierung von Schaufelprofilen. Die meisten (kommerziellen) CAD/CAM/CAE Programme bieten Schnittstellen für B-Splines in Form von definierten Austauschformaten an, beispielsweise IGES (*Initial Graphics Exchange Specification*), so daß eine Weiterverarbeitung, z.B. in rechnergestützten Fräsmaschinen, ohne Informations- und Qualitätsverlust möglich ist.

Gleichwohl existieren eine Reihe von Ansätzen, um B-Splines an spezielle Anforderungen zu erweitern bzw. zu ergänzen, die an dieser Stelle nur kurz angerissen werden können.

B-Splines fordern an den Berührungs punkten der einzelnen Segmente gewisse *analytische* Stetigkeitsbedingungen  $C^r$ . Für zwei Kurven  $C_0(t)$  mit  $t \in [t_0, t_1]$  und  $C_1(u)$  mit  $u \in [u_0, u_1]$  fordert die  $C^1$ -Stetigkeit in einem gemeinsamen Punkt  $C_0(t_1) = C_1(u_0)$  die Übereinstimmung der Ableitungen  $C'_0(t_1) = C'_1(u_0)$ . Oftmals ist jedoch diese Forderung zu strikt, statt dessen würde eine *geometrische* Stetigkeit  $GC^r$  den Ansprüchen genügen. So fordert die  $GC^1$  Stetigkeit die Bedingung, daß nur die Tangentenrichtungen an den aneinanderstoßenden Segmenten gleich sind, d.h.  $C'_0(t_1) = \alpha C'_1(u_0)$  für  $\alpha > 0$ .  $GC^2$  stetige Splines erfüllen zusätzlich an den Segmentgrenzen noch die Bedingung gleicher Krümmung. Während für B-Splines eine Reihe von effizienten Algorithmen existieren, sind bei geometrischen Splines wesentlich kompliziertere Verfahren notwendig. KUPPE (1997) vergleicht die Approximation von Schaufelprofilen mit B-Splines sowie mit geometrischen Splines.

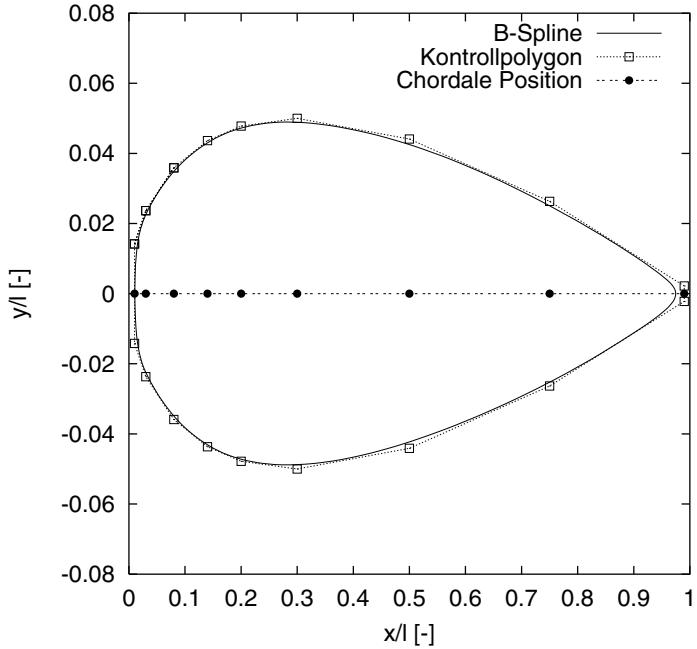


Abbildung 2.8: An den chordalen Position  $x/l = 0.01, 0.03, 0.08, 0.14, 0.2, 0.3, 0.5, 0.75, 0.99$  ist jeweils für die Ober- und Unterseite ein Konturpunkt berechnet worden (NACA-4 Dickenverteilung). Die so berechneten Punkte bilden das Kontrollpolygon des B-Splines.

Wie eingangs beschrieben, sollen die zur mathematischen Beschreibung verwendeten Funktionen einen großen Bereich von Kurven abdecken. Die hier vorgestellten B-Splines sind jedoch nicht in der Lage, Kreise oder Ellipsen exakt zu beschreiben, was jedoch durch einen Übergang zu rationalen B-Spline-Funktionen (auch als NURBS<sup>6</sup> bezeichnet) möglich ist:

$$C(t) = \sum_{i=0}^n R_{i,p}(t) \mathbf{P}_i \quad \text{mit} \quad R_{i,p}(t) = \frac{N_{i,p}(t) w_i}{\sum_{j=0}^n N_{j,p}(t) w_j} . \quad (2.35)$$

Die Funktionen  $\{R_{i,p}\}$  werden als rationale Basisfunktionen bezeichnet, die  $\{w_i\}$  sind zusätzliche Gewichte mit  $w_i > 0$ .

BARSKY (1988, 1993) führt  $\beta$ -Splines ein. Zwei zusätzliche Formparameter ermöglichen es, die Kurve global zu manipulieren.

Alle beschriebenen Erweiterungen haben den Nachteil zusätzlicher Parameter, die bestimmt werden müssen. Insbesondere bei der Optimierung wirken sich zusätzliche Frei-

<sup>6</sup>Das Akronym steht für *nonuniform rational b-spline*. Der Zusatz *nonuniform* bezieht sich auf den Knotenvektor  $\mathbf{t}$ , bei dem die einzelnen Knoten  $t_i$  nicht äquidistant verteilt sind. Die in dieser Arbeit verwendeten B-Spline Funktionen sind nicht uniform und nicht rational.

heitsgrade negativ auf die benötigte Rechenzeit aus. Ebenso sind die Algorithmen zur (interaktiven) Manipulation erweiterter Spline–Funktionen schlechter handhabbar.

## 2.3 3D–Geometrie

Zur Konstruktion einer Schaufel ist eine Folge von Profilen notwendig, die (entlang einer Achse) übereinander ausgerichtet werden (Auffädelung). Um aus diesen Profilen dreidimensionale Schaufeln und letztendlich damit Lauf– oder Leiträder zu erhalten, bedarf es eines geeigneten Verfahrens, um die Einhüllende zu berechnen.

### 2.3.1 Tensorprodukt–B–Spline–Flächen

Die zugrundeliegende Idee der B–Spline–Kurve kann ohne weiteres auf im Raum liegende Flächen ausgeweitet werden. Diese führt dann auf die folgende Definition.

#### DEFINITION 6 (TENSORPRODUKT–B–SPLINE–FLÄCHE)

Es seien  $\{N_{i,p}\}$  und  $\{N_{j,q}\}$  nach Definition 4 gebildete Basisfunktionen mit den zugehörigen Knotenvektoren  $\mathbf{u}$  und  $\mathbf{v}$  sowie den Kontrollpunkten  $\{\mathbf{P}_{i,j}\}$  im  $\mathbb{R}^3$  für  $i = 0, \dots, n$  und  $j = 0, \dots, m$ . Eine Tensorprodukt–B–Spline–Fläche wird dann durch

$$S(u, v) = \sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) \mathbf{P}_{i,j} \quad (2.36)$$

definiert.

Von den vier möglichen Fällen zur Auswahl der Knoten– und Kontrollpunkte ist für die Konstruktion von Schaufeln ein periodischer Verlauf in  $u$  und ein nicht periodischer in  $v$  Richtung von Interesse. Die Knotenvektoren ergeben sich also zu:

$$\mathbf{u} = \left\{ \underbrace{u_0, u_1, \dots, u_{p-1}}_p \text{Knoten}, u_p, \dots, u_{r-p}, \underbrace{u_{r-p+1}, \dots, u_r}_p \text{Knoten} \right\} \text{ bzw.} \quad (2.37)$$

$$\mathbf{v} = \left\{ \underbrace{v_0, v_1, \dots, v_q}_{q+1 \text{ Knoten}}, v_{q+1}, \dots, v_{s-q-1}, \underbrace{v_{s-q}, \dots, v_s}_{q+1 \text{ Knoten}} \right\}, \quad (2.38)$$

wobei die Bedingungen aus Definition 5 bezüglich der Periodizität im Fall von  $\mathbf{u}$  und der Nichtperiodizität im Fall von  $\mathbf{v}$  gelten.

Die Verbindung der einzelnen Kontrollpunkte in  $u$  und  $v$  Richtung wird als Kontrollnetz bezeichnet. Für den Grad  $p$ , die Anzahl der Kontrollpunkte  $n + 1$  in  $u$  Richtung und die Anzahl der Knoten  $r + 1$  gilt die Beziehung  $r = n + p + 1$ . Äquivalent gilt für die  $v$  Richtung  $s = m + q + 1$ .

Im nächsten Abschnitt werden die B–Spline–Funktionen verwendet, um erst zweidimensionale Konturen und daraus dann dreidimensionale Schaufeln zu definieren.

### 2.3.2 Skinning

Mit *Skinning*<sup>7</sup> (Bespannung) wird das Verfahren bezeichnet, welches dreidimensionale B-Spline-Kurven so miteinander verbindet, daß die dadurch entstehende Fläche diese interpoliert. Hierzu bieten sich die sogenannten Tensorprodukt-B-Spline-Flächen an. Das hier vorgestellte Verfahren folgt, erweitert auf periodische B-Spline-Funktionen, im wesentlichen PIEGL UND TILLER (1995b).

Es seien  $\{B_k(u)\}$  mit  $k = 0, \dots, K$  eine Menge von B-Spline-Kurven. Diese werden auch oftmals als Schnittkurven bezeichnet. Gesucht ist nun eine Tensorprodukt-B-Spline-Fläche  $S(u, v)$ , so daß für eine aufsteigende Folge  $v_k$  gilt:  $S(u, v_k) = B_k(u)$ , d.h. die B-Splines sind isoparametrische Kurven der Tensorproduktfläche. Mit

$$S(u, v_k) = \sum_{i=0}^n \sum_{j=0}^K N_{i,p}(u) N_{j,q}(v_k) \mathbf{P}_{i,j} \text{ und } B_k(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_{k_i} \text{ folgt} \quad (2.39)$$

$$\sum_{j=0}^K N_{j,q}(v_k) \mathbf{P}_{i,j} = \mathbf{P}_{k_i} \quad (2.40)$$

für  $k = 0, \dots, K$  und  $i = 0, \dots, n$ . Dieses führt zu  $n + 1$  Interpolationsproblemen in  $v$  Richtung mit jeweils  $K + 1$  Unbekannten und  $K + 1$  Bekannten. Zusammengefäßt läßt sich also die Vorgehensweise wie folgt beschreiben:

1. die einzelnen B-Spline-Kurven werden übereinander geschichtet,
2. der jeweils  $i$ -te Kontrollpunkt  $\mathbf{P}_{k_i}$  einer jeden B-Spline-Kurve  $k = 0, \dots, K$  wird durch eine B-Spline-Kurve interpoliert
3. dieses wird für alle  $n + 1$  Kontrollpunkte<sup>8</sup> durchgeführt,
4. die dadurch erhaltenen Kontrollpunkte bilden das Kontrollnetz der interpolierenden Tensorprodukt-B-Spline-Fläche.

Bisher ist stillschweigend davon ausgegangen worden, daß die einzelnen B-Spline-Kurven den gleichen Knotenvektor  $\mathbf{u}$  und den gleichen Grad  $p$  haben. Dieses ist i.allg. aber jedoch nicht der Fall. Es ist allerdings möglich, zwei oder mehr B-Spline-Kurven zueinander kompatibel umzuformen, so daß diese einen identischen Knotenvektor und einen gleichen Grad haben. Die Geometrie bleibt dabei erhalten.

Durch die Verwendung von geschlossenen, periodischen B-Spline-Funktionen bedarf es einer geeigneten Definition von Anfang und Ende der Kurve, da ansonsten durch die Interpolation der Kontrollpunkte in Spannweitenrichtung sehr stark verwundene Oberflächen entstehen. In Abbildung 2.9 ist die Problematik graphisch dargestellt. Links befinden sich zwei Profile übereinander, so daß durch die lineare Interpolation eine Oberfläche erzeugt

<sup>7</sup>Oftmals wird in der Literatur auch die Bezeichnung *lofting* synonym verwendet.

<sup>8</sup>An dieser Stelle reicht es aus, die ersten  $n + 1 - p$  Kontrollpunkte zu berechnen, da die weiteren eine periodische Fortsetzung vorangegangener bilden.

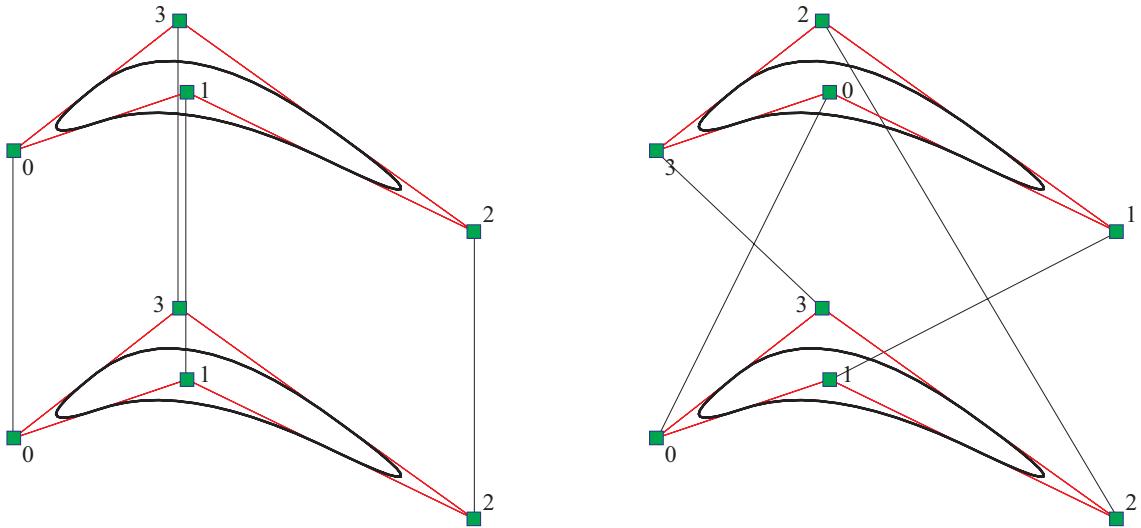


Abbildung 2.9: Optimierungsaufgabe im Skinning-Prozeß: Unterschiedliche Indizierungen ergeben verschiedenartige Verwindungen der dreidimensionalen Oberfläche.

wird, die keine oder nur eine minimale Verwindung aufweist. Im rechten Teil der Abbildung ist jedoch die Indizierung des oberen Profils entgegen dem Uhrzeigersinn um eins verschoben. Erfolgt nun eine lineare Interpolation, entsteht eine verwundene, unter Umständen sich selbst schneidende, Oberfläche. Obwohl in beiden Fällen die Ausgangsprofile geometrisch identisch sind, wird eine unterschiedliche Tensorproduktfläche generiert. Der Grund hierfür ist eine zyklische Verschiebung des Knotenvektors und der Kontrollpunkte. Da die Profile einzeln und unabhängig voneinander entworfen werden, zum Beispiel interaktiv am Bildschirm, ist nicht gewährleistet, daß eine geeignete Indizierung besteht. Eine Aufgabe im optimierten Skinning-Prozeß ist es daher, eine Indizierung zu finden, die eine zufriedenstellende Oberfläche ergibt.

Ein weiteres Problem ergibt sich bei der Wahl der  $v_k$  Werte für  $k = 0, \dots, K$ . Hierdurch wird in entscheidendem Maße die Oberflächengestalt in Spannweitenrichtung ( $v$ -Richtung) beeinflußt.

Es zeigt sich, daß der hier beschriebene *Skinning*-Ansatz für einfache dreidimensionale Geometrien zu befriedigenden Ergebnissen führt. Es ist, wie in Abschnitt 5.2.3 beschrieben, ein Verfahren zur Oberflächenoptimierung entwickelt worden. Für komplexe Schaufeln sind jedoch leistungsfähigere Methoden, wie zum Beispiel von BEDI UND VICKERS (1989) oder KAKLIS UND GINNIS (1996), notwendig.

## 2.4 Implementierte Verfahren

Im Rahmen dieser Arbeit sind eine Reihe von Werkzeugen entwickelt worden, die die beschriebenen Konzepte implementieren. Nachfolgend werden diese Programme kurz vor-

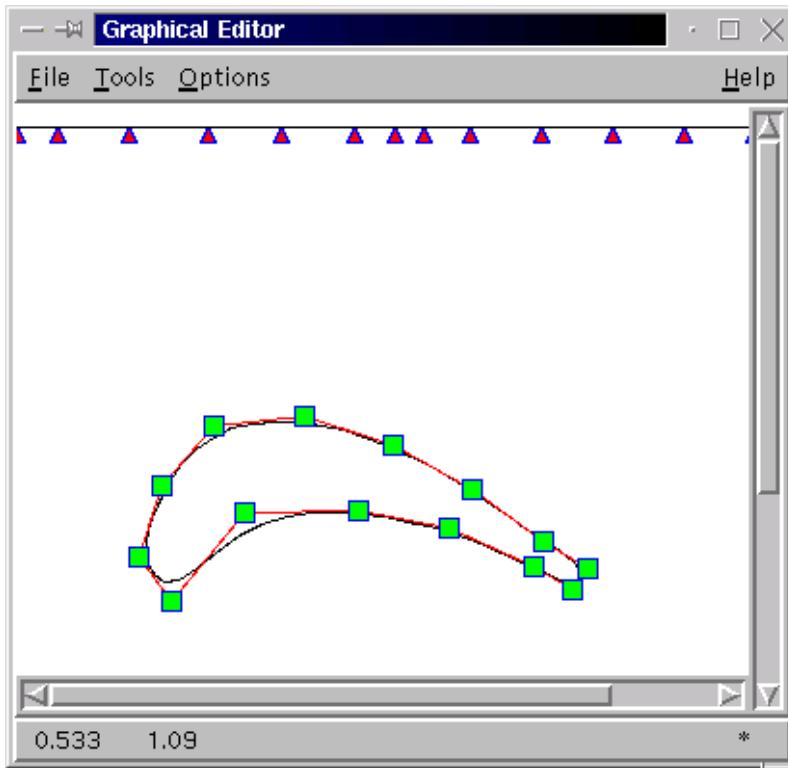


Abbildung 2.10: Der im Rahmen dieser Arbeit entwickelte `mudesigner` ermöglicht es, komfortabel Profile mit periodischen B-Spline Funktionen zu entwerfen.

gestellt.

#### `mufit`

Das Programm `mufit` ist das Kernstück zur automatischen Erzeugung von zweidimensionalen Profilen durch Approximation und Interpolation mit periodischen B-Spline-Funktionen. Hierzu sind die in Abschnitt 2.2.3 beschriebenen Verfahren implementiert worden. Durch die Kombination einer Auswahl (vgl. Anhang A.2) von Wölbungs- und Dickenverteilungen ist es möglich, eine Vielzahl von Profilen in eine B-Spline-Form zu überführen. Weiterhin besteht die Möglichkeit, Konturpunkte aus einer Datei einzulesen und weiter zu bearbeiten.

#### `mudesigner`

Die Abbildung 2.10 zeigt den im Rahmen der Arbeit entwickelten graphischen Editor. Der `mudesigner` enthält die üblichen Funktionen graphischer Editoren sowie spezielle, auf periodische B-Splines angepaßte, geometrische Operationen.

Der `mudesigner` und auch `mufit` enthalten eine Methode zur Berechnung der Wölbungs-

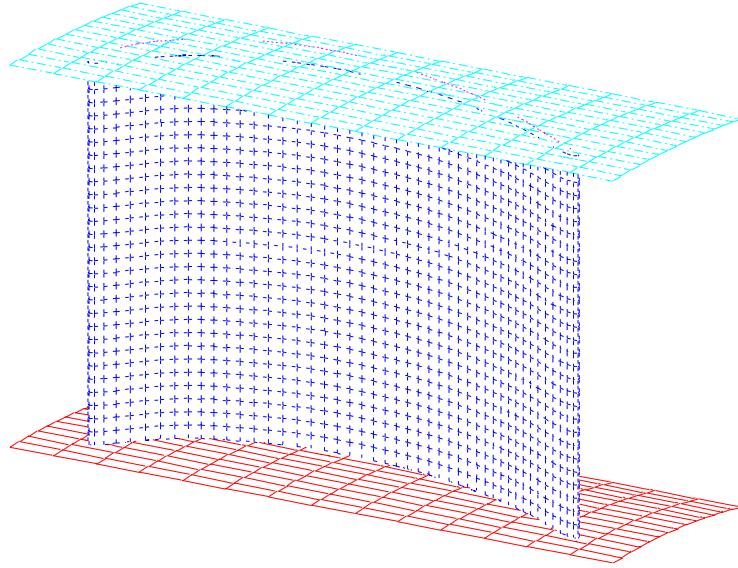


Abbildung 2.11: Das Werkzeug **mustacker** erzeugt aus einer Folge von radialen zweidimensionalen Profilschnitten eine dreidimensionale Schaufel als Tensorprodukt–B–Spline–Fläche.

linie eines als B–Spline definierten Profils. Hierdurch ist es möglich, eine Reihe von geometrischen Charakteristika, wie zum Beispiel die Vorder– und Hinterkante sowie die Metall– und Staffelungswinkel, zu ermitteln.

#### **mustacker**

Das Werkzeug **mustacker** dient zum Erstellen dreidimensionaler Schaufeln durch Auffädeln zweidimensionaler Profilschnitte, wie dieses exemplarisch in Abbildung 2.11 mit zwei Flächen für Nabe und Gehäuse dargestellt ist. Hierzu ist das in Abschnitt 2.3.2 vorgestellte Skinning–Verfahren implementiert worden.

**mustacker** ermöglicht es, aus einer beliebigen Anzahl von Profilschnitten eine Schaufel als Tensorprodukt–B–Spline–Fläche zu ermitteln und ist, wie in Abschnitt 5.2.3 beschrieben wird, durch ein zusätzliches Optimierungsverfahren erweitert worden, welches die Oberflächeneigenschaft der Schaufel bezüglich der Krümmung minimiert.

## 2.5 Andere Ansätze und Ausblick

Wie bereits eingangs erwähnt, sind Splines, insbesondere B–Spline–Kurven, ein häufig eingesetztes Werkzeug, so daß die vorhandene Literatur zu diesem Thema bereits nicht mehr vollständig darstellbar ist. Für die geometrische Datenverarbeitung sei stellvertretend auf HOSCHEK UND LASSER (1989) hingewiesen. Als umfangreichste Quelle bezüglich

nicht periodischer B-Splines ist das Buch von PIEGL UND TILLER (1995b) zu nennen, welches ihre zahlreichen Einzelpublikationen, zum Beispiel TILLER (1992) sowie PIEGL UND TILLER (1994, 1995a, 1997), zusammenfaßt und erweitert. Sämtliche Algorithmen werden ausführlich besprochen und in einer an die Programmiersprache C angelehnten Notation beschrieben.

Eine objektorientierte Beschreibung von B-Spline-Kurven sowie von Flächen wird von FUHR ET AL. (1995) vorgeschlagen. Eine Zusammenfassung wichtiger Methoden sind bei AU UND YUEN (1995) zu finden.

Bevor in den nächsten Kapiteln die hier vorgestellten periodischen B-Spline-Funktionen eingesetzt werden, sollen die verschiedenen, in der Praxis eingesetzten, Ansätze zur geometrischen Beschreibung von Turbinen- und Verdichterbeschauflungen reflektiert werden, die sich durch einen unterschiedlich starken Gestaltungsfreiraum auszeichnen.

Parametrisierte Profilfamilien (z.B. NACA oder C4), die die Kontur aus einer Dicken- und Wölbungsverteilung zusammensetzen, lassen dem Konstrukteur nur geringe Gestaltungsmöglichkeiten und werden in modernen Entwurfsprozessen zunehmend weniger verwendet. Weitaus größeren Spielraum läßt sich mit zusammengesetzten Bézier-Kurven erreichen, wobei die Vorder- und Hinterkanten oftmals durch einen Kreisbogenabschnitt definiert werden. SUBEL (1998) und BAIER (1998) folgen diesem Ansatz.

Die Anzahl der Segmente ist in den meisten Fällen *a priori* festgelegt. JERICHA ET AL. (1994) bzw. GEHRER ET AL. (1997) verwenden drei Bézier-Segmente für Saug- und Druckseite sowie für eine runde Hinterkante. KORAKIANITIS UND PANTAZOPOULOS (1993) beschreiben den Entwurf von Profilen von Turbinenschaufeln mit parametrischen Spline-Kurven. Der zweidimensionale Profilschnitt definiert sich durch jeweils vier Polynomsegmente vierten Ordnung an der Druck- und Saugseite, wobei an den Schnittpunkten bestimmte Differenzierbarkeitsbedingungen erfüllt sein müssen. Um die geometrischen Vorgaben einzuhalten, wird ein Optimierungsproblem mit einem nicht weiter genannten Suchverfahren formuliert. Ziel ist es, ein Profil mit minimalen Krümmungsgradienten zu finden. Wirkliche Freiformprofile lassen sich nur mit Bézier-Spline oder B-Spline-Kurven gestalten, wobei letztere den Vorteil haben, Stetigkeitsbedingungen einfacher zu garantieren sowie eine interaktive Manipulation mit graphischen Werkzeugen intuitiver zu gestalten. Die in dieser Arbeit entwickelte Methode bietet den flexibelsten Ansatz, da keine Einschränkungen bezüglich der Designparameter (Knoten, Kontrollpunkte und Grad) festgelegt werden. Eine weitere Erhöhung der Entwurfsmöglichkeiten ist nur durch den Übergang zu mathematisch komplizierteren und damit auch algorithmisch schwerer handhabbaren Funktionen, wie zum Beispiel NURBS, möglich.

## Kapitel 3

# Numerische Strömungsberechnung

Nachdem im vorherigen Kapitel der geometrische Entwurf von Profilen und Schaufeln vorgestellt worden ist, schließt sich nun die fluiddynamische Untersuchung an. Ziel ist es, einzelne zweidimensionale Profilschnitte aerodynamisch zu bewerten.

### 3.1 Überblick

Die strömungsmechanische Untersuchung und Bewertung von physikalischen Strömungsvorgängen wird durch eine Modellbildung und –beschreibung anhand mathematischer Beziehungen vollzogen. Hierbei handelt es sich i.allg. um partielle, analytisch bisher nicht lösbare, Differentialgleichungen.

Die Strömungsberechnung gliedert sich folglich in zwei Aufgaben: zum einen in die Diskretisierung des Strömungsraumes (Gittergenerierung) und zum anderen in die Aufstellung und numerische Behandlung der notwendigen Gleichungen zur Beschreibung der Strömungsphänomene.

Die Verfahren zur Gittergenerierung zerfallen grob in zwei Klassen. Die algebraischen Ansätze verwenden hauptsächlich verschiedene Interpolationsmethoden, die den Vorteil einer schnellen Berechnung besitzen. Im Gegensatz hierzu führt die zweite Klasse die Gittergenerierung auf die Lösung einer, i.allg. elliptischen, Differentialgleichung zurück. Die elliptische Gittergenerierung hat den Vorteil, daß sich die einzelnen Gitterlinien glatt an den umströmten Körper schmiegen. Die Grundlagen der Gittergenerierung sind u.a. in dem Standardwerk von THOMPSON ET AL. (1985) dargestellt.

Trotz der ständigen Steigerung der Leistungsfähigkeit der jeweils aktuellen Rechnergeneration sind zur Rechenzeitminimierung Vereinfachungen in der Modellbildung zu treffen. Im Hinblick auf eine Optimierung sind aussagekräftige Ergebnisse innerhalb von Minuten zu berechnen. Bei dem hier vorgestellten zweidimensionalen Stromfunktionsverfahren sind u.a. der Einfluß der Kompressibilität sowie die Grenzschichtausbildung auf der Profiloberfläche hervorzuheben. Die in dieser Arbeit entwickelte  $S_1$ -Stromflächenrechnung bedient sich der in LÜCKING (1982, 1989) bzw. RÖPER (1994) vorgestellten Methoden.

### 3.2 Elliptische Gittergenerierung

Die Voraussetzung zur eigentlichen numerischen Strömungsberechnung (vgl. nachfolgenden Abschnitt 3.3) ist die Diskretisierung des Strömungsraums insbesondere des Bereichs des umströmten Körpers (in diesem Fall des zweidimensionalen Schaufelprofils). Bei dieser Gittergenerierung<sup>1</sup> sind vornehmlich die folgenden Aspekte zu berücksichtigen:

- Gebiete, wie zum Beispiel die Vorder- und Hinterkanten des Profils, an denen große Veränderungen der Strömungsgrößen zu erwarten sind, sind feiner aufzulösen als homogene Randbereiche.
- Ein möglichst unkompliziertes Numerierungsschema zur Bestimmung von Nachbar-elementen gewährleistet, Gradienten durch finite Differenzen einfach zu berechnen.
- Die Diskretisierung des Strömungsraums sollte innerhalb eines angemessenen Zeit-raumes durchführbar sein.

Obwohl die Gittergenerierung inzwischen ein eigenes Forschungsfeld darstellt, kann diese nicht unabhängig von der eigentlichen Strömungsberechnung betrachtet werden. So sind zum Beispiel Randbedingungen der Strömungsberechnung mitzuberücksichtigen. Im vorliegenden Fall wird ein zweidimensionales Verfahren verwendet, welches auf der Lösung einer elliptischen partiellen Differentialgleichung (Poisson-Gleichung) beruht. Dieser Ansatz ist u.a. von CARSTENS (1988), SONAR (1989) bzw. KNUPP UND STEINBERG (1994) ausführlich beschrieben worden (vgl. ebenfalls HOLTMANN (1983), ANDERSON ET AL. (1996) und LEHTIMÄKI (1999)). An dieser Stelle werden die grundlegenden Beziehungen kurz dargestellt. Die nachfolgend aufgeführten Erweiterungen und Ergänzungen werden ausführlich diskutiert.

- Eine automatische Kantendetektion und das Aufschneiden des Profils an der Hin-terkante ermöglicht, die Gittergenerierung ohne Anwenderinteraktion durchzuführen.
- Es zeigt sich, daß insbesondere die Konturpunktverteilung Einfluß auf das Konvergenzverhalten der Strömungsberechnung hat. Durch einen neuen Ansatz werden die Konturpunkte anhand von Gewichtsfunktionen bestimmt.
- Die Erweiterung auf mehrstufige Gitter dient zur Beschleunigung der Konvergenz in der nachfolgenden Strömungsberechnung.

Der in dieser Arbeit entwickelte Gittergenerator ermöglicht, zusätzlich die von WANG UND SONI (1991) beschriebenen geometrischen und statistischen Koeffizienten zu berechnen. Eine qualitative Bewertung liefert somit Aussagen über mögliche Schwachstellen im Gitter, die in der Strömungsberechnung nicht erwünscht sind und sich i.allg. negativ auf die Konvergenzgeschwindigkeit auswirken.

---

<sup>1</sup>In dieser Arbeit werden die Bezeichnungen Gitter und Netz sowie die zugehörige Gitter- und Netzge-nerierung synonym verwendet.

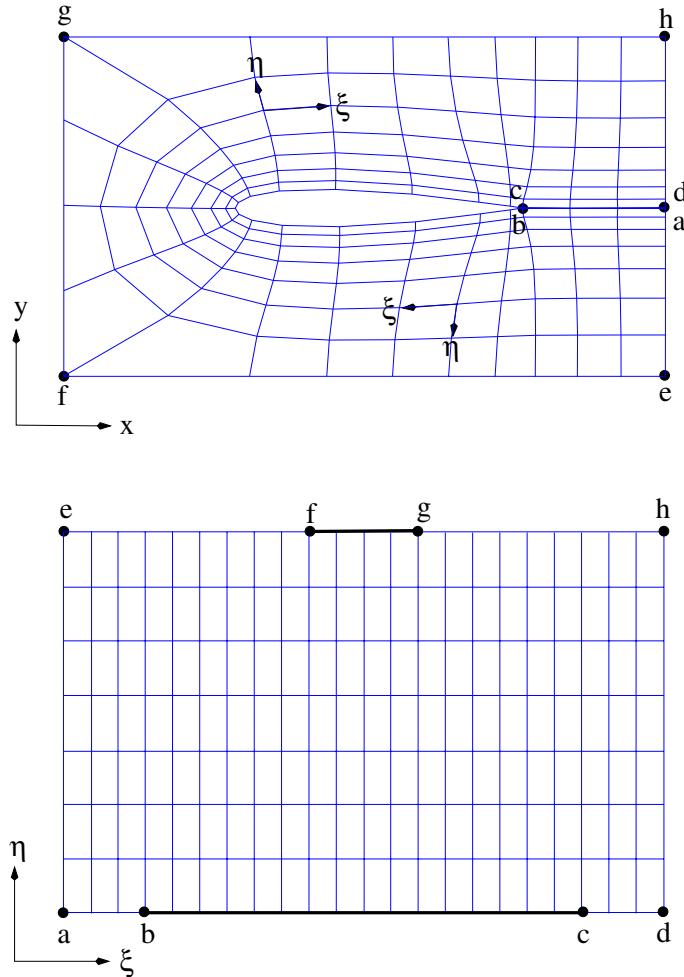


Abbildung 3.1: Durch eine Transformation wird ein krummliniges Koordinatensystem in den Rechenraum überführt. Bei dem dargestellten Typ handelt es sich um ein sogenanntes C–Gitter.

### 3.2.1 Transformation

Die Aufgabe der im folgenden beschriebenen Gittergenerierung ist die Bestimmung eines konturangepaßten Gitters, d.h. die Profilkontur selber definiert eine Gitterlinie. Hierzu wird ein krummliniges Koordinatensystem eingeführt, welches sich an den umströmten Körper schmiegt. Dessen Koordinaten  $(\xi, \eta)$  lassen sich analog wie kartesische Koordinaten im physikalischen Raum in ein rechtwinkliges System, im folgenden als Rechenraum bezeichneten, übertragen. Gesucht ist nun eine Abbildung oder eine Transformation, die die Gitterpunkte  $(\xi, \eta)$  kartesischen Punkten  $(x, y)$  zuordnet und umgekehrt (vgl. hierzu Abbildung 3.1). Es gilt also:

$$x = x(\xi, \eta) \quad \text{bzw.} \quad y = y(\xi, \eta) \quad \text{und} \quad \xi = \xi(x, y) \quad \text{bzw.} \quad \eta = \eta(x, y). \quad (3.1)$$

Zur Verdeutlichung der noch zu bestimmenden Transformation sind charakteristische Gitterpunkte in den beiden korrespondierenden Koordinatensystemen eingezeichnet. Bei dem dargestellten und in dieser Arbeit verwendeten Typ handelt es sich um ein sogenanntes *C*-Gitter (im Gegensatz zu *H*- oder *O*-Gittertypen). Die Gitterlinien in  $\xi$ -Richtung umlaufen die Kontur in Form eines Halbkreises (oder eben in Form des Buchstabens *C*). Die Strecken  $\overline{ab}$  bzw.  $\overline{cd}$  definieren eine Schnittkante, d.h. die entsprechenden Gitterpunkte sind identisch (vgl. Abbildung 3.1).

Zur Beschreibung der Transformation wird ein Ansatz in Form zweier Poissons-Gleichungen mit den Kontrollfunktionen  $P$  und  $Q$  verwendet:

$$\frac{\partial^2 \xi}{\partial x^2} + \frac{\partial^2 \xi}{\partial y^2} = P(\xi, \eta) \quad \text{und} \quad \frac{\partial^2 \eta}{\partial x^2} + \frac{\partial^2 \eta}{\partial y^2} = Q(\xi, \eta) . \quad (3.2)$$

Die beiden partiellen Differentialgleichungen beschreiben die Abhängigkeit der krummlinigen Koordinaten von denen des physikalischen Raums. Da diese jedoch unbekannt und daher zu bestimmen sind, ist eine Invertierung der beiden Gleichungen notwendig (die Herleitung ist z.B. in HOLTmann (1983) zu finden). Der funktionale Zusammenhang der invertierten Gleichungen beschreibt die der elliptischen Gittergenerierung zugrundeliegenden Differentialgleichungen in der folgenden Form:

$$\alpha \frac{\partial^2 x}{\partial \xi^2} - 2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 x}{\partial \eta^2} = -J^2 \left( \frac{\partial x}{\partial \xi} P(\xi, \eta) + \frac{\partial x}{\partial \eta} Q(\xi, \eta) \right) \quad (3.3)$$

$$\alpha \frac{\partial^2 y}{\partial \xi^2} - 2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 y}{\partial \eta^2} = -J^2 \left( \frac{\partial y}{\partial \xi} P(\xi, \eta) + \frac{\partial y}{\partial \eta} Q(\xi, \eta) \right) \quad (3.4)$$

$$\text{mit } \alpha = \left( \frac{\partial x}{\partial \eta} \right)^2 + \left( \frac{\partial y}{\partial \eta} \right)^2 , \quad \beta = \frac{\partial x}{\partial \xi} \frac{\partial x}{\partial \eta} + \frac{\partial y}{\partial \xi} \frac{\partial y}{\partial \eta} , \quad \gamma = \left( \frac{\partial x}{\partial \xi} \right)^2 + \left( \frac{\partial y}{\partial \xi} \right)^2 \quad (3.5)$$

$$\text{sowie } J = \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} . \quad (3.6)$$

Zur Lösung sind neben den Randbedingungen die Kontrollfunktionen  $P$  und  $Q$  zu definieren. Der folgende Abschnitt beschreibt eine Methode, die es gestattet, Kontraktion und Orthogonalität an den Rändern zu erhalten.

### 3.2.2 Der Ansatz von Steger und Sorensen

Wie bereits eingangs erwähnt, ist es für die Strömungsberechnung notwendig, eine Kontraktion der Gitterlinien, insbesondere an der Vorder- und Hinterkante, sowie Orthogonalität zwischen den Gitterlinien in  $\xi$ - und  $\eta$ -Richtung zu fordern. Da jedoch neben den Positionen auf dem Rand auch die Winkel und damit die Steigung vorgegeben werden, ist das aus der Diskretisierung folgende (vgl. Abschnitt 3.2.3) Gleichungssystem überbestimmt (Dirichlet und Neumann Randbedingungen). Bei der von Steger und Sorensen vorgeschlagenen Methode (vgl. CARSTENS (1988)) wird die Position festgehalten und der

Winkel  $\theta$  (wegen der geforderten Orthogonalität gilt meistens  $\theta = 90^\circ$ ) sukzessive angepaßt. Es werden hierzu die beiden Kontrollfunktionen als

$$P(\xi, \eta) = p(\xi) e^{-a(\eta - \eta_0)} + r(\xi) e^{-b(\eta_{n-1} - \eta)} \quad (3.7)$$

und

$$Q(\xi, \eta) = q(\xi) e^{-c(\eta - \eta_0)} + t(\xi) e^{-d(\eta_{n-1} - \eta)} \quad (3.8)$$

definiert. Die Funktion  $P$  ist für den Winkel und die Funktion  $Q$  für den Abstand der ersten beiden Gitterlinien in  $\eta$  Richtung verantwortlich. Das Abklingverhalten kann durch die Konstanten  $a, b$  bzw.  $c, d$  gesteuert werden. Die einzelnen Funktionen  $p, q, r$  und  $t$  ergeben sich zu:

$$p(\xi) = \frac{1}{J^3} \left( \frac{\partial x}{\partial \eta} R_y - \frac{\partial y}{\partial \eta} R_x \right) \Big|_{\eta=\eta_0} \quad r(\xi) = \frac{1}{J^3} \left( \frac{\partial x}{\partial \eta} R_y - \frac{\partial y}{\partial \eta} R_x \right) \Big|_{\eta=\eta_{n-1}} \quad (3.9)$$

$$q(\xi) = -\frac{1}{J^3} \left( \frac{\partial x}{\partial \xi} R_y - \frac{\partial y}{\partial \xi} R_x \right) \Big|_{\eta=\eta_0} \quad t(\xi) = -\frac{1}{J^3} \left( \frac{\partial x}{\partial \xi} R_y - \frac{\partial y}{\partial \xi} R_x \right) \Big|_{\eta=\eta_{n-1}} \quad (3.10)$$

mit

$$R_x = \alpha \frac{\partial^2 x}{\partial \xi^2} - 2\beta \frac{\partial^2 x}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 x}{\partial \eta^2} \quad R_y = \alpha \frac{\partial^2 y}{\partial \xi^2} - 2\beta \frac{\partial^2 y}{\partial \xi \partial \eta} + \gamma \frac{\partial^2 y}{\partial \eta^2} \quad . \quad (3.11)$$

In den verschiedenen Termen tauchen u.a. die partiellen Ableitungen  $\partial/\partial\xi$ ,  $\partial/\partial\eta$ ,  $\partial^2/\xi^2$ ,  $\partial^2/\eta^2$  sowie  $\partial^2/\partial\xi\partial\eta$  auf, die am Rand für  $\eta = \eta_0$  bzw.  $\eta = \eta_{n-1}$  zu bestimmen sind. Die Ableitungen in  $\eta$ -Richtung werden aus den geometrischen Randbedingungen, dem Winkel  $\theta$  und dem Abstand  $\Delta s$  der ersten beiden Gitterlinien für  $\eta_0$  und  $\eta_1$ , berechnet:

$$\frac{\partial x}{\partial \eta} = \Delta s \left( -\frac{\partial x}{\partial \xi} \cos \theta - \frac{\partial y}{\partial \xi} \sin \theta \right) \Big|_{\eta=\eta_0} \quad (3.12)$$

$$\frac{\partial y}{\partial \eta} = \sqrt{\left( \frac{\partial x}{\partial \xi} \right)^2 + \left( \frac{\partial y}{\partial \xi} \right)^2} \left( -\frac{\partial y}{\partial \xi} \cos \theta + \frac{\partial x}{\partial \xi} \sin \theta \right) \Big|_{\eta=\eta_0} \quad (3.13)$$

Für die Bestimmung der partiellen Ableitungen am äußeren Rand ( $\eta = \eta_{n-1}$ ) ist analog vorzugehen. In den Beispielen von Abschnitt 3.2.6 ist der Einfluß der verschiedenen Parameter deutlich zu erkennen.

### 3.2.3 Diskretisierung und iterative Lösung

Zur Lösung der Gleichung (3.3) bzw. (3.4) läßt sich eine Diskretisierung mit zentralen Differenzen und einer anschließenden Punktiteration durchführen. Hierbei ist auch in jedem Iterationsschritt die rechte Seite der Gleichung zu aktualisieren.

Zur Lösung der partiellen Differentialgleichungen sind die in Abbildung 3.2 skizzierten

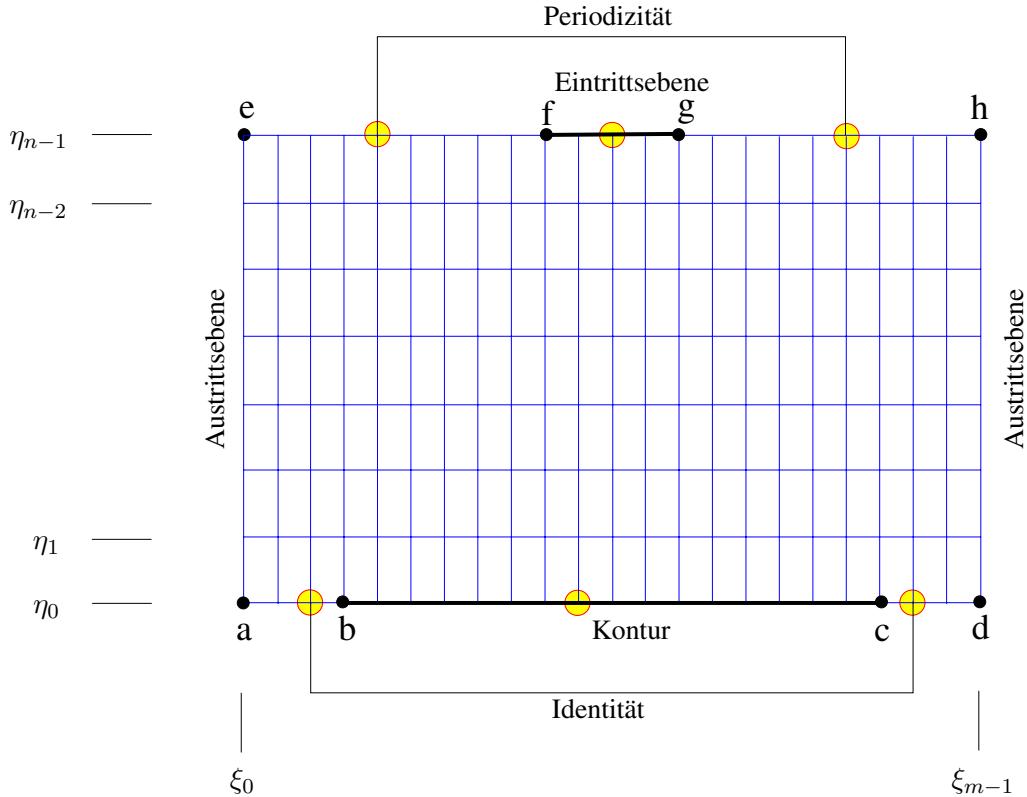


Abbildung 3.2: Bei der Gittergenerierung sind die dargestellten Randbedingungen zu berücksichtigen. Zusätzlich wird der Winkel  $\theta$  und der Abstand  $\Delta s$  der ersten bzw. letzten beiden Gitterlinien in  $\eta$  Richtung vorgegeben (iterative Methode).

Randbedingungen zu erfüllen. Die  $x$ -Koordinate der Eintritts- und Austrittsebene werden konstant gehalten und nur in der  $y$ -Richtung aktualisiert, um eine Kontraktion zu verhindern. Zusätzlich ist die periodische Fortsetzung und die Identität an der Schnittkante  $\overline{ab}$  bzw.  $\overline{cd}$  zu berücksichtigen. Die Punkte  $x(\xi, \eta)$  bzw.  $y(\xi, \eta)$  für  $\eta = \eta_0$  werden zu Beginn durch ein gesondertes Verfahren (vgl. nachfolgenden Abschnitt) berechnet und bleiben während der Iteration unverändert.

### 3.2.4 Konturpunktverteilung

Um die Genauigkeit in der Strömungsberechnung zu verbessern, kann die Anzahl der Gitterpunkte erhöht werden. Der damit einhergehende zusätzliche Rechenaufwand kann dadurch vermieden werden, daß stattdessen die Gebiete mit zu erwartenden großen Veränderungen der Strömungsgrößen feiner als die homogenen Bereiche aufgelöst werden. Insbesondere an der Vorder- und Hinterkante ist eine Konzentration der Gitterknoten notwendig. CARSTENS (1988) beschreibt ein Verfahren, welches in Abhängigkeit der Variablen  $\xi$  eine Konturpunktverteilung berechnet. Nachteil ist jedoch, daß es sich hierbei um eine Gewich-

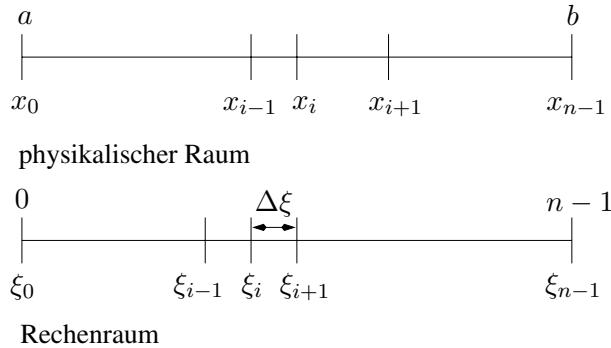


Abbildung 3.3: Bei der eindimensionalen Gittergenerierung wird ein Intervall  $[a, b]$  entsprechend einer Gewichtung in einzelne Segmente  $[x_i, x_{i+1}]$  aufgeteilt. Analog zum zweidimensionalen Fall wird eine Transformation des physikalischen Raums in den Rechenraum benötigt.

tung der Variablen des Rechenraums handelt und daher Eigenschaften im physikalischen Raum, wie zum Beispiel Krümmungen der Profilkontur, nicht berücksichtigt werden.

Der in dieser Arbeit entwickelte Gittergenerator ermöglicht es, entsprechend einer Gewichtsfunktion des physikalischen Raumes auf dem Profil eine Konturpunktverteilung zu berechnen. Es handelt sich hierbei um eine Erweiterung des von KNUPP UND STEINBERG (1994) vorgestellten eindimensionalen Gittergenerators.

### Eindimensionale Gittergenerierung

Bei der eindimensionalen Gittergenerierung wird ein Intervall  $[a, b]$  entsprechend einer Gewichtsfunktion  $w = w(x) > 0$  in einzelne Segmente  $[x_i, x_{i+1}]$  zerlegt. Analog zum oben beschriebenen zweidimensionalen Fall wird eine Transformation zwischen einem physikalischen Raum und einem Rechenraum gesucht, so daß gilt:  $x = x(\xi)$  bzw.  $\xi = \xi(x)$  (vgl. Abbildung 3.3). Zur Vereinfachung wird angenommen, daß  $\xi$  nur ganzzahlige Werte annehmen kann, d.h.  $\xi_i = i$  für  $i = 0, \dots, n - 1$ .

Für die Länge eines Segmentes soll gelten:

$$x_{i+1} - x_i = K w \left( \frac{x_{i+1} + x_i}{2} \right) \quad , \quad (3.14)$$

wobei  $w$  ein Gewicht in Abhängigkeit des Mittelpunkts des Segments  $[x_i, x_{i+1}]$  und  $K$  eine Konstante ist. Unter Berücksichtigung der Definition von  $x$  und für  $K = C\Delta\xi$  mit einer weiteren Konstante  $C$ , ergibt sich

$$\frac{x(\xi_i + \Delta\xi) - x(\xi_i)}{\Delta\xi} = C w \left( \frac{x(\xi_i + \Delta\xi) + x(\xi_i)}{2} \right) \quad (3.15)$$

bzw. allgemein für ein  $\xi \in [0, n - 1]$

$$\frac{x(\xi + \Delta\xi) - x(\xi)}{\Delta\xi} = Cw\left(\frac{x(\xi + \Delta\xi) + x(\xi)}{2}\right) \quad (3.16)$$

Wird mit  $\lim \Delta\xi \rightarrow 0$  der Grenzwert gebildet, entsteht die folgende Differentialgleichung:

$$\frac{dx(\xi)}{d\xi} \frac{1}{w(x(\xi))} = C \quad . \quad (3.17)$$

Eine weitere Differentiation nach  $\xi$  beschreibt dann schließlich die Transformation zwischen dem physikalischen Raum und dem Rechenraum.

$$\frac{d^2x(\xi)}{d\xi^2} - \frac{dx(\xi)}{d\xi} \frac{dw(x(\xi))}{d\xi} \frac{1}{w(x(\xi))} = 0 \quad (3.18)$$

Als Randbedingung gelten  $x(0) = a$  und  $x(n - 1) = b$ .

### Diskretisierung und iterative Lösung

Die Beziehung (3.18) lässt sich mit zentralen Differenzen zu

$$x_{i+1} - 2x_i + x_{i-1} - \frac{1}{4w(x_i)}(x_{i+1} - x_{i-1})(w(x_{i+1}) - w(x_{i-1})) = 0 \quad (3.19)$$

diskretisieren. Zur Lösung kann ein Punkt- oder Linieniterationsverfahren verwendet werden.

### Verallgemeinerung

Um bei zweidimensionalen parametrischen Kurven  $c(t) = (c_x(t), c_y(t))$  für  $t \in [t_0, t_1]$  eine Partitionierung anhand der beschriebenen Methode zu erreichen, muß ein neuer Parameter  $s$ , der die Bogenlänge der Kurve  $c$  beschreibt, eingeführt werden:

$$s(t) = \int_{t_0}^t \sqrt{\left(\frac{dc_x(t)}{dt}\right)^2 + \left(\frac{dc_y(t)}{dt}\right)^2} \quad (3.20)$$

Es bezeichne  $s^{-1}$  im folgenden die Umkehrfunktion, die zu einer Bogenlänge  $s$  den entsprechenden Kurvenparameter  $t$  liefert. Die Umkehrfunktion ist definiert, solange die Funktion  $s(t)$  streng monoton steigend ist. Für die in der Praxis relevanten Profile ist diese Bedingung gegeben. Bei der Implementierung des genannten Verfahrens bietet sich die Verwendung einer  $t, s$ -Wertetabelle an, über die zur Bestimmung der Umkehrfunktion  $s^{-1}$  mittels eines binären Suchalgorithmus effizient und schnell zugegriffen werden kann.

Die gesamte Bogenlänge beschreibt den physikalischen Raum  $[s(t_0), s(t_1)]$ , der mit dem oben beschriebenen Verfahren in einzelne Segmente  $[s_i, s_{i+1}]$  zerlegt wird. Aus den berechneten Segmenten lassen sich nun mittels der Umkehrfunktion die zugehörigen Parameterwerte und damit die Konturpunkte  $c_i$  ermitteln (vgl. Abbildung 3.4).

Damit eine gewünschte Konturpunktverteilung erzielt wird, sind geeignete Gewichtsfunktionen zu definieren.

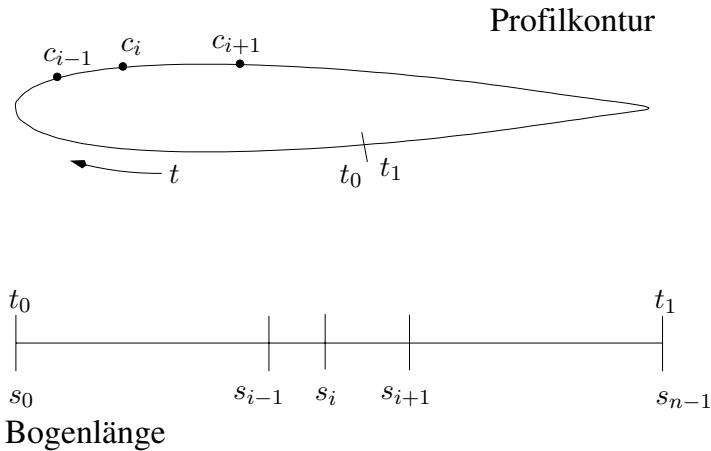


Abbildung 3.4: Durch die Einführung der Bogenlänge lässt sich mit einem eindimensionalen Gittergenerator eine Konturpunktverteilung berechnen.

### Gewichtsfunktion

Im einfachsten Fall einer Gewichtsfunktion  $w(s) = \text{const}$  ergibt sich eine äquidistante Verteilung der Konturpunkte. Um jedoch geometrische Eigenschaften des Profils im physikalischen Raum bei der Konturpunktberechnung zu berücksichtigen, bieten sich u.a. die folgenden Gewichtsfunktionen an.

Da die hier vorgestellten Gewichtsfunktionen jedoch in Abhängigkeit des Kurvenparameters  $t$  und nicht, wie eingangs dargestellt, bezüglich der Variablen des physikalischen Raumes  $s$  definiert sind, werden die Gewichtsfunktionen mit einem Überstrich gekennzeichnet. Weiterhin gilt:  $S = s(t_1) - s(t_0)$ .

### Gradient

$$\bar{w}_1(t) = \frac{1}{\sqrt{1 + \epsilon_1^2 (c'_x(t) + c'_y(t))^2}} \quad \text{mit } \epsilon_1 \geq 0 \quad (3.21)$$

### Krümmung

$$\bar{w}_2(t) = \frac{1}{\sqrt{1 + \epsilon_2^2 \kappa(t)^2}} \quad \text{mit } \kappa(t) = \frac{c'_x(t)c''_y(t) - c'_y(t)c''_x(t)}{(c'_x(t)^2 + c'_y(t)^2)^{3/2}} \quad \text{und } \epsilon_2 \geq 0 \quad (3.22)$$

### Krümmung und Gradient

$$\bar{w}_3(t) = \frac{1}{\sqrt{1 + \epsilon_1^2 (c'_x(t) + c'_y(t))^2 + \epsilon_2^2 \kappa(t)^2}} \quad \text{mit } \epsilon_1 \geq 0, \epsilon_2 > 0 \quad (3.23)$$

### Exponential

$$\bar{w}_4(t) = \begin{cases} \alpha(1 - e^{-\frac{\lambda}{S}|s(t) - s(t')|}) & \text{falls } t \neq t' \\ k & \text{sonst} \end{cases} \quad \text{mit } \alpha > 0, \lambda > 0, k > 0 \text{ und } t' \in [t_0, t_1] \quad (3.24)$$

### Kosinus

$$\bar{w}_5(t) = \begin{cases} 0.5 \left( 1 - \cos \left( \pi \frac{|s(t) - s(t')|}{S} \right) \right) & \text{falls } t \neq t' \\ k & \text{sonst} \end{cases} \quad \text{mit } k > 0 \quad (3.25)$$

### Tangens hyperbolicus

$$\bar{w}_6(t) = \begin{cases} \tanh \left( \alpha \frac{|s(t) - s(t')|}{S} \right) \tanh(\alpha)^{-1} & \text{falls } t \neq t' \\ k & \text{sonst} \end{cases} \quad \text{mit } k > 0, \alpha > 0 \quad (3.26)$$

### Binär

$$\bar{w}_7(t) = \begin{cases} k & \text{falls } t \in [t'_0, t'_1] \\ 1 & \text{sonst} \end{cases} \quad \text{mit } k > 0 \text{ und } [s(t'_0), s(t'_1)] \subset [s(t_0), s(t_1)] \quad (3.27)$$

### Euklidischer Abstand

$$\bar{w}_8(t) = \sqrt{(x - c_x(t))^2 + (y - c_y(t))^2} \quad \text{mit } \forall t \in [t_0, t_1] : c_x(t) \neq x \wedge c_y(t) \neq y \quad (3.28)$$

### Produkt

$$\bar{w}_9(t) = \prod_{i=0}^n \bar{w}_{i,j}(t) \quad \text{mit } n \geq 0 \text{ und } j \in \{1, \dots, 8\} \quad (3.29)$$

Die ersten drei aufgeführten Funktionen berücksichtigen geometrische Aspekte (Gradient bzw. Krümmung) bei der Berechnung der Gewichtung. Generell gilt, je kleiner das Gewicht, umso kleiner die Segmente. Die Gewichtsfunktionen *Exponential*, *Kosinus* und *Tangens hyperbolicus* definieren einen Kontraktionspunkt mittels  $t'$  auf der Kurve, der eine Verdichtung der Nachbarssegmente hervorruft. Um sicherzustellen, daß die Gewichte nicht null werden, wird eine Konstante  $k$  eingeführt. Die binäre Gewichtsfunktion definiert ein Intervall, innerhalb dessen eine konstante Gewichtung die Segmentlänge bestimmt. Die *Euklidischer-Abstand*-Funktion gewichtet bezüglich des Euklidischen Abstandes zu einem Kontraktionspunkt. Dieser darf nicht auf der Kontur liegen, da ansonsten die Gewichtung mit  $w > 0$  nicht möglich ist.  $\bar{w}_9(t)$  ist eine Produktgewichtsfunktion, die unterschiedliche Gewichtsfunktionen  $\bar{w}_{i,j}(t)$  in Beziehung zueinander setzt. Im Anhang A.3 werden für eine Ellipse Konturverteilungen in Abhängigkeit der unterschiedlichen Gewichtsfunktionen dargestellt.

### 3.2.5 Mehrstufige Gitter

Die Verwendung von Mehrgittermethoden zur numerischen Strömungsberechnung (vgl. Abschnitt 3.3) ist bereits bei der Gittergenerierung zu berücksichtigen, da dadurch bei der Dimensionierung gewisse Anforderungen notwendig werden.

Ein mehrstufiges Gitter besitzt die Möglichkeit, durch unterschiedliche Schrittweiten den diskretisierten physikalischen Raum in verschiedenen Abstufungen aufzulösen. Durch die sukzessive Berechnung und Verfeinerung ergibt sich so bei der Ermittlung der Strömung eine Beschleunigung der Konvergenzgeschwindigkeit. Ein  $k + 1$ -stufiges Gitter besitzt die Auflösung 0 (fein, Schrittweite 1) bis  $k$  (grob, Schrittweite  $2^k$ ). Damit die Struktur des Gitters nicht zerstört wird, sind bei der Dimensionierung Abhängigkeiten zu beachten. Die verschiedenen notwendigen Bedingungen<sup>2</sup> leiten sich aus der Überlegung ab, daß bei einem Gitter mit der Auflösung  $i$  nur jeder  $2^i$ -te Gitterknoten betrachtet wird. Um die Grundstruktur des Gitters bei den unterschiedlichen Auflösungen beizubehalten, werden die verschiedenen Punkte  $a, \dots, h$  als Fixpunkte betrachtet (vgl. Abbildung 3.1).

Für ein  $m \times n$  Gitter mit  $(k + 1)$  Auflösungsstufen sind bei der Dimensionierung die folgenden Bedingungen einzuhalten:

#### Dimension in $\eta$ -Richtung

$$(n - 1) \bmod 2^k = 0 \quad (3.30)$$

#### Dimension in der Eintrittsebene

$$(m_i + 1) \bmod 2^k = 0 \quad (3.31)$$

#### Dimension an den periodischen Rändern

$$(m - m_i) \bmod 2 = 0 \quad (3.32)$$

$$((m - m_i) \bmod 2 - 1) \bmod 2^k = 0 \quad (3.33)$$

#### Dimension auf der Kontur bzw. im Abströmbereich

$$(m - m_p) \bmod 2 = 0 \quad (3.34)$$

$$((m - m_p) \bmod 2 - 1) \bmod 2^k = 0 \quad (3.35)$$

$m_i$  ist die Anzahl der Gitterpunkte in der Eintrittsebene (Beginn des physikalischen Raums), jedoch ohne die beiden Eckpunkte  $f$  und  $g$ . Die Anzahl der Gitterpunkte auf der Kontur inklusive des zweifachen Hinterkantenpunktes  $b, c$  wird durch  $m_p$  beschrieben.

Der in dieser Arbeit entwickelte zweidimensionale Gittergenerator ermöglicht es, vorgegebene Dimensionen in Hinblick auf die Erfüllung der Bedingungen anzupassen (vgl. nächsten Abschnitt).

---

<sup>2</sup>Im folgenden bezeichnet der Operator  $\bmod$  das ganzzahlige Ergebnis und  $\bmod$  den ganzzahligen Rest bei einer Division (modulo).

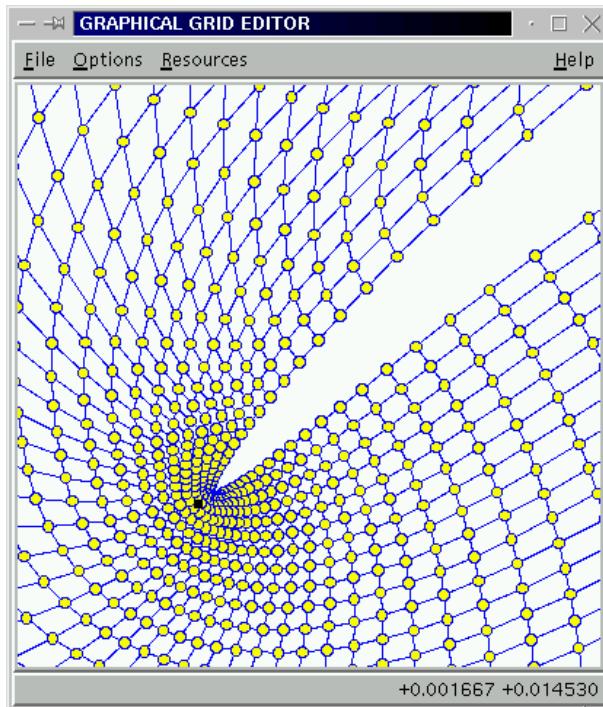


Abbildung 3.5: Durch die Definition eines Kontraktionspunktes an der Vorderkante ist eine feine Auflösung des Strömungsraumes im Bereich des Staupunktes möglich. Der graphische Gittereditor liefert zu jedem Knotenpunkt statistische Informationen (Orthogonalität, Längen–Höhen–Verhältnis, ...), so daß Schwachstellen im C–Netz behoben werden können.

### 3.2.6 Implementierte Verfahren

Im Rahmen dieser Arbeit ist ein Gittergenerator **muspider** entwickelt worden, der die beschriebenen Konzepte implementiert. Zusätzlich ist ein graphischer Editor **mugrideditor** entstanden, der dem Anwender eine bedienfreundliche Steuerung ermöglicht.

Bevor die der Gittergenerierung zugrundeliegende Differentialgleichung iterativ gelöst werden kann, ist ein Startgitter zu berechnen. Hierzu ist in einem ersten Schritt die Schnitt- und Hinterkante sowie die Konturpunktverteilung zu ermitteln. Bei der Hinterkantendetektierung wird versucht, einen gleichschenkeligen Keil in den hinteren Bereichs des Profils so einzupassen, daß etwaige Unstetigkeiten minimiert werden.

Zur Berechnung der Konturpunktverteilung (vgl. obigen Abschnitt 3.2.4) werden drei Gewichtsfunktionen überlagert. Für eine Verdichtung an der Vorderkante wird die *Euklidischer–Abstand–* Funktion verwendet. Durch eine (interaktive) Manipulation eines Kontraktionspunktes ist es möglich, eine notwendige feine Auflösung an der Profilvorderkante zu erhalten. An der aufgeschnittenen Hinterkante bestimmen zwei, je eine für die Ober– bzw. Unterseite ( $t' = t_0$  bzw.  $t' = t_1$ ), *Exponential–*Gewichtsfunktionen (mit  $\alpha = 0.01$  und  $\lambda = 1$ ) die Konturpunktverteilung. Die Abstände auf der Abströmlinie zwischen Hinter–

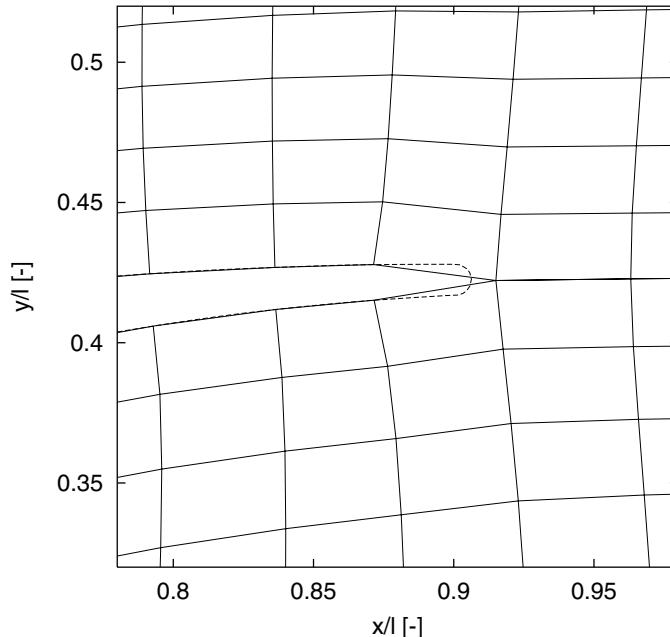


Abbildung 3.6: Der Gittergenerator schneidet das Profil an der Hinterkante auf und legt einen gleichschenkeligen Keil an der Ober- und Unterseite an. Zur Verdeutlichung ist das ursprüngliche Profil eingezeichnet.

kante und Austrittsebene werden durch eine geometrische Reihe bestimmt. Hierbei werden die Winkel so angepaßt, daß wellige Gitterlinien in Umfangsrichtung vermieden werden. In einem weiteren Schritt werden dann anschließend die äußeren (periodischen) Ränder durch B-Splines festgelegt. Durch eine einfache Interpolation werden die inneren Gitterknoten für das Startgitter berechnet.

Der graphische Gittereditor in Abbildung 3.5 ermöglicht es, die von WANG UND SONI (1991) vorgeschlagenen Bewertungskriterien für jeden einzelnen Gitterpunkt abzufragen. Bei diesen Maßzahlen handelt es sich im wesentlichen um metrische Koeffizienten. Weiterhin lassen sich die Gradienten der Maßzahlen sowie statistische Parameter (Minimum, Maximum, Mittelwert,...) untersuchen.

Durch eine automatische Anpassung der verschiedenen Dimensionen lassen sich beliebig vielstufige Gitter erzeugen. Diese Eigenschaft wird schon bei der Lösung der elliptischen Differentialgleichung ausgenutzt, so daß bereits bei der Gittergenerierung durch die sukzessive Verfeinerung eine Konvergenzbeschleunigung eintritt.

Beispielhaft ist in den Abbildungen 3.6 und 3.7 für das NACA-65-K48 Profil aus WATZLAWICK (1991) ein mit **muspider** erzeugtes C-Netz dargestellt.

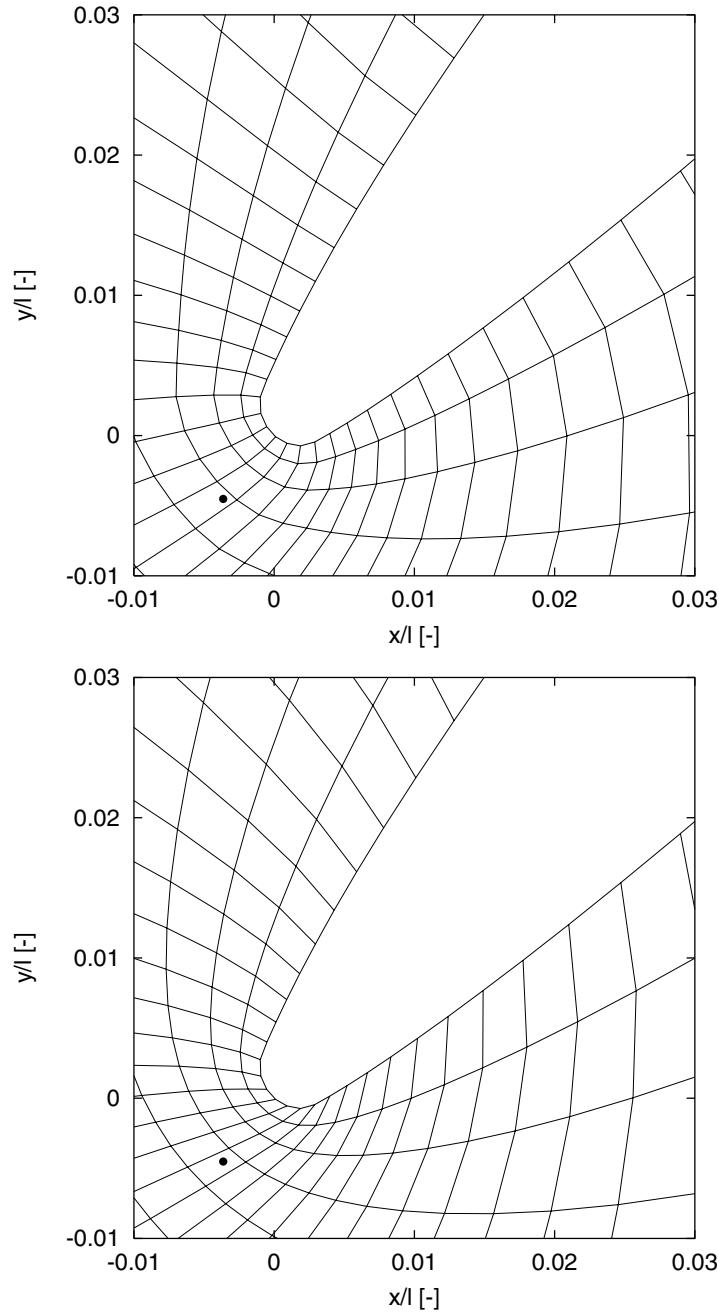


Abbildung 3.7: Das C–Netz zeigt an der Vorderkante eine deutliche Verdichtung der Gitterpunkte. Zusätzlich ist der Kontraktionspunkt für die Konturpunktverteilung eingezeichnet. Als Parameter für die Kontrollfunktionen  $P$  und  $Q$  sind  $a = 0.5$  und  $c = 0.6$  gewählt worden. Bei dem unteren Gitter ist auf die Einhaltung der Orthogonalität verzichtet worden.

### 3.3 Stromfunktionsverfahren

Das Stromfunktionsverfahren bildet die Basis zur Berechnung der kompressiblen und reibungsfreien zweidimensionalen Strömung. Um Reibungsphänomene an der Profiloberfläche zu beschreiben, wird dieses anschließend mit einem Grenzschichtverfahren gekoppelt.

#### 3.3.1 Einführung und Grundgleichungen

Für die weiteren fluiddynamischen Betrachtungen sind die folgenden Gleichungen (3.36) bis (3.39) Voraussetzung. Die Darstellung erfolgt in Zylinderkoordinaten. Es werden nur stationäre Strömungszustände betrachtet.

GLEICHUNG 1 (KONTINUITÄTSGLEICHUNG)

$$\frac{1}{r} \frac{\partial r \rho w_r}{\partial r} + \frac{1}{r} \frac{\partial \rho w_u}{\partial \varphi} + \frac{\partial \rho w_z}{\partial z} = 0 \quad (3.36)$$

Aus der Impulsbilanz in Verbindung mit der Gibbsschen Fundamentalgleichung lassen sich die folgenden Bewegungsgleichungen ableiten (vgl. LÜCKING (1982)). Hierbei wird ein mit konstanter Geschwindigkeit  $\omega$  um die Maschinenachse ( $z$ -Richtung) rotierendes System vorausgesetzt.

GLEICHUNG 2 (BEWEGUNGSGLEICHUNGEN)

*Radiale Richtung:*

$$\frac{w_u}{r} \left( \frac{\partial r w_u}{\partial r} - \frac{\partial w_r}{\partial \varphi} \right) - w_z \left( \frac{\partial w_r}{\partial z} - \frac{\partial w_z}{\partial r} \right) = T \frac{\partial s}{\partial r} - \frac{\partial h_{rot}}{\partial r} \quad (3.37)$$

*Umfangsrichtung:*

$$w_z \left( \frac{1}{r} \frac{\partial w_z}{\partial \varphi} - \frac{\partial w_u}{\partial z} \right) - \frac{w_r}{r} \left( \frac{\partial r w_u}{\partial r} - \frac{\partial w_r}{\partial \varphi} \right) = T \frac{1}{r} \frac{\partial s}{\partial \varphi} - \frac{1}{r} \frac{\partial h_{rot}}{\partial \varphi} \quad (3.38)$$

*Axiale Richtung:*

$$w_r \left( \frac{\partial w_r}{\partial z} - \frac{\partial w_z}{\partial r} \right) - w_u \left( \frac{1}{r} \frac{\partial w_z}{\partial \varphi} - \frac{\partial w_u}{\partial z} \right) = T \frac{\partial s}{\partial z} - \frac{\partial h_{rot}}{\partial z} , \quad (3.39)$$

wobei die Rothalpie  $h_{rot}$  durch

$$h_{rot} = h + \frac{w^2}{2} - \frac{u^2}{2} \quad (3.40)$$

definiert wird.

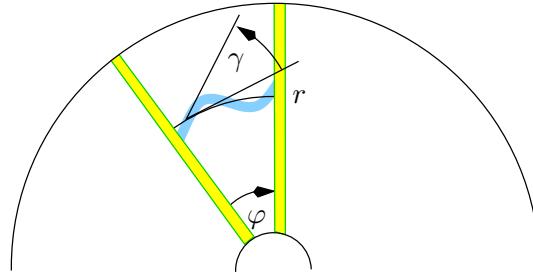


Abbildung 3.8: Die Stromflächensteigung  $\lambda_1 = -\tan \gamma$  beschreibt die Veränderung des Radius in Abhängigkeit des Winkels  $\varphi$ . Zur Vereinfachung bietet es sich an, eine zur Maschinenachse rotationssymmetrische  $S_1$ -Fläche mit  $\lambda_1 = 0$  vorauszusetzen.

### 3.3.2 Geometrie der $S_1$ -Stromfläche

Das in dieser Arbeit verwendete Verfahren berechnet die Strömung auf den beschriebenen  $S_1$ -Stromflächen. Geometrische Eigenschaften dieser Flächen sind die Steigung  $\lambda_1$  in Umfangsrichtung sowie die axiale Steigung  $\lambda_2$  und die Schichtdicke  $b$  (vgl. die Abbildungen 3.8 und 3.9).

In einem dreidimensionalen Koordinatensystem ergibt sich die radiale Komponente für eine  $S_1$ -Stromfläche aus den beiden anderen, d.h.  $r = r(\varphi, z)$ . Eine beliebige Strömungsgröße  $q$  lässt sich also als Funktion  $q = q(\varphi, z, r(\varphi, z))$  darstellen. Durch die Einführung von  $\bar{\varphi}$  und  $\bar{z}$  mit  $r = r(\bar{\varphi}, \bar{z})$  lässt sich die Geometrie einer Stromfläche durch zwei unabhängige Variablen beschreiben. Für die Steigungen der Stromfläche gelten dann die folgenden Beziehungen:

$$\tan \gamma = \frac{1}{r} \frac{\partial r}{\partial \bar{\varphi}} = -\lambda_1 \quad \text{und} \quad \tan \epsilon = \frac{\partial r}{\partial \bar{z}} = -\lambda_2 \quad . \quad (3.41)$$

Die Ableitungen entlang den beiden unabhängigen Koordinaten lassen sich definieren als:

$$\frac{1}{r} \frac{\partial}{\partial \bar{\varphi}} \equiv \frac{1}{r} \frac{\partial}{\partial \varphi} + \frac{1}{r} \frac{\partial r}{\partial \varphi} \frac{\partial}{\partial r} = \frac{1}{r} \frac{\partial}{\partial \varphi} - \lambda_1 \frac{\partial}{\partial r} \quad \text{für } \bar{z} = \text{const und} \quad (3.42)$$

$$\frac{\partial}{\partial \bar{z}} \equiv \frac{\partial}{\partial z} + \frac{\partial r}{\partial z} \frac{\partial}{\partial r} = \frac{\partial}{\partial z} - \lambda_2 \frac{\partial}{\partial r} \quad \text{für } \bar{\varphi} = \text{const} \quad (3.43)$$

Für die radiale Geschwindigkeitskomponente gilt dann:  $w_r = -\lambda_1 w_u - \lambda_2 w_z$ .

### 3.3.3 Hauptgleichung des Stromfunktionsverfahrens

Die Bewegungsgleichungen (3.37) bis (3.39) in Verbindung mit den Ableitungen entlang der Stromfläche und einem Stromfunktionsansatz mit

$$\frac{\partial \psi}{\partial \bar{\varphi}} = rb\rho w_z \quad \text{und} \quad \frac{\partial \psi}{\partial \bar{z}} = -b\rho w_u \quad (3.44)$$

führt auf die folgende Hauptgleichung für die Strömung einer  $S_1$ -Stromfläche.

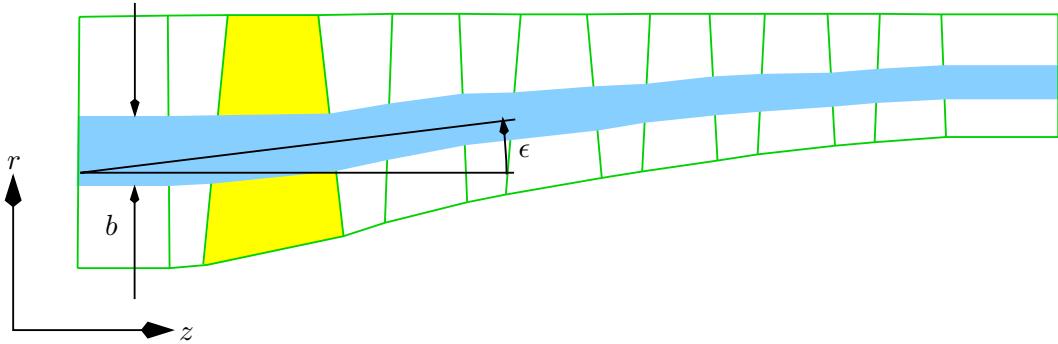


Abbildung 3.9: Im Meridianschnitt werden die Charakteristika einer  $S_1$ -Stromfläche, die Schichtdicke  $b$  und die Steigung  $\lambda_2 = -\tan \epsilon$ , deutlich. Das Beispiel zeigt die Spur einer Stromfläche eines dreistufigen Axialverdichters (Institut für Strahlantriebe und Turboarbeitsmaschinen, RWTH Aachen).

### GLEICHUNG 3 (HAUPTGLEICHUNG DER $S_1$ -STROMFLÄCHE)

$$A \frac{\partial^2 \psi}{\partial \bar{z}^2} + B \frac{\partial^2 \psi}{\partial \bar{\varphi}^2} + C \frac{\partial^2 \psi}{\partial \bar{\varphi} \partial \bar{z}} + D \frac{\partial \psi}{\partial \bar{z}} + E \frac{\partial \psi}{\partial \bar{\varphi}} = F \quad (3.45)$$

mit

$$\begin{aligned} A &= 1 + \lambda_1^2 \\ B &= \frac{1}{r^2} (1 + \lambda_2^2) \\ C &= -\frac{2}{r} \lambda_1 \lambda_2 \\ D &= \lambda_1 \frac{\partial \lambda_1}{\partial \bar{z}} - \frac{\lambda_2}{r} \left( 1 + \frac{\partial \lambda_1}{\partial \bar{\varphi}} \right) \\ E &= \frac{\lambda_2}{r^2} \frac{\partial \lambda_2}{\partial \bar{\varphi}} + \frac{\lambda_1}{r} \left( \frac{1}{r} - \frac{\partial \lambda_2}{\partial \bar{z}} \right) \\ F &= \frac{1}{r} \frac{\partial b \rho}{\partial \bar{\varphi}} (w_z - \lambda_2 w_r) - \frac{\partial b \rho}{\partial \bar{z}} (w_u - \lambda_1 w_r) - b \rho \left( \frac{1}{w_z r} \left( T \frac{\partial s}{\partial \bar{\varphi}} - \frac{\partial h_{rot}}{\partial \bar{\varphi}} \right) + \lambda_2 2\omega \right) \end{aligned}$$

Zur Vereinfachung wird angenommen, daß die  $S_1$ -Flächen zur Maschinenachse rotations-symmetrisch sind, d.h mit  $\partial r / \partial \bar{\varphi} = 0$  folgt dann  $\lambda_1 = 0$ . Die vereinfachte Hauptgleichung der rotationssymmetrischen  $S_1$ -Fläche ergibt sich dann mit konstanter Rothalpie und konstanter Entropie zu:

$$L = \frac{\partial^2 \psi}{\partial \bar{z}^2} + \frac{1 + \lambda_2^2}{r^2} \frac{\partial^2 \psi}{\partial \bar{\varphi}^2} - \frac{\lambda_2}{r} \frac{\partial \psi}{\partial \bar{z}} - w_z \frac{1 + \lambda_2^2}{r} \frac{\partial b \rho}{\partial \bar{\varphi}} + w_u \frac{\partial b \rho}{\partial \bar{z}} + 2b \rho \lambda_2 \omega = 0 \quad . \quad (3.46)$$

Durch Umsortieren der verschiedenen Terme lässt sich diese Beziehung in eine Divergenzform  $L = \nabla \cdot \vec{x} + R$  mit

$$\vec{x} = \left( 0, \frac{1 + \lambda_2^2}{r} \frac{\partial \psi}{\partial \bar{\varphi}}, \frac{\partial \psi}{\partial \bar{z}} \right) \text{ und } R = -\frac{\lambda_2}{r} \frac{\partial \psi}{\partial \bar{z}} - w_z \frac{1 + \lambda_2^2}{r} \frac{\partial b\rho}{\partial \bar{\varphi}} + w_u \frac{\partial b\rho}{\partial \bar{z}} + 2b\rho\lambda_2\omega \quad (3.47)$$

bringen. Die Integration über ein Kontrollvolumen  $V$  unter Anwendung des Gaußschen Integralsatzes liefert sodann

$$\int_V L \, dV = \int_V \nabla \cdot \vec{x} \, dV + \int_V R \, dV = \int_S \vec{x} \, d\vec{s} + \int_V R \, dV \quad , \quad (3.48)$$

wobei  $S$  die Oberfläche des Kontrollvolumens bezeichnet.

Durch Anwendung der allgemeinen Kettenregel

$$\frac{\partial \psi}{\partial \bar{\varphi}} = \frac{\partial \psi}{\partial s} \frac{\partial s}{\partial \bar{\varphi}} + \frac{\partial \psi}{\partial t} \frac{\partial t}{\partial \bar{\varphi}} \quad \text{und} \quad \frac{\partial \psi}{\partial \bar{z}} = \frac{\partial \psi}{\partial s} \frac{\partial s}{\partial \bar{z}} + \frac{\partial \psi}{\partial t} \frac{\partial t}{\partial \bar{z}} \quad (3.49)$$

lässt sich eine Transformation in das krummlinige Koordinatensystem  $(s, t)$  durchführen. Unter der Voraussetzung, daß auch die Umkehrung der Transformation gilt, lassen sich die beiden Beziehungen

$$\begin{pmatrix} \frac{\partial \psi}{\partial \bar{\varphi}} \\ \frac{\partial \psi}{\partial \bar{z}} \end{pmatrix} = \begin{pmatrix} \frac{\partial s}{\partial \bar{\varphi}} & \frac{\partial t}{\partial \bar{\varphi}} \\ \frac{\partial s}{\partial \bar{z}} & \frac{\partial t}{\partial \bar{z}} \end{pmatrix} \begin{pmatrix} \frac{\partial \psi}{\partial s} \\ \frac{\partial \psi}{\partial t} \end{pmatrix} \quad \text{und} \quad \begin{pmatrix} \frac{\partial \psi}{\partial s} \\ \frac{\partial \psi}{\partial t} \end{pmatrix} = \begin{pmatrix} \frac{\partial \bar{\varphi}}{\partial s} & \frac{\partial \bar{z}}{\partial s} \\ \frac{\partial \bar{\varphi}}{\partial t} & \frac{\partial \bar{z}}{\partial t} \end{pmatrix} \begin{pmatrix} \frac{\partial \psi}{\partial \bar{\varphi}} \\ \frac{\partial \psi}{\partial \bar{z}} \end{pmatrix} \quad (3.50)$$

herleiten. Die beiden Matrizen sind invers zueinander, so daß die Berechnung der metrischen Koeffizienten in Gleichung (3.49) sich ergibt zu:

$$\frac{\partial s}{\partial \bar{\varphi}} = \frac{1}{J} \frac{\partial \bar{z}}{\partial t}, \quad \frac{\partial t}{\partial \bar{\varphi}} = -\frac{1}{J} \frac{\partial \bar{z}}{\partial s}, \quad \frac{\partial s}{\partial \bar{z}} = -\frac{1}{J} \frac{\partial \bar{\varphi}}{\partial t} \quad \text{und} \quad \frac{\partial t}{\partial \bar{z}} = \frac{1}{J} \frac{\partial \bar{\varphi}}{\partial s} \quad \text{mit} \quad J = \begin{vmatrix} \frac{\partial \bar{\varphi}}{\partial s} & \frac{\partial \bar{z}}{\partial s} \\ \frac{\partial \bar{\varphi}}{\partial t} & \frac{\partial \bar{z}}{\partial t} \end{vmatrix}.$$

### 3.3.4 Dichtebestimmung

Die Lösung der beschriebenen Differentialgleichungen ergibt die Massenstromdichte  $\rho w$ , die anhand der folgenden Beziehungen in die Dichte  $\rho$  und die Geschwindigkeit  $w$  aufgeteilt werden kann:

$$\Sigma = \left( 1 - \frac{\Phi}{\Sigma^2} \right)^{1/(\kappa-1)} \quad (3.51)$$

mit

$$\Sigma = \frac{\rho}{\rho_1} \left( \frac{h_1}{h_{\text{rot},1} + \frac{u^2}{2}} \right)^{1/(\kappa-1)} \quad (3.52)$$

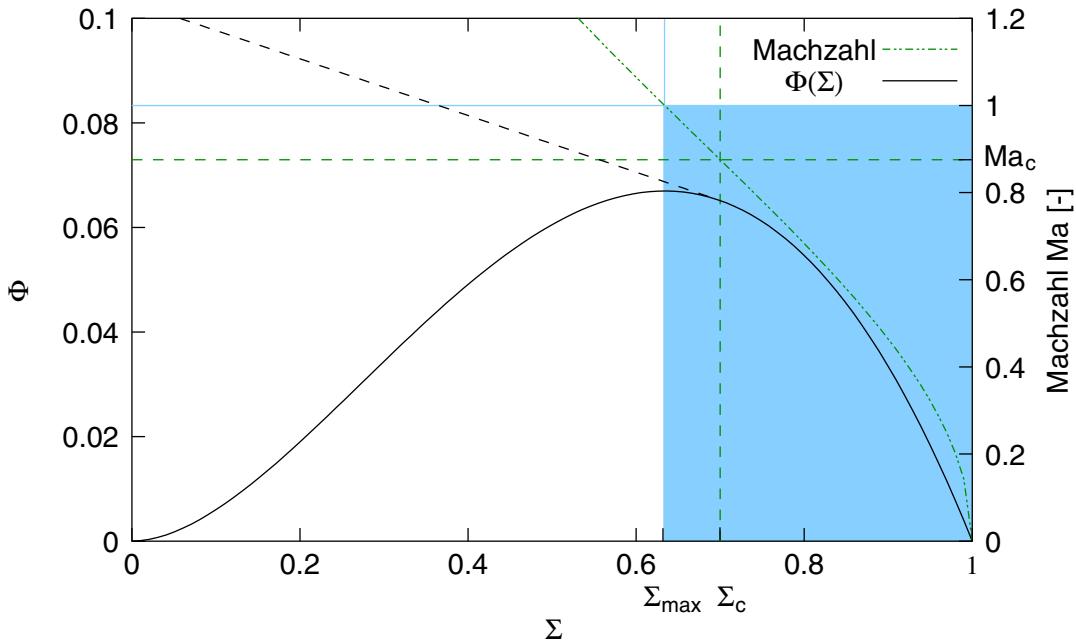


Abbildung 3.10: Zur Bestimmung der Dichte  $\rho$  wird zuerst  $\Phi$  und dann daraus  $\Sigma$  bestimmt. Jedem  $\Sigma$ -Wert lässt sich eine Machzahl zuordnen. Der grau unterlegte Bereich markiert den für diese Arbeit besprochenen Fall der Unterschallströmung ( $\kappa = 1.4$ ). Zur numerischen Stabilität wird der Bereich nahe  $Ma = 1$  durch eine Gerade approximiert.

und

$$\Phi = \frac{(\rho w)^2}{\rho_1^2 2 h_1} \left( \frac{h_1}{h_{\text{rot},1} + \frac{u^2}{2}} \right)^{(\kappa+1)/(\kappa-1)}. \quad (3.53)$$

Die mit tiefgestelltem Index 1 bezeichneten Größen sind die als bekannt vorausgesetzten Eintrittsgrößen. Durch die oben definierten Gleichungen werden jedem  $\Phi$ -Wert zwei  $\Sigma$ -Werte, jeweils einer für Unterschall- bzw. Überschallströmung, zugeordnet. In dieser Arbeit werden nur Unterschallströmungen untersucht, womit nur der rechte Zweig der Kurve Anwendung findet (vgl. Abbildung 3.10).

Zur schnellen Bestimmung des  $\Sigma$ -Wertes aus  $\Phi$  bietet sich der einmalige Aufbau einer  $\Sigma$ - $\Phi$ -Wertetabelle an, auf die mittels eines binären Suchalgorithmus in maximal logarithmischer Zeit zugegriffen werden kann.

Da für Geschwindigkeiten in einem Bereich um  $Ma = 1$  kleine Veränderungen von  $\Phi$  große Dichteänderungen zur Folge haben, wird zur Erhöhung der numerischen Stabilität der Bereich mit  $\Sigma < \Sigma_c$  bzw.  $Ma > Ma_c$  durch eine Tangente approximiert.

Zur Bestimmung der Maximalposition  $\Sigma_{\max}$  wird die Gleichung (3.51) abgeleitet und null

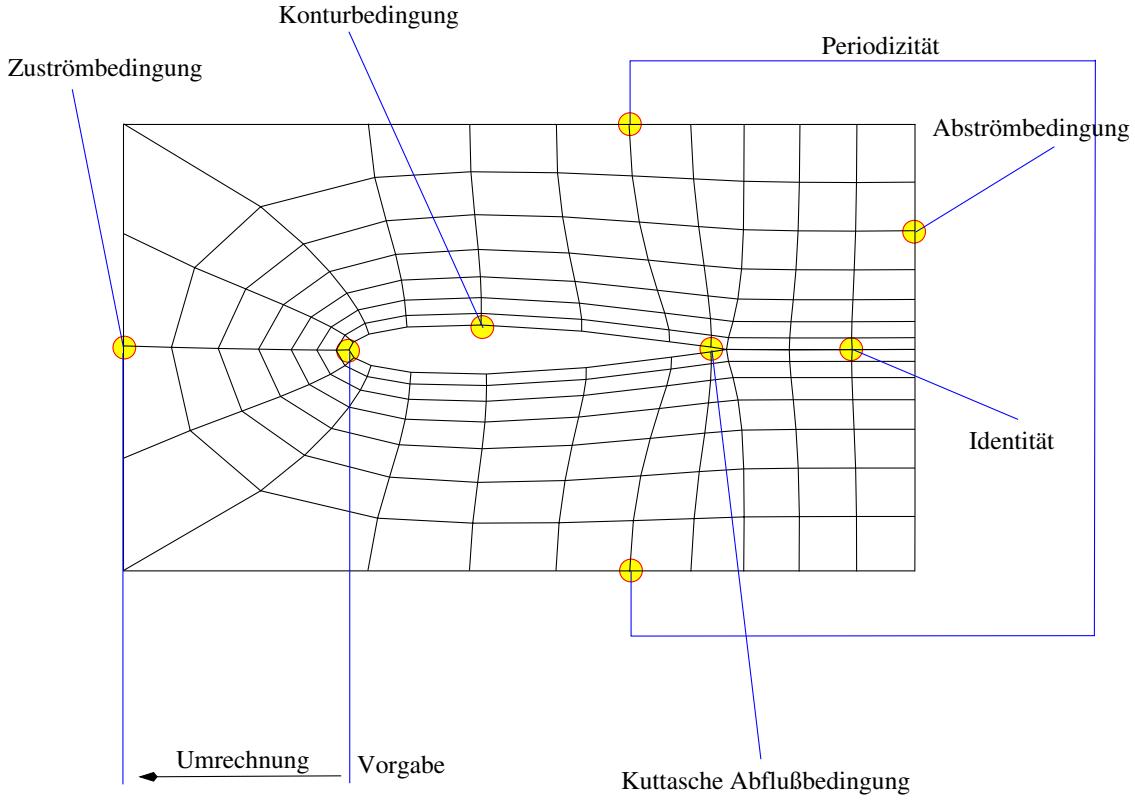


Abbildung 3.11: Zur Lösung der Hauptgleichung einer S<sub>1</sub>-Stromfläche sind die dargestellten Randbedingungen zu beachten.

gesetzt. Es gilt dann:

$$\Sigma_{\max} = \left( \frac{2}{\kappa + 1} \right)^{1/(\kappa-1)} \quad \text{bzw. mit } \kappa = 1.4 \text{ folgt } \Sigma_{\max} \approx 0.63394. \quad (3.54)$$

Für die Berechnung der jeweiligen Machzahl  $Ma$  ergibt sich der folgende funktionale Zusammenhang:

$$Ma^2 = \frac{2}{\kappa - 1} (\Sigma^{1-\kappa} - 1) \quad . \quad (3.55)$$

In dem dargestellten Beispiel ergibt sich somit für  $\Sigma_c = 0.7$  und  $\kappa = 1.4$  eine zugehörige Machzahl mit  $Ma_c \approx 0.87564$  (vgl. Abbildung 3.10). Für das implementierte Stromfunktionsverfahren gilt  $\Sigma_c = 1.05 \cdot \Sigma_{\max}$  mit einer zugehörigen Machzahl  $Ma_c \approx 0.94023$ .

### 3.3.5 Randbedingungen

Zur Lösung des in Gleichung (3.45) aufgestellten Differentialgleichungssystems sind die folgenden Randbedingungen zu beachten (vgl. Abbildung 3.11). Die ebenfalls zu berücksichtigende *Kuttasche Abflußbedingung* wird in Abschnitt 3.3.6 ausführlich dargestellt.

### Zuströmbedingung

Neben der Geometrie der  $S_1$ -Stromfläche werden weitere Größen, wie die Totaltemperatur  $T_t$ , der Totaldruck  $p_t$  und der Massenstrom  $\dot{m}$  am Gittereintritt als bekannt vorausgesetzt. Diese können zum Beispiel mit einem Stromliniengeometrieverfahren vorher bestimmt werden sein. Anschließend werden diese Größen auf die Eintrittsebene umgerechnet.

### Konturbedingung

Da an der Profiloberfläche keine Masse ein- oder austreten kann, definiert diese eine Stromlinie mit konstanten  $\psi$ -Wert für alle Konturpunkte.

### Abströmbedingung

Unter der Annahme der gleichmäßigen, ungestörten Strömung in hinreichender Entfernung zur Hinterkante wird eine gleichverteilte  $\psi$ -Verteilung in Umfangsrichtung am rechten Gitterrand vorgegeben.

### Periodizität

Die Periodizitätsbedingung ergibt sich aus der Tatsache, daß rotationssymmetrische Schaufern berechnet werden und somit das diskretisierte Berechnungsgebiet in Umfangsrichtung genau einer Teilung entspricht. Die Differenz zweier Stromfunktionswerte gleicher axialer Koordinate an der oberen und unteren Gitterkante entspricht genau dem Eingangsstrom  $\dot{m}_{in}$ .

### Identität

Aus der Schnittkante des C-Gitters (vgl. Abbildung 3.1) ergibt sich die Tatsache, daß jeweils zwei Berechnungspunkte identisch sind.

#### 3.3.6 Kuttasche Abflußbedingung

Neben den bisher oben beschriebenen Randbedingungen ist die Erfüllung der sogenannten *Kuttaschen Abflußbedingung* zur Beschreibung realer Strömungsvorgänge notwendig. Diese auf Beobachtungen beruhende Bedingung sagt aus, daß sich an einer spitzen Hinterkante ein glattes Abströmen einstellt (vgl. ANDERSON (1991)). Abbildung 3.12 verdeutlicht diesen Sachverhalt. Sind die Geschwindigkeiten  $w|_{4a}$  an der Unterseite (4a) und  $w|_{4b}$  an der Oberseite (4b) identisch, ist die Kuttasche Bedingung erfüllt. Diese Forderung fließt bei der numerischen Lösung des diskretisierten Differentialgleichungssystem als zusätzliche Randbedingung mit ein.

Mit den Beziehungen (3.44) und  $w_r = -\lambda_1 w_u - \lambda_2 w_z$  ergibt sich für einen beliebigen

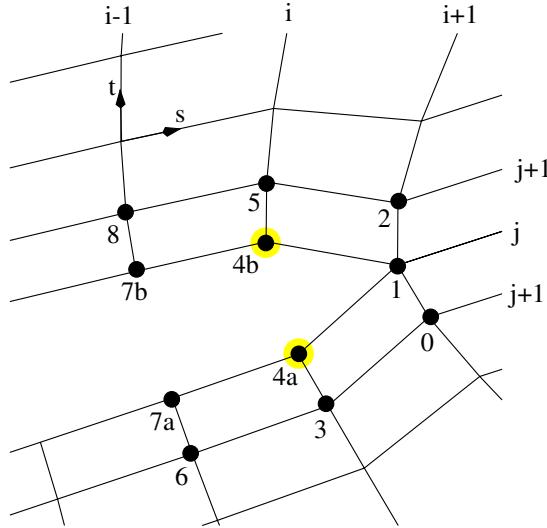


Abbildung 3.12: Die Kutta–Bedingung verlangt, daß bei einer spitzen Hinterkante die Geschwindigkeiten an den Punkten 4a und 4b identisch sind.

Gitterpunkt die nachfolgende Beziehung:

$$\begin{aligned}
 (b\rho w)^2 &= (b\rho w_u)^2 + (1 + \lambda_2^2)(b\rho w_z)^2 \\
 &= \left( \frac{\partial \psi}{\partial \bar{z}} \right)^2 + \frac{1 + \lambda_2^2}{r} \left( \frac{\partial \psi}{\partial \bar{\varphi}} \right)^2 \\
 &= \left( \frac{\partial \psi}{\partial s} \frac{\partial s}{\partial \bar{z}} + \frac{\partial \psi}{\partial t} \frac{\partial t}{\partial \bar{z}} \right)^2 + \frac{1 + \lambda_2^2}{r} \left( \frac{\partial \psi}{\partial s} \frac{\partial s}{\partial \bar{\varphi}} + \frac{\partial \psi}{\partial t} \frac{\partial t}{\partial \bar{\varphi}} \right)^2
 \end{aligned} \quad (3.56)$$

Die Kuttasche Bedingung lässt sich nun als  $(b\rho w)^2|_{4a} = (b\rho w)^2|_{4b}$  darstellen.

### 3.3.7 Diskretisierung

Die Diskretisierung der Integralgleichung für ein Kontrollvolumen wird an dieser Stelle anhand eines zentralen Knotens durchgeführt (vgl. Abbildung 3.13).

#### Diskretisierung des Oberflächenintegrals

Zur Approximation des in Gleichung (3.48) aufgeführten Oberflächenintegrals für einen Knoten an der Position  $(i, j)$  werden die vier Seiten getrennt voneinander behandelt:

$$\int_S \vec{x} d\vec{s} \approx \vec{n} \cdot \vec{x}|_\alpha + \vec{n} \cdot \vec{x}|_\beta + \vec{n} \cdot \vec{x}|_\gamma + \vec{n} \cdot \vec{x}|_\delta \quad , \quad (3.57)$$

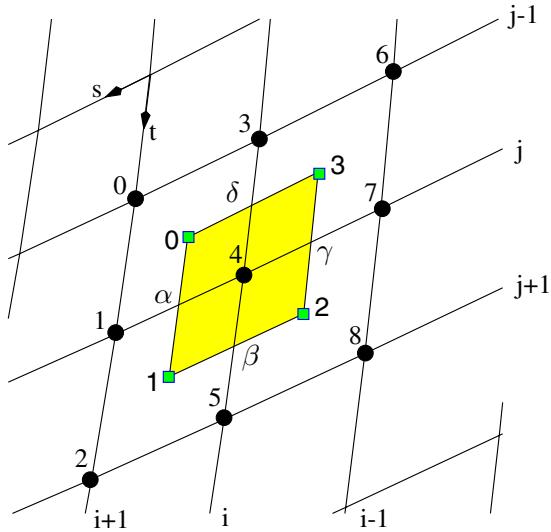


Abbildung 3.13: Ein zentrales Kontrollvolumen besteht aus vier Ecken  $0,1,2,3$  und ebenso vielen Seiten  $\alpha, \beta, \gamma, \delta$ . Bei der Verwendung von zentralen Differenzen werden die neun Nachbarknoten  $0, \dots, 8$  herangezogen.

wobei  $\vec{n}$  der Normalenvektor<sup>3</sup> der entsprechenden Seite ist. Der Vektor  $\vec{n} = (n_r, n_u, n_z)$  berechnet sich unter Anwendung des Vektorprodukts der beiden aufspannenden Vektoren  $\vec{p} = (b, 0, 0)$  und  $\vec{q} = (0, \Delta u, \Delta z)$  zu  $\vec{n} = (0, -b\Delta z, b\Delta u)$ . Die Variable  $b$  gibt die Höhe (radiale Richtung),  $\Delta u$  die Länge in Umfangsrichtung und  $\Delta z$  die axiale Breite der Seite an (siehe auch Abbildung 3.9). Da die radiale Komponente von  $\vec{x}$  verschwindet, reicht es aus, im folgenden nur die Umfangs-  $u$  bzw. Axialkomponente  $z$  der Terme zu betrachten.

Für die spätere numerische Implementierung bietet es sich an, die konstanten Anteile zusammenzufassen. Die Variable  $v_{i,k}^k$  beschreibt nachfolgend das Gewicht des  $k$ -ten Nachbarknotens für das Kontrollvolumen<sup>4</sup> an der Stelle  $(i, j)$  für  $k = 0, \dots, 8$ . Es bezeichne  $\psi_{i,j}^k$  den Stromfunktionswert des Nachbarknotens  $k$  für das betreffende Kontrollvolumen, also zum Beispiel für  $k = 2$  gilt:  $\psi_{i,j}^2 = \psi_{i+1,j+1}$ . Ziel ist es, einen Vektor  $\vec{v}_{i,j} = (v_{i,j}^0, v_{i,j}^1, \dots, v_{i,j}^8)$  zu erhalten, der die einzelnen Stromfunktionswerte  $\psi_{i,j}^k$  der neun Nachbarknoten  $k$  mit einem Faktor  $v_{i,j}^k$  gewichtet.

Unter Berücksichtigung von Gleichung (3.49) ergibt sich für eine beliebige Seite der folgenden Zusammenhang

$$\vec{n}\vec{x} = \left( n_u, n_z \right) \cdot \left( \frac{\lambda_2^2 + 1}{r} \frac{\partial \psi}{\partial \bar{\varphi}}, \frac{\partial \psi}{\partial \bar{z}} \right) = \left( c s_u, c s_z \right) \cdot \left( \frac{\partial \psi}{\partial s}, \frac{\partial \psi}{\partial t} \right) , \quad (3.58)$$

<sup>3</sup>Alle Flächenvektoren müssen die gleiche Orientierung haben, z.B. ins Kontrollvolumen zeigend.

<sup>4</sup>Der Nachbar mit  $k = 4$  bezeichnet das Kontrollvolumen an der Stelle  $(i, j)$ , für welches die Bilanzgleichung diskretisiert wird.

wobei für die konstanten (metrischen) Koeffizienten gilt:

$$cs_u = \left( n_u \frac{\lambda_2^2 + 1}{r} \frac{\partial s}{\partial \bar{\varphi}} + n_z \frac{\partial s}{\partial \bar{z}} \right) \quad \text{bzw.} \quad cs_z = \left( n_u \frac{\lambda_2^2 + 1}{r} \frac{\partial t}{\partial \bar{\varphi}} + n_z \frac{\partial t}{\partial \bar{z}} \right) . \quad (3.59)$$

Anhand der Seite  $\alpha$  wird im folgenden die Diskretisierung beschrieben. Mit den Differenzen

$$\frac{\partial \psi}{\partial s} \approx \psi_{i,j}^1 - \psi_{i,j}^4 \quad \text{und} \quad \frac{\partial \psi}{\partial t} \approx \frac{1}{4}(\psi_{i,j}^2 + \psi_{i,j}^5 - \psi_{i,j}^0 - \psi_{i,j}^3) \quad (3.60)$$

lässt sich der Anteil des Oberflächenintegrals darstellen als

$$\vec{n} \cdot \vec{x} \approx v_{i,j}^0 \psi_{i,j}^0 + v_{i,j}^1 \psi_{i,j}^1 + v_{i,j}^2 \psi_{i,j}^2 + v_{i,j}^3 \psi_{i,j}^3 + v_{i,j}^4 \psi_{i,j}^4 + v_{i,j}^5 \psi_{i,j}^5 \quad (3.61)$$

mit den einzelnen Gewichten

$$v_{i,j}^0 = v_{i,j}^3 = -\frac{1}{4}cs_z, \quad v_{i,j}^2 = v_{i,j}^5 = \frac{1}{4}cs_z, \quad v_{i,j}^1 = cs_u \quad \text{und} \quad v_{i,j}^4 = -cs_u . \quad (3.62)$$

Die Gewichte für die anderen Knoten  $k \in \{6, 7, 8\}$  sind bei der Diskretisierung für die Seite  $\alpha$  null, da keine dieser in die Berechnung der partiellen Ableitungen miteinfließt. Diese Vorgehensweise wird für alle Seiten wiederholt und die Gewichte werden aufsummiert.

Das Integral aus Gleichung (3.48) lässt sich dann schließlich für einen Knoten  $(i, j)$  durch

$$\int_S \vec{x} \cdot d\vec{s} \approx \sum_{k=0}^8 v_{i,j}^k \cdot \psi_{i,j}^k \quad \text{mit} \quad \sum_{k=0}^8 v_{i,j}^k = 0 \quad (3.63)$$

beschreiben. Mit Berücksichtigung der Randbedingungen und der nachfolgend beschriebenen Diskretisierung des Volumenintegrals lässt sich sodann ein Iterationsverfahren zur Bestimmung der Stromfunktionswerte  $\psi_{i,j}$  durchführen. Für die anderen Kontrollvolumen ist analog vorzugehen (vgl. Abbildungen 3.14 bis 3.17).

### Diskretisierung des Volumenintegrals

Das Volumenintegral der Gleichung (3.48) kann durch

$$\int_V R \, dV \approx r b A \quad (3.64)$$

approximiert werden. Es bezeichne  $A$  die Grundfläche des Kontrollvolumens und  $b$  die bereits beschriebene Schichtdicke. Der diskretisierte Integrand ist  $r$ .

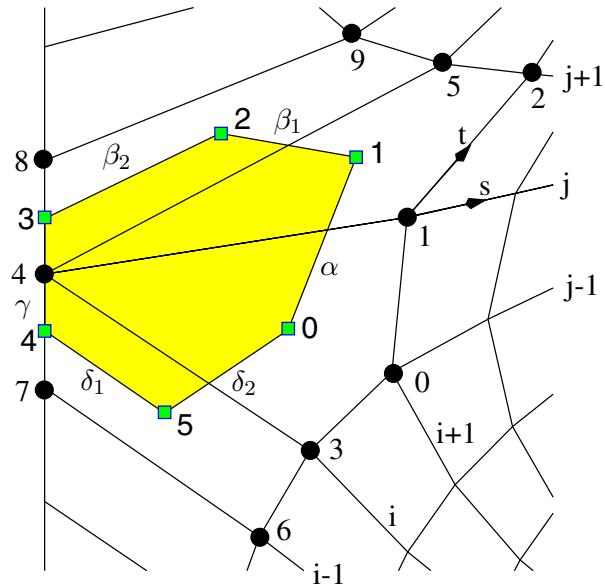


Abbildung 3.14: Zur Diskretisierung der oberen und unteren Ecke am linken Rand des Strömungsfeldes wird ein sechsseitiges Kontrollelement mit den zehn Nachbarknoten  $0, 1, \dots, 9$  eingeführt. Neben der Randbedingung am Eintritt ist auch die Periodizität in Umfangsrichtung zu beachten.

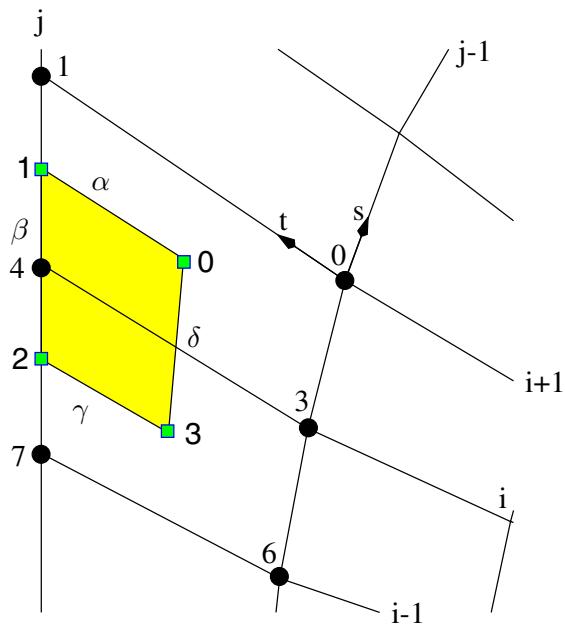


Abbildung 3.15: Bei der Diskretisierung eines Kontrollelementes am linken Rand fließt für die Seite  $\beta$  die Randbedingung der Eintrittsebene ein. Für die Berechnung der Ableitungen in  $t$ -Richtung sind einseitige Differenzen zu verwenden.

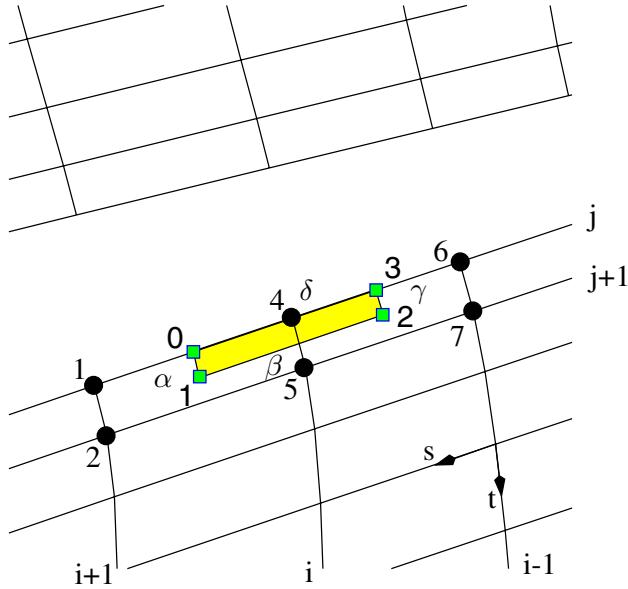


Abbildung 3.16: Für das Kontrollvolumen an der Profiloberfläche verschwindet der Anteil des Oberflächenvolumens für die Seite  $\delta$ , da vorausgesetzt wird, daß keine Masse aus dem Profil austritt.

### Anmerkung zur Kuttaschen Abflußbedingung

In Abschnitt 3.3.6 ist die Randbedingung für die Implementierung der notwendigen Kuttaschen Abflußbedingung abgeleitet, die sich demnach zu

$$(b\rho w)^2|_{4a} = (b\rho w)^2|_{4b} \quad (3.65)$$

ergibt. Die Diskretisierung für den Knoten  $4b$  kann dann in eine Form

$$v_{i,j}^3 \cdot \psi_{i,j}^3 + v_{i,j}^{4b} \cdot \psi_{i,j}^{4b} + v_{i,j}^5 \cdot \psi_{i,j}^5 = R \quad \text{für} \quad v_{i,j}^3 = v_{i,j}^5 = -\frac{1}{2} \quad \text{und} \quad v_{i,j}^{4b} = 1 \quad (3.66)$$

überführt und in das zu lösende Gleichungssystem eingesetzt werden. Die Variable  $R$  enthält u.a. eine Reihe von nichtlinearen Termen.

In Abbildung 3.17 ist das Kontrollvolumen an der Hinterkante dargestellt. Das Element umläuft die ganze Hinterkante und hat durchströmte Flächen auf der Unter- ( $\alpha_1, \beta$ ) und der Oberseite ( $\alpha_2, \delta$ ) des Profils. Die Diskretisierung für den Punkt  $4b$  lässt sich analog zu einem zentralen Kontrollvolumen durchführen. Hierbei wird angenommen, daß die Stromfunktionswerte  $\psi$  für die Knoten  $1a$  und  $1b$  sowie für die Knoten  $4a$  und  $4b$  identisch sind. Dieses folgt unmittelbar aus der Randbedingung an die Profilkontur, aus der keine Masse ein- oder austritt. Somit ergibt sich für das Hinterkantenelement ebenso eine Nachbarschaft von 9 Knoten. Bei der Bestimmung des Gleichungssystems wird demnach der Punkt  $4b$  nach dem Kontrollvolumen diskretisiert und für den Punkt  $4a$  die diskretisierte

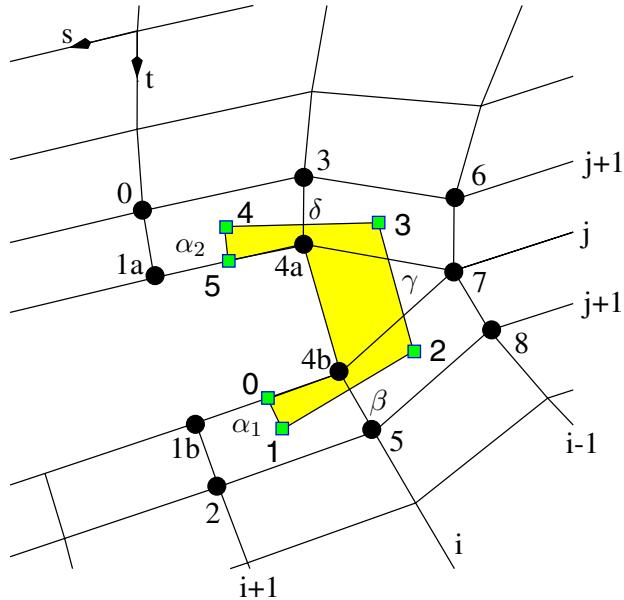


Abbildung 3.17: Das Kontrollvolumen an der Hinterkante umschließt die Unter- und Oberseite des Profils. Entsprechend der Umlaufrichtung ist die Bezeichnung der Knoten im Vergleich zum Kontrollelement in Abbildung 3.12 gedreht.

Kuttasche Bedingung eingesetzt.

In dieser Arbeit durchgeführte Untersuchungen zur Kuttaschen Bedingung an der Hinterkante haben gezeigt, daß es zur Erfüllung der Abflußbedingung hinreichend ist, den Punkt  $4b$  zu diskretisieren und den Punkt  $4a$  als normalen Konturpunkt mit  $\psi_{4a} = \psi_{1a}$  zu betrachten. Dieses bedeutet, daß eine explizite Formulierung der Kuttaschen Abflußbedingung *nicht* notwendig ist, sondern implizit durch die geschickte Diskretisierung des Hinterkantenelementes erfüllt ist.

Ein fundierter mathematischer Beweis für dieses überraschende Ergebnis steht noch aus. Es hat sich fernerhin gezeigt, daß die Netzstruktur an der Hinterkante entscheidenden Einfluß auf das numerische Verhalten hat. So hat es sich als vorteilhaft erwiesen, die Abstände  $\overline{34a}$  und  $\overline{54b}$  gleich groß zu wählen. Dieses kann durch den Gittergenerator erzwungen werden. In Abbildung 3.18 ist das Ergebnis einer Parameterstudie zu den zwei Implementierungsformen der Kuttaschen Bedingung aufgeführt.

### 3.3.8 Iterative Lösung

Zur Lösung der diskretisierten Stromfunktionsgleichung kann ein *explizites* Verfahren (Punktiteration) verwendet werden. Zur Beschleunigung der Konvergenz bietet es sich jedoch an, ein *implizites* Verfahren (Linieniteration) mit einem Relaxationsansatz zu verwenden, zumal die entstehenden Tridiagonalmatrizen mittels des *Thomas-Algorithmus* effizient gelöst werden können (vgl. AMES (1977)).

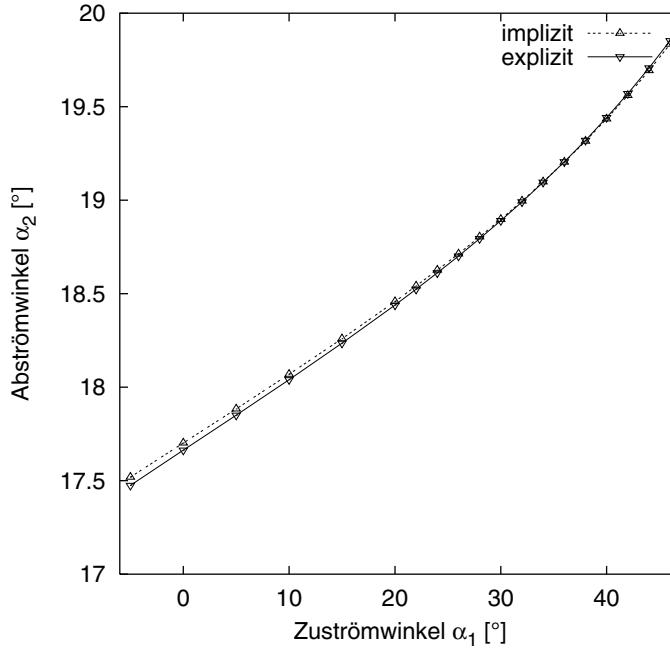


Abbildung 3.18: Untersuchungen zur Implementierung der Kuttaschen Abflußbedingung zeigen keine signifikanten Unterschiede zwischen der impliziten und expliziten Möglichkeit. Dazu ist für das Profil aus MERTENS (1995) eine Variation des Zuströmwinkels durchgeführt und der Abströmwinkel mit dem implementierten Stromfunktionsverfahren (inkompressibel) bestimmt worden.

Für ein  $m \times n$  Gitter wird für jede Gitterlinie  $j = 0, \dots, n-1$  ein lineares Gleichungssystem der Form  $\Upsilon \cdot \vec{\psi} = \vec{\zeta}$  aufgestellt und gelöst:

$$\begin{pmatrix} v_{0,j}^4 & v_{0,j}^1 \\ v_{1,j}^7 & v_{1,j}^4 & v_{1,j}^1 \\ v_{2,j}^7 & v_{2,j}^4 & v_{2,j}^1 \\ \ddots & \ddots & \ddots \\ v_{i,j}^7 & v_{i,j}^4 & v_{i,j}^1 \\ \ddots & \ddots & \ddots \\ v_{m-2,j}^7 & v_{m-2,j}^4 & v_{m-2,j}^1 \\ v_{m-1,j}^7 & v_{m-1,j}^4 & v_{m-1,j}^1 \end{pmatrix} \cdot \begin{pmatrix} \psi_{0,j} \\ \psi_{1,j} \\ \psi_{2,j} \\ \vdots \\ \psi_{i,j} \\ \vdots \\ \psi_{m-2,j} \\ \psi_{m-1,j} \end{pmatrix} = \begin{pmatrix} \zeta_{0,j} \\ \zeta_{1,j} \\ \zeta_{2,j} \\ \vdots \\ \zeta_{i,j} \\ \vdots \\ \zeta_{m-2,j} \\ \zeta_{m-1,j} \end{pmatrix} \quad (3.67)$$

Die Gewichte für die Randelemente mit  $i = 0$  bzw.  $i = m-1$  ergeben sich durch die Implementierung der Abströmbedingung. Die Berechnung der rechten Seite  $\zeta_{i,j}$  ist abhängig von

der gewählten Diskretisierung bzw. von der Implementierung der o.g. Randbedingungen. Für zentrale Kontrollvolumen bildet sich  $\zeta_{i,j}$  aus den gewichteten Stromfunktionswerten der benachbarten Gitterlinie  $j - 1$  und  $j + 1$  sowie des diskretisierten Volumenintegrals:

$$\zeta_{i,j} = -\left(\sum_k v_{i,j}^k \cdot \psi_{i,j}^k\right) - r_{i,j} b A \quad \text{mit } k \in \{0, 2, 3, 5, 6, 8\}. \quad (3.68)$$

Zur Berechnung der kompressiblen Strömung ist der Term  $R$  in Gleichung (3.47) aufgrund der veränderlichen Dichte  $\rho$  in jedem Iterationsschritt zu aktualisieren. Dazu ist in einem ersten Schritt die Massenstromdichte  $\rho w$  in ihre beiden Faktoren entsprechend des Verfahrens aus Abschnitt 3.3.4 zu zerlegen. Aus den Beziehungen (3.44) und unter Berücksichtigung der Gleichung (3.49) und des bekannten Radius  $r$  bzw. der Schichtdicke  $b$  der  $S_1$ -Fläche für einen Gitterpunkt ergibt sich

$$\rho w = \sqrt{(b\rho w_z)^2 + (b\rho w_u)^2 + (b\rho w_r)^2}/b \quad . \quad (3.69)$$

Die Bestimmung der Ableitung  $\partial\psi/\partial t$  an der Profiloberfläche wird im Gegensatz zu den anderen Kontrollvolumina nicht aus (zentralen) Differenzen, sondern direkt aus der Bilanzgleichung ermittelt. Hierzu wird die Gleichung (3.48) für ein Kontrollvolumen an der Oberfläche (vgl. auch Abbildung 3.16) aufgestellt, diskretisiert (analog Abschnitt 3.3.7) und für  $\partial\psi/\partial t|_\delta$  gelöst. Das Iterationsverfahren wird solange für alle Gitterlinien wiederholt, bis eine geforderte Genauigkeit erreicht worden ist. Als Startlösung wird eine lineare  $\psi$ -Verteilung angenommen.

Wie bereits in Abschnitt 3.2.5 hingewiesen wurde, bietet es sich zur Konvergenzbeschleunigung des Iterationsverfahrens an, einen einfachen Mehrgitteralgorithmus zu verwenden. Hierzu wird die Stromfunktionsgleichung zuerst auf einem groben Gitter gelöst und dann sukzessive die Auflösung verfeinert und das Iterationsverfahren durchgeführt. Die  $\psi$ -Werte für die neu hinzukommenden Gitterknoten werden linear interpoliert.

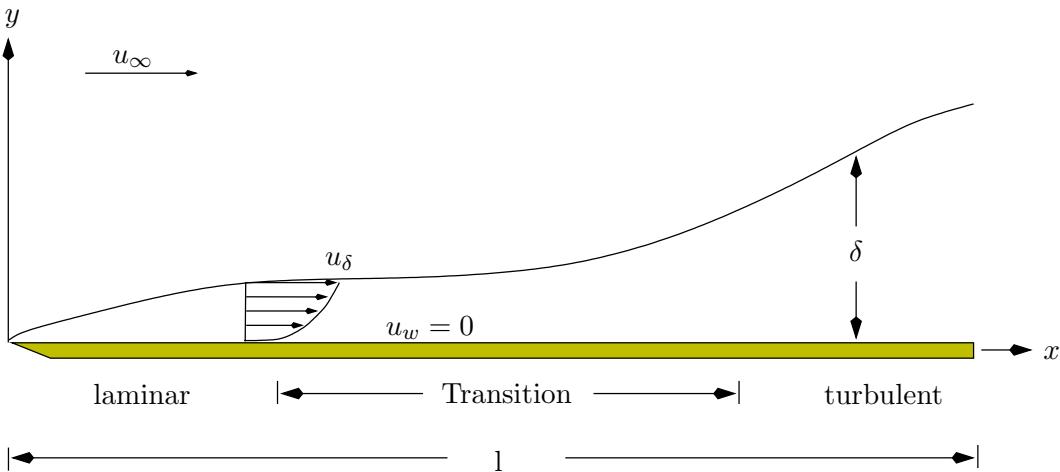


Abbildung 3.19: Die Strömung entlang einer ebenen Platte führt zur Ausbildung einer stromabwärts anwachsenden dünnen Grenzschicht mit  $\delta \ll l$ . Hierbei findet ein Übergang vom laminaren in den turbulenten Zustand statt.

### 3.4 Grenzschichtverfahren

Bei den bisherigen Betrachtungen sind die Wechselwirkungen unterschiedlicher Schichten von Teilchen des Fluids nicht berücksichtigt worden. Insbesondere bei der Bewegung entlang einer festen Wand (z.B. entlang der Profiloberfläche) beeinflusst die, durch die Zähligkeit  $\mu$  induzierte, Schubspannung  $\tau$  die Strömung und führt aufgrund der Haftbedingung an der Wand zur Ausbildung einer Grenzschicht.

Die folgenden Betrachtungen bedienen sich der Strömung entlang einer ebenen Platte als Modell (vgl. Abbildung 3.19)<sup>5</sup>. Wird solch eine Platte mit einer Geschwindigkeit  $u_\infty$  längs angeströmt, so bildet sich, beginnend an der (spitzen) Vorderkante, eine laminare Grenzschicht aus, innerhalb derer das Fluid sich weiterhin auf nicht kreuzenden Stromlinien bewegt. Aufgrund der Haftbedingung bildet sich ein charakteristisches Geschwindigkeitsprofil aus. Die Dicke  $\delta_{99}$  der Grenzschicht wird über die zugehörige Geschwindigkeit  $u_\delta$  definiert, die 99% der Geschwindigkeit der ungestörten Außenströmung entspricht. Die laminare Grenzschicht geht in einen turbulenten Zustand über, bei dem eine unregelmäßige, zufällige Vermischung des Fluids stattfindet<sup>6</sup>. Obwohl der Übergang (Transition), wie dargestellt, stetig erfolgt, bietet es sich für die numerische Behandlung an, einen Umschlagpunkt zu definieren. Die Grenzschicht übt eine Verdrängungswirkung auf die Strömung aus; wie sich später zeigen wird, führt dieses zu einer Verschiebung der Stromlinien senkrecht zur Kontur.

In den folgenden Abschnitten wird auf die Beschreibung der laminaren und turbulenten

<sup>5</sup>Krümmungseinflüsse bleiben somit unberücksichtigt.

<sup>6</sup>Turbulente Strömungen sind definitionsgemäß instationär (vgl. z.B. WALZ (1966)). In dieser Arbeit werden nur stationäre, d.h. zeitlich gemittelte, Zustände betrachtet.

ten Grenzschicht, der Bestimmung des Umschlagpunktes sowie auf die Implementierung und Koppelung mit dem  $S_1$ -Stromfunktionsverfahren eingegangen. Anhand von Beispielen werden Rechenergebnisse experimentellen Daten gegenübergestellt.

### 3.4.1 Grundgleichungen

Zur Beschreibung der Grenzschicht wird zum einen die Kontinuitätsgleichung

$$\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = 0 \quad (3.70)$$

und zum anderen die Impulsgleichung (Prandtlsche Grenzschichtgleichung) in  $x$ -Richtung

$$\rho u \frac{\partial u}{\partial x} + \rho v \frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{\partial \tau}{\partial y} \quad (3.71)$$

herangezogen (vgl. WALZ (1966)). Nach Sutherland ergibt sich die Viskosität  $\mu$  (oder Zähigkeit) eines Mediums als Funktion der Temperatur  $T$  zu

$$\mu(T) = C_1 \frac{T^{3/2}}{T + C_2} \text{ mit } C_1 = 1.486 \cdot 10^{-4} \text{ Ns/m}^2 \text{ K}^{-1/2} \text{ und } C_2 = 110.6 \text{ K für Luft.} \quad (3.72)$$

Die Schubspannung  $\tau$  ergibt sich daraus zu

$$\tau = \mu \frac{\partial u}{\partial y}. \quad (3.73)$$

Der Quotient aus Viskosität und Dichte bezeichnet die kinematische Viskosität  $\nu = \mu/\rho$ . Zur Beschreibung mechanisch ähnlicher Strömungen wird die dimensionslose Reynolds-Zahl mit  $Re = l\rho u/\mu = lu/\nu$  eingeführt. Die Geschwindigkeit  $u$  und die Länge  $l$  stellen hierin charakteristische Größen des Strömungsproblems dar.

Die Integration der beiden Gleichungen (3.70) und (3.71) in  $y$ -Richtung für  $y = 0$  bis zum Grenzschichtrand  $y = \delta$  führt auf die beiden folgenden Differentialgleichungen.

#### GLEICHUNG 4 (ENTRAINMENTGLEICHUNG)

$$\frac{\partial \delta_E}{\partial x} = E - \delta_E \frac{1}{\rho_\delta u_\delta} \frac{\partial(\rho_\delta u_\delta)}{\partial x} \quad (3.74)$$

Die Entrainmentdicke ist definiert für  $\delta_E = \delta - \delta_1$  mit

$$\delta_1 = \delta - \int_0^\delta \frac{\rho u}{\rho_\delta u_\delta} dy \quad \text{als Verdrängungsdicke.} \quad (3.75)$$

Fernerhin wird mit dem Entrainmentkoeffizienten  $E = \frac{\partial \delta}{\partial x} - \frac{v_\delta}{u_\delta}$  eine weitere Variable eingeführt.

### GLEICHUNG 5 (IMPULSGLEICHUNG)

$$\frac{\partial \delta_2}{\partial x} + \delta_2 \left[ \frac{1}{\rho_\delta u_\delta^2} \frac{\partial(\rho_\delta u_\delta^2)}{\partial x} + \frac{\delta_1}{\delta_2} \frac{1}{u_\delta} \frac{\partial u_\delta}{\partial x} \right] = \frac{\tau_w}{\rho_\delta u_\delta^2} = \frac{c_f}{2} \quad (3.76)$$

Die Impulsverlustdicke  $\delta_2$  ist hierbei definiert als

$$\delta_2 = \int_0^\delta \frac{\rho u(u_\delta - u)}{\rho_\delta u_\delta^2} dy. \quad (3.77)$$

Eine Vereinfachung der Schreibweise ergibt sich durch Einführung eines Formfaktors  $h_{12} = \delta_1/\delta_2$ .  $\tau_w$  bezeichnet die Wandschubspannung, während der Reibungsbeiwert mit  $c_f$  angegeben wird.

Die Dichte  $\rho_\delta$  sowie die Geschwindigkeit  $u_\delta$  am Grenzschichtrand werden als bekannt (z.B. als Ergebnis der  $S_1$ -Stromfunktionsrechnung) vorausgesetzt. Die Lösung der beiden Gleichungen kann durch ein Runge–Kutta–Verfahren erfolgen<sup>7</sup>.

### 3.4.2 Algorithmus

Im folgenden wird das algorithmische Verfahren zur Lösung der o.g. Differentialgleichungen beschrieben. Ziel ist es, an  $n$  Stützstellen  $x_0, \dots, x_{n-1}$  die Entrainmentdicke  $\delta_E$  sowie die Impulsverlustdicke  $\delta_2$  zu bestimmen. Die Dichte  $\rho_\delta$  und Geschwindigkeit  $u_\delta$  am Grenzschichtrand ist als bekannt vorausgesetzt. Zusätzlich wird die Temperaturverteilung zur Bestimmung der Viskosität benötigt. Des weiteren bedarf es zur Berechnung von Zwischenwerten  $x \in [x_i, x_{i+1}]$  bei der Integration mit dem Runge–Kutta–Verfahren und zur Berechnung der Ableitungen in  $x$ –Richtung einer geeigneten Interpolationsmethode.

In Abbildung 3.20 ist der Algorithmus in Pseudocode dargestellt. Neben der Lösung für den Staupunkt sind zwei Bereiche (laminare und turbulente Grenzschicht) zu unterscheiden. Verschiedene Tests untersuchen den aktuellen Zustand der Grenzschicht auf mögliche Veränderungen.

### Staupunktlösung

Zur Berechnung der Startlösung an der Stützstelle  $x_0$  wird die Bestimmung der Grenzschichtparameter im Staupunkt auf den Ansatz von Falkner und Skan zurückgeführt. Durch die Einführung der Ähnlichkeitsvariablen

$$\eta = \frac{y}{\delta_N(x)} \quad \text{mit} \quad \delta_N(x) = \sqrt{\frac{2\nu x}{u_\delta(m+1)}} \quad (3.78)$$

<sup>7</sup>Zur Lösung  $y(x)$  einer Differentialgleichung der Art  $y'(x) = f(x, y(x))$  an diskreten Stützstellen  $x_k$  mit der Anfangsbedingung  $y(x_0) = y_0$  kann diese in eine äquivalente Integralgleichung  $y(x_{k+1}) - y(x_k) = \int_{x_k}^{x_{k+1}} f(x, y(x)) dx$  überführt werden. Diese sogenannte *Anfangswertaufgabe* lässt sich durch ein Runge–Kutta–Verfahren lösen (vgl. SCHWARZ (1993)).

In dieser Arbeit wurde auf die `derkf` Routine der SLATEC–Bibliothek VAUDEVENDER UND HASKELL (1982) zurückgegriffen, die eine automatische Anpassung der Schrittweite bei der Integration ermöglicht.

---

```

algorithm PROFILGRENZSCHICHT
begin
0   Berechnung Staupunktlösung für  $x_0$ 
1   Initialisierung laminare Grenzschicht
2    $i \leftarrow 1$ 
3   do ( $i < n \wedge \neg$  laminare Ablösung  $\wedge \neg$  turbulenter Übergang) —>
4       Berechnung laminare Grenzschicht für  $x_i$ 
5        $i \leftarrow i + 1$ 
6   od
7   Initialisierung turbulente Grenzschicht
8   do ( $i < n$ ) —>
9       Berechnung turbulente Grenzschicht für  $x_i$ 
10      Test auf turbulente Ablösung
11       $i \leftarrow i + 1$ 
12  od
end.

```

Abbildung 3.20: Der Algorithmus berechnet an  $n$  diskreten Stützstellen  $x_0, \dots, x_{n-1}$  die Profilgrenzschicht.

gelten nach SCHLICHTING UND GERSTEN (1997) für den Staupunkt die folgenden Beziehungen:

$$\delta_1 = \beta_1 \delta_N, \quad \delta_2 = \beta_2 \delta_N, \quad \frac{\tau_w}{\rho} = \frac{\nu u_\delta}{\delta_N} f''_w \quad \text{sowie} \quad \delta = \beta_{99} \delta_N \quad (3.79)$$

mit

$$m = 1, \quad \beta_{99} = 2.4, \quad \beta_2 = 0.292, \quad f''_w = 1.233 \quad \text{und} \quad h_{12} = \beta_1 / \beta_2 = 2.216. \quad (3.80)$$

### Laminare Grenzschicht

Die Integration der Impulsgleichung für die laminare Grenzschicht benötigt eine Anfangslösung  $\delta_2$  an der Position  $x = \frac{1}{2}(x_0 + x_1)$ , die sich ebenfalls aus der Lösung der Falkner–Skan–Gleichung ergibt.

Die notwendige Bestimmung des Formfaktors  $h_{12}$  und des Reibungsbeiwertes  $c_f$  erfolgt nach der Methode von Thwaites (vgl. CEBECI UND BRADSHAW (1988)). Der zusätzliche Index  $i$  gibt an, daß die Korrelationen nur für inkompressible Strömungen gelten, die anschließend für kompressible korrigiert werden:

$$h_{12i} = \begin{cases} 2.472 + \frac{0.0147}{0.107 + \lambda} & \text{für } -0.1 \leq \lambda \leq 0 \\ 2.61 - 3.75\lambda + 5.24\lambda^2 & \text{für } 0 < \lambda \leq 0.1 \end{cases} \quad (3.81)$$

sowie

$$Re_2 \frac{c_{fi}}{2} = \begin{cases} 0.225 + 1.472\lambda + \frac{0.0147\lambda}{0.107 + \lambda} & \text{für } -0.1 \leq \lambda \leq 0 \\ 0.225 + 1.61\lambda - 3.75\lambda^2 + 5.24\lambda^3 & \text{für } 0 < \lambda \leq 0.1 \end{cases} \quad (3.82)$$

mit

$$\lambda = \frac{\delta_2^2}{\nu} \frac{\partial u_\delta}{\partial x} \quad \text{und} \quad Re_2 = \frac{u_\delta \delta_2}{\nu}. \quad (3.83)$$

CEBEKI UND BRADSHAW (1988) beschreiben den Zusammenhang zwischen den inkompressiblen und kompressiblen Größen. Durch Einführung eines Rückgewinnfaktors (*recovery factor*)  $r_{\text{lam}}$  ergibt sich für  $\sigma = r_{\text{lam}}(\kappa - 1)/2Ma_\delta^2$ :

$$h_{12} = \sigma + h_{12i}(1 + \sigma). \quad (3.84)$$

Mit der Beziehung  $T_w = \sigma T_\delta$  folgt für den kompressiblen Reibungsbeiwert  $c_f$ :

$$c_f = c_{fi} \frac{T_\delta}{T_w} \frac{\mu_w}{\mu_\delta}. \quad (3.85)$$

Mit diesen empirischen Größen ist es nun möglich, die Impulsgleichung für die laminare Grenzschicht zu lösen. Für den Rückgewinnfaktor wird  $r_{\text{lam}} = 0.85$  angenommen.

### Laminares Geschwindigkeitsprofil

Zur Berechnung der Grenzschichtdicke  $\delta$  und damit schließlich auch der Entrainmentdicke  $\delta_E$  wird das laminare Geschwindigkeitsprofil an jeder Stützstelle  $x_i$  berechnet.

Zur Beschreibung des laminaren Geschwindigkeitsprofils wird nach Einführung der Variablen  $\eta = y/\delta$  der folgende Polynom-Ansatz gewählt (vgl. WALZ (1966)):

$$\frac{u}{u_\delta} = a_0 + a_1\eta + a_2\eta^2 + a_3\eta^3 + a_4\eta^4. \quad (3.86)$$

Es sind die folgenden 5 Randbedingungen<sup>8</sup> mitzuberücksichtigen (vgl. Abbildung 3.21):

$$\frac{u}{u_\delta} = 1 \quad \text{für } \eta = 1 \quad (3.87)$$

$$\frac{u}{u_\delta} = 0 \quad \text{für } \eta = 0 \quad (3.88)$$

$$\frac{\partial(u/u_\delta)}{\partial\eta} = f(c_f) \quad \text{für } \eta = 0 \quad (3.89)$$

$$\frac{\partial(u/u_\delta)}{\partial\eta} = 0 \quad \text{für } \eta = 1 \quad (3.90)$$

$$\frac{\delta_1}{\delta} = \int_0^1 \left(1 - \frac{u}{u_\delta}\right). \quad (3.91)$$

---

<sup>8</sup>Bei der Bestimmung des Geschwindigkeitsprofils wird der Einfluß der Kompressibilität nicht berücksichtigt. Daher ist in diesem Fall  $\delta_1$  mit  $h_{12i}$  gebildet zu verwenden (vgl. WALZ (1966)).

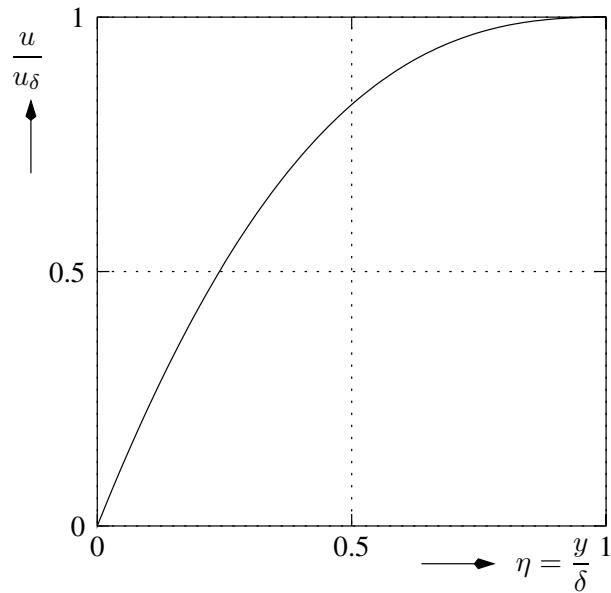


Abbildung 3.21: Das laminare Geschwindigkeitsprofil entsprechend dem Polynomansatz erfüllt die aufgeführten Randbedingungen.

In einem Iterationsverfahren wird  $\delta$  solange verändert, bis das aus Gleichung (3.86) und den Randbedingungen folgende Geschwindigkeitsprofil die linke Seite der Beziehung

$$\delta_2 - \delta \int_0^1 \frac{u}{u_\delta} \left(1 - \frac{u}{u_\delta}\right) d\eta = 0 \quad (3.92)$$

minimiert<sup>9</sup>. Zuvor ist  $\delta_2$  mit dem Runge–Kutta–Verfahren berechnet worden. Die Entrainmentdicke ergibt sich dann aus der Differenz  $\delta_E = \delta - \delta_1$ .

### Laminar–turbulenter Übergang

Der Übergang von der laminaren zur turbulenten Grenzschicht vollzieht sich nicht schlagartig, sondern ist durch einen Übergangsbereich gekennzeichnet (vgl. SCHLICHTING UND GERSTEN (1997)). Der Punkt, an dem die laminare Grenzschicht instabil zu werden beginnt, wird als *Indifferenzpunkt* bezeichnet, und es wird diesem eine Reynolds–Zahl  $Re_{2\text{ind}}$  zugeordnet. Entsprechend wird dem *Umschlagpunkt*, also dem Punkt, ab dem die Grenzschicht als vollturbulent angesehen werden kann, die kritische Reynolds–Zahl  $Re_{2\text{krit}}$  zugewiesen.

Um zu entscheiden, ob eine Grenzschicht an einer Stützstelle  $x_i$  den Übergang vollzogen

<sup>9</sup>Genau genommen handelt es sich um eine Nullstellensuche (lösbar z.B. mit Newtonverfahren oder Bisektionsmethode). Per se ist nicht gewährleistet, daß genau eine Nullstelle existiert. Existieren mehrere Nullstellen, so wird die zuerst gefundene ausgewählt. Existiert keine Nullstelle in dem physikalisch sinnvollen Suchintervall  $\delta \in [\delta_1, 10 \cdot \delta_1]$ , wird versucht, die Abweichung zu minimieren (Approximation).

hat, ist demnach die Reynolds-Zahl  $Re_2$  zu berechnen und mit der kritischen Reynolds-Zahl zu vergleichen.

Für den laminar-turbulenten Übergang der Grenzschicht hat der Störungs- oder *Turbulenzgrad* der Außenströmung einen entscheidenden Einfluß. Hierzu wird der Begriff der *Schwankungsgeschwindigkeit* eingeführt. Die Geschwindigkeit<sup>10</sup> des Fluids  $u$  lässt sich zum einen in eine gemittelte Geschwindigkeit  $\bar{u}$  und zum anderen in eine Schwankungskomponente  $u'$  einteilen, d.h.  $u = \bar{u} + u'$ . Der Turbulenzgrad  $Tu$  ist so dann definiert als:

$$Tu = \frac{\sqrt{\overline{u'^2}}}{u_\infty} \quad , \quad (3.93)$$

wobei  $\overline{u'^2}$  das Quadrat der zeitlich gemittelten Schwankungsgeschwindigkeit beschreibt. Die Abhängigkeit der kritischen Reynolds-Zahl  $Re_{2\text{krit,Platte}}$  vom Turbulenzgrad lässt sich nach GRANVILLE (1953) für eine ebene Plattengrenzschicht wie in Abbildung 3.22 aufzeigen. Die Approximation der experimentellen Daten<sup>11</sup> lässt sich durch den folgenden funktionalen Zusammenhang ausdrücken:

$$Re_{2\text{krit,Platte}} = Re_{2\text{ind}} + 20 + 1000 e^{-2.76 Tu} + 220 e^{-(1.4 Tu - 1)^2} \quad (3.94)$$

SCHLICHTING (1979) gibt für den Indifferenzpunkt die Reynolds-Zahl  $Re_{2\text{ind}} = 200$  an. Als Kriterium für den laminar-turbulenten Umschlag hat sich nach GERSTEN UND HERWIG (1992) die empirische Beziehung

$$Re_{2\text{krit}} = a \left( 1 + \frac{b}{Re} \right) Re^c \quad \text{für } a = 1.174, b = 22400 \text{ und } c = 0.46 \quad (3.95)$$

in vielen Fällen bewährt. Der Koeffizient  $a$  wird angepaßt, um den Einfluß der Turbulenz mitzuberücksichtigen. Es lässt sich  $a$  als Funktion der Reynolds-Zahl  $Re_2$  unter Ausnutzung der für die laminare Plattengrenzschicht gültigen Beziehung  $Re_2 = 0.664\sqrt{Re}$  beschreiben als:

$$a = a(Re_2) = Re_2 \left( \frac{0.664}{Re_2} \right)^{2c} \left( 1 + b \left( \frac{0.664}{Re_2} \right)^2 \right)^{-1} \quad (3.96)$$

Die Berechnung der kritischen Reynolds-Zahl für Profilgrenzschichten nach Gleichung (3.95) in Abhängigkeit des Turbulenzgrades der Außenströmung ergibt sich folglich aus der Bestimmung der kritischen Reynoldszahl  $Re_{2\text{krit,Platte}}$  für Plattengrenzschichten nach Gleichung (3.94) und der anschließenden Bestimmung des Skalierungsfaktors  $a$  mittels der Berechnungsformel (3.96).

---

<sup>10</sup>Hauptströmungsrichtung

<sup>11</sup>Die Datenbasis beruht auf mehreren von verschiedenen Autoren durchgeführten Experimenten.

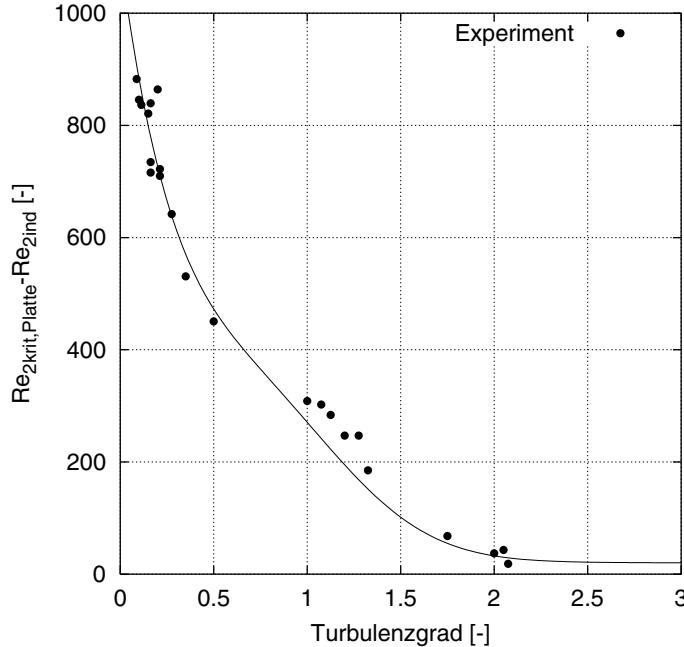


Abbildung 3.22: Die kritische Reynolds-Zahl  $Re_{2\text{krit},\text{Platte}}$  für eine ebene Plattengrenzschicht lässt sich in Abhängigkeit vom Turbulenzgrad (in %) beschreiben. Die Kurve zeigt die Approximation nach Gleichung (3.94) der Experimente aus GRANVILLE (1953).

### Laminare Ablösung

Eine laminare Ablösung ist gegeben, sobald die Wandschubspannung  $\tau_w$  verschwindet, d.h.

$$\tau_w = \mu \left. \frac{\partial u}{\partial y} \right|_w = 0. \quad (3.97)$$

Dieses ist äquivalent zu  $c_f = 0$ . Zur Vermeidung eines numerischen Problems in Gleichung (3.82) wird der Reibungsbeiwert zur Bestimmung einer laminaren Ablösung nicht direkt auf null überprüft, sondern durch  $c_{f,\min}$  abgeschätzt und mit dem berechneten Wert  $c_f$  an der Stützstelle  $x_i$  verglichen. Wird der Minimalwert unterschritten, so liegt eine laminare Ablösung vor (vgl. Abbildung 3.23).

Zur Abschätzung von Geschwindigkeitsprofilen wird angenommen, daß diese sich entsprechend dem Potenzgesetz verhalten (vgl. CEBEKI UND BRADSHAW (1988)):

$$\frac{u}{u_\delta} = \left( \frac{y}{\delta} \right)^{1/n}. \quad (3.98)$$

Hieraus lassen sich dann die folgenden Beziehungen ableiten:

$$\frac{\delta_1}{\delta} = \frac{1}{1+n} \quad \text{und} \quad \frac{\delta_2}{\delta} = \frac{n}{(1+n)(2+n)} \quad \text{sowie} \quad h_{12} = \frac{2+n}{n}. \quad (3.99)$$

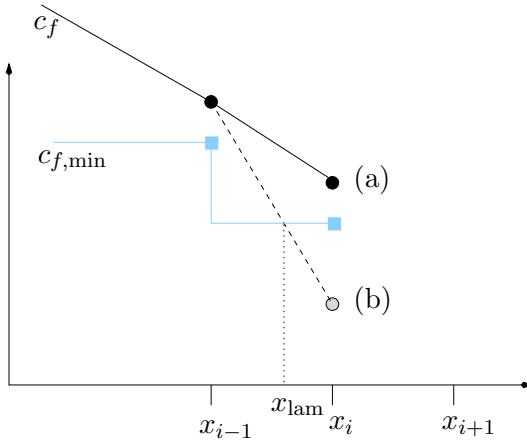


Abbildung 3.23: Bleibt der Reibungsbeiwert  $c_f$  an eine Stützstelle  $x_i$  größer als ein Minimalwert  $c_{f,\min}$ , so ist die Grenzschicht laminar anliegend (Fall (a)). Wird hingegen der Minimalwert unterschritten, lässt sich eine laminare Ablöseposition  $x_{\text{lam}}$  bestimmen (Fall (b)).

Zur Abschätzung wird ein lineares Geschwindigkeitsprofil, d.h.  $n = 1$  unterstellt. Der Minimalwert lässt sich so dann durch

$$c_{f,\min} = \frac{\tau_w}{\frac{\rho}{2} u_\delta^2} = \frac{2\mu}{\rho \delta_2 u_\delta} \frac{\delta_2}{\delta} = \frac{1}{Re_2} \frac{1}{3} \quad (3.100)$$

abschätzen.

### Turbulente Grenzschicht

Ist die Position des laminar-turbulenten Umschlags ermittelt, so bilden die aus der laminaren Rechnung stammenden Werte für  $\delta_2$  und  $\delta_E$  an der Umschlagposition  $x_{\text{tr}}$  bzw.  $x_{\text{lam}}$  die Anfangswerte der turbulenten Rechnung (Gleichung (3.74) und (3.76)).

Zur Integration der Entrainmentgleichung muß zuvor der Entrainmentkoeffizient  $E$  berechnet werden. Für einen Formfaktor  $h_1$  geben CEBECI UND BRADSHAW (1988) die folgende Korrelation an:

$$E = 0.0306 (h_1 - 3.0)^{-0.6169} \quad \text{für} \quad h_1 = \frac{\delta_E}{\delta_2} = \frac{\delta - \delta_1}{\delta_2}. \quad (3.101)$$

In IRMISCH (1991) wird der Zusammenhang zwischen dem inkompressiblen und dem kompressiblen Formfaktor  $h_1$  angegeben. Mit

$$h_1 = h_{1i} \left( 1 + 0.125 \left( \frac{\kappa - 1}{2} \right) Ma_\delta^2 \right) \quad (3.102)$$

lässt sich  $h_{1i}$  bestimmen und anschließend durch

$$h_{12i} = \begin{cases} 0.6778 + \left( \frac{h_{1i} - 3.3}{1.5501} \right)^{-1/3.064} & \text{für } h_{1i} \leq 5.2867 \\ 1.1 + \left( \frac{h_{1i} - 3.3}{0.8234} \right)^{-1/1.287} & \text{für } h_{1i} \geq 5.3092 \end{cases} \quad (3.103)$$

auch der inkompressible Formfaktor  $h_{12i}$ . Der Formfaktor  $h_{12}$  lässt sich aus der Gleichung (3.84) unter Berücksichtigung eines mit dem turbulenten Rückgewinnfaktors  $r_{\text{turb}}$  gebildeten  $\sigma$  angeben. Für den Rückgewinnfaktor wird  $r_{\text{turb}} = 0.89$  angenommen.

Analog zur laminaren Grenzschichtrechnung wird die Impulsgleichung gelöst. Für die Berechnung des Reibungsbeiwertes lässt sich die von DRELA UND GILES (1987) angegebene Korrelation für  $f_c = (1 + 0.2 Ma_\delta^2)^{-1/2}$  ausnutzen:

$$c_f = f_c \left[ \frac{0.3 e^{-1.33 h_{12i}}}{(\log_{10}(Re_2 f_c))^{1.74+0.31 h_{12i}}} + 0.00011 \left( \tanh \left( 4 - \frac{h_{12i}}{0.875} \right) - 1 \right) \right]. \quad (3.104)$$

### Turbulente Ablösung

Analog zur beschriebenen laminaren Ablösung wird die turbulente Grenzschicht untersucht, ob eine Ablösung stattgefunden hat. CEBCI UND BRADSHAW (1988) geben die folgenden Abschätzungen für eine turbulente Plattengrenzschicht an:

$$c_f = 0.059 Re^{-0.2} \quad (3.105)$$

$$\frac{\delta}{x} = 0.074 Re^{-0.2} \quad (3.106)$$

$$\frac{\delta_2}{x} = 0.036 Re^{-0.2}. \quad (3.107)$$

$Re$  ist die mit der Lauflänge  $x$  gebildete Reynolds-Zahl, die sich unter Ausnutzung der aufgeführten empirischen Beziehungen durch  $Re = (Re_2/0.036)^{1.25}$  ergibt. Analog zur laminaren Ablösung lässt sich bei Unterschreiten eines nach Gleichung (3.105) gebildeten Minimalwertes  $c_{f,\min}$  eine turbulente Ablöseposition ermitteln (vgl. auch Abbildung 3.23).

Die Berechnung der abgelösten turbulenten Grenzschicht wird ohne Druckgradienten weitergeführt, d.h. die Ableitungen  $\partial u_\delta / \partial x$  und  $\partial \rho_\delta / \partial x$  in der Entrainment- und der Impuls- gleichung werden zu null gesetzt.

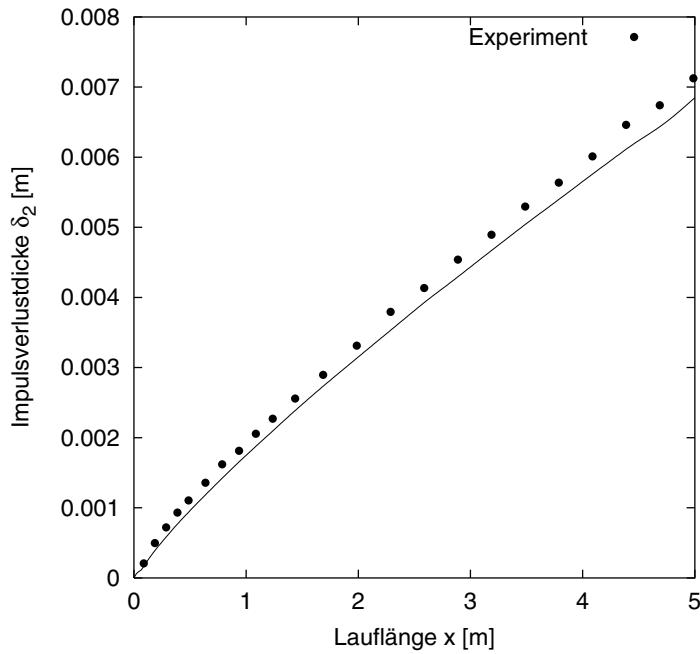


Abbildung 3.24: Die berechnete Impulsverlustdicke bei der turbulenten Grenzschicht wird im Vergleich zum Experiment aus COLES UND HIRST (1969) leicht unterbewertet.

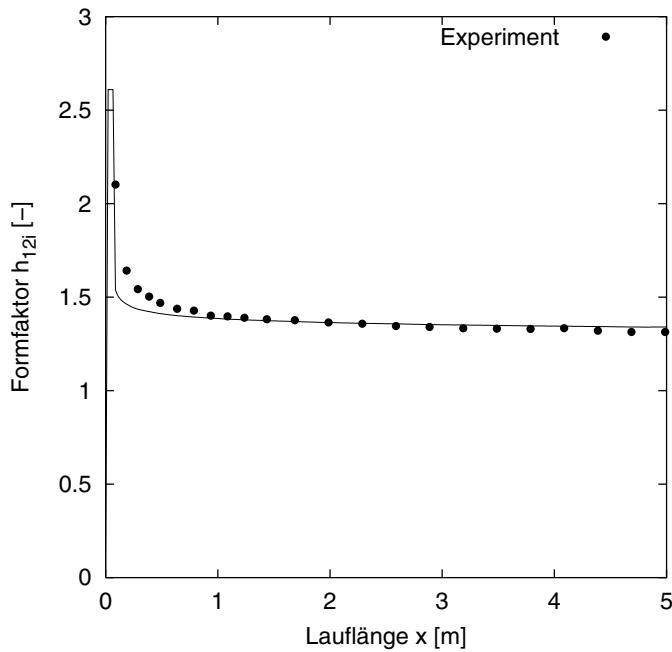


Abbildung 3.25: Die Rechnung ergibt eine mit den experimentellen Daten aus COLES UND HIRST (1969) übereinstimmendes Verhältnis von  $\delta_1/\delta_2$ . Im ersten Bereich ist die laminare Anlaufstrecke mit einem  $h_{12i} \approx 2.5$  erkennbar.

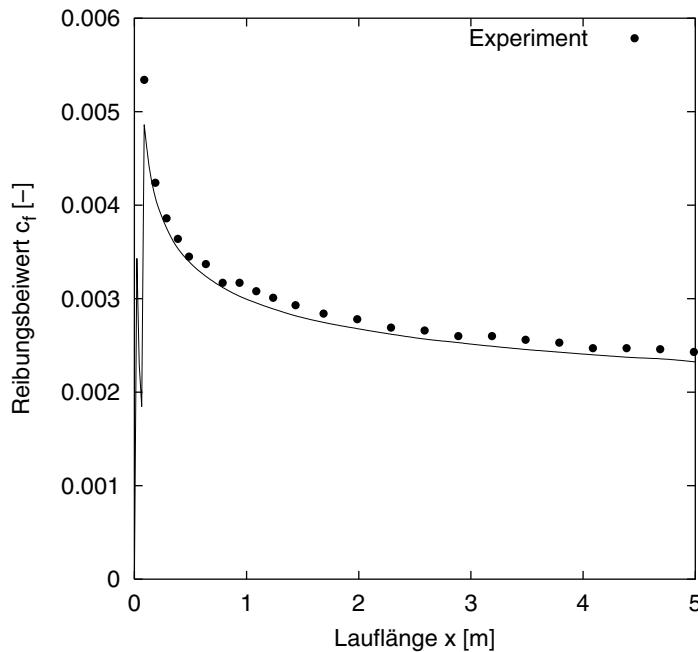


Abbildung 3.26: Der Reibungsbeiwert  $c_f$  wird ähnlich  $\delta_2$  im Vergleich zum Experiment aus COLES UND HIRST (1969) durch das rechnerische Grenzschichtverfahren leicht unterbewertet.

### 3.4.3 Beispiel einer ebenen Plattengrenzschicht

In den Abbildungen 3.24 bis 3.26 sind den experimentellen Daten (vgl. COLES UND HIRST (1969)) die mit o.g. Grenzschichtverfahren ermittelten Ergebnisse gegenübergestellt. Hierbei handelt es sich um eine parallel angeströmte ebene Platte mit einer Geschwindigkeit im ungestörten Bereich von ca. 33 m/s.

Durch einen sogenannten *Stolperdraht* an der Vorderkante der ebenen Platte wird eine turbulente Grenzschicht provoziert<sup>12</sup>. Dieses führt zu einer relativ kurzen laminaren Anlaufstrecke und einem berechneten laminar–turbulenten Umschlag bei  $x_{tr} = 0.0659$  m.

Der Vergleich mit den experimentellen Ergebnissen zeigt eine gute Übereinstimmung mit dem hier vorgestellten Grenzschichtverfahren.

### 3.4.4 Koppelung mit Stromfunktionsverfahren

Einem Iterationsschritt im bereits beschriebenen Stromfunktionsverfahren schließt sich die Grenzschichtrechnung zur Bestimmung der Verdrängungswirkung an<sup>13</sup>. Die Koppelung gliedert sich in drei Schritte. Zum einen ist der Staupunkt an der Vorderkante zu detek-

<sup>12</sup>Der Turbulenzgrad wird mit 25% angegeben.

<sup>13</sup>Bei dem hier verwendeten Mehrgitter–Algorithmus zur Beschleunigung der Konvergenz wird auf den groben Gittern keine Grenzschichtrechnung durchgeführt. Auf dem feinsten Gitter wird die ermittelte Verdrängungswirkung zur numerischen Stabilität unterrelaxiert.

tieren, damit eine Separierung der Grenzschichten für die Saug– und Druckseite möglich wird.

Die Geschwindigkeitsverteilung aus dem Stromfunktionsverfahren, die als Randbedingung in die Grenzschichtrechnung einfließt, ist im Bereich der Vorderkante zu glätten, da ansonsten die dort auftretenden Machzahlspitzen zu physikalisch unbegründeten Phänomenen führen.

Sind die Grenzschichtrechnungen für die Saug– und Druckseite durchgeführt, ist in einem dritten Schritt die Verdrängungswirkung in Form von  $\Delta\psi$ -Werten in den nächsten Iterationsschritt des Stromfunktionsverfahrens als veränderte Randbedingung zu integrieren.

### Bestimmung des Staupunktes

Zur Berechnung der Profilgrenzschichten der Saug– und Druckseite ist in einem ersten Schritt der Staupunkt an der Vorderkante zu detektieren. Die Geschwindigkeit des Fluids im Staupunkt ist per Definition null. Da aufgrund der Diskretisierung des Strömungsfeldes im allgemeinen der Staupunkt nicht aufgelöst wird, bietet es sich an, eine Minimumssuche durchzuführen. Aufgrund von auftretenden Geschwindigkeitsspitzen und eventuellen groben Diskretisierungen im Bereich der Vorderkante hat sich der folgende Ansatz als robuster herausgestellt. Durch

$$w_s = w_u \frac{1}{r} \frac{\partial \varphi}{\partial s} + w_z \frac{\partial z}{\partial s} \quad (3.108)$$

wird eine Geschwindigkeitskomponente entlang der Gitterlinie auf der Profilkontur definiert. Im Staupunkt findet ein Vorzeichenwechsel statt. Da nur *ein* Vorzeichenwechsel zu erwarten ist, kann mittels der Bisektionsmethode die Suche schnell durchgeführt werden.

### Glättung der Machzahlverteilung

Die aus dem Stromfunktionsverfahren berechnete Machzahlverteilung weist auf der Profiloberfläche im Bereich des Staupunktes an der Vorderkante lokale Machspitzen auf, die u.U. sogar weit im Überschall liegen können. Im Gegensatz zu anderen Ansätzen, bei denen der vordere Bereich unberücksichtigt bleibt, wird in dieser Arbeit eine Glättung zur Erhöhung der numerischen Stabilität des Verfahrens durchgeführt. Das physikalische Phänomen einer Beschleunigung um die Vorderkante bleibt somit erhalten.

Das Maximum  $Ma_{\max}$  der Machzahlverteilung im vorderen Bereich wird auf einen Wert  $Ma_n$  skaliert, der der 1.05-fachen Machzahl  $Ma_{1/20}$  für  $x/l_{1/20}$  an der Position 5% entfernt vom Staupunkt entspricht. Die übrigen Zwischenwerte werden anschließend durch eine Interpolation ermittelt. In Abbildung 3.27 ist die Glättung der Machzahlverteilung mit einer Machzahlspitze auf der Druckseite skizziert.

### Verdrängungswirkung

Die im Grenzschichtverfahren berechnete Verdrängungsdicke  $\delta_1$  ist bei dem folgenden Iterationsschritt des Stromfunktionsverfahrens zu berücksichtigen. Ein Ansatz ist es, die Verdrängung durch eine veränderte, aufgedickte Profilgeometrie einfließen zu lassen. Da jedoch

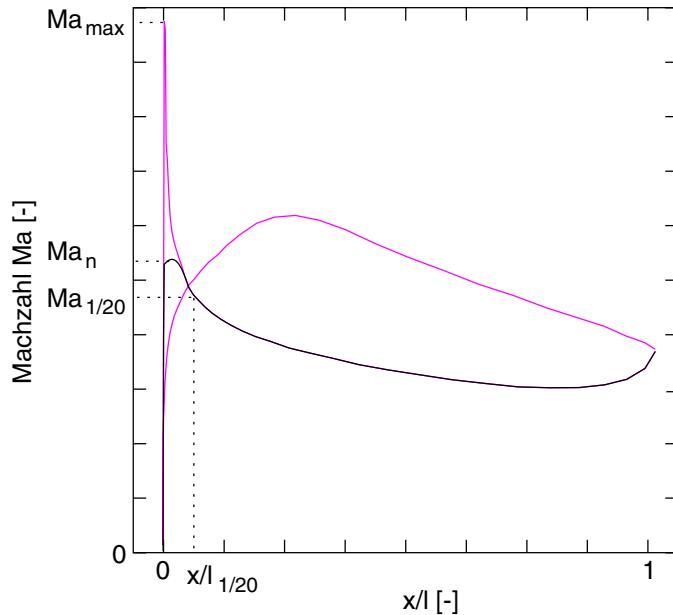


Abbildung 3.27: Die aus dem Stromfunktionsverfahren ermittelte Machzahlverteilung fließt als Randbedingung in die Grenzschichtrechnung ein. Machzahlspitzen an der Vorderkante sind zu glätten.

dadurch ebenso das C–Netz neu berechnet werden muß, ist dieser Ansatz nicht praktikabel. Erfolgversprechender ist es, die Verdrängungswirkung durch einen Massenaustritt aus dem Profil zu simulieren. Aus der Definition der Stromfunktion und der Beziehung aus THIEDE (1977)

$$\rho v = -\frac{\partial \psi}{\partial x} \quad \text{und} \quad v = u \frac{\partial \delta_1}{\partial x} \quad (3.109)$$

ergibt sich

$$-\frac{1}{\rho} \frac{\partial \psi}{\partial x} = u \frac{\partial \delta_1}{\partial x} \quad . \quad (3.110)$$

Die Diskretisierung führt anschließend auf

$$\psi_i = \psi_{i-1} - \Delta \psi_i \quad \text{mit} \quad \Delta \psi_i = \rho_i \cdot w_i (\delta_{1i} - \delta_{1i-1}). \quad (3.111)$$

Dieser Zusammenhang ist in Abbildung 3.28 verdeutlicht. Die Stromlinien für  $\psi = \text{const}$  an der Profiloberfläche werden um den Wert  $\Delta \psi$  verschoben. Dieses führt zu einer, von der Vorder– zur Hinterkante zunehmenden, Aufweitung des Profils.

### 3.4.5 Profilverluste

Zur qualitativen Bewertung von Profilen ist neben der Erfüllung der Umlenkaufgabe unter Berücksichtigung konstruktiver, mechanischer Randbedingungen der induzierte Total-

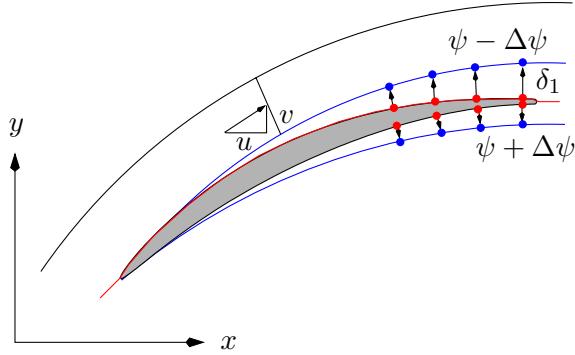


Abbildung 3.28: Die Verdrängungswirkung der Grenzschicht führt zu einer Verschiebung der Stromlinien auf der Profilkontur.

druckverlust  $\omega$  von entscheidener Bedeutung. Im Gegensatz zu RÖPER (1994) wird der Verlust nicht über die sogenannte Lieblein–Korrelation, sondern direkt aus den Ergebnissen des Grenzschichtverfahrens berechnet. Da das Stromfunktionsverfahren die reibungsfreie und damit verlustfreie Strömung berechnet, kann nicht direkt auf die Totalgrößen der verlustbehafteten Strömung zurückgegriffen werden. Zur Verlustbestimmung wird nach SCHOLZ (1965) ein Kontrollvolumen stromabwärts der Hinterkante des Profils definiert, dessen Austrittsebene  $m$  einen ausgemischten Zustand des Fluids beschreibt. Zur Ermittlung des Totaldruckverlustes

$$\omega = \frac{p_{t,1} - p_{t,m}}{p_{t,1} - p_1} \quad (3.112)$$

werden die Bilanzgleichungen für das beschriebene Kontrollvolumen berechnet und der Totaldruck  $p_{t,m}$  in der Ebene  $m$  bestimmt. Mischungsverluste werden durch diese Vorgehensweise mitberücksichtigt. Wie in Abbildung 3.29 skizziert, erstreckt sich das Kontrollvolumen, beginnend an der Hinterkante, über die gesamte Teilung  $t$ . Die Eintrittsebene zwischen den Hinterkantenpunkten  $c$  und  $d$  läuft entlang einer Gitterlinie mit  $\xi = \text{const}$ , wodurch eine Interpolation in Umfangsrichtung mit  $z = \text{const}$  vermieden wird. Die Ebene  $m$  ist in dieser Arbeit identisch mit dem rechten Rand des physikalischen Raumes und kennzeichnet eine Position weit hinter dem Gitter, in der sich Gradienten in Umfangsrichtung ausgemischt haben. Der ausgemischte Zustand wird als Randbedingung des Stromfunktionsverfahrens gefordert (vgl. Abschnitt 3.3.5).

Im ersten Schritt wird die Kontinuitätsgleichung

$$\int_S \rho \vec{c} d\vec{s} = 0 \quad \text{für} \quad d\vec{s} = (ds_z, ds_y) = (bdy, -bdz) \quad (3.113)$$

für eine durch die Eckpunkte  $a$ ,  $b$ ,  $c$  und  $d$  aufgespannten Oberfläche  $S$  betrachtet. Infolge der Periodizitätsbedingung in Umfangsrichtung reicht es für die weiteren Untersuchungen

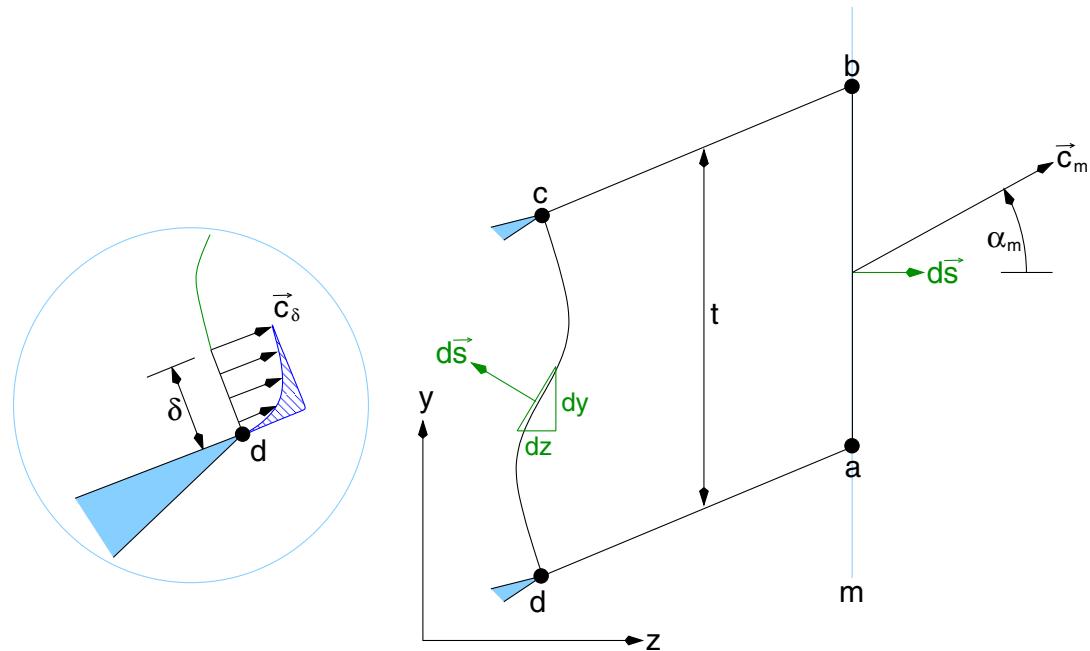


Abbildung 3.29: Zur Bestimmung der Profilverluste wird ein Kontrollvolumen mit den Eckpunkten  $a, b, c, d$  definiert. Die Ebene  $m$  definiert einen ausgemischten Zustand des Fluids. Im linken Bereich ist das Geschwindigkeitsprofil an der Hinterkante und die Differenz zur Geschwindigkeit am Grenzschichtrand (Schraffur) vergrößert dargestellt.

aus, die Anteile der Seiten  $ab$  und  $cd$  zu betrachten, d.h.

$$\underbrace{\int_a^b b\rho(c_z dy - c_u dz)}_{b_m \rho_m c_m t \cos \alpha_m} + \int_c^d b\rho(c_z dy - c_u dz) = 0 \quad . \quad (3.114)$$

Aufgrund der Definition der Austrittsebene  $m$  kann mit  $dz = 0$ ,  $c_z = c_m \cos \alpha_m$  und  $p_m = \text{const}$  das Integral über  $ab$ , wie beschrieben, vereinfacht werden.

In ähnlicher Weise werden die Bilanzen für den Impuls in axialer und in Umfangsrichtung gebildet. Aus

$$\int_S (\rho \vec{c} d\vec{s}) \cdot \vec{c} + \int_S p d\vec{s} = 0 \quad (3.115)$$

folgt für die axiale Richtung

$$\underbrace{\int_a^b b\rho c_z(c_z dy - c_u dz) + \int_a^b bpdy + \int_c^d b\rho c_z(c_z dy - c_u dz) + \int_c^d bpdy}_{(\rho_m c_m^2 \cos^2 \alpha_m + p_m)tb_m} = 0 \quad (3.116)$$

und für die Umfangsrichtung

$$\underbrace{\int_a^b b\rho c_u(c_z dy - c_u dz) - \int_a^b bpdz + \int_c^d b\rho c_u(c_z dy - c_u dz) - \int_c^d bpdz}_{b_m \rho_m c_m^2 t \sin \alpha_m \cos \alpha_m} = 0 \quad . \quad (3.117)$$

Zusammenfassend ergeben sich durch die Einführung von

$$I_1 t = \int_d^c b\rho(c_z dy - c_u dz) \quad (3.118)$$

$$I_2 t = \int_d^c b\rho c_z(c_z dy - c_u dz) + \int_d^c bpdy \quad (3.119)$$

$$I_3 t = \int_d^c b\rho c_u(c_z dy - c_u dz) - \int_d^c bpdz \quad (3.120)$$

die nachfolgenden Beziehungen

$$I_1 = b_m \rho_m c_m \cos \alpha_m, \quad I_2 = I_1 c_m \cos \alpha_m + b_m p_m \quad \text{und} \quad I_3 = I_1 c_m \sin \alpha_m \quad . \quad (3.121)$$

Mit  $T_t = T + c_m^2/(2c_p)$  und unter Berücksichtigung von  $T_t = \text{const}$  wegen der geforderten Energieerhaltung sowie mit  $p_m = RT_m \rho_m$  lässt sich die Geschwindigkeit in der Ebene des ausgemischten Zustands des Fluids als

$$c_m^2 = 2c_p \left( T_t - \frac{p_m}{R \rho_m} \right) \quad (3.122)$$

beschreiben. Die Substitution der Dichte  $\rho_m$  und des Drucks  $p_m$  aus den Gleichungen (3.121) ergibt mit  $\sin^2 \alpha_m + \cos^2 \alpha_m = 1$  schließlich

$$c_m^2 = 2c_p \left\{ T_t - \frac{1}{R} \left[ c_m \cos \alpha_m \left( \frac{I_2}{I_1} - c_m \cos \alpha_m \right) \right] \right\} \quad \text{und} \quad c_m^2 = \left( \frac{I_3}{I_1} \right)^2 + c_m^2 \cos^2 \alpha_m \quad . \quad (3.123)$$

Durch Einführung von  $q = c_m \cos \alpha_m$  lassen sich die Gleichungen (3.123) als

$$q^2 \left( 1 - \frac{2c_p}{R} \right) + q \left( \frac{2c_p}{R} \cdot \frac{I_2}{I_1} \right) + \left( \frac{I_3}{I_1} \right)^2 - 2c_p T_t = 0 \quad (3.124)$$

formulieren. Nach einigen algebraischen Umformungen unter Berücksichtigung von  $c_p/R = \kappa/(\kappa - 1)$  lässt sich abschließend der gesuchte Term als

$$q = \frac{\kappa}{\kappa + 1} \left\{ \frac{I_2}{I_1} - \sqrt{\left( \frac{I_2}{I_1} \right)^2 + \frac{\kappa + 1}{\kappa} \left[ \frac{\kappa - 1}{\kappa} \left( \frac{I_3}{I_1} \right)^2 - 2RT_t \right]} \right\} \quad (3.125)$$

darstellen.

Zur Bestimmung der Integrale, die durch die Eckpunkte  $d$  und  $c$  aufgespannten Fläche, hat sich aufgrund der fehlenden Geschwindigkeitsprofile der Grenzschicht der folgende Ansatz als praktikabel erwiesen. Die Integration an den Integrationsgrenzen erfolgt mit den Geschwindigkeiten<sup>14</sup> am Grenzschichtrand  $\vec{c}_\delta$ . Wie in Abbildung 3.29 skizziert, wird die Differenz der Geschwindigkeitsanteile zum Geschwindigkeitsprofil der Grenzschicht (schraffierte Fläche) subtrahiert. Da die Geschwindigkeitsprofile der turbulenten Grenzschicht im beschriebenen Verfahren nicht explizit berechnet werden, bietet es sich an, auf die Definitionen in Gleichung (3.75) und (3.77) zurückzugreifen. Dichtegradienten in der Grenzschicht werden dabei vernachlässigt. Da ebenfalls auch der Druck  $p$  in der Grenzschicht als konstant vorausgesetzt wird, sind bei den Impulsbilanzen nur die geschwindigkeitsbehafteten Terme zu berücksichtigen. Dieses führt damit auf die folgenden Beziehungen:

$$I_1 t = \int_d^c \{ b\rho(c_z dy - c_u dz) \}_\delta - \underbrace{\int_0^{\delta^d} b\rho(c_\delta - c) ds}_{b_\delta \rho_\delta c_\delta \delta_1 \text{ für Punkt } d} - \underbrace{\int_0^{\delta^c} b\rho(c_\delta - c) ds}_{b_\delta \rho_\delta c_\delta \delta_1 \text{ für Punkt } c} \quad (3.126)$$

$$I_2 t = \int_d^c \{ b\rho c_z(c_z dy - c_u dz) \}_\delta + \int_d^c bpd y - \underbrace{\int_0^{\delta^d} b (\rho_\delta c_\delta^2 \cos \alpha_\delta - \rho c^2 \cos \alpha) ds}_{(b_\delta \rho_\delta c_\delta^2 \cos \alpha_\delta)(\delta_1 + \delta_2) \text{ für Punkt } d} - \underbrace{\int_0^{\delta^c} b (\rho_\delta c_\delta^2 \cos \alpha_\delta - \rho c^2 \cos \alpha) ds}_{(b_\delta \rho_\delta c_\delta^2 \cos \alpha_\delta)(\delta_1 + \delta_2) \text{ für Punkt } c} \quad (3.127)$$

---

<sup>14</sup>Die entsprechenden Terme sind mit  $\{\dots\}_\delta$  bezeichnet.

$$\begin{aligned}
I_3 t = & \int_d^c \{b\rho c_u(c_z dy - c_u dz)\}_\delta - \int_d^c bpdz \\
& - \underbrace{\int_0^{\delta^d} b(\rho_\delta c_\delta^2 \sin \alpha_\delta - \rho c^2 \sin \alpha) ds}_{(b_\delta \rho_\delta c_\delta^2 \sin \alpha_\delta)(\delta_1 + \delta_2) \text{ für Punkt } d} - \underbrace{\int_0^{\delta^c} b(\rho_\delta c_\delta^2 \sin \alpha_\delta - \rho c^2 \sin \alpha) ds}_{(b_\delta \rho_\delta c_\delta^2 \sin \alpha_\delta)(\delta_1 + \delta_2) \text{ für Punkt } c} . \quad (3.128)
\end{aligned}$$

Die Berechnung des Totaldrucks  $p_{t,m}$  der Ebene  $m$  lässt sich somit wie folgt zusammenfassen. Nachdem durch das Grenzschichtverfahren die Verdrängungs- und Impulsverlustdicken für die Ober- und Unterseite an der Hinterkante des Profils berechnet worden sind, lassen sich die Integrale der Gleichungen (3.126) bis (3.128) bestimmen. In einem weiteren Schritt ergibt sich die Größe  $q$  in Gleichung (3.125), aus der dann  $\rho_m$ ,  $p_m$ ,  $T_m$  und damit schließlich der Totaldruck  $p_{t,m}$  folgt. Die Größen in der Eintrittsebene fließen als Randbedingungen in das Stromfunktionsverfahren ein (vgl. Abschnitt 3.3.5) und sind damit als bekannt vorausgesetzt.

Aufgrund von numerischen Ungenauigkeiten stellt sich bereits für  $\delta_1 = 0$  und  $\delta_2 = 0$  ein Totaldruckverlust  $\omega_{\text{inviskos}}$  ein. Dieser ist von der reibungsbehafteten Lösung  $\omega_{\text{viskos}}$  zu subtrahieren, d.h.  $\omega = \omega_{\text{viskos}} - \omega_{\text{inviskos}}$ . Die Integration entlang der Seite  $\overline{cd}$  setzt voraus, daß im Bereich der Profilgrenzschichten sich die Größen der reibungsfreien Strömung nicht ändern. Dieses trifft in der Regel für die numerische Lösung an der Hinterkante nicht zu. Aus diesem Grund hat es sich fernerhin als vorteilhaft erwiesen, die Eintrittsebene  $\overline{cd}$  nicht direkt hinter der Profilhinterkante, sondern einige wenige Gitterpunkte stromabwärts zu definieren. Die Grenzschichtdicken werden dabei als konstant betrachtet. Für eine verfeinerte Bestimmung des Totaldruckverlustes ist eine detaillierte Modellierung des Nachlaufes nötig.

Zusätzlich zum durch die Profilgrenzschichten verursachten Totaldruckverlust wird der Verlust  $\omega_d$  berechnet, der sich durch die Hinterkante des Profils mit einer Dicke  $d > 0$  ergibt. Das betrachtete Kontrollvolumen ist in Abbildung 3.30 skizziert. Für die nachfolgende eindimensionale Betrachtung wird eine uniforme Strömung durch die Ebenen  $m$  und  $\bar{m}$  unterstellt.

Aus der Kontinuitäts-, Impuls- und Energieerhaltung ergeben sich die Beziehungen

$$b_m \rho_m c_m(t-d) = b_{\bar{m}} \rho_{\bar{m}} c_{\bar{m}} t \quad (3.129)$$

$$b_m (\rho_m c_m^2(t-d) + p_m(t-d) + \tilde{p}_m d) = b_{\bar{m}} (\rho_{\bar{m}} c_{\bar{m}}^2 t + p_{\bar{m}} t) \quad (3.130)$$

$$c_p T_m + c_m^2/2 = c_p T_{\bar{m}} + c_{\bar{m}}^2/2 , \quad (3.131)$$

wobei der Druck  $\tilde{p}_m = (p_a + p_b)/2$  sich aus dem Mittelwert der Einzeldrücke auf der Saug- und Druckseite an der Hinterkante zusammensetzt. Durch die bekannten Größen

$$\bar{I}_1 = b_m \rho_m c_m (1 - d/t) \quad \text{und} \quad \bar{I}_2 = b_m (\rho_m c_m^2 (1 - d/t) + p_m (1 - d/t) + \tilde{p}_m d/t) \quad (3.132)$$

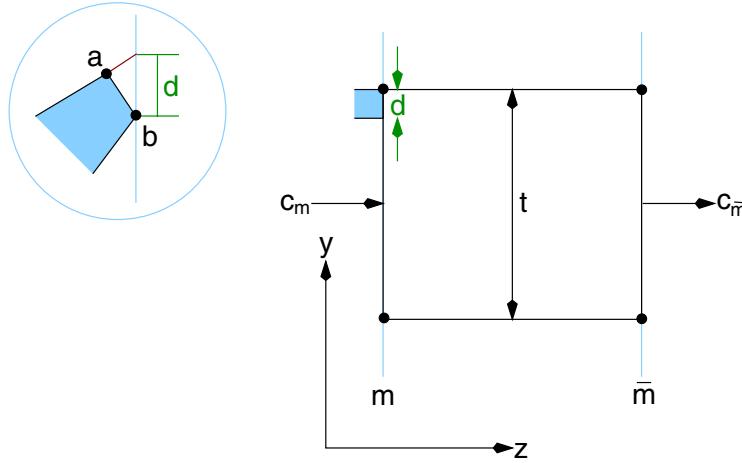


Abbildung 3.30: Zur Bestimmung des zusätzlichen Totaldruckverlustes durch die Hinterkante mit der Dicke  $d$  wird ein Kontrollvolumen, begrenzt durch die Ebenen  $m$  und  $\bar{m}$ , definiert. Im linken Bereich ist die Hinterkante vergrößert dargestellt.

analog zur Vorgehensweise der Gleichungen (3.122) bis (3.125) folgt

$$c_{\bar{m}} = \frac{\kappa}{\kappa + 1} \left\{ \frac{\bar{I}_2}{\bar{I}_1} - \sqrt{\left( \frac{\bar{I}_2}{\bar{I}_1} \right)^2 - \frac{\kappa + 1}{\kappa} 2RT_t} \right\} . \quad (3.133)$$

Aus der Geschwindigkeit der Ebene  $\bar{m}$  und mit  $\bar{I}_1 = b_{\bar{m}}\rho_{\bar{m}}c_{\bar{m}}$  sowie  $\bar{I}_2 = b_{\bar{m}}(\rho_{\bar{m}}c_{\bar{m}}^2 + p_{\bar{m}})$  lässt sich abschließend der Totaldruckverlust, verursacht durch die Profilhinterkante,

$$\omega_d = \frac{p_{t,m} - p_{t,\bar{m}}}{p_{t,1} - p_1} \quad (3.134)$$

bestimmen und zum bisherigen Verlust  $\omega$  addieren.

## 3.5 Nachrechnung

Zur Überprüfung des hier vorgestellten Stromfunktionsverfahrens auf  $S_1$ -Flächen mit dem gekoppelten Grenzschichtverfahren werden exemplarisch zwei ebene Verdichtergitter nachgerechnet und experimentellen Ergebnissen aus der Literatur gegenübergestellt.

### 3.5.1 Verdichterprofil FVV-V1

Zuerst wird die Strömung des ebenen Versuchsgitters FVV-V1 berechnet und mit den experimentellen Daten von WATZLAWICK (1991) verglichen.

Nachfolgend sind die geometrischen und aerodynamischen bzw. thermodynamischen Randbedingungen und Auslegungsdaten zusammengefaßt.

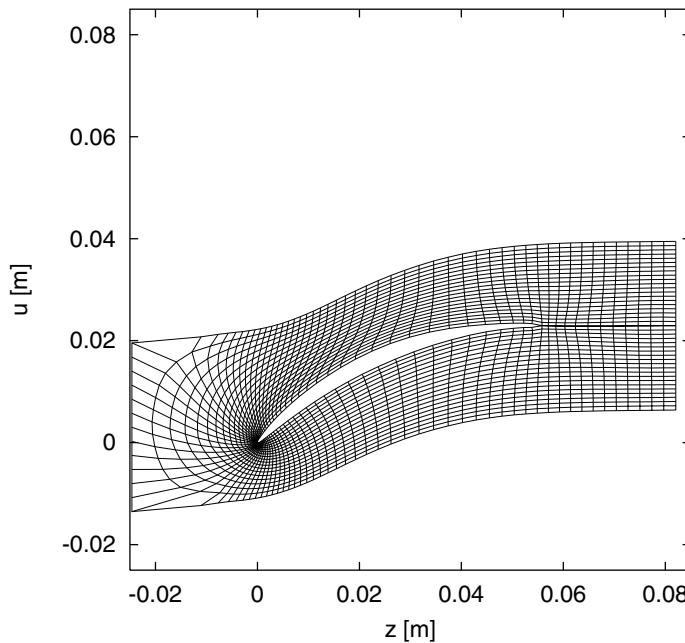


Abbildung 3.31: Das Rechengitter besteht aus  $109 \times 21$  Gitterknoten.

Zuström–Machzahl	$Ma_1$	0.67
Zuström–Reynoldszahl	$Re_1$	450000
Zuströmwinkel	$\alpha_1$	$39^\circ$ ( $42^\circ$ )
Abströmwinkel	$\alpha_2$	$6^\circ$
Sehnenlänge	$l$	0.060 m
Teilungsverhältnis	$t/l$	0.55
Totaldruck	$p_{t,1}$	$65500 \text{ Nm}^{-2}$
Totaltemperatur	$T_{t,1}$	$313.15 \text{ K}$
Turbulenzgrad	$Tu$	4%

Der Zuströmwinkel  $\alpha_1$  ist in den experimentellen Untersuchungen nicht direkt gemessen, sondern aus dem Einbauwinkel des Versuchsgitters ermittelt worden. HILDEBRANDT (1998) hat numerisch einen Winkel  $\alpha_1 = 39^\circ$  berechnet, der vom Auslegungszuströmwinkel ( $42^\circ$ ) um  $3^\circ$  abweicht.

In den Abbildungen 3.31 bis 3.33 sind die Ergebnisse des Stromfunktionsverfahrens dargestellt. Die Machzahlverteilung des Profils wird dem Experiment angemessen wiedergegeben. Der berechnete Totaldruckverlust beträgt  $\omega = 1.99\%$ . WATZLAWICK (1991) gibt den experimentell ermittelten Verlust mit  $\omega = 3.9\%$  an.

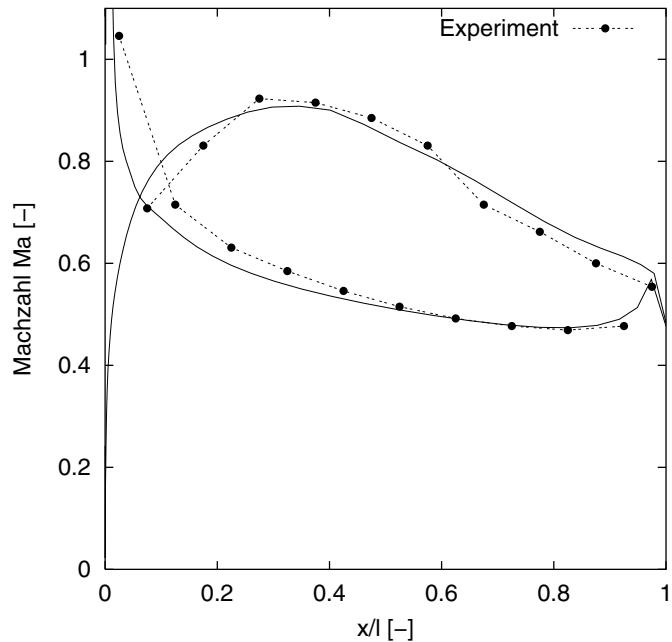


Abbildung 3.32: Die berechnete Machzahlverteilung zeigt eine gute Übereinstimmung mit den experimentellen Ergebnissen von WATZLAWICK (1991).

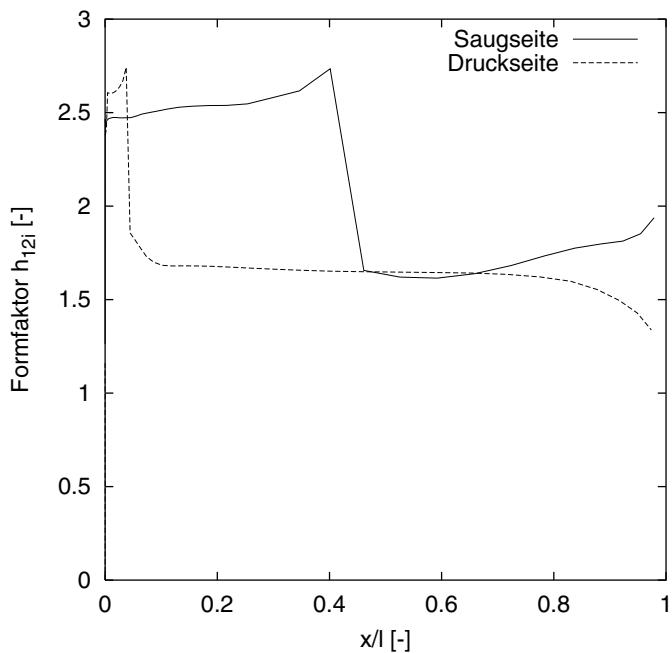


Abbildung 3.33: Die Verteilung des Formfaktors  $h_{12i}$  zeigt einen laminar–turbulenten Um- schlagpunkt auf der Saugseite bei ca. 40% der Sehnenlänge.

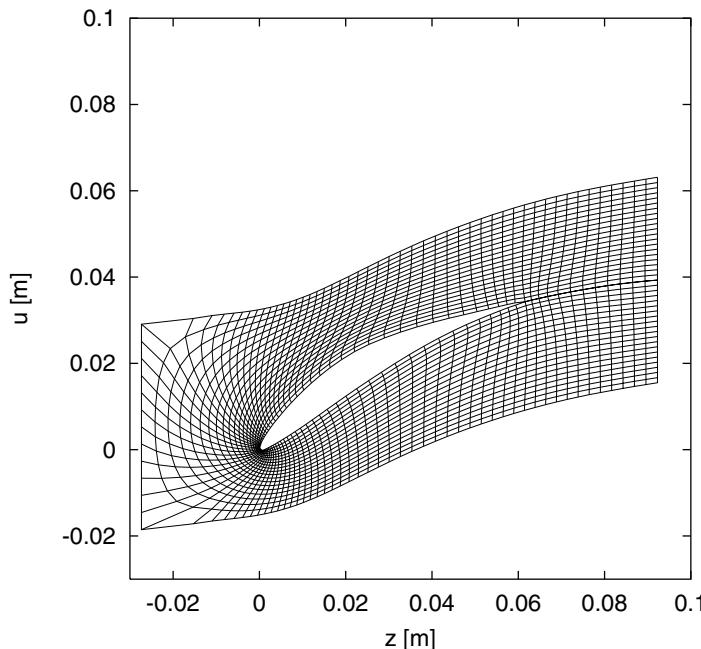


Abbildung 3.34: Der Strömungsraum wird mit  $109 \times 21$  Gitterknoten aufgelöst.

### 3.5.2 Verdichterprofil MAN–GHH–1–S1

Als weiteres Beispiel zur Überprüfung der Leistungsfähigkeit der in dieser Arbeit entwickelten und implementierten Methoden wird das von STEINERT ET AL. (1991) untersuchte Profil herangezogen. Die aerodynamischen Randbedingungen zeigt nachfolgende Tabelle:

Zuström–Machzahl	$Ma_1$	0.618
Zuström–Reynoldszahl	$Re_1$	841000
Zuströmwinkel	$\alpha_1$	$47^\circ$
Abströmwinkel	$\alpha_2$	$20.6^\circ$
Sehnenlänge	$l$	0.070 m
Teilungsverhältnis	$t/l$	0.68
Totaldruck	$p_{t,1}$	$101325 \text{ Nm}^{-2}$
Totaltemperatur	$T_{t,1}$	$287.15 \text{ K}$
Turbulenzgrad	$Tu$	1.5%

In den Abbildungen 3.34 bis 3.36 sind einige Ergebnisse dargestellt. Es zeigt sich, daß eine gute Übereinstimmung mit dem in dieser Arbeit entwickelten Stromfunktionsverfahren erzielt wird. Der berechnete Totaldruckverlust ergibt sich zu  $\omega = 1.43\%$  und ist damit etwas niedriger als der von STEINERT ET AL. (1991) angegebene Wert  $\omega = 1.86\%$  im Auslegungspunkt.

Der Vergleich der Nachrechnungen mit den experimentellen Ergebnissen zeigt, daß das Stromfunktionsverfahren zusammen mit dem Grenzschichtverfahren zur aerodynamischen Bewertung zweidimensionaler Profile hinreichend gut geeignet ist.

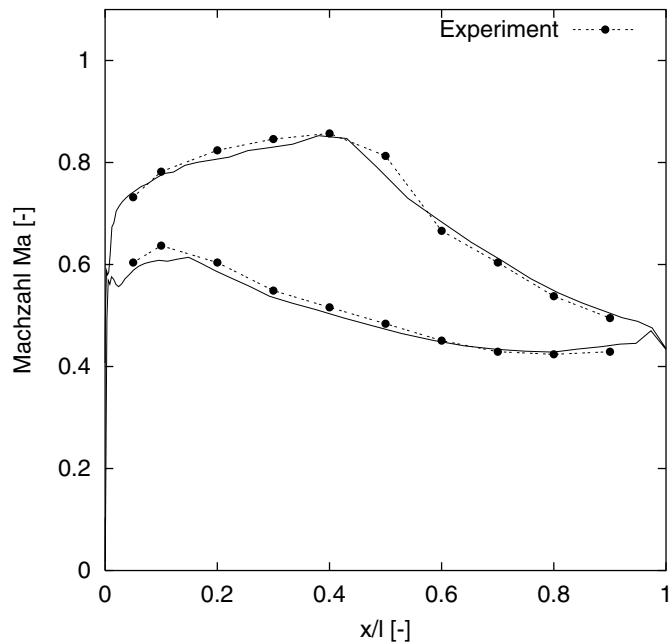


Abbildung 3.35: Die berechnete Machzahlverteilung zeigt eine gute Übereinstimmung mit den Resultaten des Experimentes von STEINERT ET AL. (1991).

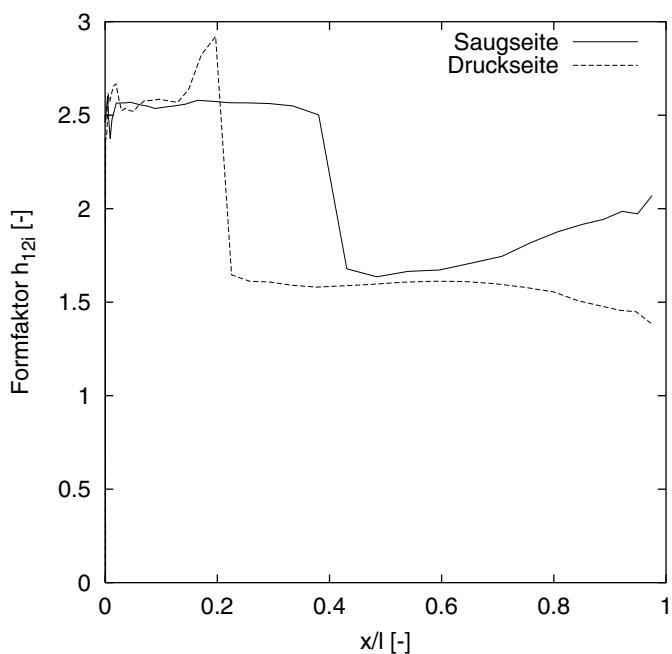


Abbildung 3.36: Der laminar–turbulente Umschlag findet bei  $x/l = 0.4$  statt und stimmt mit dem Ergebnissen des Rechenverfahrens von STEINERT ET AL. (1991) sehr gut überein.



## Kapitel 4

# Approximation mit neuronalen Netzen

Nachfolgend wird ein Verfahren mit neuronalen Netzen beschrieben, welches durch Eingabe von geometrischen und aerodynamischen Randbedingungen in der Lage ist, eine entsprechende Profilgeometrie vorzuschlagen. Zusätzlich werden weitere Eigenschaften dieses Profils vorhergesagt. Der dargestellte Ansatz dient zum schnellen Profilentwurf und als verbesserte Möglichkeit zur Bestimmung einer Startgeometrie in einer sich anschließenden Optimierung.

### 4.1 Einführung

Zum besseren Verständnis künstlicher neuronaler Netze soll an dieser Stelle das biologische Vorbild kurz beschrieben werden. Das Nervensystem lässt sich in zwei Teile untergliedern. Das periphere Nervensystem dient zur Steuerung der inneren Organe und der Skelettmuskeln sowie zur Wahrnehmung durch die Sinneszellen. Gehirn und Rückenmark bilden das Zentralnervensystem. Im Großhirn sind sogenannte Assoziationsfelder zu finden, die Informationen anderer Gehirnregionen miteinander verknüpfen. Insbesondere gehören hierzu das Gedächtnis und die Lernfähigkeit. Neben der hormonalen Steuerung ist des Nervensystems das zweite Steuerungssystem im Körper des Menschen.

Funktional setzt sich das Nervensystem aus dem Zusammenspiel von Verbänden von Nervenzellen (oder Neuronen) zusammen. Trotz der unterschiedlichen Funktionsweisen und Aufgaben lassen sich die Neuronen auf eine, wie in Abbildung 4.1 skizziert, grundlegende Struktur zurückführen. Neuronen bestehen aus einem Zellkörper und verschiedenen Fortsätzen. Diese lassen sich in den Neurit (oder Axon) und in die Dendriten unterscheiden. Die Dendriten empfangen die Ausgangssignale anderer Neuronen und leiten die Summation an den Zellkörper weiter. Wird hier ein Schwellenwert überschritten, dann *feuert* das Neuron, d.h. das Signal wird an das Axon weitergeleitet. Die Signalübertragung zwischen zwei Neuronen erfolgt über sogenannte Synapsen zwischen Dendrit und Neurit, die hemmend oder erregend wirken können (vgl. auch KNODEL UND BAYRHUBER (1985)).

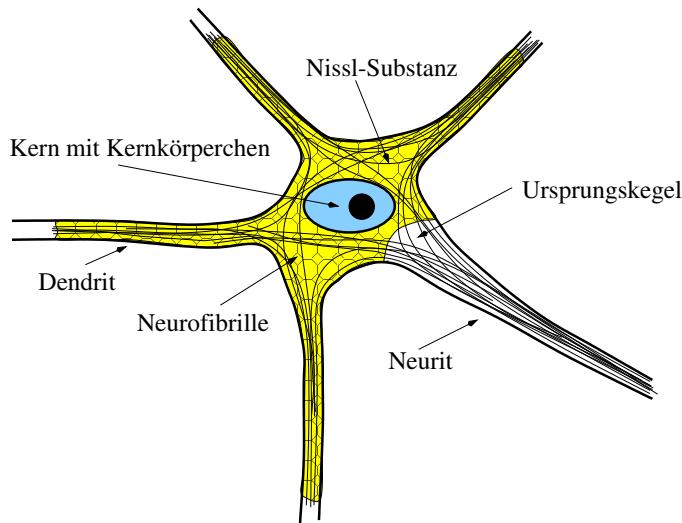


Abbildung 4.1: Das Zytoplasma eines menschlichen Neurons enthält die Nissl–Substanz zur Energieversorgung und Eiweißbildung, die im Gegensatz zum Neurit auch in den Dendriten zu finden ist. Neurofibrillen durchziehen die gesamte Zelle und ihre Fortsätze (adaptiert nach FALLER (1988)). Ein Mensch besitzt mehr als  $10^{10}$  solcher Neuronen.

## 4.2 Künstliche neuronale Netze

Wie bereits in Kapitel 1 dargestellt, ist die Motivation des Konnektionismus, Intelligenz durch eine rechneradäquate Nachbildung des (menschlichen) Gehirns zu erhalten. Obwohl die in den fünfziger Jahren hochgesteckten Ziele zum Bau intelligenter Maschinen in keinem Fall auch nur annähernd erreicht worden sind, bilden künstliche neuronale Netze eine wichtige Möglichkeit, um nichtlineare Zusammenhänge herzustellen. Insbesondere ist ihre Lernfähigkeit hervorzuheben.

### 4.2.1 Abbildung neuronaler Netze

Künstliche neuronale Netze stellen den Versuch dar, die Struktur und die Eigenschaften des biologischen Vorbilds in eine formale, berechenbare Form abzubilden. Abstrakt lässt sich ein neuronales Netz als *black box* betrachten, welches eine  $n$ –stellige Eingabe in eine  $m$ –stellige Ausgabe überführt. Insbesondere sind künstliche neuronale Netze in der Lage, nichtlineare Zusammenhänge abzubilden.

Abbildung 4.2 zeigt beispielhaft ein dreistufiges neuronales Netz. Mathematisch betrachtet, beschreibt die Netztopologie einen gerichteten und gewichteten Graphen aus Knoten (Neuronen) und Kanten (Verbindungen zweier Neuronen). Die einzelnen Neuronen lassen sich in einzelnen Ebenen anordnen, die in die Ein– und Ausgabeschicht sowie eine beliebige Anzahl von (versteckten) Zwischenschichten eingeteilt werden. Für das hier beschriebene Approximationsproblem werden nur vorwärtsgerichtete Netze ohne Rückkopplung

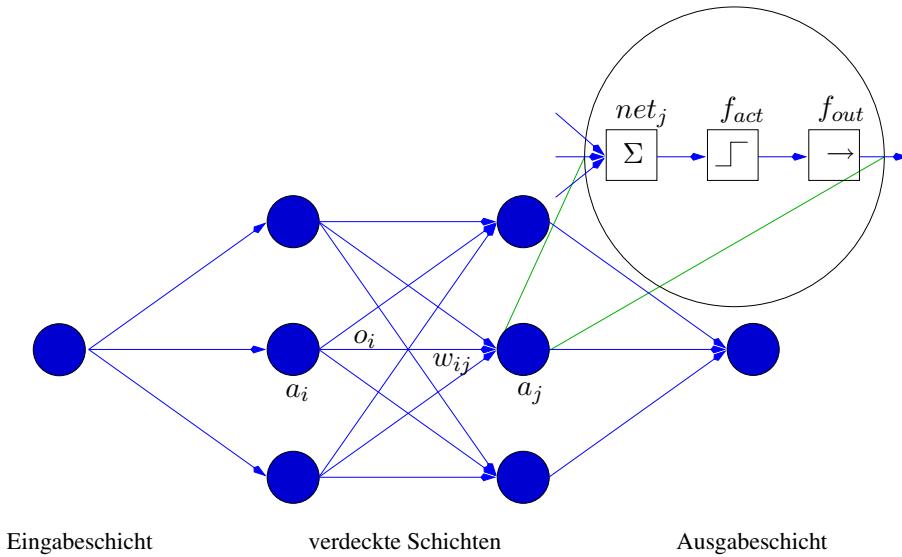


Abbildung 4.2: Modell eines dreistufigen neuronalen Netzes mit 1–3–3–1 Topologie.

betrachtet (azyklischer Graph). Analog zum biologischen Vorbild wird ein Neuron durch eine Eingabe aktiviert und gibt anschließend eine Ausgabe an die entsprechenden Neuronen nachfolgender Schichten weiter (Propagierung).

Der Aktivierungszustand  $a_j$  einer Zelle  $j$  berechnet sich aus der Netzeingabe  $net_j$  und einem Schwellenwert  $\theta_j$ , der sich hemmend auf die Eingabe auswirkt:

$$a_j = f_{act}(net_j - \theta_j) \quad . \quad (4.1)$$

Gebräuchliche Aktivierungsfunktionen sind u.a. Tangens Hyperbolicus und (binäre) Schwellenwertfunktionen. Für die hier beschriebene Anwendung wird die logistische Funktion verwendet, d.h.

$$f_{act}(x) = 1/(1 + e^{-x}) \quad . \quad (4.2)$$

Die Propagierungsfunktion  $net_j$  einer Zelle  $j$  setzt sich aus der Summe der mit  $w_{ij}$  gewichteten Ausgaben  $o_i$  einer Zelle  $i$  zusammen:

$$net_j = \sum_{i=0}^{m_j-1} w_{ij} o_i \quad , \quad (4.3)$$

wobei  $m_j$  die Anzahl der eingehenden Verbindungen einer Zelle  $j$  bezeichnet. Das Gewicht  $w_{ij}$  der Verbindung zweier Neuronen  $i$  und  $j$  ist zuvor durch ein Lernverfahren (siehe Abschnitt 4.2.2) zu bestimmen. Positive Gewichte wirken erregend, negative Gewichte hemmend.

Die Propagierung der Informationen erfolgt asynchron entsprechend einer topologischen

Ordnung von den Eingabeneuronen über die versteckten Schichten zu den Ausgabezellen. Neuronen einer Schicht arbeiten somit (quasi-) parallel.

Die Ausgabe  $o_i$  eines Neurons ist eine Funktion des Aktivierungszustandes, d.h.

$$o_i = f_{out}(a_i) \quad . \quad (4.4)$$

In den folgenden Beispielen wird die Identität als Ausgabefunktion  $f_{out}$  gewählt.

#### 4.2.2 Lernverfahren

Bevor ein neuronales Netz eingesetzt werden kann, muß es zuvor trainiert werden, d.h. das Netz muß eine bestimmte Aufgabe lernen. Hierzu werden die einzelnen Gewichte  $w_{ij}$  so lange verändert, bis der Fehler bei einer vorgegebenen Menge von Trainingsdaten (Muster) minimal wird. Ein Muster besteht aus einem Eingabe- und entsprechenden Ausgabevektor. Der Fehler für ein Muster  $p$  in der Ausgangsschicht mit  $n$  Neuronen wird als

$$E = \frac{1}{2} \sum_{i=0}^{n-1} (t_i - o_i)^2 \quad (4.5)$$

definiert, wobei  $t_i$  die gewünschte und  $o_i$  die nach Gleichung (4.4) berechnete tatsächliche Ausgabe ist. Die Aktualisierung

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t) \quad (4.6)$$

der Gewichte kann nach jedem einzelnen Muster (*online*) oder nach einer Epoche (*offline*), d.i. die Gesamtheit aller Muster, erfolgen. Dieser Vorgang wird solange wiederholt, bis eine vorgegebene Anzahl von Iterationen erreicht wird oder keine signifikante Veränderung mehr eintritt.

Dieser Algorithmus, bei dem versucht wird, das neuronale Netz so zu trainieren, daß dieses zu vorgegebenen Eingabemustern die entsprechenden Ausgabemuster approximiert, wird als überwachtes Lernen bezeichnet.

Als Ansatz zur Bestimmung der  $\Delta w_{ij}$  wird eine zur Ableitung der Fehlerfunktion proportionale Änderung durch

$$\Delta w_{ij}(t) = -\eta \frac{\partial E}{\partial w_{ij}} \quad (4.7)$$

$$= -\eta \frac{\partial E}{\partial net_j} \frac{\partial net_j}{\partial w_{ij}} \quad (4.8)$$

$$= -\eta \frac{\partial E}{\partial net_j} o_i(t) \quad (4.9)$$

gewählt. Der Proportionalitätsfaktor  $\eta$  wird als Lernrate bezeichnet. Durch Anwendung der Kettenregel ergibt sich dann:

$$-\frac{\partial E}{\partial net_j} = -\frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} \quad (4.10)$$

$$= -\frac{\partial E}{\partial o_j} f'_{act}(net_j) \quad . \quad (4.11)$$

Die Ableitung der Fehlerfunktion  $E$  in Gleichung (4.11) berechnet sich für ein Neuron  $j$  der Ausgangsschicht zu

$$-\frac{\partial E}{\partial o_j} = t_j - o_j \quad . \quad (4.12)$$

Da für die Neuronen der verdeckten Schichten keine Trainingsdaten vorliegen, wird die Berechnung auf die Modifikationen bereits berechneter Ebenen zurückgeführt. Für Neuronen der verdeckten Schicht ergibt sich somit:

$$-\frac{\partial E}{\partial o_j} = \sum_k \delta_k w_{jk} \text{ mit } \delta_k = -\frac{\partial E}{\partial net_k} \quad . \quad (4.13)$$

Wird die logistische Funktion (vgl. Gleichung (4.2)) als Aktivierungsfunktion gewählt, lassen sich die Gleichungen (4.11) – (4.13) zu

$$\Delta w_{ij}(t) = \eta o_i(t) \delta_j \quad (4.14)$$

mit

$$\delta_j = \begin{cases} o_j(1 - o_j)(t_j - o_j) & \text{falls } j \text{ Ausgabezelle} \\ o_j(1 - o_j) \sum_k \delta_k w_{jk} & \text{ansonsten} \end{cases} \quad (4.15)$$

zusammenfassen.

Das Training erfolgt also in zwei Schritten. Im ersten, vorwärtsgerichteten Schritt wird ein Muster durch die Aktivierung der Neuronen von der Eingabe– zur Ausgabeschicht propagiert. Im zweiten, rückwärtsgerichteten Schritt werden nun die Gewichte entsprechend der Gleichungen (4.6) und (4.14) angepaßt. Dieses Lernverfahren wird daher auch als (Online–) Backpropagation bezeichnet und ist das am häufigsten eingesetzte Verfahren für vorwärtsgerichtete Netze.

Wie jedes andere Gradientenabstiegsverfahren besitzt auch der Backpropagation–Algorithmus unterschiedliche Probleme bei der Suche nach dem globalen Minimum der Fehlerfunktion  $E$ . Zum einen kann es passieren, daß dieser in einem lokalen Minimum verweilt oder in flachen Ebenen stagniert. Ferner können Oszillationen in steilen Schluchten auftreten.

Eine Modifikation durch die Berücksichtigung vergangener Gewichtsveränderungen führt auf das Backpropagation mit Momentum–Term:

$$\Delta w_{ij}(t + 1) = \eta o_i(t) \delta_j + \alpha \Delta w_{ij}(t) \quad . \quad (4.16)$$

Große Gewichte erschweren die Minimierung der Fehlerfunktion  $E$ . Damit keine zu großen Gewichte zugelassen werden, wird ein Strafterm mittels

$$\Delta w_{ij}(t + 1) = \eta o_i(t) \delta_j - d |w_{ij}(t)| \quad (4.17)$$

eingeführt (Weight Decay).

Eine andere, adaptive Methode verwendet statt der Ableitungen der Fehlerfunktion nur das Vorzeichen und wird als Resilient Backpropagation bezeichnet:

$$\Delta w_{ij}(t) = \begin{cases} -\Delta_{ij}(t) & \text{falls } \frac{\partial \bar{E}(t)}{\partial w_{ij}} > 0 \\ +\Delta_{ij}(t) & \text{falls } \frac{\partial \bar{E}(t)}{\partial w_{ij}} < 0 \\ 0 & \text{ansonsten} \end{cases}, \quad (4.18)$$

wobei  $\bar{E}$  die Summe der Fehler über alle Muster ist. Im Gegensatz zu den beiden ersterwähnten gehört dieser Algorithmus daher zur Klasse der offline-Verfahren. Befindet man sich auf dem ansteigenden Ast der partiellen Ableitung, wird das Gewicht  $w_{ij}(t)$  verringert, im anderen Fall vergrößert. In einem zweiten Schritt wird nun die eigentliche Größe  $\Delta_{ij}(t)$  der Gewichtsveränderung berechnet:

$$\Delta_{ij}(t) = \begin{cases} \eta^+ \Delta_{ij}(t-1) & \text{falls } \frac{\partial \bar{E}(t)}{\partial w_{ij}} \frac{\partial \bar{E}(t-1)}{\partial w_{ij}} > 0 \\ \eta^- \Delta_{ij}(t-1) & \text{falls } \frac{\partial \bar{E}(t)}{\partial w_{ij}} \frac{\partial \bar{E}(t-1)}{\partial w_{ij}} < 0 \\ \Delta_{ij}(t-1) & \text{ansonsten} \end{cases} \quad (4.19)$$

mit

$$0 < \eta^- < 1 < \eta^+ . \quad (4.20)$$

Falls kein Vorzeichenwechsel der partiellen Ableitung stattgefunden hat, wird zur Beschleunigung der Konvergenz die Schrittweite mit  $\eta^+$  vergrößert (Fall 1). Im zweiten Fall war die letzte Schrittweite zu groß und führte dazu, daß das lokale Minimum übersprungen worden ist. Daher wird die Schrittweite verringert. Damit im darauffolgenden Schritt  $t+1$  keine Anpassung erfolgt (Fall 3), wird zusätzlich die Ableitung durch  $\partial \bar{E}(t)/\partial w_{ij} = 0$  zu null gesetzt. Häufige Werte für die beiden Lernfaktoren sind:  $\eta^- = 0.5$  bzw.  $\eta^+ = 1.2$ .

In den meisten Fällen ist eine Skalierung der Trainingsdaten sinnvoll und notwendig. Insbesondere die Ausgabevektoren sind an den Wertebereich der Aktivierungsfunktion bzw. Ausgabefunktion anzupassen. Durch

$$x^* = a^* + \frac{(x - a)}{(b - a)} \cdot (b^* - a^*) \quad (4.21)$$

wird ein Wert  $x \in [a, b]$  in  $x^* \in [a^*, b^*]$  überführt. Da im folgenden die logistische Aktivierungsfunktion verwendet wird, ist daher  $a^* = 0$  und  $b^* = 1$  zu wählen. Versuche haben gezeigt, daß eine wesentlich schnellere Konvergenz durch eine Skalierung auf das Intervall  $[0.1, 0.9]$  erreicht wird. Dieses Phänomen ist auf den Backpropagation-Algorithmus zurückzuführen, der die Ableitung  $f'_{act}$  zur Berechnung der  $\Delta w_{ij}$  verwendet. Aufgrund der

flachen Steigungen an den Rändern des Wertebereichs von  $[0, 1]$  benötigen daher Anpassungen der Gewichte  $w_{ij}$  mehr Iterationsschritte. Für die Eingangsneuronen bietet sich eine Skalierung der Werte um null an, z.B.  $[-1, +1]$ . Interessanterweise stellten sich bei einigen Versuchen in dieser Arbeit Oszillationen ein, die durch eine Skalierung auf  $[0.1, 0.9]$  vermieden werden konnten.

Eine ausführliche Diskussion zu neuronalen Netzen und Lernverfahren ist u.a. in ROJAS (1996) und ZELL (1997) zu finden. Einen kompletten, historischen Überblick über die Entwicklung, Anwendung und Forschung von neuronalen Netzen wird in den Werken von ANDERSON UND ROSENFELD (1988) bzw. ANDERSON ET AL. (1995) gegeben.

### 4.3 Stuttgarter Neuronale Netze Simulator (SNNS)

Zum Aufbau von künstlichen neuronalen Netzen wird in dieser Arbeit der Stuttgarter Neuronale Netze Simulator (SNNS) verwendet, der in den neunziger Jahren von einer Arbeitsgruppe um ZELL (1997) an der Universität Stuttgart entwickelt worden ist. Nach einem Wechsel an die Universität in Tübingen wird der Simulator weiterhin betreut und entwickelt<sup>1</sup>.

Der Simulator dient zum Erzeugen, Trainieren, Analysieren sowie zur Visualisierung von neuronalen Netzen. Hierzu stehen eine Vielzahl von Netzwerktopologien und Lernverfahren zur Verfügung. In Abbildung 4.3 ist die Architektur schematisch dargestellt. Der Simulator besteht im wesentlichen aus zwei Komponenten, dem Kern und der graphischen Bedienschnittstelle, und diversen Hilfswerkzeugen. Der Simulatorkern gliedert sich in drei Teile. Dieser verwendet eine eigene Speicher verwaltung, um effizient neuronale Netze aufzubauen. Durch die Verwendung von dynamischen Feldern ist die Größe der Topologie nur durch den Hauptspeicher des verwendeten Rechners begrenzt. Eine mittlere Ebene enthält die diversen Netzwerkfunktionen, u.a. Initialisierungs- und Aktivierungsfunktionen sowie die verschiedenen Lernverfahren, die durch benutzerdefinierte Methoden erweitert und angepasst werden können. In der oberen Ebene sind zwei Schnittstellen definiert. Zum einen lässt sich die interne Netztopologie in ein ASCII-Format schreiben und lesen und bietet somit einen Austausch mit anderen Programmen. Zum anderen existiert eine Schnittstelle zur Unterstützung der graphischen Benutzerinteraktion. Die graphische Bedienoberfläche xgui, zunächst für das X-Windows System unter UNIX entwickelt und inzwischen auf Microsoft Windows Varianten portiert, enthält neben einem Editor zum interaktiven Entwurf eine Reihe von Möglichkeiten zum Einstellen von Trainingsparametern, einen zweidimensionalen Betrachtungsmodus sowie graphische Ausgaben von Konvergenz- und Fehlerverläufen.

Eine wichtige Möglichkeit ist es, einmal trainierte neuronale Netze als C-Funktion zu exportieren und somit diese, völlig losgelöst vom Simulator und SNNS zugehörigen Funktionsbibliotheken, in eigenen Programmen weiterzuverwenden. Die in einem ASCII-Format

<sup>1</sup>Eine frühe Version gewann den Deutschen Hochschul-Software-Preis. Eine 1993 durchgeführte Schätzung ergab eine Installation von wenigstens 1000 Installationen weltweit. Die z.Zt. aktuelle Version ist 4.2. Die Benutzung für nicht kommerzielle Forschungsvorhaben ist frei.

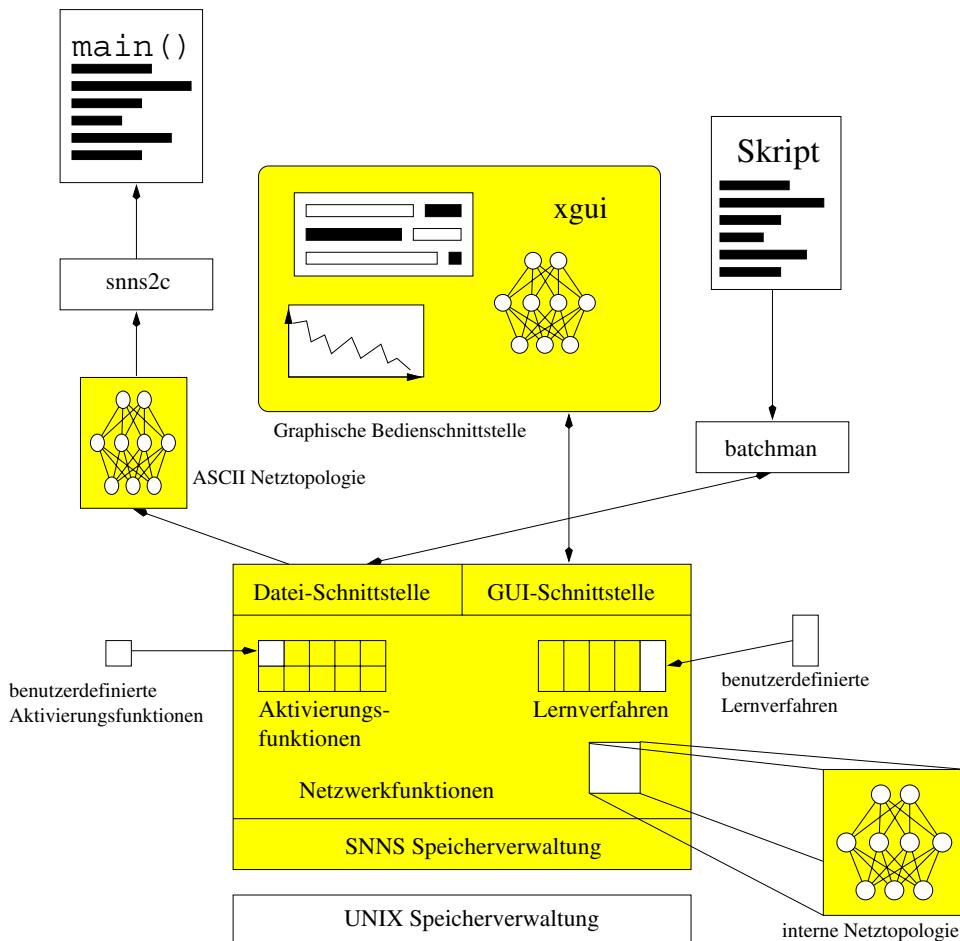


Abbildung 4.3: Die Architektur des Stuttgart Neuronale Netze Simulator gliedert sich in mehrere Komponenten und zusätzliche Hilfswerkzeuge (adaptiert nach SNNS (1998)).

vorliegende Netztopologie wird durch das Programm **snns2c** in eine einzige C-Funktion übersetzt. Der Funktion wird der Eingabevektor übergeben, durch das Netzwerk propagiert und dann der so berechnete Ausgabevektor an den Benutzer zurückgegeben.

Ein weiteres nützliches Werkzeug ist **batchman**, das es ermöglicht, komplizierte und langwierige Lernzyklen durch in einer Skriptsprache definierte Anweisungen ohne die Verwendung der graphischen Bedienschnittstelle durchzuführen. Dieses eignet sich insbesondere zur Untersuchung von Lernparametervariationen.

Zur Überprüfung der Eigenschaften des neuronalen Netzsimulators SNNS sowie zum Vergleich des Einflusses der Parameter der verschiedenen Lernverfahren und der Netzarchitektur ist von UELSCHEN ET AL. (1999) der Einsatz neuronaler Netze bei der Strömungsmessung mit Fünfloch–Drucksonden untersucht worden.

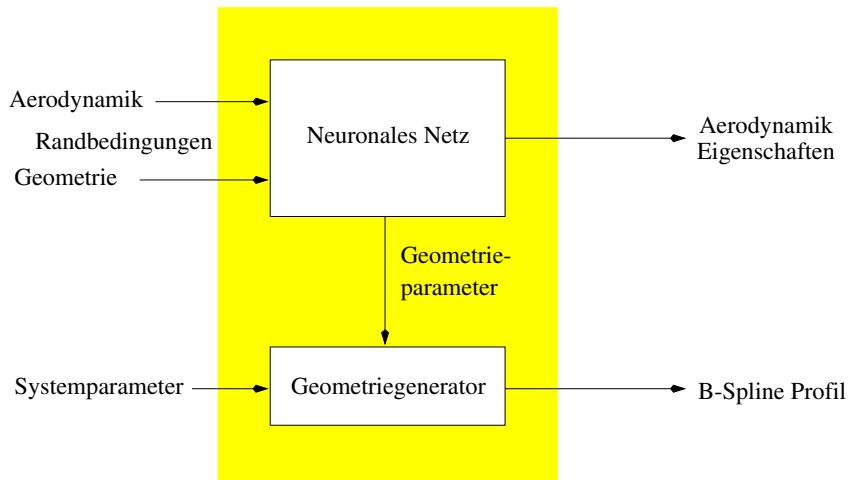


Abbildung 4.4: Das neuronale Netz bildet den Zusammenhang zwischen Aerodynamik und Geometrie ab. Der Geometriegenerator erzeugt Profile in Form periodischer B-Splines.

## 4.4 Profilentwurf

In Abbildung 4.4 ist das in dieser Arbeit entwickelte und nachfolgend beschriebene Verfahren zum Profilentwurf mit neuronalen Netzen schematisch dargestellt. Die Motivation für diesen Ansatz ist es, eine schnelle Methode zur Verfügung zu haben, die ohne aufwendige Strömungsberechnungen Profilkonturen erzeugt, die vorher definierten Anforderungen hinreichend genügen. Geometrische (z.B. maximale Dicke) und aerodynamische (z.B. Zuströmwinkel) Randbedingungen bilden die Eingabe für das neuronale Netz. Nach der Propagierung des Eingabevektors durch das zuvor trainierte Netz werden zum einen geometrische Profilparameter ausgegeben, zum anderen werden weitere anzunehmende aerodynamische Eigenschaften (z.B. Grenzschichtbelastung) des entsprechenden Profils vorhergesagt.

Zur Beschreibung der Profilgeometrie durch das neuronale Netz wird auf eine direkte Formulierung mit B-Spline-Funktionen verzichtet. Zur Reduzierung der Freiheitsgrade wird statt dessen eine Geometriesystematik mit 6 Parametern eingeführt, aus der ein Geometriegenerator eine B-Spline-Funktion nach dem in Abschnitt 2.2.3 beschriebenen Verfahren der direkten Berechnung der Kontrollpunkte erzeugt. Über zusätzliche Systemparameter kann der Geometriegenerator konfiguriert werden.

Durch diese Trennung sind die Designparameter des B-Splines (Grad, Kontrollpunkte, Knotenvektor) unabhängig von der Architektur, insbesondere von der Ausgabeschicht des neuronalen Netzes. Bei einer direkten Ausgabe ist die Anzahl der Parameter nicht variierbar, da die Architektur für ein trainiertes Netz festgelegt und nicht nachträglich modifizierbar ist.

Ein weiterer Vorteil ist es, daß durch diesen Ansatz gewährleistet ist, daß auf jeden Fall glatte B-Spline-Profilen generiert werden. Der Geometriegenerator stellt eine geeignete Pa-

rametrisierung des Knotenvektors  $\mathbf{t}$  sicher, so daß keine selbstschneidenden oder sonstige unerwünschte B-Spline-Kurven produziert werden. Die entwickelte Geometriesystematik ist speziell für Verdichterprofile ausgelegt. Eine Übertragung auf Turbinenprofile ist jedoch ohne Einschränkungen möglich.

Der in Abschnitt 2.2.3 beschriebene Geometriegenerator ist um die Komponente `muoracle` erweitert worden, welches die Koppelung zum neuronalen Netz für den Benutzer transparent implementiert. Durch die Eingabe der aerodynamischen und geometrischen Randbedingungen sowie einiger Systemparameter steht somit „auf Knopfdruck“ eine entsprechende Profilgeometrie zur Verfügung.

Neuronale Netze werden in zweierlei Hinsicht beim Entwurf von Turbomaschinen eingesetzt. Einerseits läßt sich durch Vorgabe gewisser Anforderungen eine zugehörige Geometrie des Schaufelprofils mit neuronalen Netzen realisieren. So beschreibt PFEIL (1994) ein Verfahren zur Approximation von Turbinenprofilen mit neuronalen Netzen. Im Gegensatz zum Verfahren dieser Arbeit werden jedoch nur Parameter zur geometrischen Definition des Profils durch das neuronale Netz bestimmt; eine Vorhersage zusätzlicher aerodynamischer Eigenschaften, z.B. der Grenzschicht, erfolgt nicht. Auch ist die Trainingsmenge, Profile einer Niederdruckturbine von der MTU München, eher gering, so daß nur ein sehr geringer Bereich des Parameterraums abgebildet werden konnte.

Andererseits lassen sich neuronale Netze als Module, sogenannte *black boxes*, in einem übergeordnetem Verfahren, zum Beispiel innerhalb einer Optimierungsstrategie, einsetzen. So läßt sich die i.allg. sehr zeitaufwendige Strömungsberechnung stark reduzieren. Aerodynamische Eigenschaften der Profile werden auf einige Kennzahlen oder Verteilungen reduziert und durch das neuronale Netz approximiert. Diesen Ansatz verfolgen u.a. PIERRET UND VAN DEN BRAEMBUSSCHE (1999), die die Geschwindigkeitsverteilung für je 20 Stützstellen auf der Saug- und Druckseite vorhersagen. KEPPLER (1998) setzt neuronale Netze ein, um das Betriebsverhalten kompletter stationärer Gasturbinen von Siemens/KWU in Kraftwerken abzubilden. Ein zweistufiges Netz wird zur Approximation von Umwelt- und Lastanforderungen auf der einen und von Verbrauchsparametern sowie Abgaswerten auf der anderen Seite verwendet. LO UND SHI (1992) verwenden neuronale Netze zur Klassifikation von Betriebszuständen und zur Detektion von Instabilitäten bei Verdichtern. Vergleichbar der Auslegung von Turbomaschinen ist der Ansatz von GREENMAN (1998), der ein Verfahren zur Prognose von Auftriebskräften an Tragflügeln eines Flugzeugs vorstellt. SANZ (1999) entwickelt ein System mit neuronalen Netzen für den indirekten Entwurf von Profilen.

#### 4.4.1 Netzarchitektur

Bevor auf die Ein- und Ausgabeschicht des neuronalen Netzes eingegangen wird, soll die im folgenden verwendete Geometriesystematik beschrieben werden. Der prinzipiell mögliche Freiraum zur Beschreibung wird zwar mit diesem Verfahren eingeschränkt, aber nur durch die Reduzierung der die Geometrie beschreibenden Parameter ist ein effizientes Trainieren des neuronalen Netzes möglich.

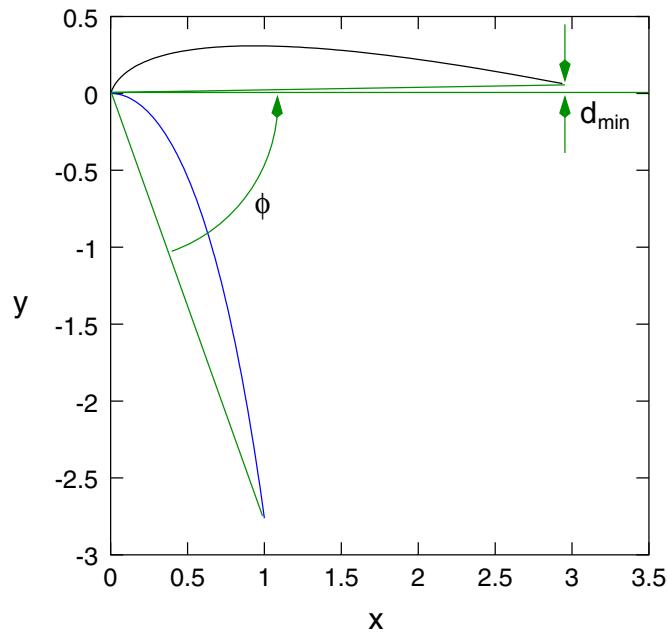


Abbildung 4.5: Die Skizze zeigt die Konstruktion der im folgenden verwendeten Dicken- und Wölbungsverteilung. Die an der  $x$ -Achse gespiegelte Cosinus Hyperbolicus Funktion wird um den Winkel  $\phi$  gedreht.

### Geometriesystematik

Bei der Beschreibung einer Kontur wird die übliche Komposition einer Wölbungslinie mit einer Dickenverteilung verwendet. Bei der mathematischen Formulierung handelt es sich im wesentlichen um eine auf die Ordinate gedrehte und gespiegelte Cosinus Hyperbolicus Funktion, die für  $t \in [0, 1]$  durch

$$f_x(t) = t \cos \phi - (\cosh(at) - 1) \sin \phi \text{ und } f_y(t) = -t \sin \phi - (\cosh(at) - 1) \cos \phi \quad (4.22)$$

gegeben ist. Der Drehwinkel ergibt sich zu  $\phi = -\arctan(\cosh(a) - 1 + d_{\min})$ , wobei  $d_{\min}$  ein vorgegebener Versatz an der Stelle mit  $t = 1$  ist. Dadurch lässt sich eine endliche Dicke an der Hinterkante gewährleisten. In Abbildung 4.5 ist dieser Zusammenhang skizziert. Für die verwendete Wölbungslinie wird  $d_{\min} = 0$  angenommen.

Die weiteren Umformungen dienen zur Skalierung der Funktion, um diese an eigene Anforderungen (z.B. Sehnenlänge, Dicke) anzupassen. Durch die Einführung einer Hilfsfunktion  $h$  wird  $f_y$  auf 1 normiert und mit einer Geraden überlagert:

$$h(t) = \frac{f_y(t)}{f_y(\hat{t})} + t \cdot d_{\min} \quad \text{für } \hat{t} = \frac{1}{a} \operatorname{arsinh} \left( -\frac{\tan \phi}{a} \right) \quad . \quad (4.23)$$

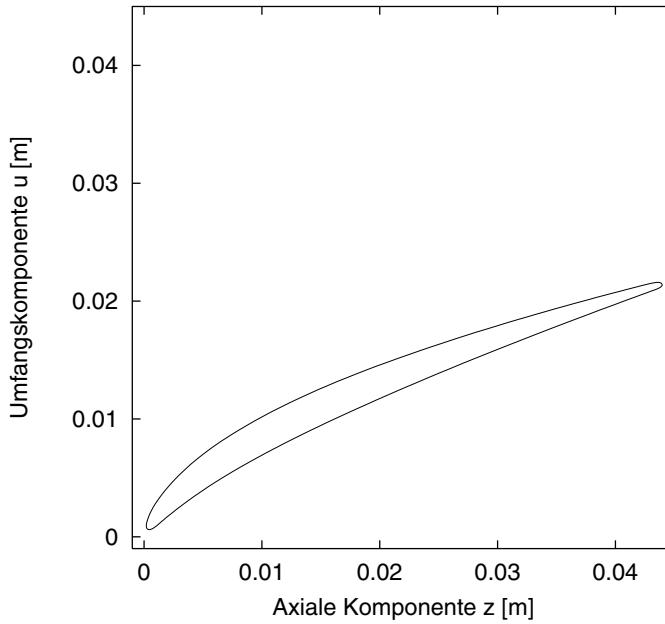


Abbildung 4.6: Beispiel einer Verdichterschaufel mit den Parametern  $a_w = 1.6$  und  $d_{\max,w}/l = 0.071$  für die Wölbungslinie sowie  $a_d = 2.0$  und  $d_{\max,d}/l = 0.06$  für die Dickenverteilung, einem Staffelungswinkel  $\beta = 26^\circ$  und einer Sehnenlänge  $l = 0.046$  m.

Für die  $x$ -Richtung ergibt sich ein Skalierungsfaktor mit  $s = 1/\sqrt{1 + (\cosh(a) - 1 + d_{\min})^2}$ . Für die Dicken- bzw. Wölbungsverteilung folgt daraus:

$$x(t) = s \cdot f_x(t) \quad \text{und} \quad y(t) = d_{\max} \cdot \frac{h(t)}{h(\bar{t})} \quad (4.24)$$

für

$$\bar{t} = \frac{1}{a} \operatorname{arsinh} \left( \frac{-\tan \phi + d_{\min} \cdot f_y(\hat{t})}{a} \right) \quad . \quad (4.25)$$

Für die Berechnung von  $y(t)$  für einen gegebenen Wert  $x$  ist mit einem Iterationsverfahren, z.B. Newton-Verfahren, zuerst der zugehörige Parameter  $t$  zu bestimmen.

Somit sind für die Profilkontur je zwei Parameter ( $a$  und  $d_{\max}$ ) für die Dicken- und Wölbungsverteilung und zusätzlich ein Staffelungswinkel  $\beta$  und die Sehnenlänge  $l$  notwendig. Zur Unterscheidung wird im nachfolgenden die Schreibweise  $a_w$  bzw.  $a_d$  und die auf die Sehnenlänge bezogene Dickenparameter  $d_{\max,w}/l$  bzw.  $d_{\max,d}/l$  für die Wölbungs- bzw. Dickenverteilung eingeführt. In Abbildung 4.6 ist dieser Zusammenhang graphisch aufbereitet.

### Eingabeschicht

Die Neuronen der Eingabeschicht definieren die aerodynamischen und geometrischen Randbedingungen für ein gewünschtes Profil. Neben den Zuström- und Abströmbedingungen kann eine Machzahlverteilung der Saugseite anhand dreier Parameter vorgegeben werden. Die Eingabeschicht besteht im einzelnen aus den folgenden Neuronen (vgl. Abbildung 4.8):

**Zuström- und Abströmbedingungen** Durch die Spezifizierung der Zuströmmachzahl  $Ma_1$  (Neuron 0, *Mach*) und der beiden Winkel  $\alpha_1$  (Neuron 1, *Inlet*) und  $\alpha_2$  (Neuron 2, *Outlet*) werden entscheidene Parameter für die geforderte Profilumlenkung festgelegt.

**Machzahlverteilung** Die gewünschte, auf das Maximum  $Ma_{\max}$  normierte, Machzahlverteilung der Saugseite wird, wie in Abbildung 4.7 skizziert, anhand dreier Parameter beschrieben. Die Festlegung der chordalen Position  $x_{Ma,\max}^* = x_{Ma,\max}/l$  (Neuron 3, *Position*) wird durch die Angabe von  $Ma_{1/3}$  (Neuron 4, *Mach@33%*) und  $Ma_{2/3}$  (Neuron 5, *Mach@67%*) ergänzt, wobei  $x_{1/3}^* = 0.33 + 0.67x_{Ma,\max}^*$  und  $x_{2/3}^* = 0.67 + 0.33x_{Ma,\max}^*$  gilt.

**Geometrische Vorgaben** Durch die Angabe der maximalen Dicke  $d_{\max}/l$  (Neuron 6, *Thickness*), der axialen Breite  $l_{ax}$  (Neuron 7, *Axiallength*) und des Teilungsverhältnisses  $t/l$  (Neuron 8, *Pitch/Chord*) werden geometrische Restriktionen gefordert.

### Ausgabeschicht

Die Ausgabeschicht enthält die Parameter für den Geometriegenerator entsprechend der beschriebenen Geometriesystematik und die prognostizierten Werte für die maximale Machzahl sowie die Grenzschichtbelastung an zwei ausgewählten Punkten der Saugseite. Die 9 Neuronen sind im einzelnen:

**Geometrieparameter** Für die Wölbungslinie und die Dickenverteilung werden durch die Parameter  $a_w$  (Neuron 0, *Geometry 0*) und  $d_{\max,w}/l$  (Neuron 1, *Geometry 1*) bzw.  $a_d$  (Neuron 2, *Geometry 2*) und  $d_{\max,d}/l$  (Neuron 3, *Geometry 3*) beschrieben. Das Profil ist um den Winkel  $\beta$  (Neuron 4, *Stagger*) zu drehen und auf eine Sehnenlänge  $l$  (Neuron 6, *Chordlength*) zu skalieren.

**Aerodynamische Eigenschaften** Der Prognosewert  $Ma_{\max}$  (Neuron 5, *MaxMach*) sowie die Formfaktoren  $h_{12i}$  für die Saugseite an den Positionen  $x = 0.5l$  (Neuron 7, *h12@50%*) und  $x = 0.9l$  (Neuron 8, *h12@90%*) beschreiben weitere Eigenschaften des durch die Geometrieparameter spezifizierten Profils.

Die Anzahl der verdeckten Schichten ist variiert worden. Dabei hat sich herausgestellt, daß mehr als 2 verdeckte Schichten zu einer sehr langsamen Konvergenz führen. Für Untersuchungen sind Konfigurationen mit 16 bzw. 27 verdeckten Neuronen pro verdeckter Schicht betrachtet worden.

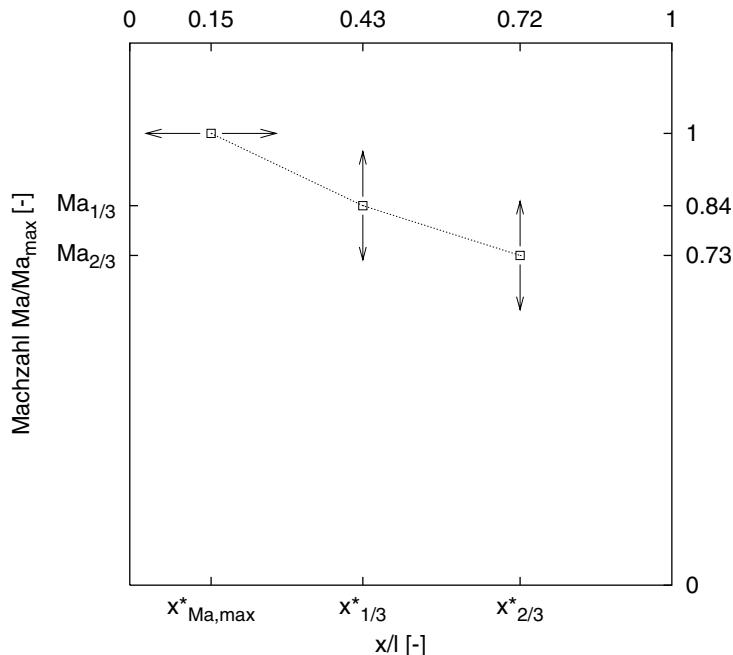


Abbildung 4.7: Die Pfeile geben die Änderungsmöglichkeiten zur Definition der geforderten Machzahlverteilung an. In diesem Beispiel gilt  $x^*_{Ma,\max} = 0.15$ ,  $(x^*_{1/3}, Ma_{1/3}) = (0.43, 0.84)$  und  $(x^*_{2/3}, Ma_{2/3}) = (0.72, 0.73)$ . Als Eingabe in das neuronale Netz fließen die chordale Koordinate  $x^*_{Ma,\max}$  sowie die Machzahlen  $Ma_{1/3}$  bzw.  $Ma_{2/3}$  ein.

#### 4.4.2 Aufbau der Datenbasis und Training

Für den Aufbau der Muster der Trainingsmenge sind insgesamt 807 verschiedene Geometrie- und Aerodynamikkonfigurationen mit dem in Kapitel 3 beschriebenen Stromfunktionsverfahren berechnet worden. Ausgehend von einer Menge von Profilgeometrien mit zugehörigen Zuströmwinkeln  $\alpha_1$  ist die Anströmmachzahl  $Ma_1 \in \{0.3, 0.4, 0.5, 0.6\}$ , das Teilungsverhältnis  $t/l \in \{0.5, 0.8, 1.0, 1.2\}$  und die Sehnenlänge  $l \in \{0.025, 0.0635, 0.1\}$  variiert worden. In die Trainingsmenge sind all jene Konfigurationen übernommen worden, deren maximale Machzahl  $Ma_{\max}$  auf der Saugseite geringer als 1.2 sind.

Bevor das Netz trainiert werden kann, sind die durch die Strömungsberechnung erhaltenen Ein- und Ausgabemuster dem Definitions- bzw. Wertebereich der Aktivierungsfunktion  $f_{act}$  zu skalieren. Hierzu sind unterschiedliche Skalierungsmethoden betrachtet worden. Für Methode I werden nur die Parameter skaliert, deren Werte außerhalb  $[0,1]$  liegen. Hierzu werden der minimale bzw. maximale Wert grob nach unten und oben abgeschätzt und dann auf dieses Intervall skaliert. Bei der zweiten Methode wird für jeden einzelnen Parameter der Ein- bzw. Ausgabeschicht das Minimum und Maximum über alle Trainingsmuster ermittelt und auf dieses Intervall skaliert. Die Tabelle in Abbildung 4.9 faßt diese zwei Ansätze zusammen. Die aufgeführten Intervalle für Methode II geben somit den

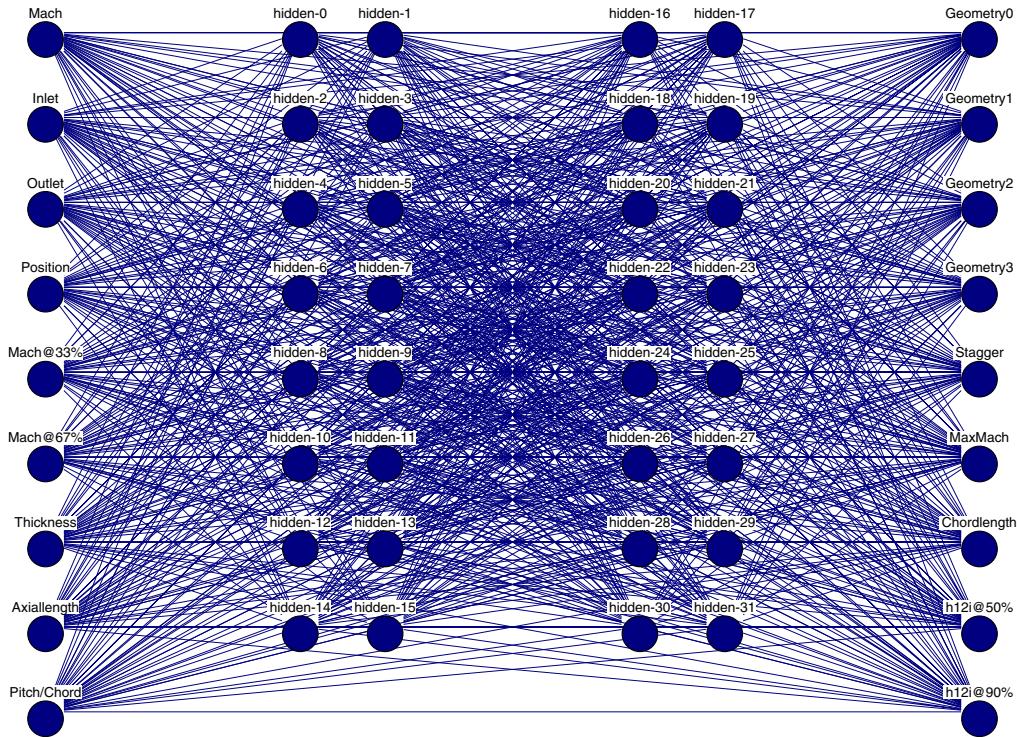


Abbildung 4.8: Zur Geometriedefinition wird ein Netz mit einer Eingabe- bzw. Ausgabeschicht von jeweils 9 Neuronen verwendet. Die Anzahl der Neuronen in den verdeckten Schichten ist variiert worden. Im Beispiel sind zwei verdeckte Schichten mit jeweils 16 Neuronen abgebildet.

Wertebereich der trainierten Daten an. Für die Neuronen ohne Angabe für die Methode I werden die Werte nicht skaliert. Die Intervallgrenzen sind für die Eingabe- und Ausgabeneuronen durch  $a^* = 0.1$  und  $b^* = 0.9$  gegeben. Methode II hat den Nachteil, daß eine Extrapolation durch das neuronale Netz nur begrenzt möglich ist. Dafür werden die Werte gleichmäßiger auf das Intervall  $[0.1, 0.9]$  verteilt.

Zum Trainieren des neuronalen Netzes sind die in Abschnitt 4.2.2 beschriebenen Verfahren untersucht worden. Die besten Ergebnisse sind mit der Momentum-Term-Variante und dem Resilient-Backpropagation-Ansatz erzielt worden. Die Anzahl der Lernzyklen beträgt 25000.

#### 4.4.3 Approximationsgüte und Generalisierungseigenschaften

Zur Bewertung des trainierten neuronalen Netzes sind zuerst die Approximationseigenschaften der 807 Trainingsmuster zu untersuchen. In Abbildung 4.10 ist in einer Histogrammdarstellung der Approximationsfehler für den Staffelungswinkel  $\beta$  abgebildet. Der

Neuron	Eingabeschicht		Ausgabeschicht		[ ]
	Methode I	Methode II	Methode I	Methode II	
0	–	[0.3, 0.6]	[ ]	[1.0, 3.0]	[1.0, 3.0]
1	[-60, 60]	[-9, 52]	[°]	–	[0.0, 0.3]
2	[-60, 60]	[-48.25, 31.82]	[°]	[1.0, 3.0]	[1.0, 3.0]
3	–	[0.05173, 0.482]	[ ]	–	[0.07, 0.2]
4	–	[0.5952, 0.9892]	[ ]	[-60, 60]	[-23.5, 30.0]
5	–	[0.4666, 0.9621]	[ ]	[0.3, 1.3]	[0.3442, 1.234]
6	–	[0.06978, 0.2001]	[ ]	[0.0, 0.2]	[0.025, 0.1]
7	[0.0, 0.2]	[0.02451, 0.09483]	[m]	[1.3, 2.8]	[1.457, 2.736]
8	[0.4, 1.6]	[0.507, 1.235]	[ ]	[1.3, 2.8]	[1.43, 1.779]

Abbildung 4.9: Die Tabelle zeigt die untersuchten Methoden zur Skalierung der Trainingsmuster. Die angegebenen Intervalle geben den Bereich  $[a, b]$  an, der nach Gleichung (4.21) auf das Intervall  $[0.1, 0.9]$  transformiert wird.

Fehler für die Trainingsdaten ist sehr gering. Weitere Ergebnisse sind in der Abbildung A.15 im Anhang zu entnehmen.

Neben der Fähigkeit des neuronalen Netzes, die gelernten Trainingsmuster zu approximieren, ist die Generalisierungseigenschaft wichtig. Um das Verhalten des neuronalen Netzes zu untersuchen, auf unbekannte Testdaten zu reagieren, ist in einem zweiten Schritt eine Validierungsmenge mit insgesamt 96 Mustern erstellt und mit dem trainierten Netz anschließend der Approximationsfehler berechnet worden. In Abbildung 4.11 sind die Ergebnisse für das Ausgabeneuron zur Prognose der maximalen Machzahl  $Ma_{\max}$  für die beiden Skalierungsmethoden in einem Histogramm dargestellt. Die Abweichung ist sehr gering, 90% aller Muster haben eine Abweichung von weniger als 0.02  $Ma_{\max}$ .

Diese eher statistischen Aussagen werden durch zwei Testkonfigurationen näher untersucht. Hierzu ist für eine Reihe von Randbedingungen eine systematische Parametervariation durchgeführt worden. Die zwei Testkonfigurationen sind im einzelnen:

Randbedingung	Konfiguration 1	Konfiguration 2
$Ma_1$	0.5	0.4
$\alpha_1$	30.0	10.0
$\alpha_2$	5.0	-10.0
$x_{Ma,\max}$	0.15	0.12
$Ma_{1/3}$	0.84	0.84
$Ma_{2/3}$	0.73	0.73
$d_{\max}/l$	0.08	0.08
$l_{ax}$	0.1	0.055
$t/l$	1.0	1.0

In den Abbildungen 4.12 bis 4.14 und im Anhang A.4 sind einige der graphisch aufbereiteten Ergebnisse dargestellt. Als ein erstes, sehr wichtiges Ergebnis kann festgehalten werden,

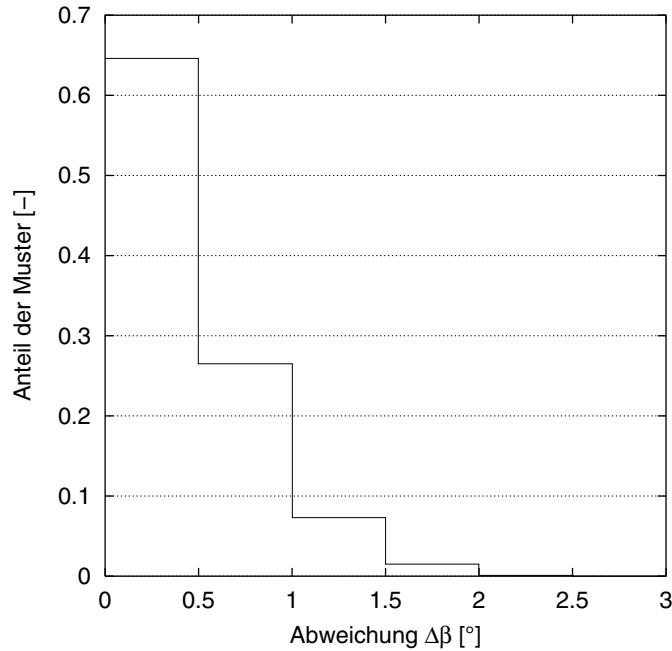


Abbildung 4.10: Der Approximationsfehler für den Staffelungswinkel  $\beta$  ist für die meisten Muster sehr klein ( $\Delta\beta < 1.5^\circ$ , Skalierungsmethode I).

daß das trainierte neuronale Netz sich *wohldefiniert* verhält. Ein kleine Veränderung einer Randbedingung erzeugt auch eine vergleichsweise kleine Änderung in der Ausgabeschicht. Ein oszillierendes oder gar chaotisches Verhalten konnte nicht festgestellt werden. Es zeigt sich, daß das neuronale Netz für die Konfiguration 1 i.allg. bessere, d.h. mit geringeren Abweichungen zu den Vorgaben bzw. den prognostizierten Werten, Profilgeometrien ermittelt. Bei der Untersuchung sind die Zuströmmachzahl, der Zuström- und Abströmwinkel, sowie das Teilungs- bzw. Dickenverhältnis variiert worden.

In Abbildung 4.14 wird die Vorhersage für die Grenzschichtbelastung untersucht. Der Zusammenhang, daß der Formfaktor  $h_{12i}$  an der Position  $x = 0.5l$  geringer als an der Position  $x = 0.9l$  ist, wird durch das neuronale Netz erkannt. Allerdings ist anzumerken, daß eine Entlastung des Profils (der Abströmwinkel  $\alpha_2$  wird dementsprechend größer) durch einen geringeren Formfaktor in der Nachrechnung zwar richtig wiedergegeben wird, das neuronale Netz jedoch eine gegenläufige, nicht gewünschte Tendenz zeigt.

In Abbildung 4.15 ist die Fähigkeit des trainierten neuronalen Netzes dargestellt, auf eine Veränderung einer Randbedingung zu reagieren. Die Geometrie des Ausgangsprofils wird konstant gehalten, und für verschiedene Zuströmwinkel  $\alpha_1 \in \{0^\circ, 5^\circ, \dots, 25^\circ\}$  wird durch eine Strömungsberechnung die Machzahl  $Ma_{1/3}$  berechnet, die z.T. erheblich von der vorgegebenen Randbedingung  $Ma_{1/3} = 0.84$  abweicht. Wird der veränderte Zuströmwinkel durch das neuronale Netz propagiert, so ergeben sich Profile, die eine wesentlich geringere Abweichung zur Vorgabe haben.

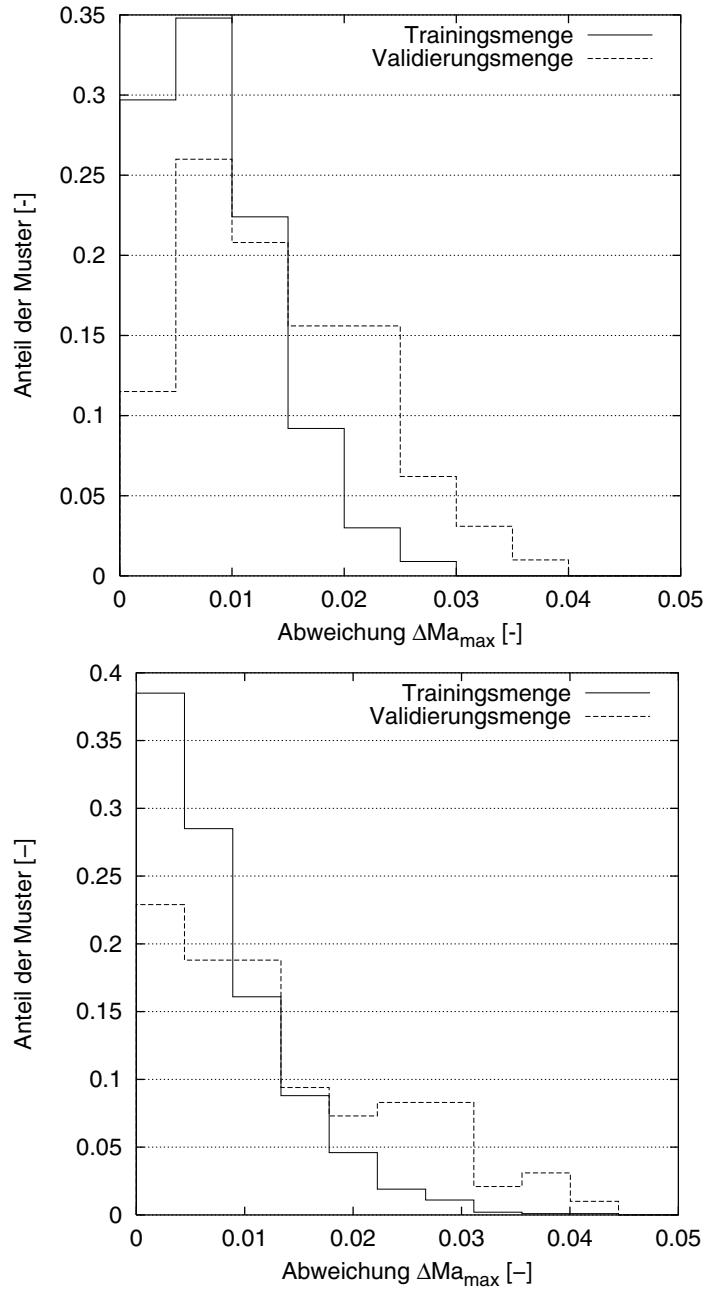


Abbildung 4.11: Der Vergleich des Approximationsfehlers für die Skalierungsmethode I (obiges Histogramm) und Methode II zeigt ein ähnliches Verhalten. Der jeweilige Fehler der Validierungsmenge ist, wie nicht anders zu erwarten war, größer.

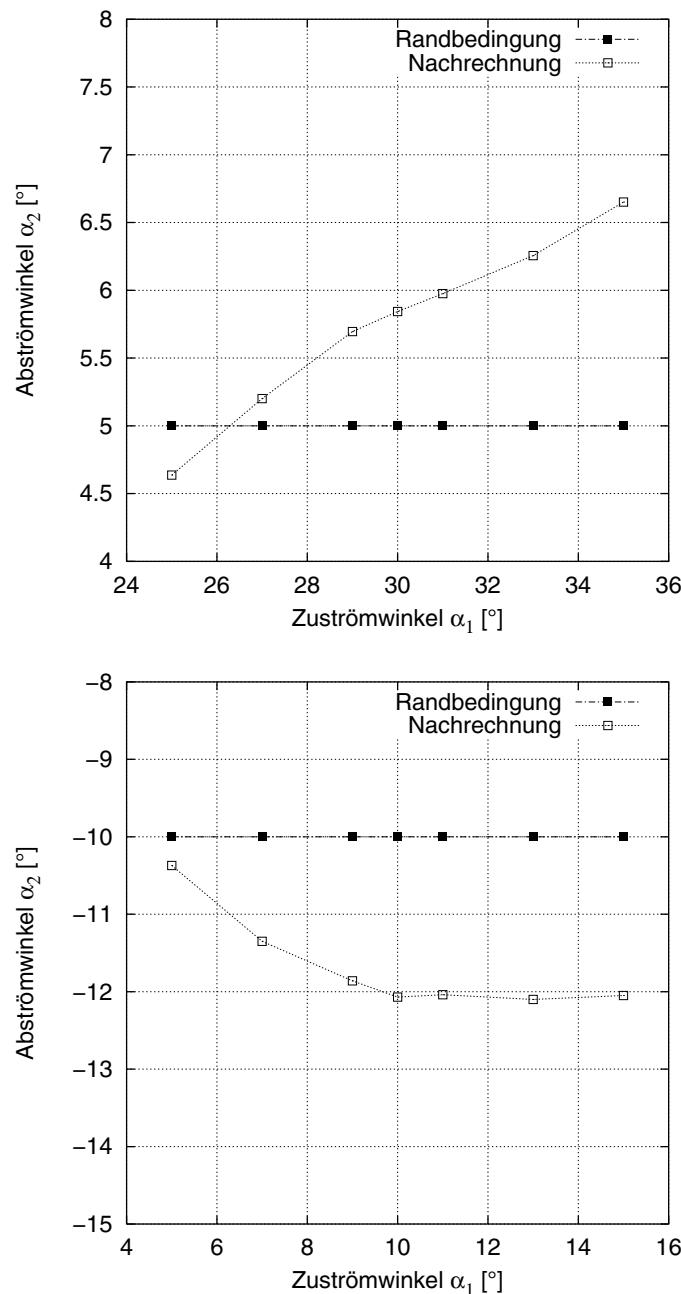


Abbildung 4.12: Die Abweichung des tatsächlichen Abströmwinkels  $\alpha_2$  ist bei einer Variation des Zuströmwinkels  $\alpha_1$  im wesentlichen geringer als  $2^\circ$ .

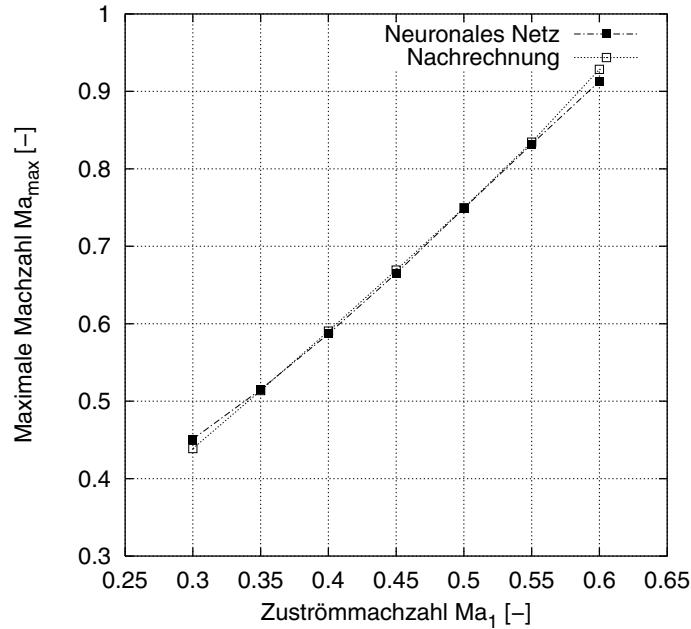


Abbildung 4.13: Bei einer Variation der Zuströmmachzahl für die Konfiguration 1 mit  $Ma_1 \in \{0.3, 0.35, \dots, 0.6\}$  zeigt beim Vergleich der tatsächlichen mit der vom neuronalen Netz prognostizierten maximalen Machzahl  $Ma_{max}$  ein sehr gute Übereinstimmung.

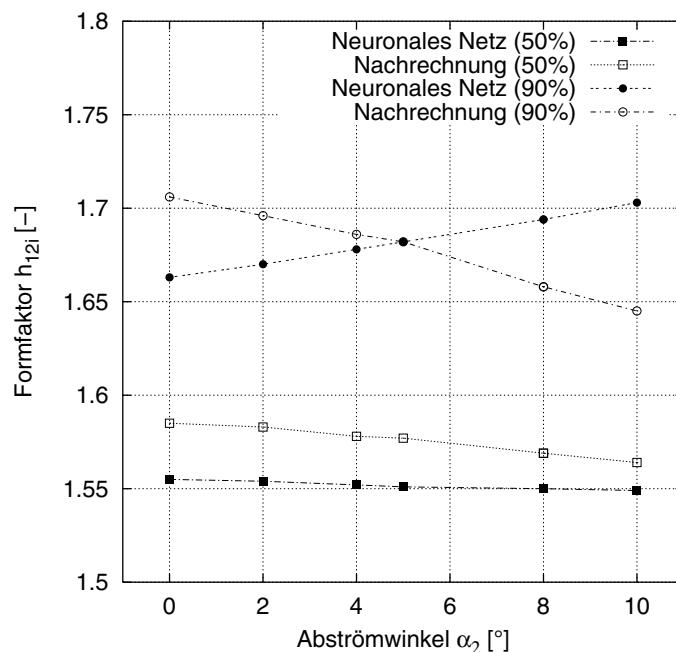


Abbildung 4.14: Die Vorhersage des Grenzschichtformfaktors  $h_{12i}$  an der Position 50% bzw. 90% der chordalen Koordinate zeigt eine Abweichung im Mittel von  $\Delta h_{12i} \approx 0.05$ .

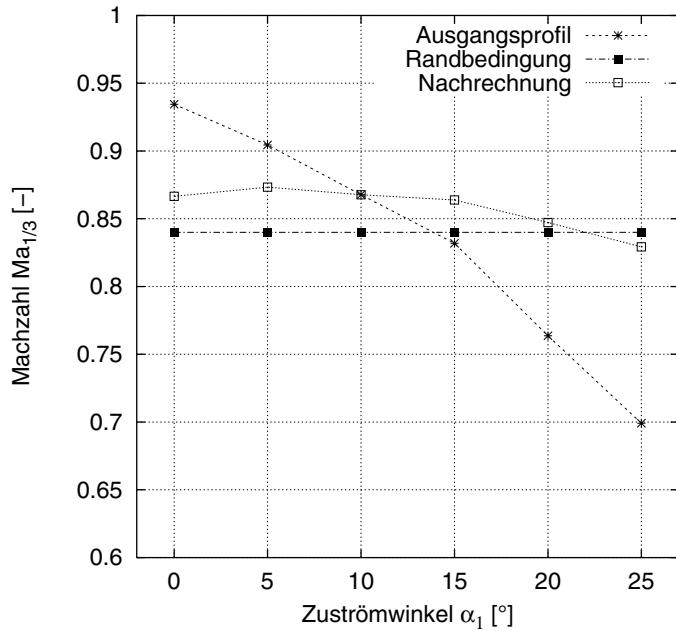


Abbildung 4.15: Das neuronale Netz hat die Fähigkeit erlernt, auf eine Veränderung des Zuströmwinkels  $\alpha_1$  entsprechend die Geometrie anzupassen.

Das neuronale Netz hat also durch das Training die Fähigkeit erworben, das Ausgangsprofil einer sich veränderten Randbedingung anzupassen.

Die gleiche Fähigkeit ist in der Abbildung 4.16 für die Geometrie des Profils graphisch aufbereitet. Anhand einer Variation des Zuströmwinkels für  $\alpha_1 \in \{0^\circ, 5^\circ, \dots, 25^\circ\}$  sind die vom neuronalen Netz produzierten Profilgeometrien dargestellt. Alle übrigen geometrischen und aerodynamischen Randbedingungen sind konstant gelassen. Deutlich ist zu erkennen, daß das neuronale Netz gelernt hat, wie die Vorderkante zu verändern ist, wenn der Anströmwinkel steiler wird. Der Metallwinkel an der Hinterkante bleibt im wesentlichen konstant.

In Abbildung 4.17 ist anhand des Dickenverhältnisses  $d_{\max}/l$  der Unterschied der Skalierungen nach Methode I und Methode II sichtbar. Die Werte für Methode II sind auf den minimalen  $a^* = 0.06978$  und maximalen  $b^* = 0.2001$  Wert skaliert, eine Extrapolation ist nur begrenzt möglich. Der Abströmwinkel  $\alpha_2$  für  $d_{\max}/l = 0.05$  ist wesentlich schlechter als für die nach Methode I skalierten Muster.

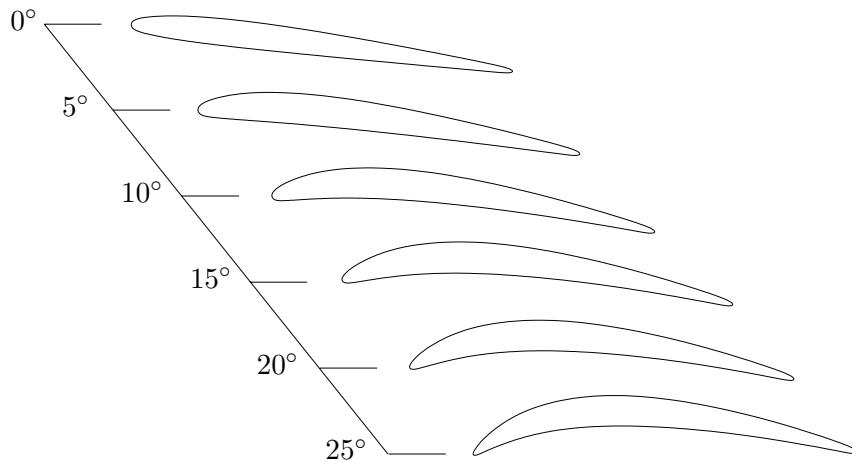


Abbildung 4.16: Eine Variation für die Testkonfiguration 2 zeigt die erlernte Fähigkeit des neuronalen Netzes, auf den veränderten Zuströmwinkel zu reagieren. Alle anderen Randbedingungen bleiben konstant.

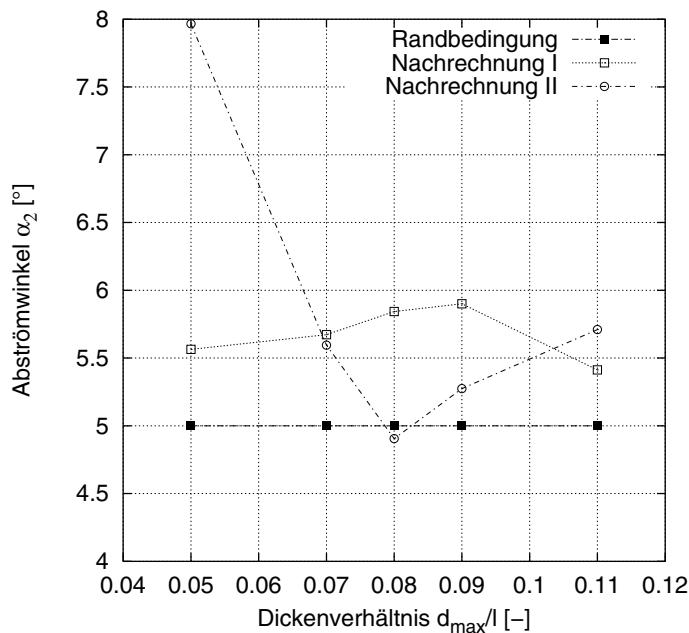


Abbildung 4.17: Eine Extrapolation für nach Methode II skalierte Muster zeigt ein wesentlich schlechteres Verhalten als für entsprechende Konfigurationen der Methode I.

## 4.5 Ausblick

Künstliche neuronale Netze werden in nahezu allen Anwendungsbereichen erfolgreich eingesetzt. Die Literatur kann auch nur übersichtsweise nicht mehr dargestellt werden. Es existieren eine Fülle von Netztopologien und Trainingsverfahren. Bei einer näheren Betrachtung stellt sich jedoch heraus, daß Erfahrungen und/oder aufwendige Tests notwendig sind, um optimale Parametereinstellungen zu erhalten (vgl. auch FREY UND SCHACH (1996)). Das in dieser Arbeit vorgestellte und verwendete Backpropagation–Verfahren mit den unterschiedlichen Varianten ist die am häufigsten verwendete Trainingsmethode für vorwärtsgerichtete Netze. Dieses ist nicht zuletzt darauf zurückzuführen, daß für die meisten Problemstellungen akzeptable Ergebnisse erzielt worden sind. Die Anzahl der Parameter und die Anforderungen an die Trainingsdaten sind gering.

Bei den hier beschriebenen Untersuchungen ist der benötigte Zeitaufwand zum Trainieren des Netzes als unkritisch betrachtet worden. Die Trainingszeiten auf einer SUN Ultra 2 betragen zwischen 10 Minuten und wenigen Stunden. Aufgrund der schnellen Entwicklung neuester Prozessoren dürfte dieses in absehbarer Zeit auf einen Bruchteil schrumpfen. Die Zeit, um einen Eingabevektor durch ein trainiertes Netz zu propagieren, kann vernachlässigt werden. Spezielle Neuro–Rechner sind für diese Aufgaben daher nicht (mehr) notwendig.

Durch den stetig steigenden Wettbewerbsdruck sind die Hersteller von Turbomaschinen zu immer kürzeren Entwicklungszeiten neuerer und besserer Produkte gezwungen. Hierdurch ist eine zunehmend parallele und vernetzte Vorgehensweise aller am Entwurf beteiligten Fachdisziplinen notwendig (*concurrent design*). Künstliche neuronale Netze können nun hierfür eingesetzt werden, um das Fachwissen der einzelnen Disziplinen zu approximieren und damit als Wissensbasis für die Gesamtheit der Entwickler zur Verfügung zu stellen. Ohne konkretes Fachwissen anderer Disziplinen zu benötigen oder aufwendige numerische Simulationen durchzuführen, steht den Entwicklungsabteilungen ein Werkzeug zur Verfügung, welches Auswirkungen veränderter Parameter unmittelbar aufzeigen und auf potentielle Schwachstellen hinweisen kann. Aufwendige Iterationszyklen wie bei einer seriellen Vorgehensweise werden dadurch vermieden. MARRA (1997) beschreibt hierzu ein Expertensystem bei Pratt & Whitney, dessen Ziel es ist, allen an der Entwicklung beteiligten Disziplinen eine Wissensbasis zur Verfügung zu stellen, so daß die Auswirkungen der Veränderung bestimmter Parameter sofort überprüft werden können.

Der in dieser Arbeit vorgestellte Ansatz ist eine Erweiterung der bestehenden Methoden, da sowohl Parameter zur Geometriedefinition von Profilen als auch aerodynamische Eigenschaften in *einem* neuronalen Netz approximiert werden. Das hier entwickelte Verfahren zum Profilentwurf kann als Grundlage für weitere Entwicklungen herangezogen werden. Durch eine breitere Basis der Trainingsdaten läßt sich der Gültigkeitsbereich erweitern und die Genauigkeit erhöhen.

In dieser Arbeit wird das trainierte neuronale Netz nicht als Bestandteil in einem übergeordneten System eingesetzt, sondern zur Bestimmung der Startkonfiguration einer, im nächsten Abschnitt beschriebenen, Optimierungsstrategie verwendet.



# Kapitel 5

# Optimierung

Im letzten Teil dieser Arbeit wird ein Verfahren vorgestellt, welches versucht, Profile zu optimieren, indem eine Zielfunktion minimiert wird. Hierzu werden geometrische und aerodynamische Beurteilungskriterien herangezogen. Zur Lösung der Optimierungsaufgabe werden Mutations–Selektions– sowie genetische Algorithmen verwendet, die eine wesentliche Gruppe der Methoden der *Computational Intelligence* darstellen.

## 5.1 Optimierungsverfahren

Unter einer Optimierungsaufgabe wird im folgenden ein Verfahren zur Minimierung einer Zielfunktion (Qualitätsfunktion)  $f$  verstanden, d.h.

$$\text{Minimiere } f(\vec{x}) \quad \text{mit} \quad f : \mathbf{M}_0 \times \mathbf{M}_1 \times \dots \times \mathbf{M}_{n-1} \rightarrow \mathbf{M}_n \quad , \quad (5.1)$$

wobei meistens für die Definitions– bzw. Wertebereiche  $\mathbf{M}_i = \mathbb{R}$  bzw.  $\mathbf{M}_i = \mathbb{N}$  angenommen wird. Zusätzlich sind  $m$  Restriktionen zu berücksichtigen, die sich in  $q$  Ungleichheits– und  $m - q$  Gleichheitsrestriktionen aufteilen lassen:

$$g_i(\vec{x}) \leq 0 \quad \text{für } i = 0, \dots, q-1 \quad \text{und} \quad h_i(\vec{x}) = 0 \quad \text{für } i = 0, \dots, m-q-1 \quad . \quad (5.2)$$

Zur Behandlung dieser Nebenbedingungen wird hier der Ansatz über eine Straffunktion gewählt. Das allgemeine Optimierungsproblem aus Gleichung (5.1) ergibt sich dann zu:

$$\text{Minimiere } f(\vec{x}) + \sum_{i=0}^q \max\{0, g_i(\vec{x})\} + \sum_{i=0}^{m-q-1} |h_i(\vec{x})| \quad . \quad (5.3)$$

Zur Lösung von Optimierungsproblemen bieten sich drei Klassen von Verfahren an. Die exakten, analytischen Verfahren sind i.allg. sehr rechenintensiv und nur für kleinste Testfunktionen praktikabel. Im Gegensatz dazu basieren heuristische Verfahren auf intuitiven Ansätzen und Erfahrungen. Für diese Arbeit wird auf Methoden der dritten Klasse, der numerischen Näherungsverfahren, zurückgegriffen. Diese lassen sich in deterministische und

```

algorithm MUTATION-SELECTION
begin
0   Wähle eine zufällige Startkonfiguration  $\vec{x}$ 
1   do ( $\neg$  terminate) —→
2       mutiere die Konfiguration  $\vec{x}$  zu  $\vec{x}'$ 
3       if ( $f(\vec{x}') < f(\vec{x})$ ) —→
4           selektiere  $\vec{x}'$  als neue Konfiguration  $\vec{x}$ 
5       fi
6   od
end.

```

Abbildung 5.1: Allgemeine Formulierung des Mutations–Selektions–Algorithmus zur Minimierung einer Zielfunktion  $f$

stochastische Verfahren unterteilen. Häufige Vertreter deterministischer Verfahren sind die Gradientenverfahren (vgl. SPELLUCCI (1993)). In dieser Arbeit werden insbesondere die stochastischen Verfahren untersucht (vgl. HAJELA (1997, 1999)), die sich an der natürlichen Evolution orientieren. Dieses auf Versuch und Irrtum basierende Prinzip hat sich bei der Entwicklung der belebten Natur als eine äußerst flexible und erfolgreiche Optimierungsstrategie erwiesen. Der Parametervektor  $\vec{x}$  wird im folgenden auch als Konfiguration bzw. Chromosom bezeichnet.

## 5.2 Mutations–Selektions–Algorithmen

Zu den einfachsten evolutionären Verfahren der stochastischen Optimierung gehört der sogenannte Mutations–Selektions–Algorithmus. Dieser lässt sich, wie in Abbildung 5.1 skizziert, so beschreiben: eine zufällig gewählte Konfiguration des zu minimierenden Systems wird so lange verändert (mutiert), bis keine Verbesserung der Zielfunktion  $f$  mehr möglich ist. Da durch den oben angegebenen Algorithmus die Gefahr besteht, in lokale Minima zu laufen, sind eine Reihe von verbesserten Varianten entwickelt und angewendet worden. Trotz ihrer Einfachheit haben sich die Mutations–Selektions–Algorithmen bei der Lösung von Optimierungsproblemen in allen Bereichen bewährt (siehe auch ABLAY (1987) und KINNEBROCK (1994)).

### 5.2.1 Der Mutationsoperator

Da der Mutationsoperator die treibende Kraft im Optimierungsablauf ist, soll zuerst auf diesen näher eingegangen werden<sup>1</sup>. Eine Möglichkeit, um an einer Konfiguration  $x \in [x_{\min}, x_{\max}]$  eine Mutation durchzuführen, ist es,  $x$  in  $x' = x + r * \delta$  zu verändern. Hierbei

---

<sup>1</sup>Zur grundlegenden Diskussion des Mutationsoperators reicht es an dieser Stelle aus, eindimensionale Probleme zu betrachten. Die Vorgehensweise lässt sich anschließend ohne weiteres auf den mehrdimensionalen Fall übertragen.

kann zum Beispiel  $r \in [-1, 1]$  eine gleichverteilte Zufallszahl sein (uniforme Mutation).  $\delta$  gibt die Variationsbreite an. Durch die Gleichverteilung der Zufallszahl  $r$  ergibt sich, daß in einer auszuführenden Mutation die Wahrscheinlichkeit einer großen Veränderung der Konfiguration gleich der Wahrscheinlichkeit einer Mutation mit einer geringeren Auswirkung ist. Dieses entspricht jedoch nicht dem Vorgehen in der Realität. Kleine Veränderungen sind häufig anzutreffen, schwerwiegende Mutationen hingegen nur sehr selten. Um dieses Verhalten bei der Mutation im Modell nachzubilden, bietet es sich an, eine *natürliche* Verteilung der Mutation zugrunde zu legen.

Eine wichtige Funktion in der Statistik und Stochastik nimmt die Normal– oder Gaußverteilung ein<sup>2</sup>. Eine Zufallsgröße, die die Summe von mehreren, voneinander unabhängigen Zufallsgrößen ist, läßt sich durch eine Normalverteilung approximieren.

So wird zum Beispiel der Durchmesser einer Schraube in der Produktion durch eine Vielzahl von Parametern (Werkstoffcharakteristika, Temperaturschwankungen, Verschleiß von Werkzeugen, ...) gestreut. Der Durchmesser der Schrauben läßt sich bei einer genügend großen Stichprobe als normalverteilt beschreiben. Mutationen in der Natur zeigen ebenfalls eine normalverteilte Variation auf. Daher ist es nur naheliegend, dieses algorithmisch zu berücksichtigen. Die Dichte der Gaußverteilung  $\mathcal{N}(\mu, \sigma)$  mit den beiden Parametern  $\mu$  (Erwartungswert) und  $\sigma$  (Standardabweichung) ergibt sich zu:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \quad (5.4)$$

Für den Gauß–Mutationsoperator folgt dann:

$$x' = x + \delta(\sigma) \quad , \text{ wobei } \delta \text{ mit } \mathcal{N}(0, \sigma) \text{ normalverteilt ist.} \quad (5.5)$$

Zur sinnvollen Bestimmung der Standardabweichung kann auf eine Eigenschaft der Gaußverteilung zurückgegriffen werden. Normalverteilte Zufallsgrößen nehmen in 68,26% aller Fälle Werte aus  $[\mu - \sigma, \mu + \sigma]$  an. In sogar 99,75% der Fälle werden Werte des Intervalls  $[\mu - 3\sigma, \mu + 3\sigma]$  angenommen (Drei–Sigma–Regel). Des weiteren bietet es sich an, mit fortlaufender Iterationsanzahl die Standardabweichung  $\sigma$  zu reduzieren.

Zur algorithmischen Berechnung von normalverteilten Zufallszahlen kann die sogenannte *Box–Muller–Methode* (auch Polarmethode) verwendet werden (vgl. BRATLEY ET AL. (1983)). Sind  $u_0$  und  $u_1$  zwei auf  $[0, 1]$  gleichverteilte Zufallszahlen, dann sind  $z_0$  und  $z_1$  zwei  $\mathcal{N}(0, 1)$ –verteilte Zufallszahlen, unter der Annahme, daß gilt:

$$z_0 = \cos(2\pi u_0) \sqrt{-2 \log(u_1)} \quad \text{und} \quad z_1 = \sin(2\pi u_0) \sqrt{-2 \log(u_1)} \quad . \quad (5.6)$$

Mit Hilfe der trivialen Transformation  $x_0 = \mu + z_0 \sigma$  bzw.  $x_1 = \mu + z_1 \sigma$  lassen aus zwei  $\mathcal{N}(0, 1)$ –verteilten Zufallszahlen sich die  $\mathcal{N}(\mu, \sigma)$ –verteilten Zufallsvariablen  $x_0$  und  $x_1$  erzeugen.

---

<sup>2</sup>Es wird unterstellt, daß sich die Konfiguration als Vektor reeller Zahlen darstellen läßt. Oftmals ist es allerdings erforderlich, einen Mutationsoperator für ganzzahlige Werte zu haben. Um einen ähnlichen Effekt wie bei Mutationen mittels der Gaußverteilung zu erhalten, bietet es sich an, auf die diskrete Binomialverteilung zurückzugreifen.

---

```

algorithm SIMULATED ANNEALING
begin
0   Wähle eine zufällige Startkonfiguration  $\vec{x}$ 
1   Wähle eine Starttemperatur  $T_{\text{start}}$  und Endtemperatur  $T_{\text{end}}$ 
2    $T \leftarrow T_{\text{start}}$ 
3   do ( $T > T_{\text{end}}$ ) —>
4     do ( $\neg$  Gleichgewichtszustand) —>
5       mutiere die Konfiguration  $\vec{x}$  zu  $\vec{x}'$ 
6       if ( $f(\vec{x}') < f(\vec{x})$ )  $\vee$  mit einer Wahrscheinlichkeit  $p$ ) —>
7          $\vec{x} \leftarrow \vec{x}'$ 
8     fi
9   od
10  Reduziere die Temperatur  $T$ 
11 od
end.

```

Abbildung 5.2: Allgemeine Form des *Simulated Annealing*–Algorithmus zur Minimierung einer Zielfunktion  $f$ .

### 5.2.2 Simulated Annealing

Die *Simulated Annealing*–Methode entspringt aus dem Bereich der Festkörperphysik und simuliert den Abkühlvorgang von Kristallen (vgl. AARTS UND KORST (1989) und KINNEBROCK (1994)). Um möglichst optimale, d.h. mit geringen Verunreinigungen durchsetzte, Kristallstrukturen zu erhalten, ist es notwendig, die erwärmte Materie in einem bestimmten Zeitplan langsam abzukühlen, um somit das absolute Energieminimum des Kristalls zu erhalten. 1980 wurde dieses Verfahren vom Physiker Scott Kirkpatrick auf Optimierungsprobleme übertragen.

Hierbei (siehe Abbildung 5.2) handelt es sich um eine Variante des bereits vorgestellten Mutations–Selektions–Algorithmus. Der Grundgedanke ist es, nicht nur bezüglich der Fitneß, d.i. der Zielfunktionswert, immer nur bessere Konfigurationen zu akzeptieren, sondern mit einer gewissen Wahrscheinlichkeit auch Verschlechterungen. Im Laufe des Optimierungsprozesses wird nun die Wahrscheinlichkeit so verändert, daß es immer unwahrscheinlicher wird, schlechte Konfigurationen zu akzeptieren, in der Hoffnung am Ende das globale Optimum zu erhalten. Dadurch, daß auch Verschlechterungen selektiert werden, soll vermieden werden, in lokalen Extrema zu enden. Die Wahrscheinlichkeit  $p$ , daß eine schlechtere Konfiguration akzeptiert wird, ist abhängig vom Parameter  $T$ , der in Anlehnung an das physikalische Vorbild des Algorithmus im weiteren als (dimensionslose) Temperatur bezeichnet wird.

**Akzeptanzwahrscheinlichkeit und Starttemperatur** Entscheidenden Einfluß auf das Konvergenzverhalten hat die Wahl der sogenannten Akzeptanzwahrscheinlichkeit, al-

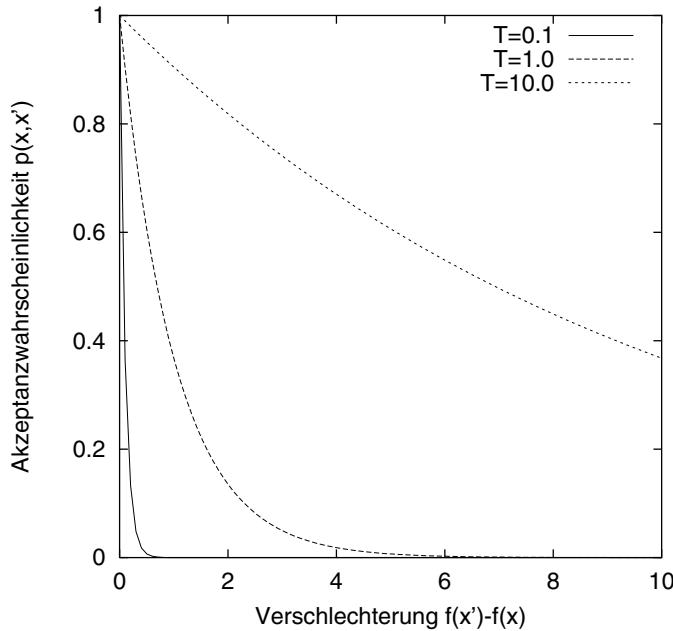


Abbildung 5.3: Für eine Verschlechterung  $f(x') - f(x) = 2$  mit  $T = 10$  ist die Akzeptanz  $p(x, x') \approx 0.82$ , mit  $T = 1$  gilt nur noch  $p(x, x') \approx 0.14$  und für  $T = 0.1$  ist  $p(x, x') \approx 10^{-9}$  vernachlässigbar gering.

so der Wahrscheinlichkeit, mit der eine Konfiguration trotz einer Verschlechterung nicht verworfen wird. Bei dem einfachen Mutations–Selektions–Algorithmus in Abbildung 5.1 ist diese null.

Die sogenannte *Metropolis*–Akzeptanzwahrscheinlichkeit ist durch

$$p(x, x') = \begin{cases} 1 & \text{falls } f(x') < f(x) \\ e^{-(f(x') - f(x))/T} & \text{falls } f(x') \geq f(x) \end{cases} \quad (5.7)$$

gegeben, wobei  $T > 0$  für die Temperatur gilt.

In Abbildung 5.3 ist die Wahrscheinlichkeitsverteilung für unterschiedliche Temperaturen aufgetragen. Erkennbar ist, daß für  $T = \text{const}$  große Verschlechterungen mit zunehmend geringer werdenden Wahrscheinlichkeiten akzeptiert werden. Auf der anderen Seite wird deutlich, daß für  $f(x') - f(x) = \text{const}$  die Temperatur sehr entscheidend dafür ist, ob eine neue Konfiguration  $x'$  akzeptiert oder verworfen wird.

Die Wahl der Temperatur ist somit der entscheidende Faktor für die Güte des Optimierungsprozesses. Ist die Temperatur zu gering, besteht die Gefahr in lokalen Extrema zu enden. Ist die Temperatur zu groß, werden zu viele schlechte Konfigurationen akzeptiert. Ein zielgerichtetes Suchen ist nicht mehr gegeben („Stochern nach einer Nadel im Heuhaufen“). Daher ist es umso wichtiger, eine sinnvolle Starttemperatur  $T_{\text{start}}$  zu finden. Die folgende einfache Überlegung kann hierbei hilfreich sein. Anstatt eine absolute Temperatur

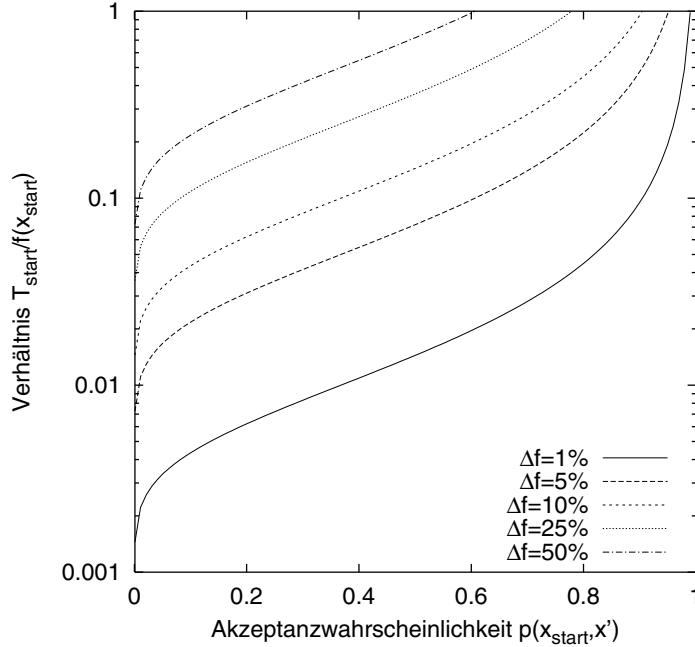


Abbildung 5.4: Zur Bestimmung der Starttemperatur wird eine Abweichung  $\Delta f$  und eine Wahrscheinlichkeit  $p$  vorgegeben. Ist die Startfitneß berechnet, ergibt sich dann anschließend die Temperatur  $T_{\text{start}}$ .

zu Beginn festzulegen, ist es möglich, einer prozentualen Abweichung von der Startfitneß  $f(x_{\text{start}})$  eine Akzeptanzwahrscheinlichkeit  $p(x_{\text{start}}, x')$  zuzuordnen. Aus Gleichung (5.7) ergibt sich die Beziehung

$$\frac{T_{\text{start}}}{f(x_{\text{start}})} = \frac{1}{\ln p(x_{\text{start}}, x')} \cdot \left[ -\left( \frac{f(x')}{f(x_{\text{start}})} - 1 \right) \right]. \quad (5.8)$$

Der Term  $\Delta f = f(x')/f(x_{\text{start}}) - 1$  ist die prozentuale Abweichung der Konfiguration  $x'$  zur Startkonfiguration  $x_{\text{start}}$ . In Abbildung 5.4 ist für verschiedene Akzeptanzwahrscheinlichkeiten und prozentuale Abweichungen der Quotient  $T_{\text{start}}/f(x_{\text{start}})$  aufgetragen. Soll zum Beispiel eine 10% Abweichung von der Startkonfiguration mit einer Wahrscheinlichkeit  $p = 0.5$  akzeptiert werden, ist der Quotient  $\approx 0.14$ . In einer Implementierung des Simulated-Annealing Algorithmus ist es daher angebracht, den Quotienten  $T_{\text{start}}/f(x_{\text{start}})$  anstatt einer absoluten Starttemperatur vorzugeben, da  $f(x_{\text{start}})$  bereits bekannt ist.

**Abkühlverhalten** Nachdem die Akzeptanzwahrscheinlichkeit und die Starttemperatur bestimmt worden sind, bedarf es einer Betrachtung der anderen Parameter. Insbesondere ist der im Algorithmus nicht näher bezeichnete Gleichgewichtszustand zu definieren.

Damit der Optimierungsprozeß das globale Optimum findet, wird die Temperatur  $T$  und damit auch die Akzeptanzwahrscheinlichkeit verringert, bis diese einen Minimalwert  $T_{\text{end}}$

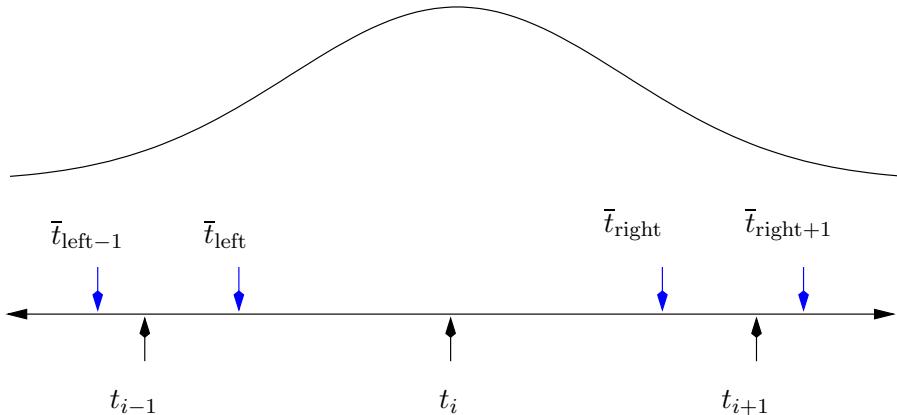


Abbildung 5.5: Der Gauß–Mutationsoperator verändert den Knoten  $t_i$  innerhalb des Intervalls  $[\bar{t}_{\text{left}}, \bar{t}_{\text{right}}]$ . Hierdurch wird gewährleistet, daß sich mindestens je ein Parameterwert  $\bar{t}_j$  im Intervall  $[t_{i-1}, t_i]$  bzw. im Intervall  $[t_i, t_{i+1}]$  befindet und damit das in der Approximation zu lösende Gleichungssystem positiv definit wird.

annimmt (üblicherweise gilt  $T_{\text{end}} = 0$ ). Der Zeitpunkt der Verringerung wird in Abhängigkeit eines Gleichgewichtszustandes bestimmt. Der *Simulated Annealing*–Prozeß befindet sich im Gleichgewicht, wenn nach einer bestimmten Anzahl von Mutationen (im folgenden mit  $N_{\text{eq}}$  bezeichnet) keine weitere Verbesserung eingetreten ist. In einer Implementierung bietet es sich an, einen Zähler einzufügen, der die Anzahl der verworfenen Konfigurationen seit dem letzten Gleichgewichtszustand summiert.

Ein häufige Methode die Temperatur zu verringern ist es, die Temperatur mit einem Abkühlfaktor  $c_f$  mit  $T \leftarrow c_f \cdot T$  zu gewichten (vgl. Zeile 10 in Abbildung 5.2). Als Richtwert ist  $c_f \in [0.5, 0.8]$  zu wählen.

### 5.2.3 Beispiel: Geometrieoptimierung

Im Rahmen dieser Arbeit wird das beschriebene *Simulated Annealing*–Optimierungsverfahren bei zwei geometrischen Aufgabenstellungen verwendet: zum einen bei der Approximation von zweidimensionalen Profilschnitten, zum anderen bei der Oberflächenoptimierung dreidimensionaler Schaufeln. An dieser Stelle werden einige Ergebnisse von UELSCHEN ET AL. (1998) vorgestellt.

#### Approximation mit B–Spline–Funktionen

Für eine aerodynamische Optimierung der Profilgeometrie ist es zweckmäßig, eine möglichst geringe Anzahl von Freiheitsgraden zu haben. Dazu wird das Approximationsverfahren aus Abschnitt 2.2.3 durch ein zusätzliches Optimierungsverfahren erweitert. Es zeigt sich, daß die Wahl des Knotenvektors  $\mathbf{t}$  (vgl. Gleichung (2.23)) einen entscheidenden Einfluß auf die Approximationsgüte hat. Daher liegt es nahe, das Approximationsverfahren durch einen *Simulated Annealing*–Optimierungsansatz, wobei die Knoten  $t_{p+1}, \dots, t_{m-p-1}$  die freien

---

```

algorithm LEAST SQUARE FIT SIMULATED ANNEALING
begin
  0 Erzeuge zentripetale Parameterisierung  $\bar{\mathbf{t}} = \{\bar{t}_0, \bar{t}_1, \dots, \bar{t}_r\}$ 
  1 Wähle eine Startkonfiguration  $\mathbf{t} = \{t_0, t_1, \dots, t_m\}$ 
  2 Wähle eine Starttemperatur  $T_{\text{start}}$  und Endtemperatur  $T_{\text{end}}$ 
  3  $T \leftarrow T_{\text{start}}$ 
  4 do ( $T > T_{\text{end}}$ ) —>
      5   do ( $\neg$  Gleichgewichtszustand) —>
          6      $i \leftarrow \mathcal{U}[p + 1, m - p - 1]$ 
          7     Bestimme  $\bar{t}_{\text{left}} \in [t_{i-1}, t_i]$  mit  $\bar{t}_{\text{left}-1} < t_{i-1}$ 
          8     Bestimme  $\bar{t}_{\text{right}} \in [t_i, t_{i+1}]$  mit  $\bar{t}_{\text{right}+1} > t_{i+1}$ 
          9      $t'_i \leftarrow \text{Gauß-Mutation}(t_i, \sigma, \bar{t}_{\text{left}}, \bar{t}_{\text{right}})$ 
         10     $\mathbf{t}' \leftarrow \{t_0, t_1, \dots, t_{i-1}, t'_i, t_{i+1}, \dots, t_m\}$ 
         11    if ( $\text{LSF}(\mathbf{Q}, n, p, \bar{\mathbf{t}}, \mathbf{t}') < \text{LSF}(\mathbf{Q}, n, p, \bar{\mathbf{t}}, \mathbf{t}) \vee$  mit Wahrscheinlichkeit  $p$ ) —>
         12       $\mathbf{t} \leftarrow \mathbf{t}'$ 
         13    fi
         14  od
         15  Reduziere die Temperatur  $T$ 
  16 od
end.

```

Abbildung 5.6: Anpassung des allgemeinen *Simulated Annealing*-Algorithmus (vgl. Abb. 5.2) zur Optimierung des Knotenvektors  $\mathbf{t}$  zur Approximation nach der Methode der kleinsten Fehlerquadrate.

Parameter sind, zu ergänzen. Bei der Mutation eines Knotens sind folgende Bedingungen einzuhalten:

1. die einzelnen  $t_i$  müssen monoton steigend sein,
2. die Periodizitätsbedingungen an den beiden Enden müssen gelten und
3. in jedem Knotenintervall  $[t_i, t_{i+1}]$  muß sich mindestens ein Parameterwert  $\bar{t}_j$  befinden.

Die ersten beiden Bedingungen folgen aus der Definition einer B-Spline Kurve (vgl. Definition 4 und 5, Seite 11ff.). Um keine Ecken in der Approximation zu erhalten, wird zusätzlich  $t_i < t_{i+1}$  verlangt, d.h. die Multiplizität eines Knotens ist eins. Hingegen ist die dritte Bedingung zur Lösung des linearen Gleichungssystems notwendig.

Abbildung 5.5 zeigt den verwendeten Gauß-Mutationsoperator. Die Anpassung auf das hier beschriebene Approximation-Optimierungsverfahren des allgemeinen *Simulated Annealing*-Prozesses ist in Abbildung 5.6 wiedergegeben.

Zu Beginn werden die  $r + 1$  Parameterwerte entsprechend einer zentripetalen Verteilung nach Gleichung (2.21) parametrisiert (Zeile 0). Anschließend sind eine Startkonfiguration für den Knotenvektor  $\mathbf{t}$  sowie die Temperaturen  $T_{\text{start}}$  und  $T_{\text{end}}$  zu wählen. Zur Be-

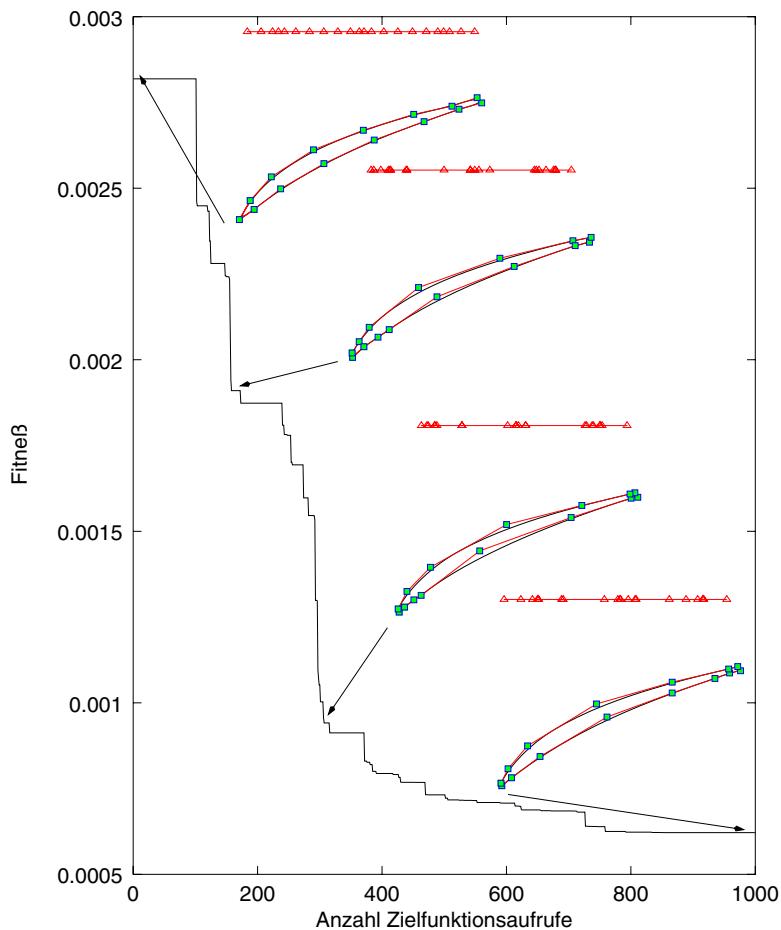


Abbildung 5.7: Für den *Simulated Annealing*–Prozeß sind die folgenden Parameter verwendet worden:  $N_{eq} = 70$ ,  $c_f = 0.5$ ,  $T_{start}/f(x_{start}) = 0.1$ . Zusätzlich zum Konvergenzverhalten ist für 4 Iterationspunkte der jeweilige B–Spline mit dem Knotenvektor dargestellt. Die Fitneß ist durch den maximalen Approximationsfehler  $E_{\max}^*$  definiert.

rechnung der Knoten kann die Gleichung (2.23) herangezogen werden. Im nun folgenden Abkühlungsprozeß ist entsprechend einer Gleichverteilung zuerst ein Index  $i$  zu bestimmen (Zeile 6). Darauffolgend wird in den Zeilen 7 und 8 die Spannweite  $[\bar{t}_{left}, \bar{t}_{right}]$  berechnet, innerhalb derer der Knoten  $t_i$  verschoben werden darf. Der Gauß–Mutationsoperator verändert nun mit Mittelwert  $\mu = t_i$  und Standardabweichung  $\sigma$  den Knoten  $t_i$  entsprechend einer Gaußverteilung. Hierbei ist  $\sigma$  im Laufe der Iterationen zu verringern. Gegebenenfalls ist noch zusätzlich die periodische Ergänzung am Anfang bzw. Ende des Knotenvektors zu aktualisieren (nicht mit aufgeführt). Für einen so mutierten Knotenvektor  $\mathbf{t}'$  wird die Approximationsaufgabe gelöst und in Abhängigkeit einer Verbesserung bzw. mit einer Wahrscheinlichkeit  $p$  akzeptiert (Zeile 11). Die Routine **LSF** ermittelt den maximalen Fehler bei einer Approximation mit den angegebenen Parametern.

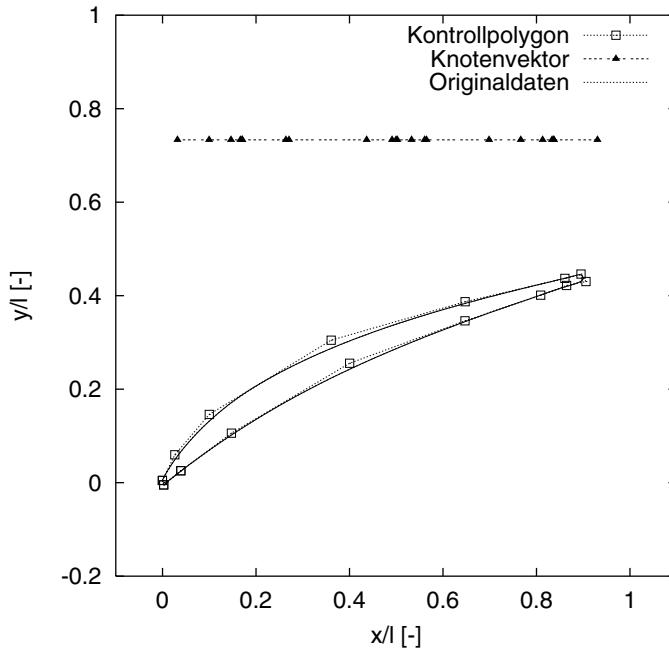


Abbildung 5.8: Die Parameter für den B–Spline sind:  $p = 3, n = 17, m = 21$  (15 nicht periodische Kontrollpunkte). Das Endergebnis der Approximation hat einen mittleren Fehler  $\bar{E}^* = 1.89 \cdot 10^{-4}$  und einen maximalen Fehler  $E_{\max}^* = 6.22 \cdot 10^{-4}$ .

In den Abbildungen 5.7 bis 5.9 ist das Ergebnis der Approximation eines Profils aus MERTENS (1995) dargestellt. Weitere Ergebnisse sind dem Anhang A.5 zu entnehmen.

### 3D–Skinning

In Abschnitt 2.3.2 ist auf die dreidimensionale Geometriedefinition mittels einer Tensorprodukt–B–Spline–Fläche eingegangen worden. Um aerodynamisch günstige Schaufeln zu erhalten, reicht es i.allg. aber nicht aus, eine Oberfläche nur aus den übereinander geschichteten Profilen innerhalb des *Skinning*–Prozesses zu berechnen. In Abbildung 2.9 ist bereits auf das Indizierungsproblem, welches bei einer ungünstigen Numerierung der Kontrollpunkte entstehen kann, hingewiesen worden.

Daher ist es unerlässlich, ein Glattheitsmaß (*Strakmaß*) zu definieren, welches die Qualität der dreidimensionalen Schaufeloberfläche beschreibt. Es bietet sich an, auf die in der Flächentheorie definierten Begriffe der Flächenhauptkrümmungen  $\kappa_1$  und  $\kappa_2$  zurückzugreifen. Insbesondere das Produkt der Hauptkrümmungen (Gaußsche Krümmung) und das arithmetische Mittel (mittlere Krümmung)

$$K = \kappa_1 \kappa_2 \quad \text{und} \quad H = \frac{1}{2}(\kappa_1 + \kappa_2) \quad (5.9)$$

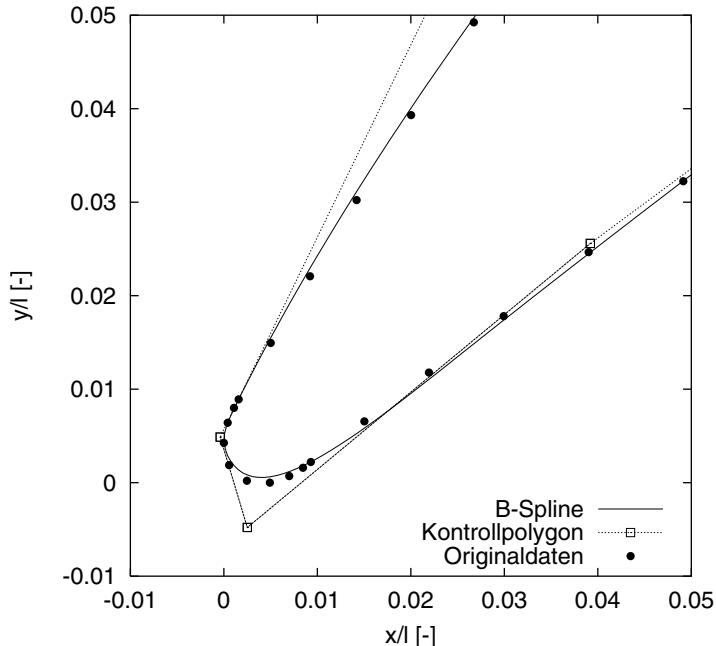


Abbildung 5.9: Eine Vergrößerung der Vorderkante des approximierten Profils zeigt die maximalen Fehler. Durch eine Verdichtung der Punkte kann der Approximationsfehler reduziert werden.

sind zur Beschreibung der Oberflächeneigenschaften verwendbar. Einzelheiten zur Berechnung von Flächenkrümmungen sind zum Beispiel in TASCHNER (1977) und SCHÖNE (1978) zu finden. Sinnvoll erscheint es, das *Skinning*–Verfahren mit einem Optimierungsansatz zu koppeln, in dem versucht wird, die Krümmung der Schaufeloberfläche zu minimieren. Als freie Parameter in der Optimierung steht zum einen die Numerierung der Kontrollpunkte  $\{\mathbf{P}_{ij}\}$  (Permutation) und zum anderen der Knotenvektor  $\mathbf{v}$  in Spannweitenrichtung zur Verfügung.

Im Rahmen dieser Arbeit ist ein *Simulated Annealing*–Optimierungsverfahren zur Minimierung der Gesamtkrümmung der dreidimensionalen Schaufel entwickelt worden. Das bereits in Abschnitt 2.4 beschriebene Programm `mustacker` wurde hierzu durch das beschriebene *Skinning*–Optimierungsverfahren erweitert. Es zeigt sich, daß für einfache Geometrien befriedigende Ergebnisse mit dem beschriebenen Ansatz erreicht werden. Bei komplexeren und realitätsnäheren Schaufeln versagt das Verfahren. Insbesondere bei einer größeren Anzahl ( $> 4$ ) von Ebenen liefert die Optimierung nur lokale Minima (vgl. hierzu UELSCHEN ET AL. (1998)).

Überlegenswert ist es, das beschriebene Verfahren zur zweidimensionalen Approximation von punktförmig gegebenen Zusammenhängen auf den dreidimensionalen Fall zu erweitern. Hierzu sind in jeder Ebene eine Folge von Konturpunkten zu berechnen, die dann in der radialen Richtung durch eine Tensorprodukt–B–Spline–Fläche zu approximieren ist. In

der Zielfunktion des Optimierungsprozesses ist zum einen der Minimierung des Approximationfehlers und zum anderen der Minimierung der Oberflächenkrümmung Rechnung zu tragen. Durch Nebenbedingungen ist es so dann möglich, weitere konstruktive Restriktionen, zum Beispiel Kühlkanäle, in den *Skinning*-Prozeß einfließen zu lassen.

#### 5.2.4 Weitere Ansätze

INGBER (1993) beschreibt eine adaptive Variante des Simulated Annealing Verfahrens, welches sich insbesondere für mehrdimensionale Optimierungsprobleme eignet. Im Rahmen dieser Arbeit durchgeführte Untersuchungen an Testfunktionen zeigten keine Verbesserungen gegenüber dem Standardverfahren.

DUECK UND SCHEUER (1990) stellen eine weitere Variante *Threshold Accepting* des oben eingeführten Mutations–Selektions–Algorithmus vor, der im Gegensatz zum Simulated Annealing Verfahren auf das stochastische Element bei der Akzeptanzentscheidung verzichtet und alle Konfigurationen, die einen Schwellenwert nicht überschreiten, akzeptiert. Der Sintflut–Algorithmus *The Great Deluge* ist vergleichbar mit dem Durchschreiten einer Berglandschaft, in der es ohne Unterbrechung regnet. Es wird von einer Bergspitze zur nächsten gesprungen ohne dabei durch bereits überflutete Täler zu gehen. Mit steigendem Wasserpegel bleiben nur die höchsten Berge (Optima) übrig. Ein weiterer Mutations–Selektions–Algorithmus ist das sogenannte *Record-to-Record-Travel*–Verfahren. Hierbei werden neue Konfigurationen genau dann akzeptiert, wenn sich diese vom bisherigen Spitzenwert (*record*) höchstens um einen bestimmten Wert unterscheiden. Dieser Spitzenwert wird jedesmal aktualisiert, wenn eine neuer, besserer Spitzenwert gefunden worden ist. Nähere Einzelheiten sind in DUECK (1993) bzw. in KINNEBROCK (1994) zu finden.

In KUPPE (1997) bzw. SPEER ET AL. (1998) werden verschiedene deterministische Approximationsverfahren untersucht. BAIER (1998) verwendet eine Evolutionsstrategie zur Minimerung des Approximationfehlers.

### 5.3 Genetische Algorithmen

Evolutionäre Algorithmen dienen als Strategie zur Lösung von Optimierungsproblemen, indem Prinzipien der natürlichen, biologischen Evolution abstrahiert und im Rechner modelliert werden.

Im wesentlichen haben sich zwei Modelle der Evolution in den letzten 30 Jahren herauskristallisiert. In Deutschland wurde von RECHENBERG (1994) und SCHWEFEL (1995) die sogenannte Evolutionsstrategie seit den sechziger Jahren entwickelt. Zu Beginn wurde die Evolution sehr vereinfachend abgebildet und entspricht den bereits vorgestellten Mutations–Selektions–Verfahren (siehe auch ABLAY (1987)). Anwendungsgebiet war hierbei von Anfang an die numerische Optimierung.

Parallel und unabhängig voneinander<sup>3</sup> fanden in den USA die genetischen Algorithmen

---

<sup>3</sup>Erstaunlicherweise kommt es erst seit 1990 zu einem Austausch der Forschungsgruppen diesseits und jenseits des Atlantiks.

von HOLLAND (1975) und GOLDBERG (1989) breite Anwendung. Im Unterschied zur Evolutionsstrategie wurden zu Beginn Operationen auf Bitfeldern eingesetzt. Die Erweiterung auf reellwertige Felder, wie sie i.allg. bei der numerischen Optimierung zu finden sind, wurde erst später angestellt.

Weitere evolutionäre Algorithmen sind die evolutionäre Programmierung und die genetische Programmierung, die aber an dieser Stelle nicht weiter vorgestellt werden. Eine einführende Darstellung der verschiedenen Verfahren ist u.a. in BLÜMECKE (1991), DAS-GUPTA UND MICHALEWICZ (1997a) oder DE FALCO (1997a) zu finden.

### 5.3.1 Einführung

Noch stärker als im Bereich der künstlichen neuronalen Netze wird sich im Anwendungsbereich der genetischen Algorithmen der Terminologie der Biologie zur Beschreibung der Verfahren und Methoden bedient. Bevor auf die berechnete Evolution eingegangen wird, werden daher die Elemente des biologischen Vorbildes erläutert.

#### Das biologische Vorbild

Die Evolution beschreibt den zeitlichen Prozeß der Entstehung, Entwicklung und Verbreitung von Lebewesen. Ein wesentliches Prinzip der Evolution ist es, daß mehr Nachkommen produziert werden, als zur Erhaltung der Art notwendig sind. Somit kommt es zu einem ständigen Wettbewerb der Individuen um die zur Verfügung stehenden Ressourcen (z.B. Nahrung oder Lebensraum). Das von Darwin Mitte des 19. Jahrhunderts entdeckte Prinzip des *survival of the fittest* beschreibt ein natürliches Selektionsverfahren, welches postuliert, daß sich die Individuen einer Art durchsetzen, die sich den Gegebenheiten der Umwelt am besten angepaßt haben.

Die Evolution bedient sich der Populationsgenetik als Grundlage zur Beschreibung der Phänomene. Wesentliche Voraussetzung für den Darwinschen Ausleseprozeß ist die Tatsache, daß die Erbinformationen, das sogenannte Genom, bei den Individuen einer Art nicht identisch, sondern über eine Bandbreite gestreut sind. Die Gene sind einzelne funktionale Einheiten des Genoms, die auf den Chromosomen linear angeordnet sind. Während der Reifeteilung zweier Zellen eines aus Vater und Mutter bestehenden Elternpaares kommt es zur Aufspaltung der Chromosomen in zwei Chromatiden und zur Überkreuzung (*crossover*) je eines väterlichen und mütterlichen Chromatidenstranges. An bestimmten Stellen brechen die beiden Chromatiden auf, genetisches Material wird ausgetauscht und die Bruchstellen schließen sich wieder. Die nachkommende Generation enthält dadurch Erbgut vom Vater und der Mutter. Diese Rekombination von Erbgut ist ein zentraler Vorgang der Evolution.

Im Gegensatz dazu ist die Mutation ein spontaner und wenig häufig auftretender Prozeß. Hierbei kommt es zu einer Veränderung der Chromosomen. Es gibt verschiedene Varianten der Mutation, wobei insbesondere die Gen- oder Punktmutation für die weiteren Betrachtungen von Bedeutung ist.

Ein wichtiger Umstand der Evolution bzw. der Genetik ist es, daß die entscheidenden Prozesse zufällig ablaufen. So ist weder die Anzahl noch die Position auf den Chromosomen

```

algorithm GENETIC ALGORITHM
begin
0   Wähle eine zufällige Startpopulation  $\mathbf{P} = \{\vec{x}_0, \vec{x}_1, \dots, \vec{x}_{n-1}\}$ 
1   Werte die Population  $\mathbf{P}$  aus
2   do ( $\neg$  terminate) —>
3      $\mathbf{P}' \leftarrow \{\}$ 
4     do ( $|\mathbf{P}'| < |\mathbf{P}|$ ) —>
5       selektiere Elternpaar  $\vec{x}_i, \vec{x}_j$  aus  $\mathbf{P}$  mit Heiratsschema
6       rekombiniere  $\vec{x}_i, \vec{x}_j$  zu  $\vec{x}'_i, \vec{x}'_j$  mit Wahrscheinlichkeit  $p_r$ 
7       mutiere  $\vec{x}'_i, \vec{x}'_j$  mit Wahrscheinlichkeit  $p_m$ 
8        $\mathbf{P}' \leftarrow \mathbf{P}' \cup \{\vec{x}'_i, \vec{x}'_j\}$ 
9   od
10  Werte die Population  $\mathbf{P}'$  aus
11  ersetze Individuen in  $\mathbf{P}$  durch Nachkommen aus  $\mathbf{P}'$  mit Ersetzungsschema
12 od
end.

```

Abbildung 5.10: Allgemeine Form des *Genetischen Algorithmus* in Pseudocode

bei der Rekombination und Mutation deterministisch festgelegt.

### Die berechnete Evolution

Ähnlich den künstlichen neuronalen Netzen wird bei den genetischen Algorithmen von den biologischen Vorbildern abstrahiert und nur die wesentlichen Aspekte modelliert. Hierzu sind insbesondere die Kodierung des Genoms und die Implementierung der Evolutionsfaktoren Selektion, Rekombination und Mutation zu betrachten.

In Abbildung 5.10 ist eine Erweiterung des in den vorangegangenen Abschnitten beschriebenen Mutations–Selektions–Algorithmus in der Pseudocode–Schreibweise aufgeführt, der im wesentlichen das Grundgerüst eines jeden genetischen Algorithmus darstellt.

Zu Beginn wird eine zufällige Startpopulation  $\mathbf{P}$  mit  $n$  Individuen gewählt und jedem Individuum eine Fitneß zugewiesen. In einem Iterationsverfahren wird nun die Evolution simuliert. Aus der Elternpopulation wird nach einem Heiratsschema ein Elternpaar ausgewählt (Selektion) und mit einer Wahrscheinlichkeit  $p_r$  rekombiniert. Anschließend werden dieses zufällig mutiert und als Nachkommen in eine Menge  $\mathbf{P}'$  eingefügt. Sind genügend Nachkommen produziert worden, dann wird die nächste Elterngeneration aus den Nachkommen entsprechend einem Ersetzungsschema erzeugt.

Detaillierte Einführungen in die Grundlagen der genetischen Algorithmen mit Beispielen sind u.a. in dem Klassiker von GOLDBERG (1989), in SCHÖNEBURG ET AL. (1994), HEISTERMANN (1994) oder MICHALEWICZ (1996) zu finden. KINNEBROCK (1994) verwendet einen leicht modifizierten Ansatz, der für ein Individuum exklusiv nur Mutation, Rekombination oder Reproduktion zuläßt.

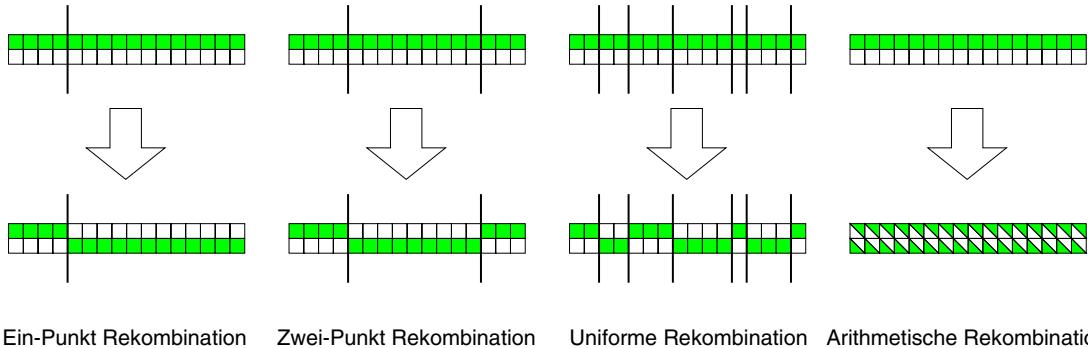


Abbildung 5.11: Die Bruchstellen für die Rekombination werden gleichverteilt ausgewählt. In den ersten drei Fällen setzen sich die Nachkommen aus den Allelen der Elternchromosomen zusammen. Bei der arithmetischen Rekombination werden hingegen neue Werte berechnet.

**Kodierung des Genoms** Die in der Optimierung freien Parameter bilden analog zu den Konfigurationen bei den einfacheren Mutations–Selektions–Verfahren das Genom, welches verändert, bewertet und ausgewählt oder verworfen wird. In der ursprünglichen Form der genetischen Algorithmen werden nur binäre Kodierungen verwendet, d.h. diskrete oder reellwertige Parameter werden in eine Binärdarstellung umgewandelt. Inzwischen haben sich jedoch analog zu den Evolutionsstrategien auch reelle Genome durchgesetzt. In MICHALEWICZ (1996) sind ausführliche Vergleiche zwischen den beiden Varianten durchgeführt worden mit dem Ergebnis, daß reelle Kodierungen den binären überlegen sind (höhere Präzision bei geringerer Rechenzeit).

**Zielfunktion und Fitneß** Die Zielfunktion  $f$  ist der eigentliche Gegenstand der Optimierung, d.h. die Funktion, die es zu minimieren oder maximieren gilt und im wesentlichen problemabhängig ist. Die Funktion  $f$  erhält als Argument ein Genom und liefert einen (reellen) Wert zurück, der widerspiegelt, wie gut das entsprechende Individuum das Optimum erreicht. Im Gegensatz dazu beschreibt die Fitneß  $f^*$ , die Chance eines Individuums bei der Selektion berücksichtigt zu werden. Die Fitneß ist somit eine Funktion vom Zielfunktionswert, der nicht zwingend mit diesem gleichzusetzen ist. Evolutionsstrategien kennen keinen Unterschied.

**Selektion mit Heiratsschema** Die Reproduktion zweier Individuen (Elternpaar) ist der erste Schritt zur Berechnung der Nachkommen. Das Heiratsschema beschreibt das Verfahren, nach dem die Individuen der Elterngeneration zu selektieren sind. Das Roulette–Auswahl–Verfahren bestimmt die Wahrscheinlichkeit  $p(x_i)$ , daß ein Individuum  $x_i$  selektiert wird, anhand der Fitneß  $f^*(x_i)$  bezogen auf die Summe der Fitneß aller Individuen

der Population durch

$$p(x_i) = \frac{f^*(x_i)}{\sum_{j=0}^{n-1} f^*(x_j)} . \quad (5.10)$$

Das Turnier–Auswahl–Verfahren bestimmt zwei Individuen anhand der beschriebenen Roulette–Methode und wählt von diesen dann das Individuum als ein Elternteil aus, welches die bessere Fitneß hat.

**Rekombination** Nachdem das Elternpaar selektiert ist, kommt es zum Austausch von genetischen Informationen. Die häufigsten Rekombinationsoperatoren sind in Abbildung 5.11 skizziert. Eine Besonderheit stellt die arithmetische Variante dar, bei der die einzelnen Allele durch

$$x'_i = (1 - \lambda) \cdot x_i + \lambda \cdot x_j \quad (5.11)$$

$$x'_j = (1 - \lambda) \cdot x_j + \lambda \cdot x_i \quad (5.12)$$

für  $\lambda \in [0, 1]$  verändert werden. Zu beachten ist, daß für  $\lambda = 0.5$  die beiden Nachkommen identisch sind, wodurch die Streuung der Chromosomen in der Population abnimmt. Dieses führt i.allg. zu einem schlechten Konvergenzverhalten. MÜHLENBEIN (1995) erweitert diesen Ansatz für  $\lambda \in [-d, 1+d]$ . Mit  $d = 0.2$  erzielen VICINI UND QUAGLIARELLA (1999) gute Ergebnisse.

**Mutation** Die Mutation ist bereits in Abschnitt 5.2.1 bei der Vorstellung der Mutations–Selektions–Algorithmen ausführlich besprochen worden. Numerische Ergebnisse haben gezeigt, daß es sinnvoll ist, die Variationsbreite  $\delta$  bei der Mutation in Abhängigkeit des Fortschreitens des Optimierungsprozesses zu verkleinern, da ansonsten eine gewisse Genauigkeit nicht überschritten wird. MICHALEWICZ (1996) schlägt den folgenden nichtuniformen Mutationsoperator vor.

$$x' = \begin{cases} x + \delta(i, x_{\max} - x) & \text{falls eine binäre Zufallszahl } z \text{ gleich 0 ist} \\ x - \delta(i, x - x_{\min}) & \text{ansonsten} \end{cases} \quad (5.13)$$

Die Variationsbreite  $\delta$  ergibt sich in Abhängigkeit des Iterationsschrittes  $i$  zu

$$\delta(i, y) = y \left\{ 1 - r \left( 1 - \frac{i}{I} \right)^b \right\} , \quad (5.14)$$

wobei  $I$  die maximal mögliche Anzahl von Iterationsschritten,  $r$  eine auf  $[0, 1]$  gleichverteilte Zufallszahl und  $b$  ein Systemparameter ist, der den Einfluß des Iterationsschrittes gewichtet. Die Kurven in Abbildung 5.12 verdeutlichen diesen Zusammenhang. Mit fortlaufenden Iterationsschritten wird die Wahrscheinlichkeit, große Veränderungen innerhalb des vorgegebenen Intervalls  $y$  auf einem Allel auszuführen, immer geringer.

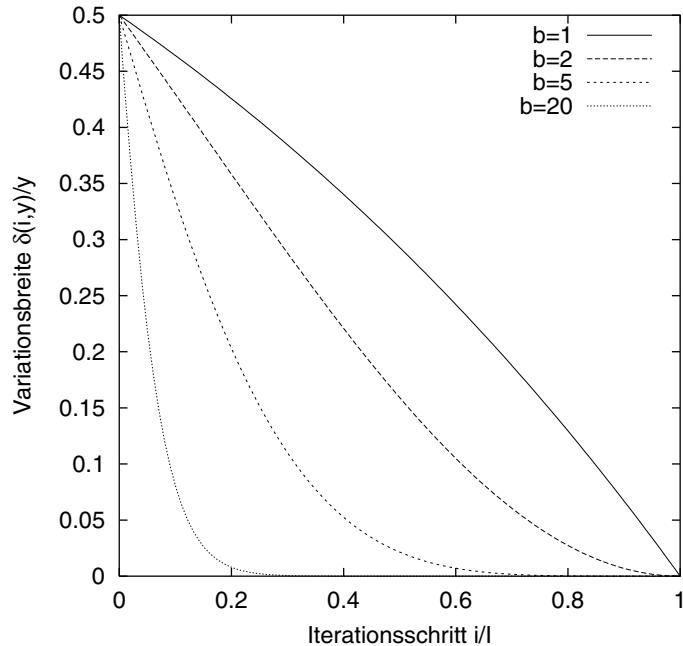


Abbildung 5.12: Die Kurven verdeutlichen das Abklingverhalten des nichtuniformen Mutationsoperators in Abhängigkeit des Parameters  $b$ . Für  $r = 0.5$  (mittlere zu erwartende Zufallszahl) nimmt mit fortlaufenden Iterationen die Variationsbreite ab.

**Ersetzung mit Ersetzungsschema** Im letzten Schritt innerhalb der simulierten Evolution wird nach einem Verfahren bestimmt, vergleichbar dem Heiratsschema, welche Nachkommen in die nächste Generation übernommen werden. Im einfachsten Fall erneuern alle Nachkommen die aktuelle Elterngeneration. Da hierdurch auch die besten Individuen der Eltern ersetzt werden, wird das Prinzip der Eliten (Elitismus) eingeführt. Dieses Schema reproduziert die jeweils besten, zum Beispiel die ersten beiden, der Elterngeneration und übernimmt diese anschließend ohne Veränderung in die nächste Generation.

### 5.3.2 Die C++ Klassenbibliothek GAlib (MIT)

Zur Implementierung der im nachfolgenden ausführlich dargestellten Optimierung von zweidimensionalen Profilschnitten wird die von WALL (1996) am MIT<sup>4</sup> entwickelte Klassenbibliothek verwendet<sup>5</sup>. Aufgrund des objektorientierten Ansatzes lässt sich die Bibliothek an eigene Problemstellungen unkompliziert anpassen. Im einfachsten Fall ist nur die gewünschte Zielfunktion zu definieren.

<sup>4</sup>Massachusetts Institute of Technology

<sup>5</sup>Für nicht kommerzielle Anwendungen ist die Verwendung der Bibliothek frei.

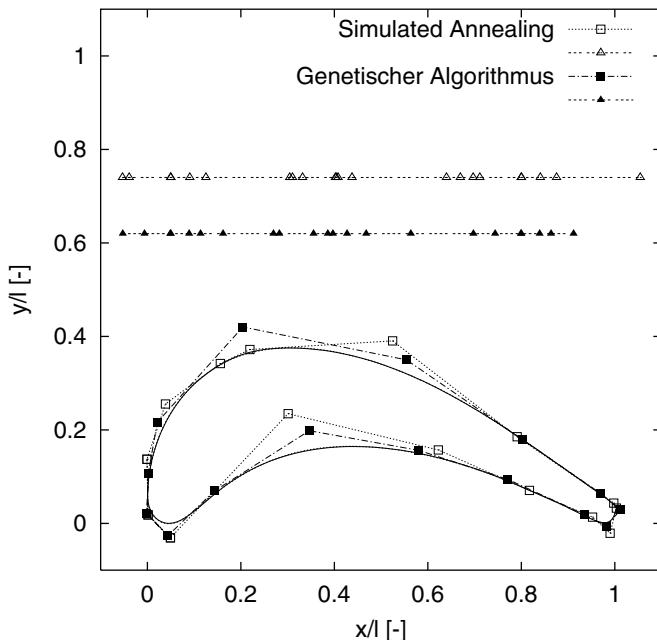


Abbildung 5.13: Die unterschiedlichen Kontrollpolygone der B-Splines sind das Ergebnis der Rauhigkeit der Zielfunktion.

### 5.3.3 Approximation mit B-Spline-Funktionen

In Abschnitt 5.2.3 ist das mit einem Optimierungsansatz gekoppelte Approximationsverfahren ausführlich vorgestellt worden. Diese Problematik wird nun auf ein Lösungsverfahren mit genetischen Algorithmen übertragen und mit der *Simulated Annealing*-Methode verglichen.

Der Mutationsoperator kann direkt vom Simulated Annealing Ansatz übernommen werden. Der arithmetische Rekombinationsoperator ist zu erweitern, um die notwendigen Bedingungen zu gewährleisten, damit das in der Approximation auftretende Gleichungssystem lösbar wird. Bei der Rekombination werden zwei Knotenvektoren  $\mathbf{s}$  und  $\mathbf{t}$  durch

$$\mathbf{s} = \{s_0, s_1, \dots, s_m\} \quad \rightarrow \quad \mathbf{s}' = \{s'_0, s'_1, \dots, s'_m\} \quad (5.15)$$

$$\mathbf{t} = \{t_0, t_1, \dots, t_m\} \quad \rightarrow \quad \mathbf{t}' = \{t'_0, t'_1, \dots, t'_m\} \quad (5.16)$$

mit  $s_i \in [\bar{s}_{i,\text{left}}, \bar{s}_{i,\text{right}}]$  und  $t_i \in [\bar{t}_{i,\text{left}}, \bar{t}_{i,\text{right}}]$  für  $i = 0, \dots, m$  durch

$$t'_i = \min(\bar{t}_{i,\text{right}}, \max(\bar{t}_{i,\text{left}}, (1 - \lambda) \cdot t_i + \lambda \cdot s_i)) \quad (5.17)$$

$$s'_i = \min(\bar{s}_{i,\text{right}}, \max(\bar{s}_{i,\text{left}}, (1 - \lambda) \cdot s_i + \lambda \cdot t_i)) \quad (5.18)$$

zwei neue Nachkommen berechnet. Die Bestimmung der Intervallgrenzen, innerhalb derer die Knoten verändert werden können, erfolgt analog der Vorgehensweise des Mutationsoperators.

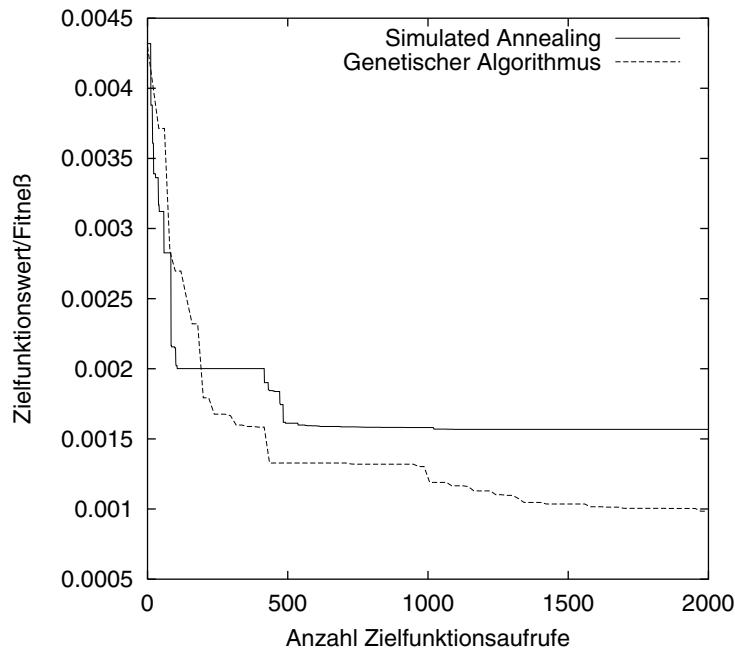


Abbildung 5.14: Das mit einem genetischen Algorithmus gekoppelte Approximationsverfahren zeigt eine bessere Konvergenz als die entsprechende *Simulated Annealing*–Variante.

In den Abbildungen 5.13 bis 5.15 ist das Ergebnis der Approximation mit einem kombinierten Optimierungsansatz dargestellt. Hierzu wurde das VKI–Profil aus WATZLAWICK (1991) mit den beiden Verfahren approximiert. Es findet keine Skalierung des Zielfunktionswertes statt, so daß Zielfunktion  $f$  und Fitneß  $f^*$  identisch sind. Es zeigt sich, daß die Optimierung mit genetischen Algorithmen ein besseres Ergebnis, d.h. ein B–Spline–Profil mit einem geringeren Approximationsfehler, liefert. Abbildung 5.13 stellt die mit den beiden Optimierungsverfahren approximierten B–Spline–Kurven dar. Deutlich ist zu erkennen, daß die Knoten und damit auch die Kontrollpunkte zum Teil sehr unterschiedlich sind. Es ist somit davon auszugehen, daß beide Verfahren aufgrund der Rauhigkeit der Zielfunktion nur lokale Minima gefunden haben.

Die in diesem Abschnitt beschriebenen Konzepte und Erfahrungen werden nun nachfolgend versucht, auf die aerodynamische Optimierung von zweidimensionalen Profilen zu übertragen.

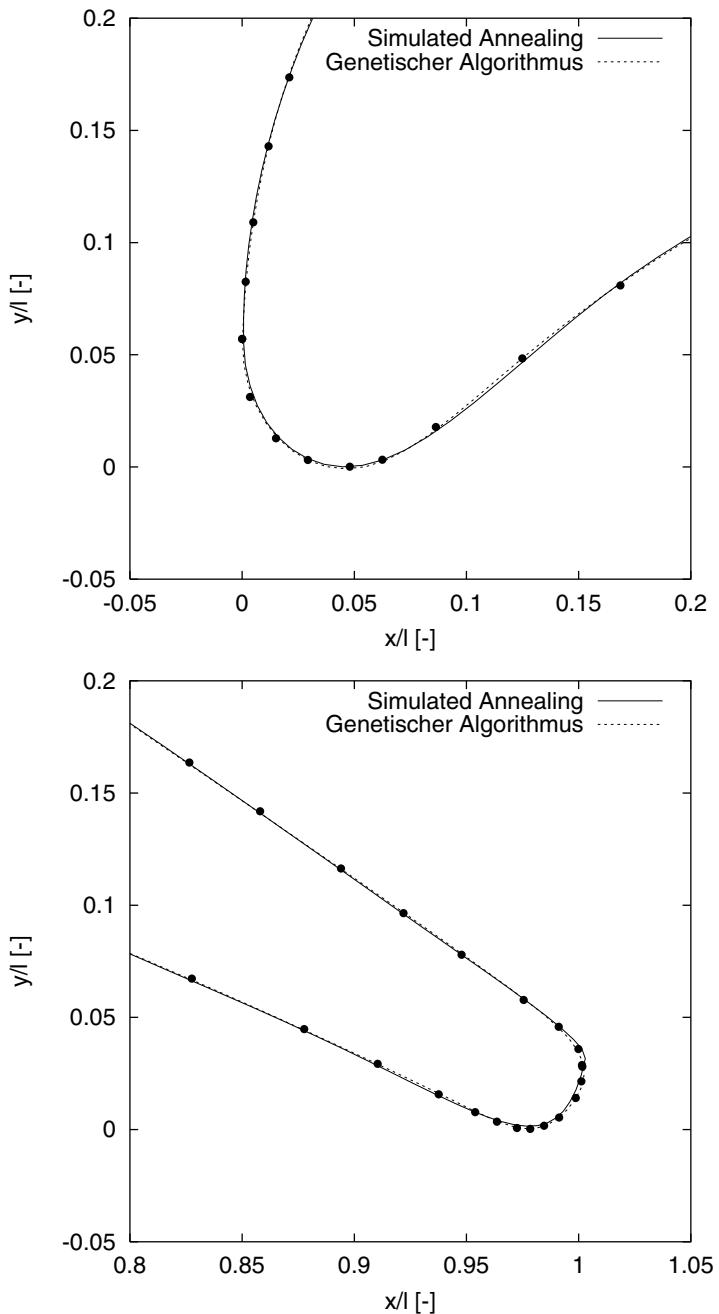


Abbildung 5.15: Eine Vergrößerung der Auflösung der Vorder- und Hinterkante läßt erkennen, daß der genetische Algorithmus mit einem geringeren Fehler gegenüber der *Simulated Annealing*-Methode die vorgegebenen Konturpunkte approximiert.

## 5.4 Optimierung von Profilen

Zur aerodynamischen Optimierung von Profilen werden sowohl deterministische, das sind i.allg. Gradientenabstiegsverfahren, als auch stochastische, vornehmlich Simulated Annealing und genetische Algorithmen, verwendet. SADREHAGHIGHI ET AL. (1995); NERURKAR ET AL. (1996); SUBEL (1998) setzen Gradientenverfahren zur Optimierung ein. Eine wichtige Klasse bilden hierbei die Quasi–Newton–Verfahren (vgl. GILL ET AL. (1981)).

DREYER ET AL. (1995); SCHULZ ET AL. (1996); EGARTNER UND SCHULZ (1998) verwenden einen neuartigen Ansatz, bei dem zu einer vorgegebenen Geschwindigkeitsverteilung das zugehörige Schaufelprofil simultan mit der Strömungsberechnung mittels eines SQP–Verfahrens (sequentielle quadratische Programmierung, vgl. SPELLUCCI (1993)) ermittelt wird. Als Nebenbedingungen fließen neben geometrischen Restriktionen (z.B. Sehnenlänge) somit die Strömungsgleichungen ein. OROZCO UND GHATTAS (1996) verwenden ebenfalls einen Simultanoptimierungsansatz.

PÉRIAUX ET AL. (1995); OBAYASHI UND TSUKAHARA (1997) vergleichen verschiedene Optimierungsverfahren und geben den genetischen Algorithmen den Vorzug, da diese, trotz der i.allg. höheren Anzahl von Zielfunktionsauswertungen, bessere Resultate erzielen. QUAGLIARELLA (1995) sieht zusätzlich die Stabilität gegenüber Rauhigkeiten der Zielfunktion, die insbesondere bei komplexen Systemen auftreten, als wesentlichen Vorteil gegenüber den klassischen, gradientenbasierten Methoden. Umfangreiche Untersuchungen sind auch in BAIER (1998) zu finden, wobei insbesondere eine Erweiterung des Simulated Annealing Verfahrens von INGBER UND ROSEN (1992) gute Sucheigenschaften bei der Optimierung von Turbinenprofilen aufweist. VICINI UND QUAGLIARELLA (1999) kombinieren den genetischen Algorithmus mit einem deterministischen Verfahren, um die Effizienz der Optimierung zu erhöhen.

Da über die Glattheit der Zielfunktion bei der aerodynamischen Optimierung von Profilen nur schwerlich Aussagen zu treffen sind, wird in dieser Arbeit ein genetischer Algorithmus verwendet. Zur Reduzierung der Zielfunktionsaufrufe wird die Startgeometrie, wie im vorherigen Kapitel beschrieben, durch ein neuronales Netz ermittelt. Somit ist es als Fortführung und Erweiterung der Verfahren von SELIG UND COVERSTONE-CARROLL (1995); QUAGLIARELLA UND CIOPPA (1995); OBAYASHI UND TAKANASHI (1996); VICINI UND QUAGLIARELLA (1997); KUIPER ET AL. (1998); TRIGG ET AL. (1999) zu verstehen, die genetische Algorithmen erfolgreich zur Optimierung von Tragflügel– und Turbinenprofilen einzusetzen. LAWRENZ (1995) verwendet hierzu die Evolutionsstrategie. Eine algorithmische Einführung evolutionärer Verfahren anhand der Optimierung von NACA–Profilen gibt DE FALCO (1997a,b).

Der Ansatz mit neuronalen Netzen zur Bestimmung der Startgeometrie ist wesentlich flexibler als die Suche in einer Datenbank von PIERRET UND VAN DEN BRAEMBUSSCHE (1999), da durch die Generalisierungseigenschaften des Netzes auch unbekannte, nicht trainierte Startprofile generiert werden können.

### 5.4.1 Zielfunktion und Genom

Die Zielfunktion  $f$  zur Bewertung eines Profils, durch  $\vec{x}$  beschrieben, setzt sich mittels

$$f(\vec{x}) = \frac{1}{w} \sum_{i=0}^n w_i \frac{f_i(\vec{x})}{r_i} \quad \text{mit} \quad w = \sum_{i=0}^n w_i \quad (5.19)$$

aus geometrischen und aerodynamischen Anteilen zusammen. Um die einzelnen Anteile  $f_i(\vec{x})$  vergleichbar zu machen, werden diese jeweils auf einen Referenzwert  $r_i$  normiert. Zusätzlich wird ein Gewichtungsfaktor  $w_i$  eingeführt, der es erlaubt, die einzelnen Anteile zueinander in Beziehung zu setzen. Einzelne geometrische (z.B. maximale Dicke  $d_{\max}/l$ ) oder aerodynamische (z.B. Abströmwinkel  $\alpha_2$ ) Zielvorgaben  $y_t$  (Sollwert) lassen sich durch

$$f_y = \left( 1 - \frac{y}{y_t} \right)^2 \quad (5.20)$$

bewerten, wobei  $y$  der aktuelle Istwert des betrachteten Profils  $\vec{x}$  ist. Durch Vorgabe einer normierten Folge an  $m + 1$  diskreten Stützstellen  $(x_i, Ma_{t,i}^*)$  ist es möglich, mit

$$f_{\text{Ma}} = \sum_{i=0}^m |Ma_{t,i}^* - Ma_i^*| \cdot \Delta x \quad \text{für} \quad \Delta x = 0.5 \cdot \begin{cases} x_1 - x_0 & \text{falls } i = 0 \\ x_m - x_{m-1} & \text{falls } i = m \\ x_{i+1} - x_{i-1} & \text{ansonsten} \end{cases} \quad (5.21)$$

die Fehlerfläche der Machzahlverteilung für die Saug- bzw. der Druckseite zu bestimmen und gegebenenfalls zu minimieren. Als Bezugsgröße für die Normierung wird das Machzahlmaximum  $Ma_{\max}$  gewählt. Die Funktion

$$f_{\text{Ma,max}} = \max(0, Ma_{\max} - Ma_{t,\max}) \quad (5.22)$$

kann zusätzlich als Strafterm bei Überschreiten einer Machzahl  $Ma_{t,\max}$  ergänzt werden.

Neben der Erfüllung einer Umlenkung der Strömung ist die Minimierung der Profilverluste die zentrale Aufgabe in der Optimierung. Hierzu fließt der Totaldruckverlust  $\omega$  (vgl. Abschnitt 3.4.5) als direkte Größe  $f_\omega = \omega$  in die Zielfunktion mit ein. Das in dieser Arbeit entwickelte Modell der Profilgrenzschicht kann Phänomene infolge abgelöster Strömung (z.B. Verluste) nur bedingt wiedergeben. Aus diesem Grund wird ein zusätzlicher Strafterm eingeführt, der eine Ablösung der Strömung auf der Saug- und Druckseite bestraft.

Das Genom  $\vec{x}$  wird durch die zweidimensionalen Kontrollpunkte der B-Spline-Funktion des Profils definiert. Alle weiteren Designparameter, z.B. der Grad  $p$  und der Knotenvektor  $\mathbf{t}$ , bleiben konstant. Hierdurch wird die Anzahl der freien Optimierungsparameter reduziert.

### 5.4.2 Inverser Entwurf

Zur Überprüfung der Leistungsfähigkeit und zur Bestimmung des Einflusses der verschiedenen Parameter des genetischen Algorithmus wird versucht, durch Vorgabe einer bekannten, normierten Machzahlverteilung und des Abströmwinkels das zugehörige Profil

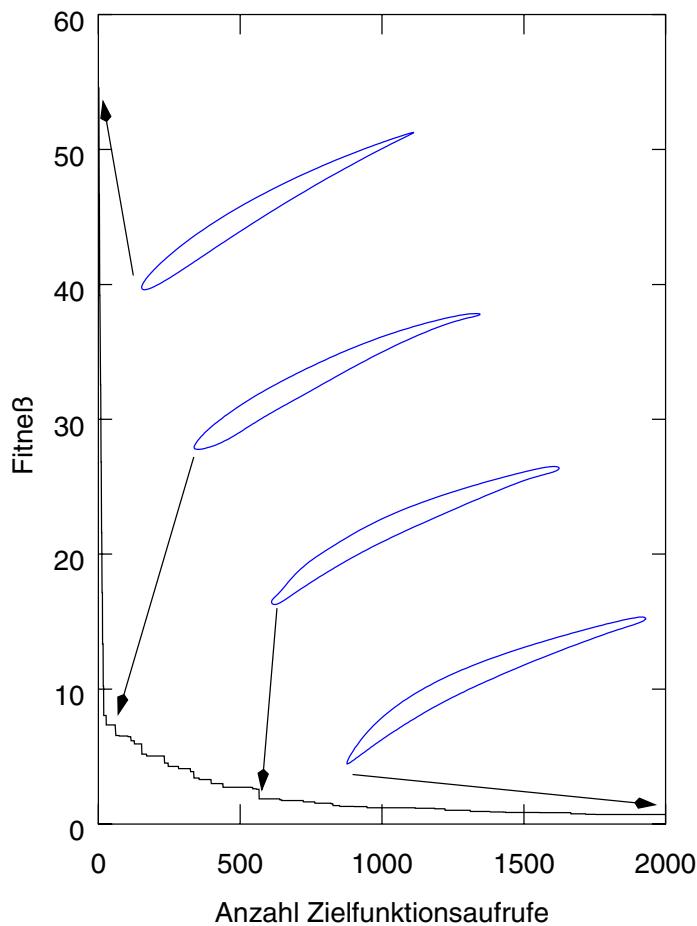


Abbildung 5.16: Der Konvergenzverlauf für 2000 Zielfunktionsaufrufe entspricht ungefähr 100 Generationen mit jeweils 20 Individuen. Für den Schritt 1, 61, 567 und 2005 ist das zugehörige Profil eingezeichnet.

zu rekonstruieren. Dieses entspricht der Vorgehensweise des inversen Entwurfs.

Als Beispiel wird hierzu das Profil aus MERTENS (1995) gewählt und mit dem implementierten Stromfunktionsverfahren die Strömung bestimmt. Die Zielfunktion für die Optimierung setzt sich aus der Vorgabe einer normierten Machzahlverteilung entsprechend Gleichung (5.21) mit 31 bzw. 33 Stützstellen auf der Saug- bzw. Druckseite und der Vorgabe eines Abströmwinkels entsprechend Gleichung (5.20) mit  $y_t = 19.5^\circ$  zusammen. In Abbildung 5.16 ist der Konvergenzverlauf für 2000 Zielfunktionsaufrufe dargestellt. Dieses entspricht ungefähr 100 Generationen bei einer Population von 20 Individuen.

In den Abbildungen 5.17 und 5.18 ist die Geometrie bzw. die Machzahlverteilung des Startprofils und des optimierten Profils dargestellt. Zusätzlich ist die Zielvorgabe der Machzahlverteilung bzw. die zugehörige Zielgeometrie aufgetragen. Es zeigt sich, daß das Profil hinreichend genau rekonstruiert wird. Größere Abweichungen treten insbesondere an der

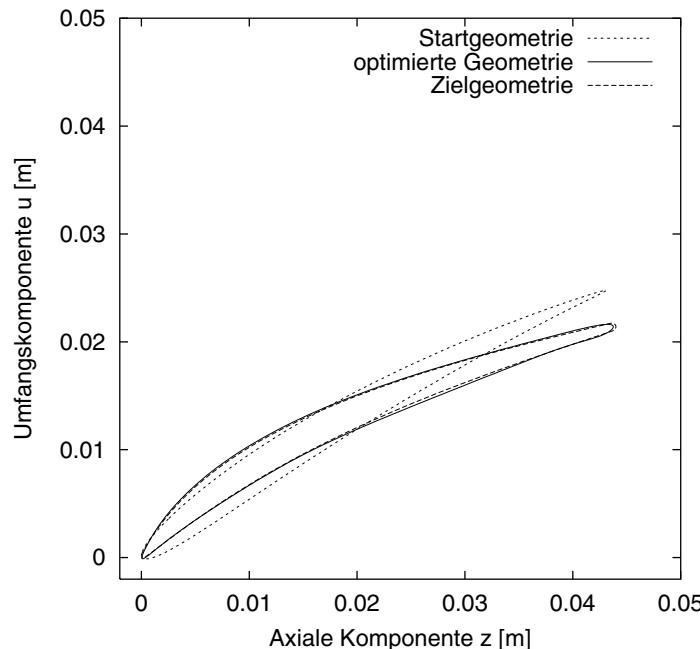


Abbildung 5.17: Ausgehend von einer Startgeometrie wird durch den genetischen Algorithmus das Profil aus MERTENS (1995) rekonstruiert.

Hinterkante auf. Die Ursache hierfür ist die in dieser Arbeit verwendete Diskretisierung in Form eines C–Netzes, bei der die Hinterkante abgeschnitten wird. Unterschiedliche Hinterkanten sind daher unter Umständen nicht zu berücksichtigen.

Weiterhin ist für dieses Beispiel zu bemerken, daß, obwohl die Originalgeometrie nicht in der mathematischen Form einer B–Spline–Funktion definiert ist, durch den inversen Entwurf die Profilgeometrie mit einer periodischen B–Spline–Funktion mit 16 Kontrollpunkten gut approximiert werden kann.

#### 5.4.3 Direkter Entwurf

Während beim inversen Entwurf die Machzahlverteilung a priori bekannt sein muß, entspricht der direkte Entwurf der i.allg. üblichen Vorgehensweise, bei der ein Profil mit einer definierten Umlenkung bei möglichst geringem Verlust gefordert wird. Hierbei sind weiterhin geometrische Randbedingungen zu berücksichtigen.

Im folgenden wird ein Beispiel mit den Zuströmbedingungen  $Ma_1 = 0.5$ ,  $\alpha_1 = 30^\circ$  und  $Tu = 2\%$  betrachtet. Ausgehend von einem Startprofil wird nun eine optimierte Geometrie gesucht, die den Verlust unter Gewährleistung einer geforderten Abströmung  $\alpha_2 = 5^\circ$  minimiert. Die Zusammensetzung der Zielfunktion sowie die erzielten Ergebnisse sind nachfolgend tabelliert. Durch die Vorgabe eines Machzahlniveaus wird die Ausbildung von Machzahlspitzen, wie diese durch eine Fehlanströmung an der Vorderkante entstehen, verhindert. Die Gewichtungsfaktoren  $w_i$  sind identisch 1.

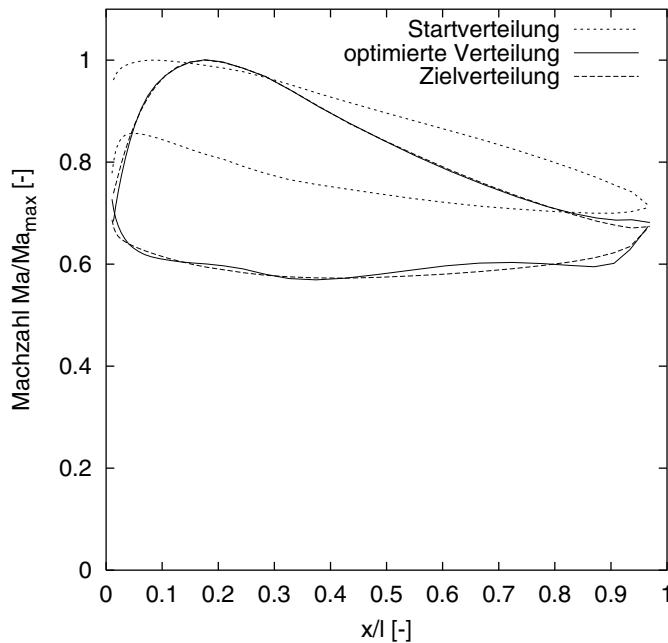


Abbildung 5.18: Abweichungen bei der Optimierung einer vorgegebenen Machzahlverteilung ergeben sich insbesondere auf der Druckseite des Profils.

Anteil	Zielfunktion		Geometrie	
	Vorgabe	Referenzwert	Start	optimiert
Sehnenlänge $l$	0.0674 m	0.0015	0.0674 m	0.0673 m
maximale Dicke $d_{\max}/l$	7%	0.02	6.2%	6.8%
Abströmwinkel $\alpha_2$	5°	0.04	4.3°	5.0°
Profilverlust $\omega$	0.0	0.0025	3.75%	1.03%
Machzahlniveau $Ma_{t,\max}$	0.9	0.1	0.76	0.75

Als Rekombinationsoperator des genetischen Algorithmus wird die arithmetische Variante mit dem Parameter  $d = 0.05$  (siehe Seite 122) bei einer Wahrscheinlichkeit  $p_r = 0.7$  sowie der nichtuniforme Mutationsoperator mit  $p_m = 0.1$  gewählt. Eine Population besteht aus 20 Individuen, die über 50 Generationen entwickelt wird. Die Selektion der Eltern in jedem Iterationsschritt erfolgt mit dem Turnier–Auswahl–Verfahren. Als Startgeometrie ist ein gekrümmtes Profil mit konstanter Dicke verwendet worden. Das Profil besteht aus 14 zweidimensionalen Kontrollpunkten, die das Genom bilden.

Weitere Ergebnisse sind in den Abbildungen 5.19 bis 5.21 wiedergegeben. Während der Abströmwinkel und die maximale Dicke konstant gehalten werden konnten, ist insbesondere an der Vorder- und Hinterkante die Geometrie durch die Optimierungsstrategie verändert worden. Durch die Verjüngung der Vorderkante wird erreicht, daß die Strömung länger laminar anliegend bleibt und somit die Ausbildung einer Grenzschicht verringert wird. Als direkte Folge wird der Totaldruckverlust reduziert. Dieses wird auch in Abbildung 5.21 sehr deutlich. Der laminar–turbulente Übergang verschiebt sich von 10% auf über 40%

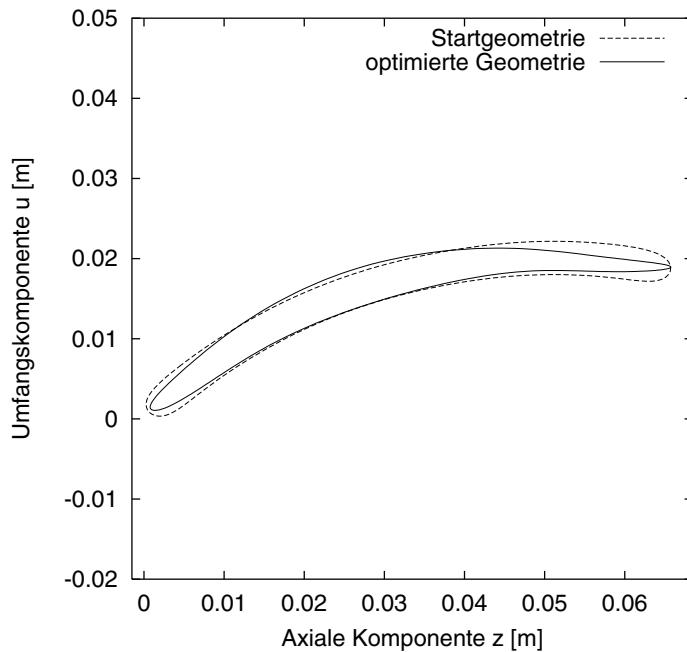


Abbildung 5.19: In der Optimierung wird die Vorder- und Hinterkante verjüngt, wodurch sich der Profilverlust reduziert.

der Sehnenlänge. Die dünne Hinterkante sorgt im wesentlichen für eine Verringerung des Totaldruckverlustes, der durch die sehr dicke Hinterkante der Startgeometrie entstanden ist.

Im Rahmen dieser Arbeit durchgeföhrte Untersuchungen haben gezeigt, daß Schwächen im physikalischen Modell zur Beschreibung der Strömungsvorgänge von den Optimierungsstrategien bei der Suche nach geeigneten Profilschnitten ausgenutzt werden. Dieses führte in vielen Fällen zu unzweckmäßigen Geometrien. In die Zielfunktion zur Bewertung von Profilen werden hauptsächlich aerodynamische Kriterien (Totaldruckverlust, Abströmwinkel) herangezogen. Mechanische, konstruktive Eigenschaften hingegen werden durch das physikalische Modell nicht hinreichend modelliert und nur über die maximale Dicke  $d_{\max}/l$  bewertet. Da aerodynamisch betrachtet, unendlich dünne Profile den geringsten Verlust aufweisen, versucht die Optimierungsstrategie solche Profile als optimal auszuwählen. Nachfolgend wird an einigen Beispielen diese grundsätzliche Problematik verdeutlicht und die Lösungsansätze, soweit diese in der vorliegenden Arbeit Anwendung finden, beschrieben.

Zu Beginn der Untersuchungen sind insbesondere solche Geometrien als optimal befunden worden, die eine möglichst frühe turbulente Ablösung aufweisen. Da die Verluste infolge der abgelösten Strömung nicht modelliert werden, ist im wesentlichen nur die Verdrängungsdicke  $\delta_1$  verantwortlich, die bei einer Ablösung direkt hinter der Vorderkante sehr gering ist. Erst durch Einföhrung eines Strafterms bei einer Ablösung konnte dieses Problem

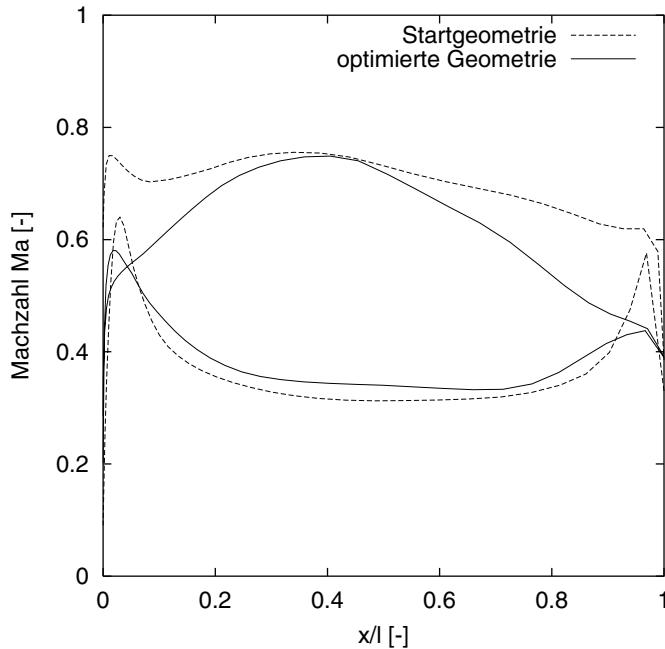


Abbildung 5.20: Vergleich der Machzahlverteilung der Start– und optimierten Geometrie

gelöst werden. Weitere, zukünftige Entwicklungen sollten das aerodynamische Modell um eine Behandlung der abgelösten Strömung erweitern.

Eine weitere Unvollständigkeit des Modells führte in der Optimierung zu Geometrien, die durch eine Keilform, d.h. spitze Vorder– und stumpfe Hinterkante, gekennzeichnet sind. Die Folge ist eine langanhaltend laminare Strömung mit geringen Grenzschichtdicken und damit geringen Verlusten. Erst durch die Einführung des Totaldruckverlustes  $\omega_d$  nach Gleichung (3.134) konnte die Unzulänglichkeit der abgebildeten Physik behoben werden.

Ein weiteres Problem bei der Formulierung der Zielfunktion ist die Beschränkung in dieser Arbeit auf den Entwurf des Profils in *einem* Auslegungspunkt. Da i.allg. innerhalb eines Verdichters jedoch eine Bandbreite von Strömungszuständen, z.B. eine Variation des Zuströmwinkels in einem festgelegten Intervall, vorliegt, werden Geometrien optimiert, die nur genau für die definierten Randbedingungen im Auslegungspunkt den Verlust minimieren. Zukünftige Entwicklungen sollten daher in Hinblick auf eine Mehrpunkt–Optimierung ergänzt werden.

Anzumerken ist, daß die beschriebenen Probleme die Folge des abgebildeten physikalischen Modells und nicht der Optimierung, insbesondere nicht die Folge der Eigenschaften des genetischen Algorithmus, sind. Jede Strategie kann nur innerhalb der modellierten Physik optimieren. Menschliche Konstrukteure besitzen hingegen Erfahrungen, die solche untauglichen Profilgeometrien ausschließen. Zukünftige Entwicklungen sollten daher versuchen, mechanische Phänomene angemessen zu berücksichtigen, die anschließend zusammen mit der aerodynamischen Bewertung in die Zielfunktion miteinfließen.

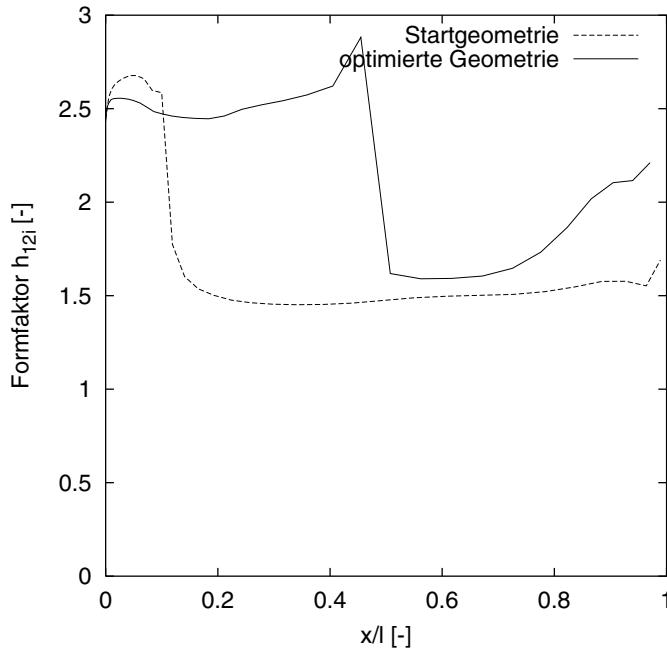


Abbildung 5.21: Der Vergleich der Grenzschichtbelastung auf der Saugseite zeigt einen verschobenen laminar–turbulenten Übergang der optimierten Geometrie gegenüber dem Startprofil.

Der in dieser Arbeit entwickelte Ansatz zum geometrischen Entwurf mit periodischen B–Spline–Funktionen stellt einen hohen Freiheitsgrad in der Optimierung dar. Dieses gewährleistet, daß neue Lösungen gefunden werden können, die durch konventionelle Entwurfsmethoden, z.B. durch parametrisierte Profilfamilien, nicht möglich sind. Auf der anderen Seite ist ein hinreichend genaues aerodynamisches und mechanisches Modell wichtig, um untaugliche Geometrien, wie beschrieben, bei der Optimierung zu vermeiden. Nachfolgend wird auf die bereits in Abschnitt 4.4.1 beschriebene Geometriesystematik beim direkten Entwurf zurückgegriffen. Probleme im physikalischen Modell werden hierdurch reduziert. Es werden jedoch nur solche Geometrien gefunden, die innerhalb dieser Systematik als optimal bewertet werden.

Als Beispiel wird die Optimierung eines Profils mit den Zuströmbedingungen durch  $Ma_1 = 0.437$  und  $\alpha_1 = 40^\circ$  betrachtet. Als Optimierungsstrategie wird zum einen ein genetischer Algorithmus, mit einer Ein–Punkt bzw. einer arithmetischen Rekombination, und zum Vergleich die Simulated Annealing Strategie herangezogen. Aufgrund des nicht sehr umfangreichen Genoms infolge der verwendeten Geometriesystematik ist für den genetischen Algorithmus eine kleinere, 10 Individuen umfassende, Population, die über 50 Generationen entwickelt wird, ausreichend. Für das Simulated Annealing Verfahren terminiert die Iteration dementsprechend nach 500 Zielfunktionsaufrufen.

Abbildung 5.22 und die nachfolgende Tabelle zeigen, daß die Optimierung mit genetischen Algorithmen und durch Simulated Annealing zu weitgehend identischen Profilen führt.

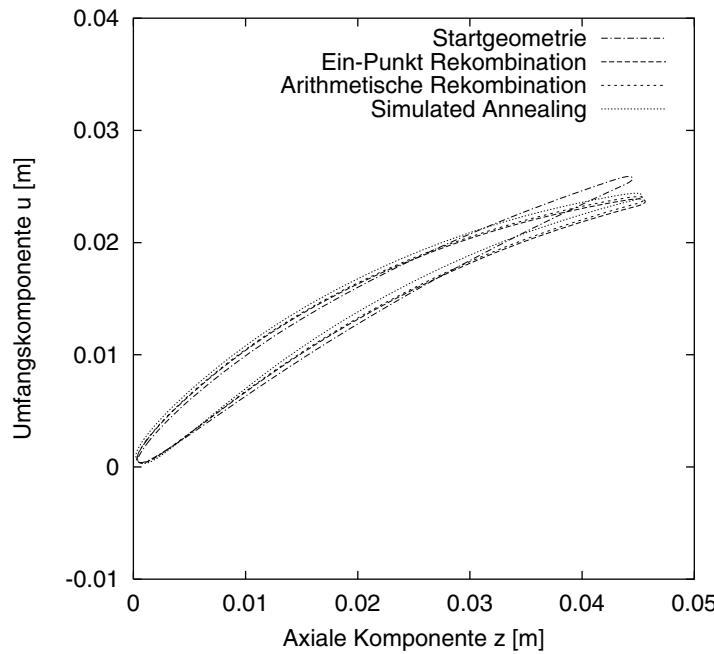


Abbildung 5.22: Die unterschiedlichen Optimierungsstrategien führen zu identischen Geometrien.

	Ein-Punkt Rekombination	Arithmetische Rekombination	Simulated Annealing
Abströmwinkel $\alpha_2$	$20.0^\circ$	$20.2^\circ$	$20.2^\circ$
Profilverlust $\omega$	1.08%	1.07%	1.10%

Die Machzahl- sowie die Formfaktorverteilung ist den Abbildungen 5.23 bzw. 5.24 zu entnehmen. Durch die Bestrafung eines Machzahlniveaus über  $0.55 Ma$  wird eine Fehlströmung vermieden. Der laminar-turbulente Übergang findet früh, bereits bei 20% der Sehnenlänge, statt. Die Belastung der Saugseite steigt moderat an und bleibt ohne eine turbulente Ablösung.

Da alle Verfahren zu einer fast identischen Geometrie führen, lässt dieses vermuten, daß tatsächlich das optimale Profil innerhalb der Geometriesystematik für die geforderten Randbedingungen bestimmt worden ist.

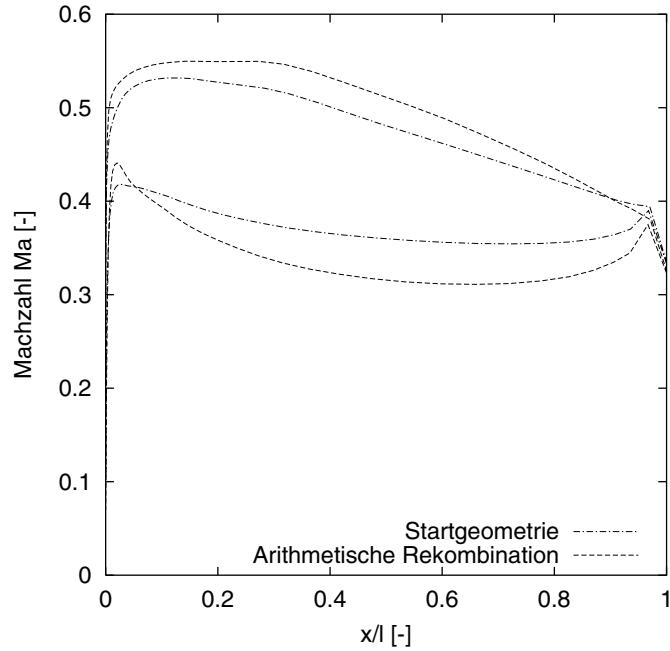


Abbildung 5.23: Machzahlverteilung der optimierten Geometrie

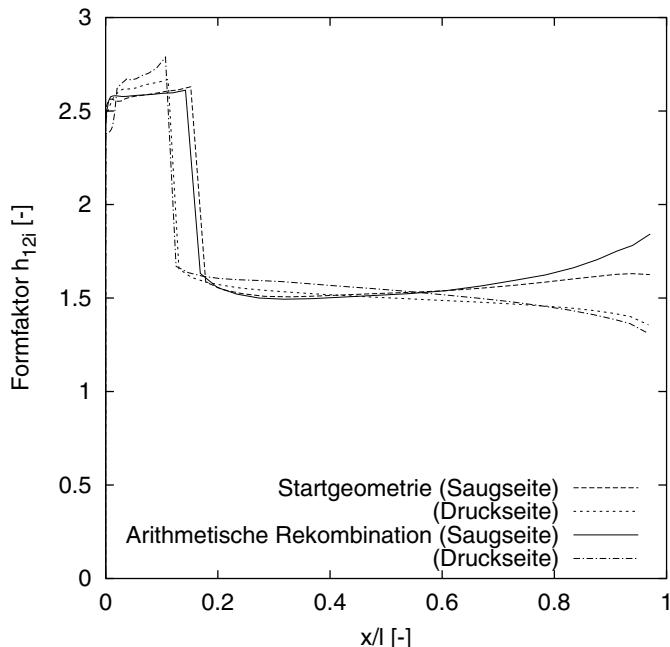


Abbildung 5.24: Grenzschichtbelastung der optimierten Geometrie

## 5.5 Ausblick

Moderne Verdichter und Turbinen sind schon hochgradig für ihre jeweiligen Anforderungen optimiert. Der verstärkte Wettbewerbsdruck erzwingt es, effiziente, den Vorgaben angepaßte, Turbomaschinen in stetig kürzeren Entwicklungsphasen zu entwerfen. Bei der Optimierung von Strömungsmaschinen ist daher neben einer Verfeinerung der aerodynamischen Modelle eine Parallelisierung des Auslegungsprozesses notwendig.

In dieser Arbeit wird ein zweidimensionales Verfahren zur numerischen Bestimmung der Strömung verwendet. Um eine Reihe aerodynamischer Phänomene zu analysieren und damit in der Optimierung zu berücksichtigen, reichen diese jedoch nicht aus. Ein Einsatz von dreidimensionalen Navier–Stokes–Lösungsverfahren scheint in Anbetracht der Rechenzeit in der näheren Zukunft mehr als fraglich. Sinnvoller erscheint es zum Beispiel daher, die  $S_1$ –Rechnung mit einem  $S_2$ –Verfahren zu koppeln (siehe RÖPER (1994)).

Ein nicht zu unterschätzendes Problem ergibt sich bei der automatischen dreidimensionalen Gittergenerierung, welches sich bereits im zweidimensionalen Fall widerspiegelt. In dieser Arbeit sind einige Anstrengungen unternommen worden, um insbesondere die Vorderkante des Profils für alle Individuen einer Population ähnlich fein aufzulösen. Nur hierdurch ist es gewährleistet, daß die Beschleunigungsgebiete an der Vorderkante aufgelöst werden. Ansonsten besteht die Gefahr, daß Rauhigkeiten der Zielfunktion den Optimierungsprozeß in lokalen Minima enden lassen.

Bei der Auslegung von Turbomaschinen sind eine Reihe unterschiedlicher Disziplinen, z.B. Aerodynamik, Thermodynamik, Konstruktion, Festigkeit und Kostenkontrolle, beteiligt, was oftmals zu einem iterativen, sequentiellen Prozeß führt. Bei einer frühzeitigen Integration der verschiedenen Fachabteilungen in einem gemeinsamen Optimierungsprozeß (*multidisciplinary optimization*) lassen sich zeitaufwendige Iterationsschritte reduzieren. NADON ET AL. (1999) zeigen beispielhaft die Optimierung eines kompletten Triebwerks mit genetischen Algorithmen. Die Zielfunktion setzt sich dabei aus thermodynamischen und aerodynamischen Anteilen zusammen.

Das biologische Vorbild der Evolution ist ein parallel ablaufender Prozeß. Aufgrund der i.allg. vorhandenen Ein-Prozessor–Rechner ist bei der Realisierung von genetischen Algorithmen jedoch eine sequentielle Formulierung notwendig. Eine wesentliche Verringerung der Rechenzeit ist durch den Einsatz von Parallelrechner möglich. Prinzipiell lassen sich hierbei zwei Rechnerarchitekturen unterscheiden. PÉRIAUX ET AL. (1998) verwenden eng gekoppelte, massiv parallele Rechner zur Parallelisierung der numerischen Strömungsbe-rechnung. QUAGLIARELLA (1995) stellt zudem ein Lastausgleichsverfahren vor, um die Berechnungen auf die Prozessoren zu verteilen. Aufgrund der heutigen heterogenen Rechnerlandschaft in Firmen und Universitäten besitzt das lose gekoppelte, verteilte Rechnen ein größeres Potential. Insbesondere ist hierbei der *Parallel Virtual Machine*–Ansatz (PVM) von GEIST ET AL. (1994) interessant, der es ermöglicht, einzelne Rechner zu einer parallelen, virtuellen Maschine zusammenzufügen. Der Anwender erhält einen transparenten Blick, der die Heterogenität, insbesondere bezüglich der einzelnen Betriebssysteme und Prozessorgeschwindigkeiten sowie den physikalischen Standort, verbirgt (siehe auch FISCHER (1999)). Zudem ist PVM hochgradig skalierbar. DOU UND PHAN-THIEN (1998)

entwickeln hierzu ein Verfahren, um die Strömungsberechnung auf die verschiedenen Rechnerknoten zu verteilen.

Nach DOORLY (1995) lassen sich zwei Methoden zur parallelen Ausführung von genetischen Algorithmen unterscheiden. Bei dem klassischen Modell lässt sich die Auswertung (Zeile 10 im Algorithmus von Abbildung 5.10) auf eine Anzahl von Prozessoren verteilen. Jeder Prozessor erhält ein Individuum zur Analyse und Bestimmung der Fitneß. Erfolgreich ist auch das andere Modell eingesetzt worden, welches hingegen eine Vielzahl von Populationen verwendet, wobei jedem Prozessor eine Gruppe von Individuen zugeordnet wird. Durch eine Erweiterung des genetischen Algorithmus kommt es zur Migration von einzelnen, den jeweils besten, Individuen zwischen den Populationen. Die in dieser Arbeit verwendete GAlib Bibliothek von WALL (1996) implementiert beide Modelle und ist für den Einsatz mit PVM vorbereitet.

TONG UND GREGORY (1990); TONG ET AL. (1992) skizzieren mit dem *Engineous*-System die zukünftige Richtung bei der Auslegung. *Engineous* setzt auf vorhandenen Programmen auf und definiert eine Schnittstelle (*shell*) zum Konstrukteur zur automatischen Auslegung. Mit Methoden der symbolischen Informationsverarbeitung aus dem Bereich der künstlichen Intelligenz (vgl. hierzu auch Abschnitt 1.2) lassen sich Wissen und Methoden des Konstrukteurs in eine Wissensdatenbank übertragen. Die implementierten Optimierungsverfahren, Gradientenverfahren sowie Simulated Annealing und genetische Algorithmen, greifen dazu bei der Auslegung zurück. Die Eingriffe des Menschen werden dadurch minimiert; im Idealfall beschränkt sich die Interaktion auf die Vorgabe der Randbedingungen und Zielvorgaben. LEE ET AL. (1993) beschreiben einen mit *Engineous*-Werkzeugen entwickelten Verfahren zur automatisierten Auslegung von Turbinen, welches sowohl aerodynamische als auch mechanische Randbedingungen berücksichtigt.

# Kapitel 6

## Resümee

Der Schwerpunkt der vorliegenden Arbeit ist die Entwicklung von Verfahren des geometrischen Entwurfs und der aerodynamischen Optimierung zweidimensionaler Turbomaschinenbeschaufelungen. Hierbei wurden insbesondere zwei Methoden der *Computational Intelligence*, künstliche neuronale Netze und genetische Algorithmen, untersucht. Die implementierten Verfahren können als Basis für weitere Untersuchungen herangezogen und erweitert werden.

Zur geometrischen Definition moderner Freiformprofile ist in dieser Arbeit ein neuer Ansatz mit periodischen B-Spline-Funktionen entwickelt worden. Die Profilkontur wird durch eine umlaufende Kurve beschrieben. Durch Approximations- und Interpolationsalgorithmen lassen sich vorhandene Profile in Form einer periodischen B-Spline-Funktion darstellen. Durch einen kombinierten Optimierungsansatz mittels Simulated Annealing lässt sich die Anzahl der B-Spline-Parameter reduzieren. Die Designparameter können geometrisch interpretiert und visualisiert werden. Mit Hilfe eines graphischen Editors lassen sich die Profile interaktiv entwerfen und manipulieren.

Zur aerodynamischen Bewertung ist ein Stromfunktionsverfahren entwickelt worden, welches die zweidimensionale, reibungsfreie, kompressible Strömung auf  $S_1$ -Flächen berechnet. Durch die Lösung einer elliptischen Differentialgleichung wird zuvor der Strömungsraum in Form eines C-Netzes diskretisiert. Durch ein neues Verfahren zur Bestimmung der Gitterknoten auf der Profiloberfläche durch Gewichtsfunktionen kann die Vorderkante besonders fein aufgelöst werden. Reibungssphänomene werden durch ein Grenzschichtverfahren modelliert. Die Koppelung mit dem Stromfunktionsverfahren geschieht durch eine simulierte Aufdickung des Profils und einer Verschiebung der Stromlinien entlang der Profiloberfläche. Die Leistungsfähigkeit des kombinierten Stromfunktions- und Grenzschichtverfahrens ist beispielhaft anhand zweier Verdichterprofile überprüft worden. Es zeigt sich eine gute Übereinstimmung mit experimentell bestimmten Ergebnissen.

Eine spätere Übertragung auf Radialmaschinen ist durch eine veränderte Definition einer Entwurfsebene der zweidimensionalen Profilschnitte möglich.

Zum schnellen Profilentwurf ist ein Modul auf der Basis neuronaler Netze entwickelt worden. Nach Aufbau einer umfangreichen Datenbasis lassen sich Profile durch Angabe geome-

trischer und aerodynamischer Randbedingungen unmittelbar („auf Knopfdruck“) erzeugen. Zusätzlich werden weitere Eigenschaften vorhergesagt. Durch Aufbau einer zweiten Datenbasis ist der Approximationsfehler des trainierten neuronalen Netzes statistisch betrachtet worden. Exemplarisch wurde eine Studie durchgeführt, um die Sensitivität und das Verhalten des Netzes auf veränderte Randbedingungen zu untersuchen.

Zukünftig wäre zu untersuchen, ob sich Fuzzy-Control-Ansätze, weitere, nicht betrachtete Methoden der *Computational Intelligence*, mit den neuronalen Netzen kombinieren lassen und dadurch ihre Aussagefähigkeit und Präzision weiter erhöhen können.

Vorhandene oder durch neuronale Netze erzeugte Profile lassen sich weiter optimieren. Hier ist ein evolutionäres Verfahren auf der Basis genetischer Algorithmen implementiert und auf Fähigkeit zur Minimierung einer Zielfunktion überprüft worden. Diese setzt sich aus geometrischen und aerodynamischen Zielvorgaben zusammen. Anhand von Beispielen des inversen und direkten Entwurfs werden die Optimierungseigenschaften genetischer Algorithmen beschrieben. Die Untersuchungen haben gezeigt, daß eine Erweiterung der abgebildeten Physik um mechanische Aspekte der Profilauslegung notwendig ist.

Zur weiteren Beschleunigung bietet es sich an, die genetischen Algorithmen zu parallelisieren und die Auswertung der Zielfunktion auf mehreren, verteilten Rechnern durchzuführen.

Die in dieser Arbeit umfangreich durchgeführten Untersuchungen zur Approximation mit neuronalen Netzen bieten ein großes Entwicklungspotential und stehen erst am Anfang einer umfassenden Anwendung bei der Auslegung von Turbomaschinen. In dieser Arbeit ist gezeigt worden, daß schon einfache Netze wesentliche Zusammenhänge und Abhängigkeiten von Entwurfsparametern erkennen können.

Inhalt weiterer Forschungsaktivitäten sollte in Hinblick auf eine weitestgehende Ersetzung der Strömungsberechnung durch neuronale Netze erfolgen. Die rechenintensive Strömungsrechnung ist somit nur zum Aufbau einer umfangreichen Trainingsdatenmenge notwendig. Solche Netze, gekoppelt mit Optimierungsstrategien, würden die Rechenzeit drastisch verkürzen; die Optimierung eines zweidimensionalen Profils Größenordnungen weniger Rechenzeit in Anspruch nehmen als die bisher konventionelle Optimierung mit mindestens einer durchzuführenden Strömungsberechnung pro Iterationsschritt.

Ein weiterer, zukünftiger Schritt ist die Approximation dreidimensionaler Strömungsphänomene mit neuronalen Netzen. Zum Aufbau einer Datenbasis wird ein dreidimensionaler Strömungslöser benötigt. Fernerhin ist eine Reduktion der Resultate auf einige Kenngrößen notwendig.

Diese Arbeit hat deutlich werden lassen, daß sowohl neuronale Netze als auch genetische Algorithmen vorteilhaft bei der Auslegung von Turbomaschinen eingesetzt werden können. Bereits heute läßt sich absehen, daß beide Ansätze sich zukünftig zu Standardmethoden des Entwurfs und der Optimierung entwickeln werden.

## Anhang A

# Algorithmen und Bildanhang

### A.1 Algorithmus von de Boor

Der Algorithmus in Abbildung A.1 implementiert die Gleichung (2.14) und dient zur Berechnung eines Kurvenpunktes  $B(t)$  einer B-Spline-Funktion für einen vorgegebenen Parameterwert  $t$ . Der Grundgedanke des Algorithmus ist es, nur die Anteile zu betrachten, die ungleich null sind. Im ersten Unterprogramm (Zeile 0–14) wird durch einen binären Suchalgorithmus der Index  $k$  bestimmt, so daß gilt:  $t \in [t_k, t_{k+1}]$ . Da der Knotenvektor  $\mathbf{t} = \{t_0, \dots, t_m\}$  per Definition aufsteigend sortiert ist, kann in maximal logarithmischer Zeit der Index  $k$  ermittelt werden. Zur Bestimmung der Basisfunktionen werden die in Abschnitt 2.2.2 aufgeführten Eigenschaften von B-Spline-Funktionen ausgenutzt. In den letzten Anweisungen (Zeile 31–35) werden nun die zuvor ermittelten Werte mit den entsprechenden Kontrollpunkten multipliziert und aufsummiert.

```

algorithm B-SPLINE-KURVENPUNKT
0 routine Binäre-Suche-k ( $t$ ) ⟨Bestimme  $k$  mit  $t \in [t_k, t_{k+1}]$ ⟩
1   if ( $t = t_{n+1}$ ) —>
2     return  $n$ 
3   fi
4    $low \leftarrow p$  ;  $high \leftarrow n + 1$  ;  $mid \leftarrow \lfloor (low + high)/2 \rfloor$ 
5   do ( $t < t_{mid} \vee t \geq t_{mid+1}$ ) —>
6     if ( $t < t_{mid}$ ) —>
7        $high \leftarrow mid$ 
8     else
9        $low \leftarrow mid$ 
10    fi
11     $mid \leftarrow \lfloor (low + high)/2 \rfloor$ 
12  od
13  return  $mid$ 
14 end

15 routine Basisfunktion ( $k, t$ ) ⟨Berechne die Basisfunktionen  $\neq 0$ ⟩
16    $N_0 \leftarrow 1$  ;  $j \leftarrow 1$ 
17   do ( $j \leq p$ ) —>
18      $left_j \leftarrow t - t_{k+1-j}$  ;  $right_j \leftarrow t_{k+j} - t$  ;  $saved \leftarrow 0$  ;  $r \leftarrow 0$ 
19     do ( $r < j$ ) —>
20        $temp \leftarrow N_r / (right_{r+1} + left_{j-r})$ 
21        $N_r \leftarrow saved + right_{r+1} * temp$ 
22        $saved \leftarrow left_{j-r} * temp$ 
23        $r \leftarrow r + 1$ 
24     od
25      $N_j \leftarrow saved$  ;  $j \leftarrow j + 1$ 
26   od
27   return  $N$ 
28 end

begin
29    $k \leftarrow$  Binäre-Suche-k( $t$ )
30    $N \leftarrow$  Basisfunktion( $k, t$ )
31    $C \leftarrow 0$  ;  $i \leftarrow 0$ 
32   do ( $i \leq p$ ) —>
33      $C \leftarrow C + N_i * \mathbf{P}_{k-p+i}$ 
34      $i \leftarrow i + 1$ 
35   od
end.

```

Abbildung A.1: Algorithmus von de Boor zur Berechnung von B-Spline Kurvenpunkten.

## A.2 Ausgewählte Wölbungs- und Dickenverteilungen

Der konventionelle Entwurf von zweidimensionalen Profilschnitten erfolgt durch die Definition einer Skeletlinie  $y_c$  und einer Dickenverteilung  $y_t$ . Die Skeletlinie beschreibt die Wölbung und teilt das Profil in der Mitte in eine obere  $(x_u, y_u)$  und untere Kontur  $(x_l, y_l)$  auf (vgl. auch KÄPPELI (1987)). Die Variable  $x \in [0, 1]$  beschreibt die auf die Sehnenlänge  $l$  bezogene Position. Ausgeföhrte Profile sind anschließend daher in  $x$  und  $y$  Richtung mit der Länge  $l$  zu skalieren. Die Profilkordinaten ergeben sich zu

$$x_u = x - y_t \sin \delta \quad x_l = x + y_t \sin \delta \quad (\text{A.1})$$

$$y_u = y_c + y_t \cos \delta \quad y_l = y_c - y_t \cos \delta \quad . \quad (\text{A.2})$$

Hierbei beschreibt  $\delta = \arctan y'_c$  den lokalen Steigungswinkel der Skeletlinie. Eine Staufelung entspricht der Drehung des Profils um den Winkel  $\beta$ . So ergeben sich die Punkte  $(x_{us}, y_{us})$  auf der Profiloberseite zu

$$x_{us} = x_u \cos \beta - y_u \sin \beta \quad y_{us} = x_u \sin \beta + y_u \cos \beta \quad . \quad (\text{A.3})$$

Ähnliche Profile werden in Profilfamilien zusammengefaßt. Hierzu gehören insbesondere die Profile der unterschiedlichen NACA Serien, die z.T. bereits um 1930 entstanden sind (vgl. ABBOTT UND VON DOENHOFF (1959)). Mit dem von LADSON ET AL. (1996) entwickelten Programm AIRFOLS lassen sich diverse NACA Wölbungs- und Dickenverteilungen kombinieren.

Trotz der eingeschränkten Parametrisierung bilden die nachfolgend aufgeführten Profilfamilien weiterhin die Grundlage zur Definition von Profilen mit Splinefunktionen. So lassen diese sich zum Beispiel als Startgeometrie in einem Optimierungsprozeß verwenden.

### Wölbungsverteilungen

**Kreisbogen** Die Skeletlinie beschreibt einen Kreisbogen mit einem Wölbungswinkel (Metallumlenkung) von  $2\alpha$ . Der Steigungswinkel beträgt an der Vorderkante  $\alpha$  bzw.  $-\alpha$  an der Hinterkante. Die Skeletlinie berechnet sich zu

$$y_c = y_0 + r \cos \arcsin \left( \frac{x - x_0}{r} \right) \quad , \quad (\text{A.4})$$

wobei der Kreismittelpunkt mit  $x_0 = 0.5$  und  $y_0 = -r \cos \alpha$  und der zugehörige Radius durch  $r = (2 \sin \alpha)^{-1}$  gegeben ist.

**Polynom** Analog zum Kreisbogen wird durch das folgende Polynom eine symmetrische Skeletlinie beschrieben:

$$y_c = c_0 x - (2c_0 + c_1)x^2 + (c_0 + c_1)x^3 \quad . \quad (\text{A.5})$$

$c_0 = \tan \alpha$  und  $c_1 = \tan -\alpha$  definieren die Steigungen an der Vorder- bzw. Hinterkante.

**NACA–4 Serie** Die häufig verwendete NACA–4 Serie *PMDD* wird durch

$$y_c = \begin{cases} \frac{p}{m^2}(2mx - x^2) & \text{für } 0 \leq x < m, \\ \frac{p}{(1-m)^2} [(1-2m) + 2mx - x^2] & \text{für } m \leq x \leq 1 \end{cases} \quad (\text{A.6})$$

beschrieben.  $p = 0.1 \cdot P$  gibt die maximale auf die Sehnenlänge bezogene Wölbung an.  $m = 0.1 \cdot M$  ist die Position der maximalen Wölbung (Wölbungsrücklage).

**NACA–16 Serie** Die aus der 6–Serie (vgl. LADSON ET AL. (1996)) abgeleitete Wölbungsverteilung wird über den Entwurfsauftriebsbeiwert  $c_l$  parametrisiert:

$$y_c = -\frac{c_l}{4\pi}(1-x)^2 \log(1-x) \quad . \quad (\text{A.7})$$

**NACA–65 Serie** Im Turbomaschinenbau wird häufig eine durch

$$y_c = -\frac{c_l}{4\pi} [(1-x) \log(1-x) + x \log(x)] \quad (\text{A.8})$$

definierte Skeletlinie verwendet.  $c_l$  beschreibt analog den Entwurfsauftriebsbeiwert.

### Dickenverteilungen

**Ellipse** Die einfachste Dickenverteilung wird durch eine Ellipse

$$y_t = d/2 \sin \arccos(2x - 1) \quad (\text{A.9})$$

beschrieben.  $d$  gibt die maximale Dicke (Dickenrücklage) an.

**C4 Serie** Die C4 Serie verwendet ursprünglich eine Kreisbogenskeletlinie. Die Dickenverteilung ergibt sich zu

$$y_t = A \frac{1.6435\sqrt{x} - 1.1153x - 0.6439x^2 + 0.2952x^3}{1 - 0.9336x} \quad , \quad (\text{A.10})$$

wobei  $A = d(1-x)$  gilt.

**NACA–4 Serie** Die zur Skeletlinie zugehörige Dickenverteilung der NACA–4 Serie ist durch

$$y_t = d(1,4845\sqrt{x} - 0.6300x - 1.7580x^2 + 1.4215x^3 - 0.5075x^4) \quad (\text{A.11})$$

definiert. Für die maximale Dicke gilt  $d = 0.01 \cdot DD$ .

**NACA–65 Serie** Durch

$$y_t = A \frac{1.0675\sqrt{x} - 0.2758x + 2.4478x^2 - 2.8385x^3}{1 - 0.1760x} \quad (\text{A.12})$$

wird mit  $A = d(1-x)$  die Dickenverteilung der NACA–65 Serie beschrieben.

### A.3 Konturpunktverteilung

Die nachfolgenden Abbildungen zeigen Konturverteilungen in Abhängigkeit der unterschiedlichen, in Abschnitt 3.2.4 vorgestellten, Gewichtsfunktionen  $\bar{w}_i$  am Beispiel einer Ellipse. Durch

$$x(t) = r_1 \cos(t) \quad \text{und} \quad y(t) = r_2 \sin(t) \quad (\text{A.13})$$

für  $t \in [-\pi, +\pi]$  wird eine Ellipse definiert. Der Umfang (Bogenlänge) ergibt sich zu:  $S \approx \pi[1.5(r_1 + r_2) - \sqrt{r_1 r_2}]$  (vgl. BRONŠTEJN UND SEMENDJAEV (1989)).

Für die folgenden Beispiele sind jeweils 40 Konturpunkte auf dem Umfang einer Ellipse mit  $r_1 = 0.95$  und  $r_2 = 0.15$  berechnet worden. Zur Verdeutlichung sind die Intervallgrenzen  $t_0$  bzw.  $t_1$  und die Bogenlänge mit den einzelnen Segmentlängen aufgeführt.

In Abbildung A.2 ist die *Exponential*–Gewichtsfunktion dargestellt. Auf der Hälfte der Bogenlänge ist ein Bezugspunkt  $t'$  gewählt, wodurch die Segmente mit  $t > t'$  bzw.  $t < t'$  zunehmend größer werden.

Die *Euklidscher Abstand*–Gewichtsfunktion definiert einen Kontraktionspunkt, wie in Abbildung A.3 skizziert, der an einer beliebigen Position in der  $x, y$ –Ebene, jedoch nicht auf der Kurve, liegen kann. Im Gegensatz zur *Exponential*–Gewichtsfunktion berechnet sich die Segmentlänge nicht aus dem Abstand entlang der Bogenlänge, sondern durch den Abstand zu diesem Punkt. Durch die interaktive Manipulation des Kontraktionspunktes steht eine komfortable Methode zur Verfügung, um eine gewünschte Konturpunktverteilung zu erreichen.

Das Ergebnis einer Kombination mehrerer Funktionen ist in Abbildung A.4 wiedergegeben. Das Produkt dieser drei Gewichtsfunktionen entspricht dem in dieser Arbeit implementierten Verfahren zur Berechnung der Konturpunktverteilung mit dem Gittergenerator **muspider**.

In den Abbildungen A.5 und A.6 ist die *Binär*–Gewichtsfunktion dargestellt, die sicherstellt, daß in einem definierten Intervall die Abstände benachbarter Konturpunkte äquidistant sind.

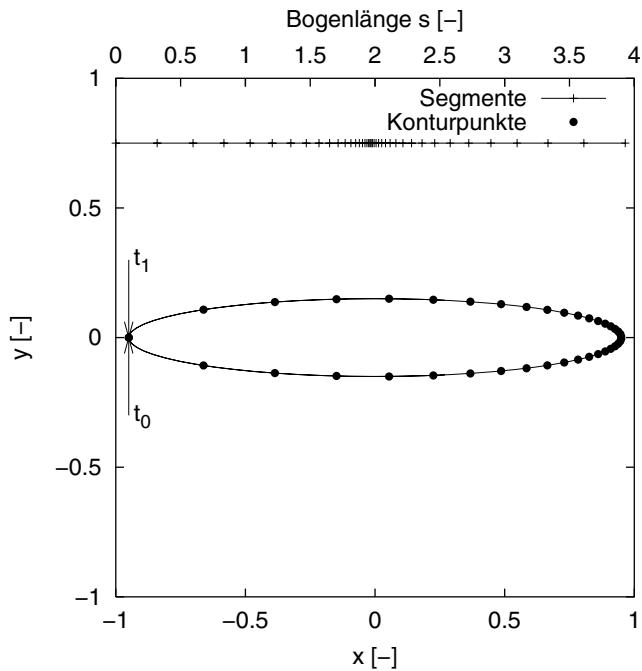


Abbildung A.2: Für die Gewichtsfunktion *Exponential* ( $\alpha = 0.1, \lambda = 0.1$ ) wurde ein Bezugspunkt  $t'$  mit  $s(t') = 0.5(s(t_0) + s(t_1))$  gewählt.

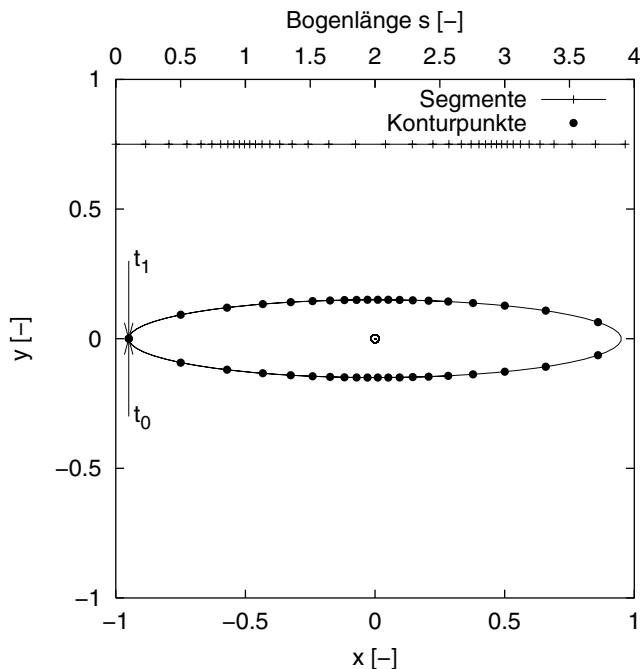


Abbildung A.3: Bei der Gewichtsfunktion *Euklidscher Abstand* befindet sich an der Position  $(0, 0)$  ein Kontraktionspunkt.

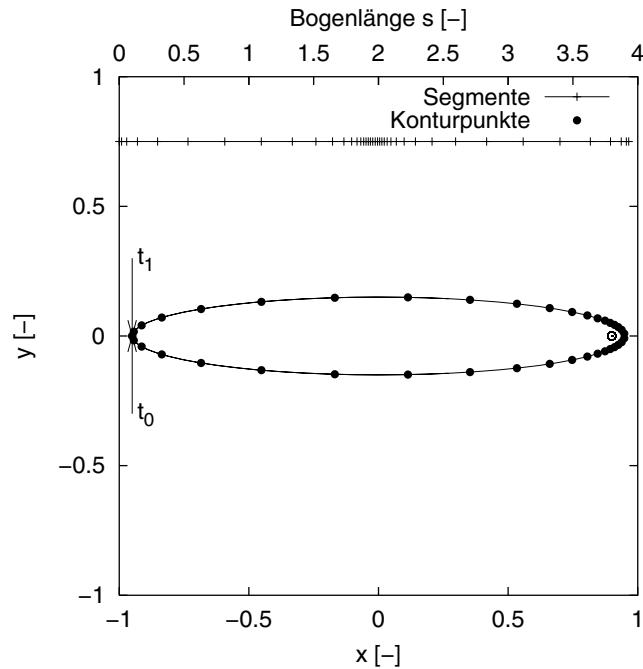


Abbildung A.4: Das Produkt zweier Gewichtsfunktionen *Exponential* ( $\alpha = 0.2, \lambda = 1$ ) und einer Gewichtsfunktion *Euklidscher-Abstand* führt zu drei Kontraktionspunkten.

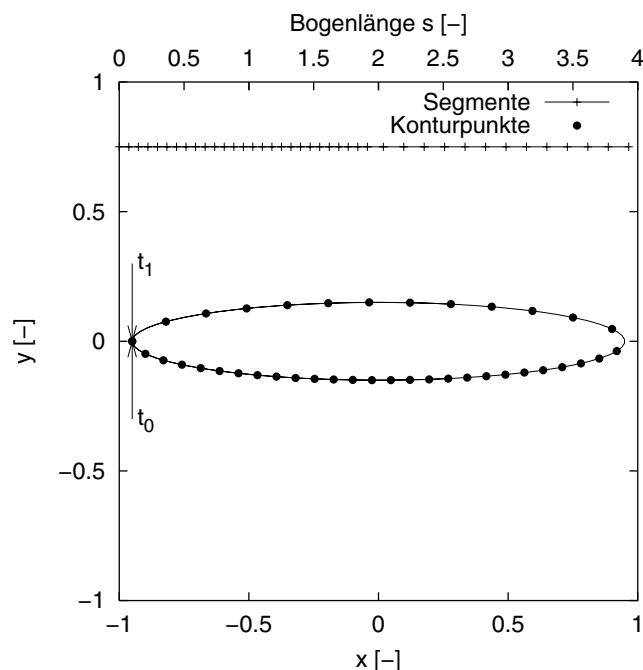


Abbildung A.5: Bei der Gewichtsfunktion *Binär* ist für  $t \in [-\pi, 0]$  ein Gewicht mit 0.5 verwendet. Die Segmente der Unterseite sind halb so lang wie die der Oberseite.

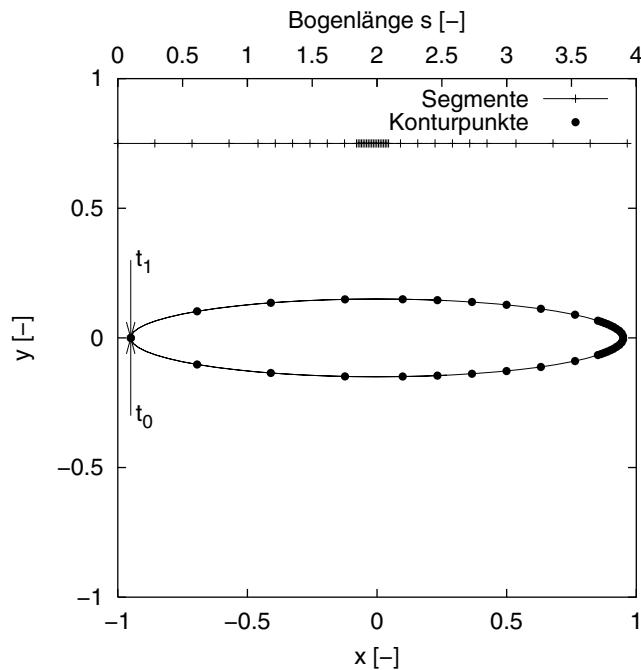


Abbildung A.6: Durch die Überlagerung zweier Gewichtsfunktionen *Binär* mit  $t \in [-\frac{\pi}{2}, +\frac{\pi}{2}]$  und einem Gewicht 0.5 bzw.  $s(t) \in [0.45 * (s(t_0) + s(t_1)), 0.55 * (s(t_0) + s(t_1))]$  und einem Gewicht 0.25 entstehen drei Bereiche mit unterschiedlichen Segmentlängen.

#### A.4 Profilentwurf mit neuronalen Netzen

In den nachfolgenden Abbildungen werden weitere Ergebnisse der in Abschnitt 4.4.3 besprochenen Variation zur Untersuchung der Approximation mittels neuronaler Netze dargestellt. Soweit nicht anders angegeben, ist jeweils in der oberen Hälfte die Konfiguration 1, in der unteren Hälfte zum Vergleich die zweite Testkonfiguration aufgeführt.

Es zeigt sich, daß die Approximation durch das neuronale Netz sich *wohldefiniert*, d.h. im weiteren Sinne stetig, verhält. Dieses ist insbesondere in Hinblick auf eine mögliche Optimierung wichtig, da rauhe Suchräume i.allg. die Optimierungsstrategien dazu verleiten, in lokalen und nicht globalen Extremen zu verweilen.

Die Erfüllung einer geforderten Umlenkaufgabe ist eine zentrale Forderung an das Profil im Auslegungsprozeß. In den Abbildungen A.7 bis A.9 ist der geforderte und tatsächliche Abströmwinkel in Abhängigkeit der Zuströmmachzahl, des Teilungs- und Dickenverhältnis aufgeführt. Die Differenz der Winkel ist i.allg. geringer als  $2^\circ$ .

In diesem Zusammenhang sind insbesondere die Resultate in Abbildung A.10 interessant, die dokumentieren, daß die Tendenz bei der Variation des geforderten Abströmwinkels sehr gut wiedergegeben wird. Das neuronale Netz hat also die Fähigkeit durch Training erlernt, auf eine veränderte aerodynamische Belastung des Profils zu reagieren und die Geometrie dementsprechend anzupassen.

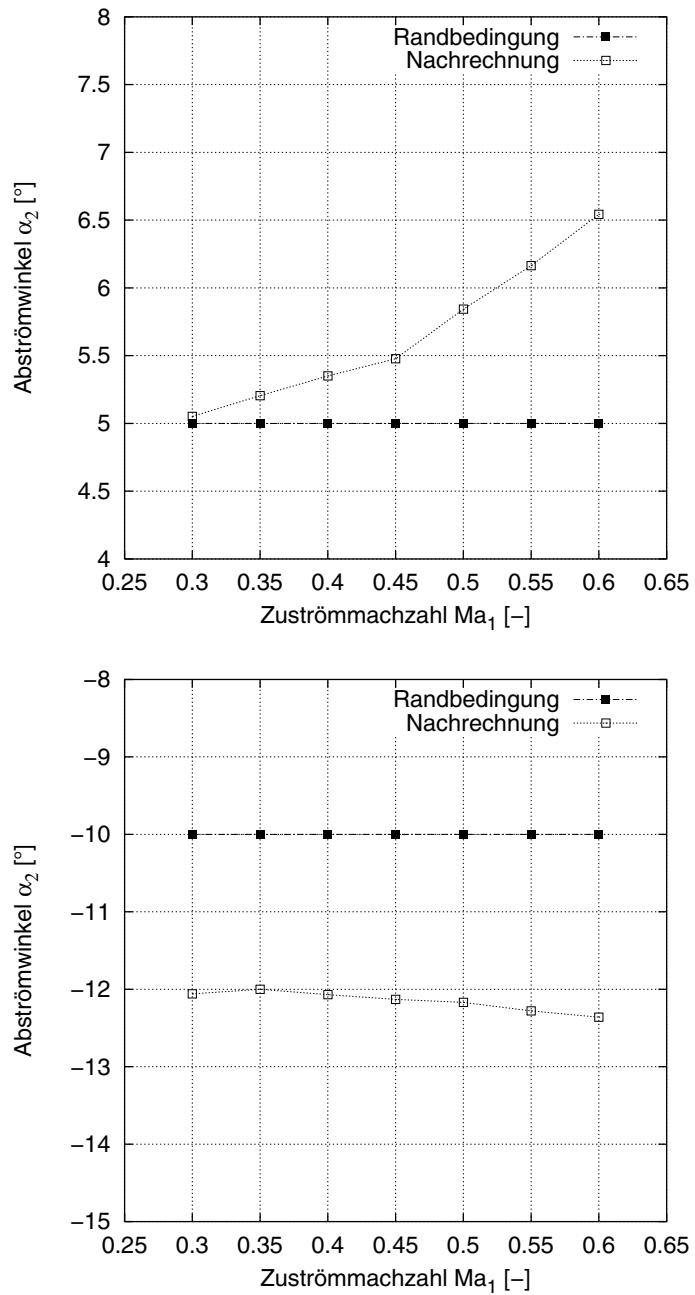


Abbildung A.7: Die maximale Abweichung zwischen der aerodynamischen Randbedingung des Abströmwinkels  $\alpha_2$  und dem tatsächlichen Winkel beträgt bei einer systematischen Variation der Zuströmmachzahl  $Ma_1$  zwischen  $1^\circ$  und  $2^\circ$ .

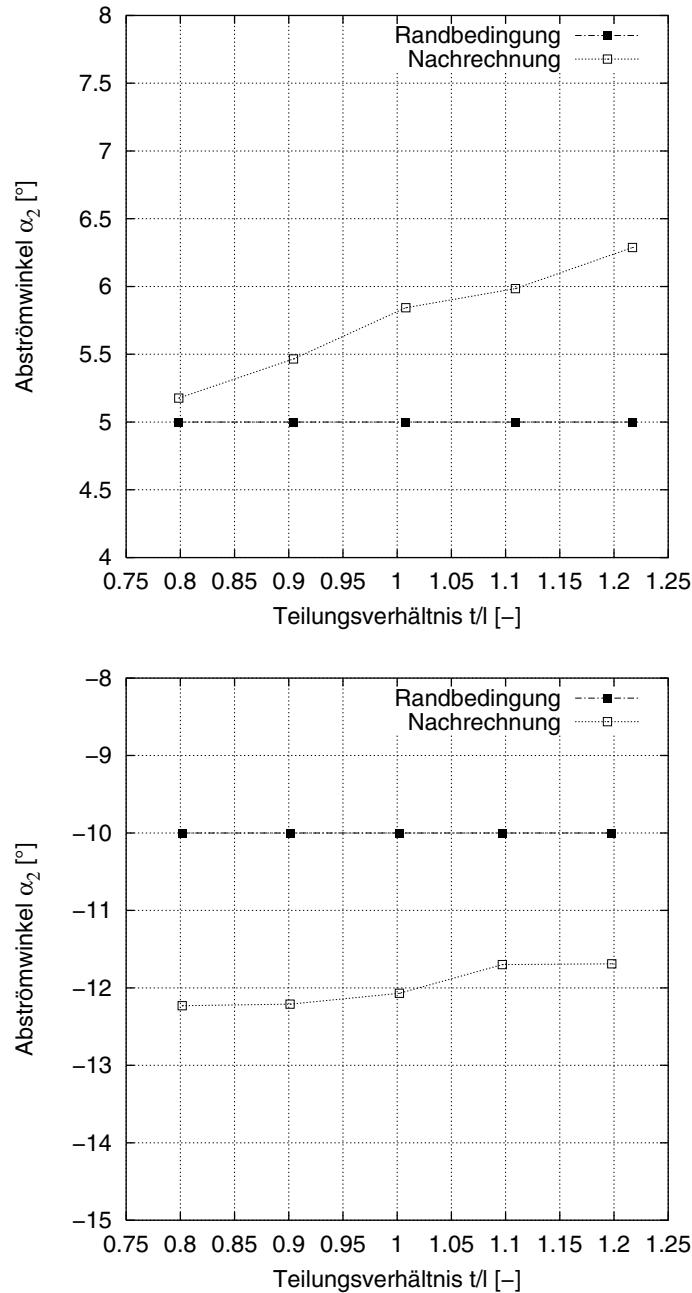


Abbildung A.8: Das Ergebnis der Variation des Teilungsverhältnisses  $t/l$  zeigt eine Abweichung vom geforderten Abströmwinkel  $\alpha_2$  zwischen  $1^\circ$  und  $2^\circ$ .

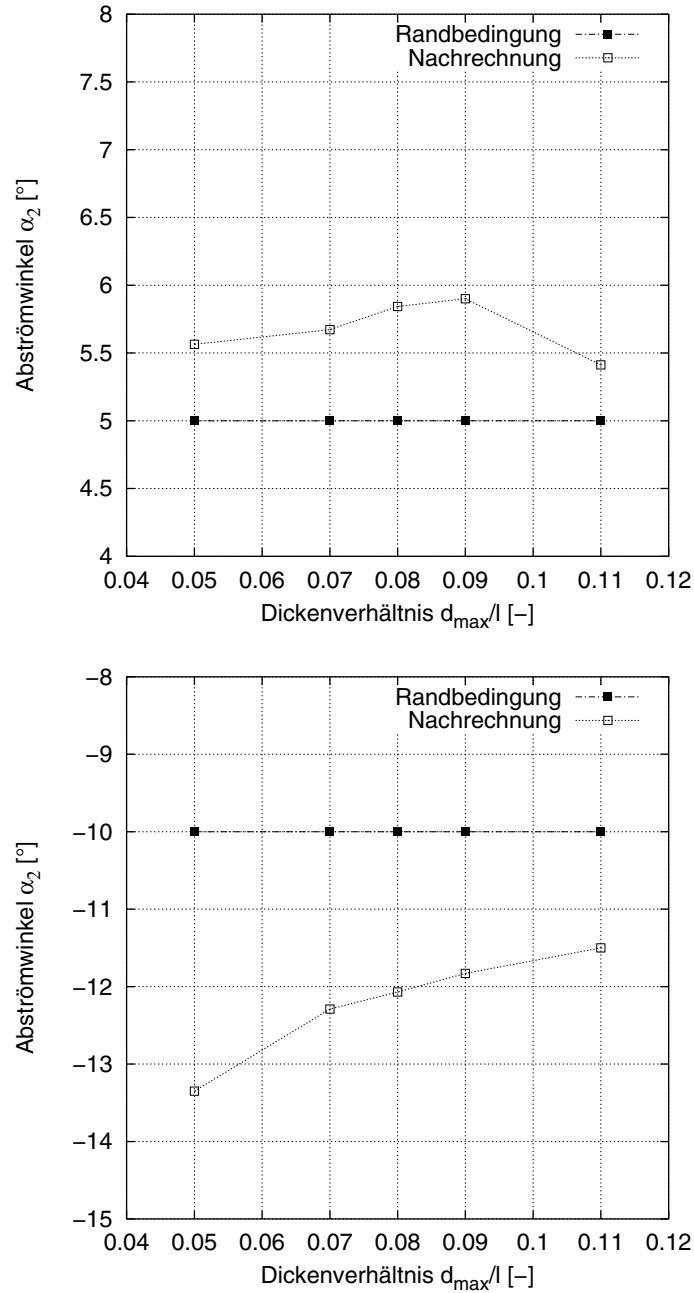


Abbildung A.9: Während bei der ersten Konfiguration eine Variation des Dickenverhältnisses  $d_{\max}/l$  wenig Einfluß auf den Fehler des Abströmwinkels hat, ist im zweiten Fall dieser für besonders dünne Profile größer als  $2^\circ$ .

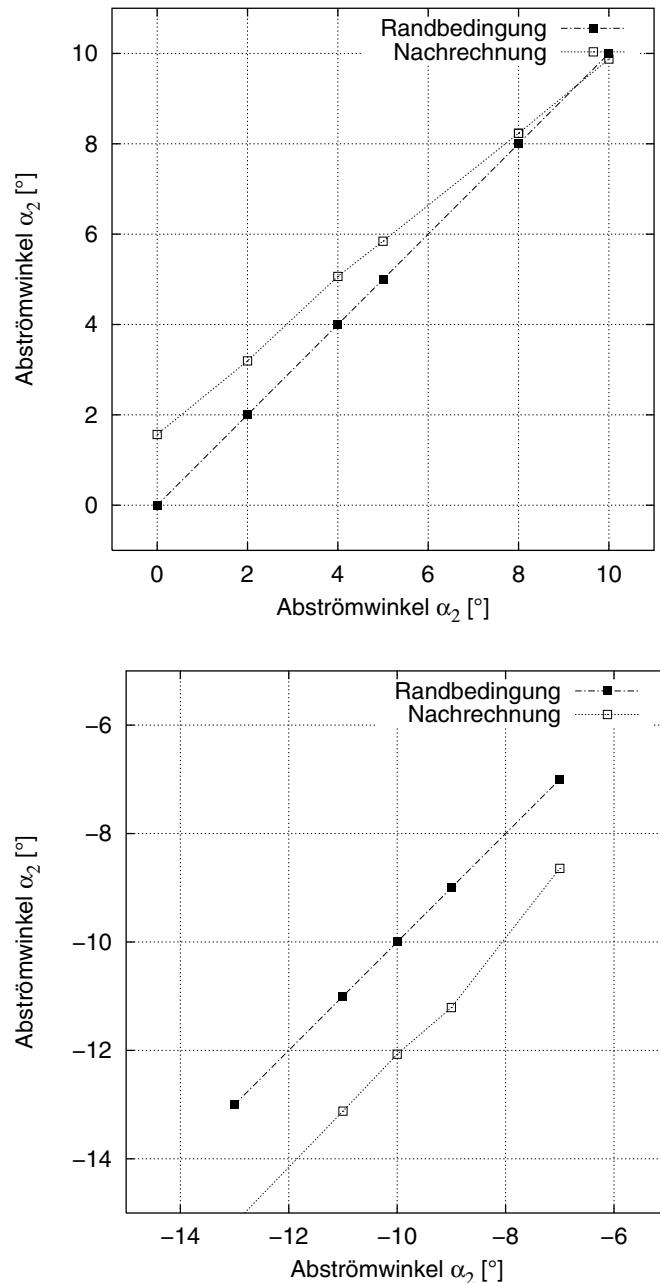


Abbildung A.10: Bei einer Veränderung der Randbedingung des Abströmwinkels  $\alpha_1$  bleibt die Abweichung zum tatsächlichen Wert kostant bzw. wird sogar geringer.

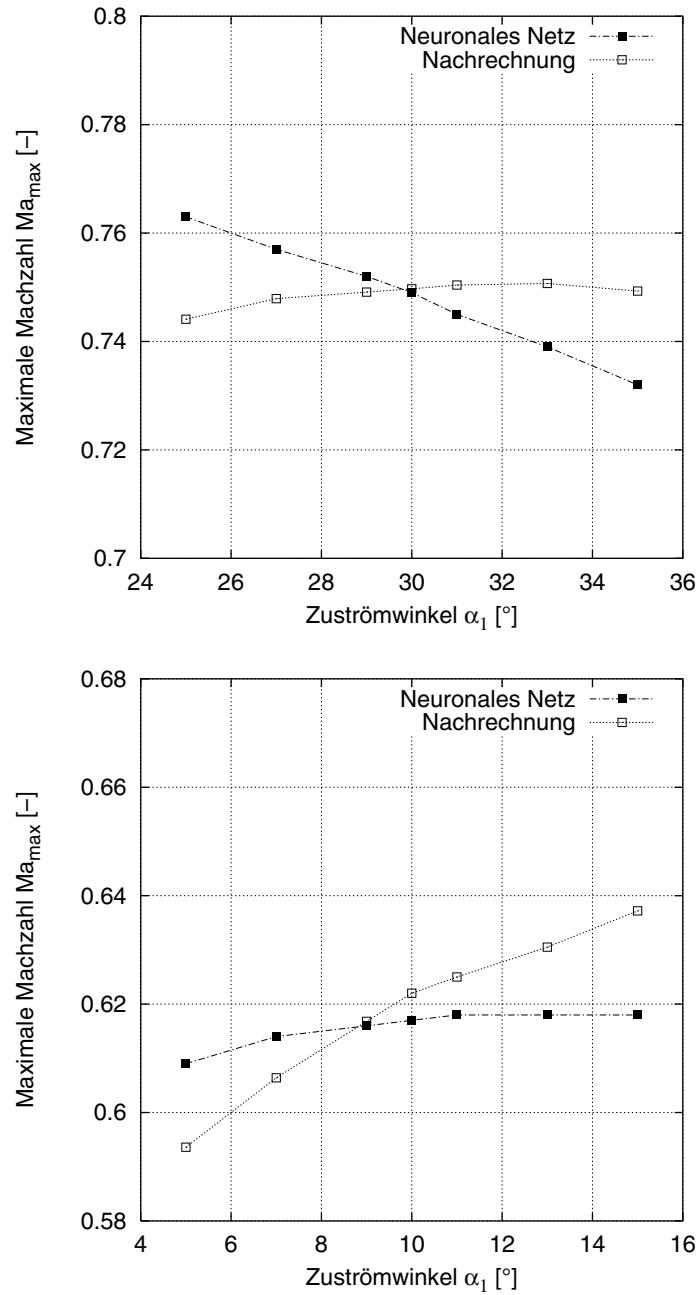


Abbildung A.11: Wird der Zuströmwinkel  $\alpha_1$  variiert, so ergibt sich beim Vergleich zwischen der tatsächlichen maximalen Machzahl  $Ma_{max}$  und dem vom neuronalen Netz vorhergesagten Wert eine geringe Abweichung von  $\approx 0.02Ma$ .

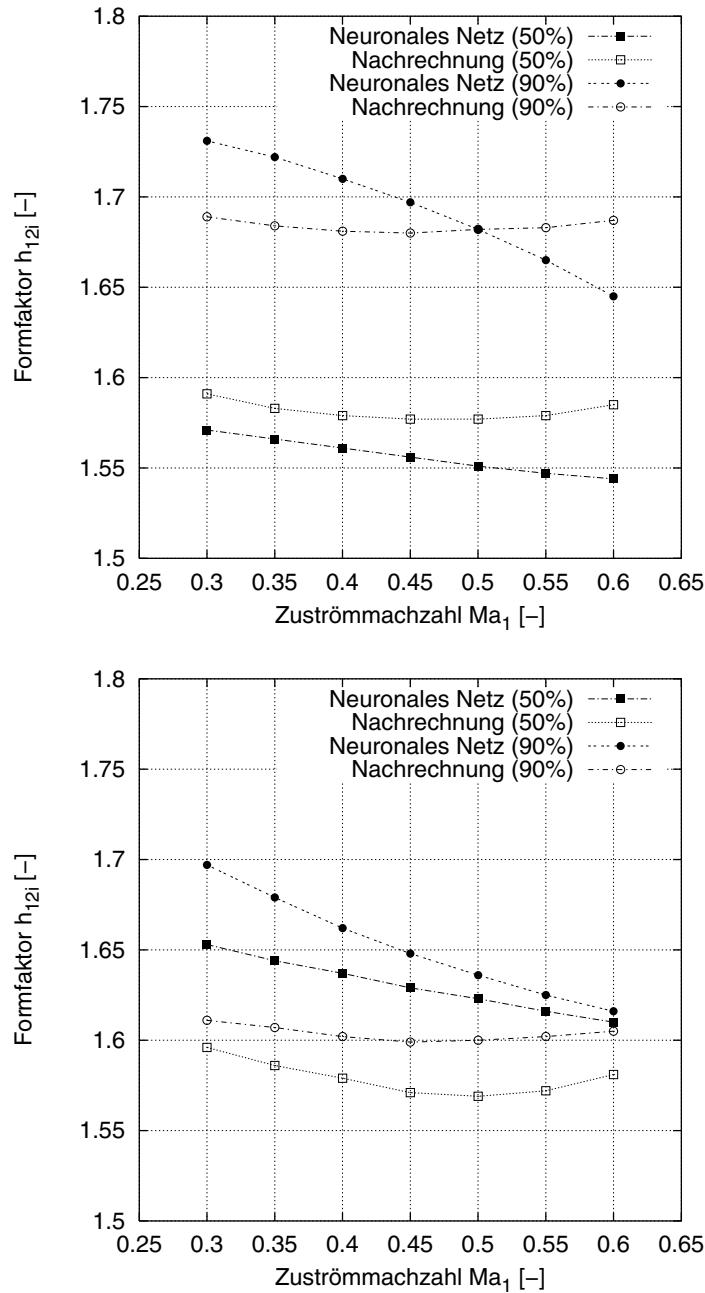


Abbildung A.12: Die Abweichungen zwischen dem prognostizierten und tatsächlichen Formfaktor  $h_{12i}$  sind bei einer Variation der Zuströmmachzahl  $Ma_1$  bei der ersten Konfiguration wesentlich geringer als im zweiten Fall.

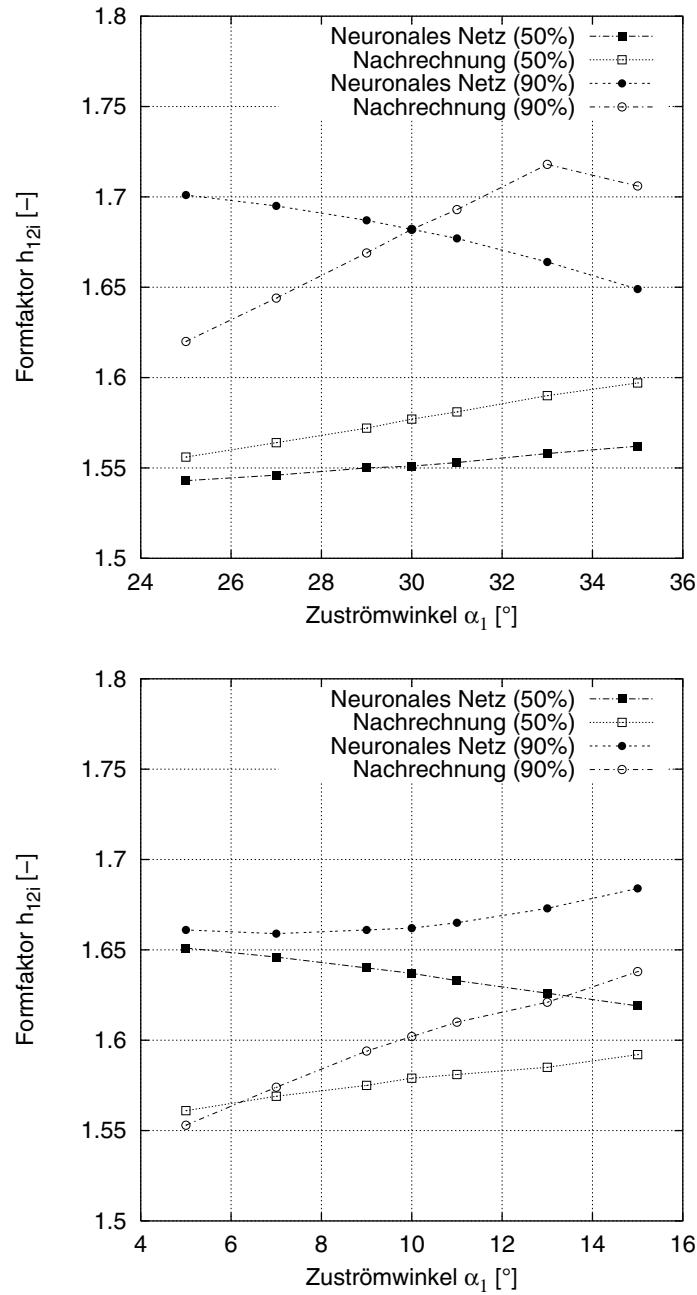


Abbildung A.13: Das vom neuronalen Netz für einen Zuströmwinkel  $\alpha_1 = 35^\circ$  berechnete Profil zeigt an der Hinterkante eine Ablösung der Grenzschicht. Die Formfaktoren für die zweite Konfiguration werden zu groß prognostiziert.

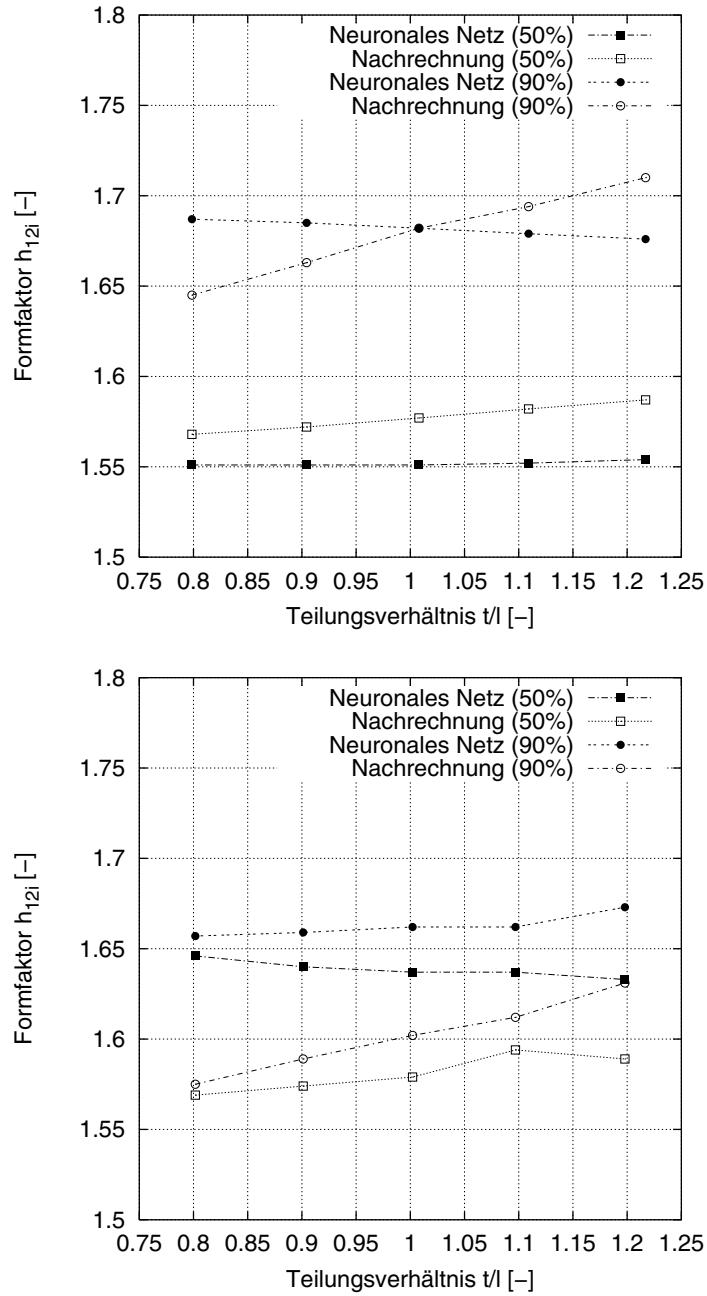


Abbildung A.14: Wie bei den vorherigen Variationen zeigt sich, daß das neuronale Netz für die Konfiguration 1 bessere prognostizierte Werte für die Belastung der Grenzschicht ermittelt.

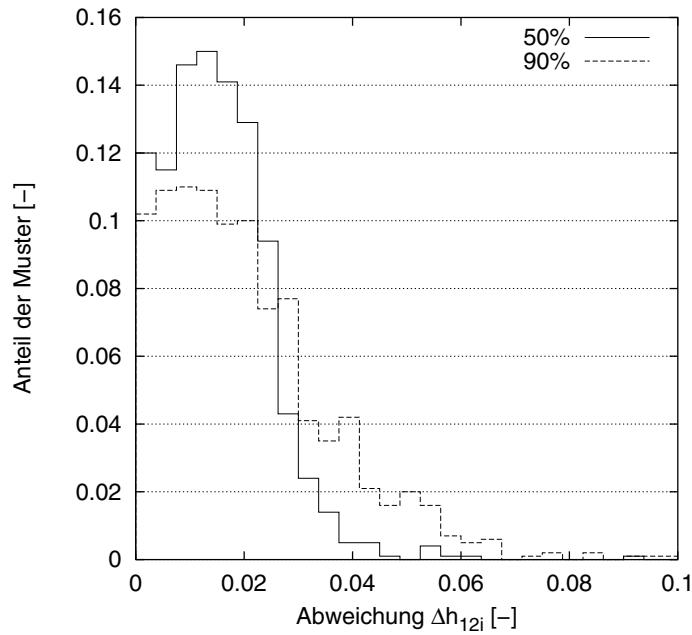


Abbildung A.15: Die Histogrammdarstellung für den Approximationsfehler der Trainingsmenge für den Formfaktor  $h_{12i}$  zeigt einen deutlichen Unterschied in Abhängigkeit der chordalen Position (Skalierungsmethode I).

In den Abbildungen A.11 bis A.14 ist die Prognosefähigkeit bezüglich der maximalen Machzahl  $Ma_{\max}$  bzw. der Grenzschichtbelastung, beschrieben durch den Formfaktor  $h_{12i}$ , dargestellt. Bei einer Variation des Zuströmwinkels ist der Fehler zwischen prognostiziertem und tatsächlichem Maximum der Machzahl sehr gering ( $< 0.02Ma$ ).

Die Prognose der Grenzschichtbelastung auf der Saugseite bei einer Variation der Zuströmmachzahl, des Zuströmwinkels und des Teilungsverhältnisses zeigt z.T. deutliche Abweichungen (vgl. auch Abbildung A.15). Während die Tatsache, daß die Belastung an der Hinterkante i.allg. größer als auf der Hälfte des Profils ist, durch das neuronale Netz korrekt erlernt worden ist, werden Tendenzen z.T. gegenläufig wiedergegeben. Weitere Entwicklungen sollten daher eine verbesserte Approximation des Grenzschichtverhaltens durch das neuronale Netz, z.B. durch eine größere Trainingsmenge, in Erwägung ziehen.

In Abbildung A.16 ist an zwei weiteren Beispielen die Fähigkeit demonstriert, auf veränderte Randbedingungen angemessen zu reagieren. Bei einer Variation des Zuströmwinkels ist der Fehler eines unveränderten Ausgangsprofils bezüglich der geforderten chordalen Position der maximalen Machzahl groß (bis zu 0.08). Hingegen reagiert das neuronale Netz auf die Veränderung von  $\alpha_1$  und versucht, die Forderung durch eine Modifikation des Profils zu erreichen. Ein ähnliches Verhalten wird auch am Beispiel einer geforderten Machzahl  $Ma_{2/3}$  deutlich.

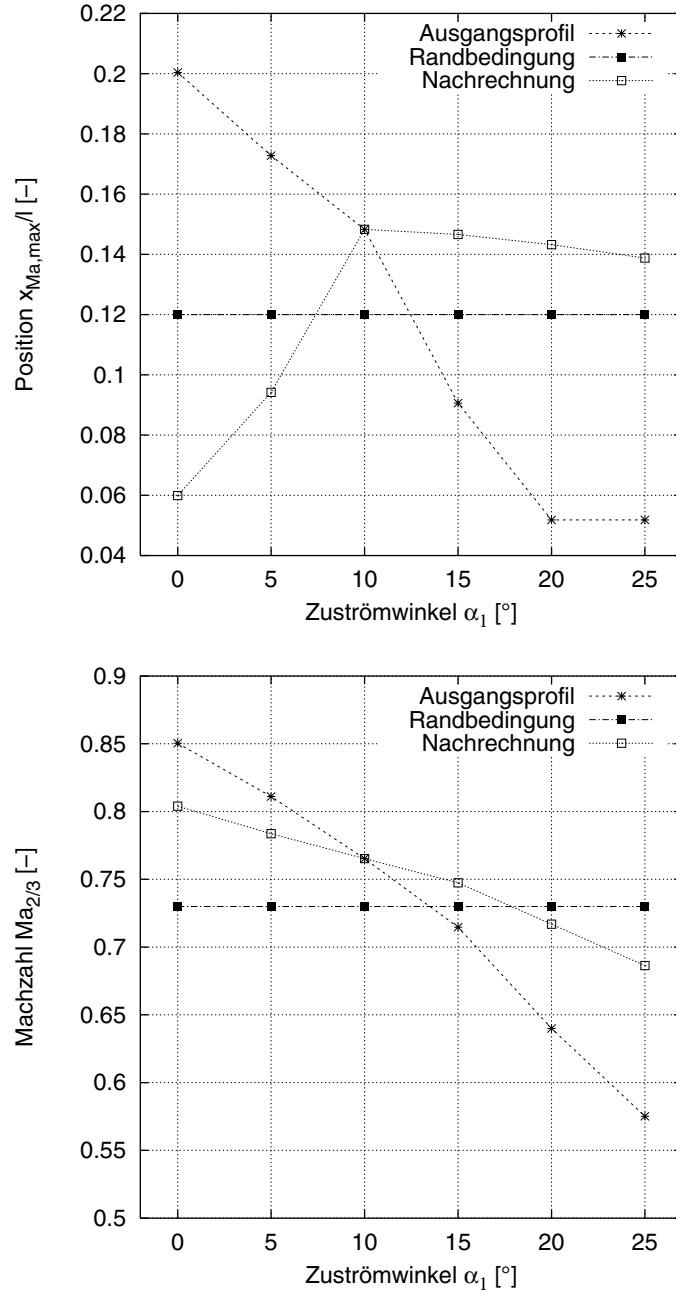


Abbildung A.16: Das neuronale Netz hat die Fähigkeit erlernt, das Ausgangsprofil bei einer Variation des Zuströmwinkels  $\alpha_1$  so zu verändern, daß die Abweichung zur vorgegebenen Randbedingung  $x_{Ma,\max}$  bzw.  $Ma_{2/3}$  verringert wird.

## A.5 Approximation mit Simulated Annealing

In den Abbildungen A.17 bis A.20 sind weitere Ergebnisse der Approximation vorhandener Profile mit periodischen B-Spline-Funktionen anhand dreier Beispiele dargestellt. Zur Reduzierung der notwendigen Parameter ist die Approximation mit einer Simulated Annealing Optimierung gekoppelt (vgl. Abschnitt 5.2.3). Der mittlere  $\bar{E}^*$  und der maximale Fehler  $E_{\max}^*$  ist nach den Gleichungen (2.28) und (2.29) definiert.

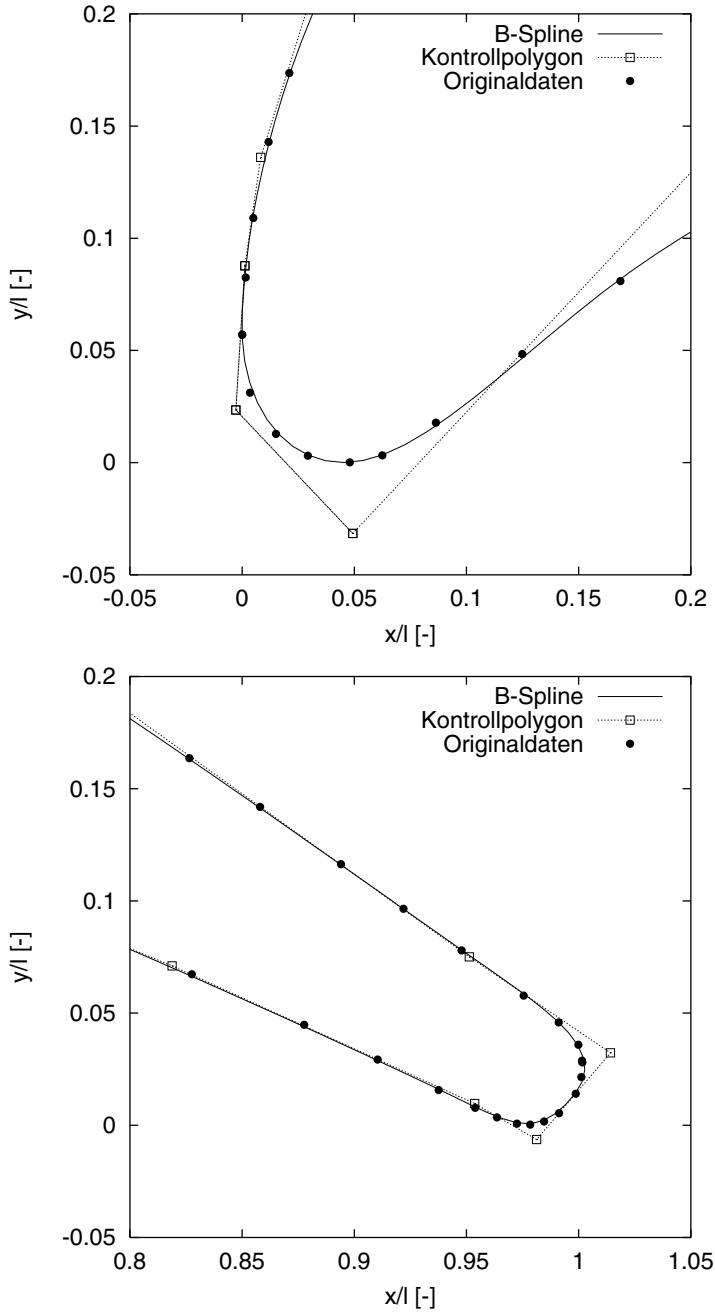


Abbildung A.17: Die Approximation des Profils aus WATZLAWICK (1991) liefert einen mittleren Fehler  $\bar{E}^* = 5.12 \cdot 10^{-4}$  und einen maximalen Fehler  $E_{\max}^* = 1.39 \cdot 10^{-3}$ . Als Parameter für den Simulated Annealing Prozeß sind  $c_f = 0.6$ ,  $N_{\text{eq}} = 70$  sowie  $T_{\text{start}}/f(x_{\text{start}}) = 0.1$  verwendet worden.

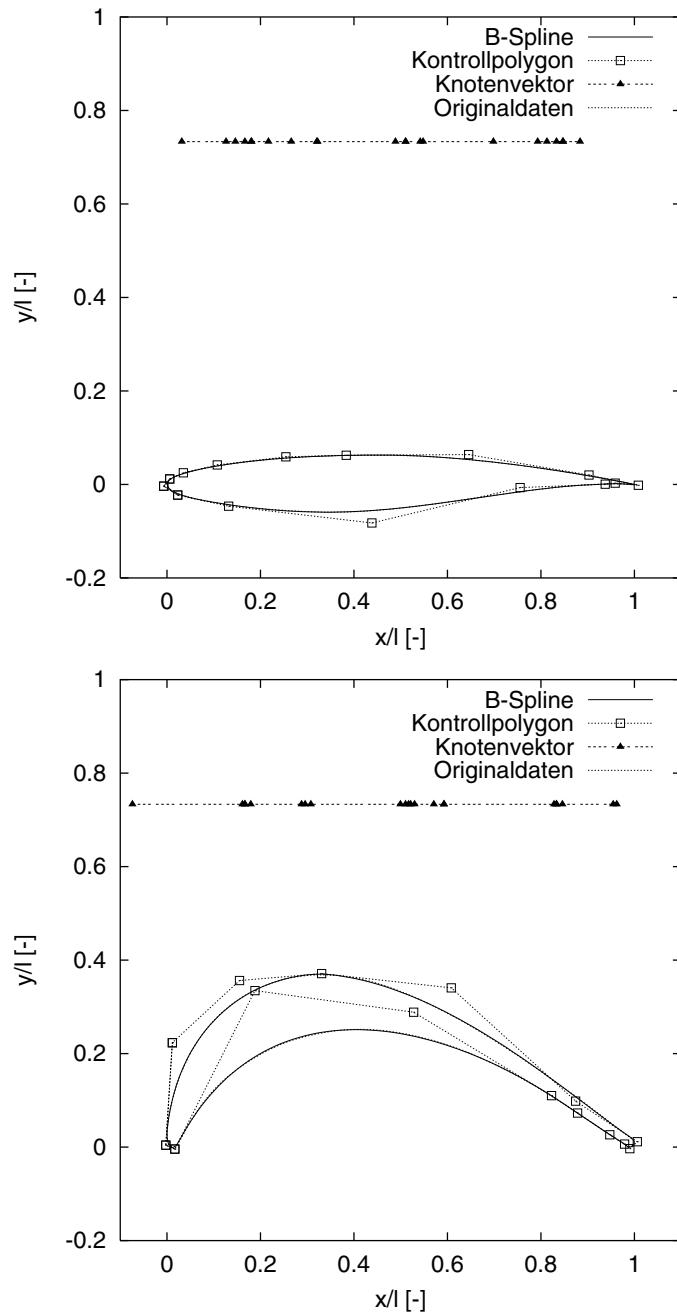


Abbildung A.18: In zwei weiteren Untersuchungen ist das Profil RAE2822 sowie das Turbinenprofil T106 durch einen periodischen B-Spline approximiert worden.

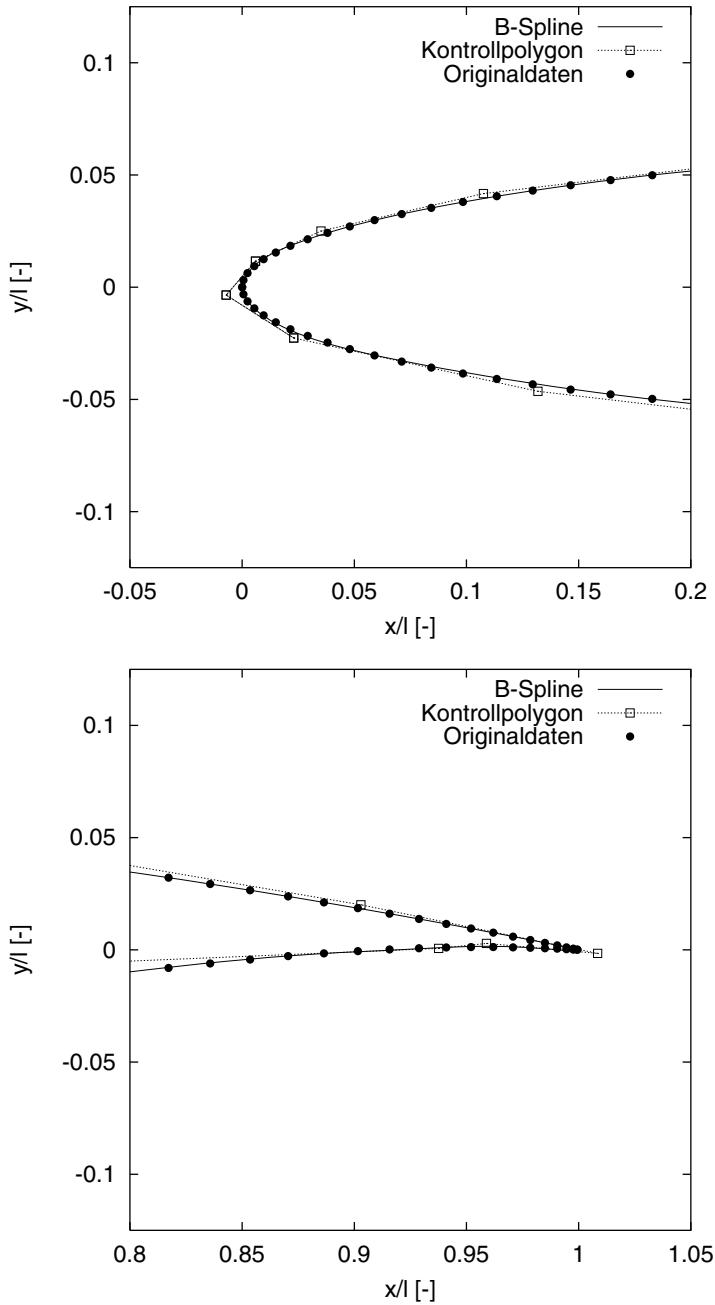


Abbildung A.19: Die Approximation hat einen mittleren Fehler  $\bar{E}^* = 1.91 \cdot 10^{-4}$  und einen maximalen Fehler  $E_{\max}^* = 5.89 \cdot 10^{-4}$ . Die spitze Hinterkante des Profils wird gut approximiert.

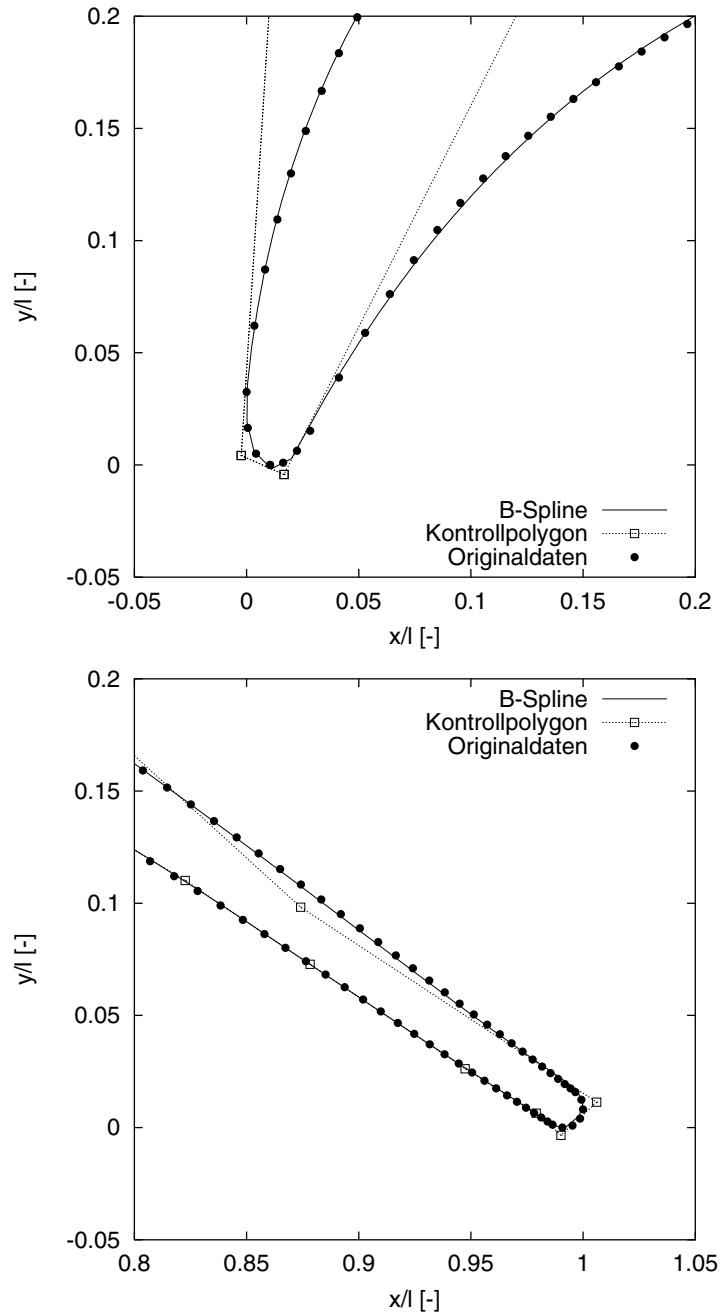


Abbildung A.20: Die Approximation des Turbinenprofils T106 hat mit  $\bar{E}^* = 7.11 \cdot 10^{-5}$  bzw.  $E_{\max}^* = 1.92 \cdot 10^{-4}$  einen geringen Fehler.



# Literaturverzeichnis

- AARTS, EMILE UND JAN KORST (1989). *Simulated Annealing and Boltzmann Machines*. John Wiley & Sons.
- ABBOTT, IRA H. UND ALBERT E. VON DOENHOFF (1959). *Theory of Wing Sections*. Dover Publications, New York.
- ABLAY, PAUL (1987). Optimieren mit Evolutionsstrategien. *Spektrum der Wissenschaft*, 7:104–115.
- AMES, WILLIAM F. (1977). *Numerical Methods for Partial Differential Equations*. Academic Press, New York, San Francisco, zweite Auflage.
- ANDERSON, JAMES A., ANDRAS PELLIONISZ UND EDWARD ROSENFELD, Herausgeber (1995). *Neurocomputing 2: Directions for Research*. MIT Press, Cambridge.
- ANDERSON, JAMES A. UND EDWARD ROSENFELD, Herausgeber (1988). *Neurocomputing: Foundations of Research*. MIT Press, Cambridge.
- ANDERSON, JOHN D. (1991). *Fundamentals of Aerodynamics*. McGraw–Hill, zweite Auflage.
- ANDERSON, JOHN D., GERARD DEGREZ, ERIK DICK UND ROGER GRUNDMANN (1996). *Computational Fluid Dynamics*. Springer–Verlag.
- AU, CHI-KIT UND MATTHEW M. F. YUEN (1995). Unified Approach to NURBS Curve Shape Modification. *Computer–Aided Design*, 27(2):85–93.
- BAIER, RALF-DIETMAR (1998). EVOTECH — Einsatz der Evolutionsstrategie in Wissenschaft und Technik, Teilprojekt Formoptimierung von Turbinenschaufeln mit evolutionären Algorithmen. Abschlußbericht, MTU Abt. TPAT Turbinenaerodynamik, Daimler–Benz Aerospace München.
- BARSKY, BRIAN A. (1988). *Computer Graphics and Geometric Modeling Using Beta-Splines*. Springer Verlag.
- BARSKY, BRIAN A. (1993). Rational Beta–Splines for Representing Curves and Surfaces. *IEEE Computer Graphics and Applications*, Seiten 24–32.

- BEDI, S. UND G. W. VICKERS (1989). Surface Lofting and Smoothing with Skeletal–Lines. *Computer Aided Geometric Design*, 6:87–96.
- BLÜMECKE, THOMAS (1991). Wunder der Evolution. *Magazin für Computertechnik (c't)*, 12:228–235.
- BOEHM, WOLFGANG (1987). Bézier Presentation of Airfoils. *Computer Aided Geometric Design*, 4:17–22.
- BÖHME, GERT (1990). *Algebra*. Springer–Verlag, 6. Auflage.
- BOTHE, HANS-HEINRICH (1998). *Neuro–Fuzzy–Methoden: Einführung in die Theorie und Anwendungen*. Springer–Verlag Berlin.
- BRATLEY, PAUL, BENNETT C. FOX UND LINUS E. SCHRAGE (1983). *A Guide to Simulation*. Springer–Verlag New York.
- BRONŠTEJN, IL'JA N. UND KONSTANTIN A. SEMENDJAEV (1989). *Taschenbuch der Mathematik*. Verlag Harri Deutsch, Thun und Frankfurt/Main, 24. Auflage.
- CARSTENS, VOLKER (1988). Two–Dimensional Elliptic Grid Generation for Airfoils and Cascades. Forschungsbericht DFVLR–FB 88–52, Deutsche Forschungsanstalt für Luft– und Raumfahrt (DLR).
- CASEY, M. V. (1994). Computational Methods for Preliminary Design and Geometry Definition in Turbomachinery. In *Turbomachinery Design using CFD (La Conception des Turbomachines par l'Aérodynamique Numérique)*, Nummer 195 in AGARD Lecture Series, Seiten 1–22. North Atlantic Treaty Organization (NATO).
- CEBECI, TUNCER UND PETER BRADSHAW (1988). *Physical and Computational Aspects of Convective Heat Transfer*. Springer–Verlag, New York.
- COFER IV, JOHN I. (1995). Advances in Steam Path Technology. In *POWER–GEN Europe '95*, Seiten 303–351. Amsterdam.
- COLES, D. E. UND E. A. HIRST (1969). Proceedings of Turbulent Boundary Layers. In *AFOSR–IFP–Stanford Conference, 18–25 Aug. 1968*, Band II (Compiled Data). Department of Mechanical Engineering, Stanford University, California.
- DASGUPTA, DIPANKAR UND ZBIGNIEW MICHALEWICZ (1997a). *Evolutionary Algorithms — An Overview*, Seiten 3–28. In DASGUPTA UND MICHALEWICZ (1997b).
- DASGUPTA, DIPANKAR UND ZBIGNIEW MICHALEWICZ, Herausgeber (1997b). *Evolutionary Algorithms in Engineering Applications*. Springer–Verlag Berlin Heidelberg.
- DE FALCO, I. (1997a). An Introduction to Evolutionary Algorithms and their Application to the Aerofoil Design Problem — Part I: The Algorithms. In VAN DEN BRAEMBUSSCHE UND MANNA (1997).

- DE FALCO, I. (1997b). An Introduction to Evolutionary Algorithms and their Application to the Aerofoil Design Problem — Part II: The Results. In VAN DEN BRAEMBUSSCHE UND MANNA (1997).
- DEUFLARD, PETER UND ANDREAS HOHMANN (1993). *Numerische Mathematik I — Eine algorithmisch orientierte Einführung*. Walter de Gruyter.
- DOORLY, DENIS (1995). Parallel Genetic Algorithms for Optimisation in CFD. In WINTER ET AL. (1995), Seiten 251–270.
- DOU, H.-S. UND N. PHAN-THIEN (1998). On the Scalability of Parallel Computations on a Network of Workstations. *Computational Mechanics*, 22:344–354.
- DRELA, MARK UND MICHAEL B. GILES (1987). ISES: A Two-dimensional Viscous Aerodynamic Design And Analysis Code. AIAA Paper 86-0424, American Institute of Aeronautics and Astronautics.
- DREYER, THOMAS, VOLKER SCHULZ UND HANS GEORG BOCK (1995). Ein mathematisches Verfahren zur schnellen Formoptimierung von Turbinenschaufeln. Schlussbericht AG-TURBO Interimsphase, Projekt 1.110, Interdisziplinäres Zentrum für Wissenschaftliches Rechnen, Universität Heidelberg.
- DREYFUS, HUBERT L. UND STUART E. DREYFUS (1996). *Einen Geist bauen gegen ein Gehirn modellieren: Künstliche Intelligenz wieder an der Abzweigung*, Seiten 15–43. In GRAUBARD (1996).
- DUECK, GUNTER (1993). New Optimization Heuristics — The Great Deluge Algorithm and the Record-to-Record Travel. *Journal of Computational Physics*, Seiten 86–92.
- DUECK, GUNTER UND TOBIAS SCHEUER (1990). Threshold Accepting: A General Purpose Optimization Algorithm Appearing Superior to Simulated Annealing. *Journal of Computational Physics*, Seiten 161–175.
- EGARTNER, WOLFGANG UND VOLKER SCHULZ (1998). Partially Reduced SQP Methods for Optimal Turbine and Compressor Blade Design. In H. G. Bock, G. Konschat, R. Rannacher, F. Brezzi, R. Glowinski, Y. A. Kuznetsov und J. Periaux, Herausgeber, *ENUMATH 97, Proceedings of the 2nd European Conference on Numerical Mathematics and Advanced Applications*, Seiten 286–293. World Scientific Publishers.
- FALLER, ADOLF (1988). *Der Körper des Menschen: Einführung in Bau und Funktion*. Georg Thieme Verlag Stuttgart.
- FISCHER, MARKUS (1999). Parallelstrategie. *Magazin für Computertechnik (c't)*, 9:148–151.
- FREY, HANS GEORG UND JUSTUS SCHACH (1996). Modern geknotet: Neuro-Netze: Strukturen, Typen, Anwendungen. *Magazin für Computertechnik (c't)*, Seiten 256–264.

- FUHR, RICHARD D., LWO HSIEH UND MICHAEL KALLAY (1995). Object–Oriented Paradigm for NURBS Curve and Surface Design. *Computer–Aided Design*, 27(2):95–100.
- GEHRER, A., H. PASSRUCKER, H. JERICHA UND J. LAND (1997). Blade Design and Grid Generation for Computational Fluid Dynamics (CFD) with Bézier–Curves and Bézier–Surfaces. In R. Decuyper und G. Dibelius, Herausgeber, *2nd European Conference Turbomachinery, Fluid Dynamics and Thermodynamics*. Antwerpen.
- GEIST, AL, ADAM BEGUELIN, JACK DONGARRA, WEICHENG JIANG, ROBERT MANCHEK UND VAIDY SUNDERAM (1994). *PVM: Parallel Virtual Machine — A Users' Guide and Tutorial for Networked Parallel Computing*. MIT Press Cambridge. [http://www.epm.ornl.gov/pvm/pvm\\_home.html](http://www.epm.ornl.gov/pvm/pvm_home.html)
- GERSTEN, KLAUS UND HEINZ HERWIG (1992). *Strömungsmechanik*. Vieweg, Braunschweig Wiesbaden.
- GILL, PHILIP. E., WALTER MURRAY UND MARGARET H. WRIGHT (1981). *Practical Optimization*. Academic Press London.
- GOLDBERG, DAVID E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison–Wesley.
- GRANVILLE, PAUL S. (1953). The Calculation of the Viscous Drag of Bodies of Revolution. Report 849, Navy Department, The David W. Taylor Model Basin.
- GRAUBARD, STEPHEN R., Herausgeber (1996). *Probleme der künstlichen Intelligenz: eine Grundlagendiskussion*. Springer–Verlag Wien.
- GREENMAN, ROXANA M. (1998). Two–Dimensional High–Lift Aerodynamic Optimization Using Neural Networks. Technical Memorandum 1998–112233, NASA.
- GUTKOWSKI, W., Herausgeber (1997). *Discrete Structural Optimization*. CISM Courses and Lectures No. 373. Springer Verlag Wien.
- HAJELA, PRABHAT (1997). *Stochastic Search in Discrete Structural Optimization Simulated Annealing, Genetic Algorithms and Neural Networks*, Kapitel 2, Seiten 55–133. In GUTKOWSKI (1997).
- HAJELA, PRABHAT (1999). Nongradient Methods in Multidisciplinary Design Optimization — Status and Potential. *Journal of Aircraft*, 36(1):255–265.
- HÄMMERLIN, GÜNTHER UND KARL-HEINZ HOFFMAN (1989). *Numerische Mathematik*. Grundwissen Mathematik. Springer–Verlag.
- HEISTERMANN, JOCHEN (1994). *Genetische Algorithmen*. B. G. Teubner Stuttgart.
- HILDEBRANDT, THOMAS (1998). *Weiterentwicklung von 3D Navier–Stokes–Strömungsrechenverfahren zur Anwendung in hochbelasteten Verdichter– und Turbinengittern*. Dissertation, Fakultät für Luft– und Raumfahrttechnik, Universität der Bundeswehr München.

- HOLLAND, JOHN H. (1975). *Adaption in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The University of Michigan Press, Ann Arbor.
- HOLTMANN, HORST (1983). *Ein numerisches Verfahren zur Berechnung der instationären, zweidimensionalen Strömung durch schwingende Schaufelungen axialer Turbomaschinen*. Dissertation, RWTH Aachen.
- HOSCHEK, JOSEF UND DIETER LASSER (1989). *Grundlagen der geometrischen Datenverarbeitung*. B.G. Teubner.
- INGBER, LESTER (1993). Simulated Annealing: Practice Versus Theory. *Mathematical and Computer Modelling*, 18(11):29–57.
- INGBER, LESTER UND BRUCE ROSEN (1992). Genetic Algorithms and Very Fast Simulated Reannealing: A Comparison. *Mathematical and Computer Modelling*, 16(11):87–100.
- IRMISCH, S. (1991). Berechnung der 2D-Schaufelgitterströmung gekoppelt mit Grenzschichtverfahren zur Berücksichtigung der Kühltausblasung aus der Schaufeloberfläche. Abschlußbericht Turbotech, Verlustminderung, Teilvorhaben 1.2.1.5, Institut für Luftfahrtantriebe, Universität Stuttgart.
- JAANINEH, GEORG UND MARCUS MAIJOHANN (1996). *Fuzzy-Logik und Fuzzy-Control*. Vogel-Verlag.
- JERICHA, H., W. SANZ UND H. PASSRUCKER (1994). Analytische Beschreibung und Optimierung dreidimensionaler verwundeter Schaufelprofile. In *Thermische Strömungsmaschinen: Fortschritte in der Strömungsmaschinentechnik*, Nummer 1109 in VDI-Berichte, Seiten 113–126. VDI-Gesellschaft Energietechnik.
- KAKLIS, P. D. UND A. I. GINNIS (1996). Sectional-Curvature Preserving Skinning Surfaces. *Computer Aided Geometric Design*, 13:601–619.
- KÄPPELI, ERNST (1987). *Strömungslehre und Strömungsmaschinen*. Verlag Harri Deutsch.
- KEPPLER, JOACHIM (1998). Einsatzoptimierung von Gas- und Dampfturbinenkraftwerken mit Genetischen Algorithmen und Neuronalen Netzen. In VDI (1998), Seiten 87–99.
- KINNEBROCK, WERNER (1994). *Optimierung mit genetischen und selektiven Algorithmen*. R. Oldenbourg Verlag, München.
- KNODEL, HANS UND HORST BAYRHUBER, Herausgeber (1985). *Linder Biologie*. J.B. Metzlersche Verlagsbuchhandlung, 19. Auflage.
- KNUPP, PATRICK M. UND STANLEY STEINBERG (1994). *The Fundamentals of Grid Generation*. CRC Press Boca Raton.
- KOCH, MARIO, THOMAS KUHN UND JÜRGEN WERNSTEDT (1996). *Fuzzy Control: optimale Nachbildung und Entwurf optimaler Entscheidungen*. Oldenbourg Verlag München.

- KÖLLER, ULF (1999). Entwicklung einer fortschrittlichen Profilsystematik für stationäre Gasturbinenverdichter. Forschungsbericht 1999–20, Deutsches Zentrum für Luft- und Raumfahrt (DLR).
- KORAKIANITIS, T. UND G. I. PANTAZOPOULOS (1993). Improved Turbine–Blade Design Techniques Using 4th–Order Parametric–Spline Segments. *Computer–Aided Design*, 25(5):289–299.
- KUIPER, H., A. J. VAN DER WEES, C. F. W. HENDRIKS UND TH. E. LABRUJÈRE (1998). Part II. Application of Genetic Algorithms to the Design of Airfoil Pressure Distributions. In PÉRIAUX ET AL. (1998).
- KUPPE, MARKUS (1997). *Zwei Methoden zur Approximation mit B–Splines und ein Ansatz zur Übertragung auf geometrische Splines*. Diplomarbeit, Fachbereich Mathematik, Arbeitsgruppe Differentialgeometrie und Kinematik, Technische Hochschule Darmstadt.
- LADSON, CHARLES L., W. BROOKS JR., ACCUILLA S. HILL UND DARRELL W. SPROLES (1996). Computer Program to Obtain Ordinates for NACA Airfoils. Technical Memorandum 4741, NASA Langley Research Center.
- LAWERENZ, MARTIN (1995). Aerodynamische Optimierung von Strömungsmaschinen unter Verwendung direkter numerischer Optimierungsverfahren. Abschlußbericht HTGT TURBOTECH, Arbeitspaket Nr. 1.1.2.13, MTU München.
- LEE, HAUHUA, SANJAY GOEL, SIU S. TONG, BRENT GREGORY UND SCOTT HUNTER (1993). Toward Modeling the Concurrent Design of Aircraft Engine Turbines. In *International Gas Turbine and Aeroengine Congress and Exposition*, Nummer 93–GT–193 in ASME Paper.
- LEHTIMÄKI, REIJO (1999). Elliptic Grid Generation. *AIAA Journal*, 37(6):768–770.
- LO, CHING F. UND G. Z. SHI (1992). Wind–Tunnel Compressor Stall Monitoring Using Neural Networks. *Journal of Aircraft*, 29(4):736–738.
- LÜCKING, PETER (1982). *Numerische Berechnung der dreidimensionalen reibungsfreien und reibungsbehafteten Strömung durch Turbomaschinen*. Dissertation, RWTH Aachen.
- LÜCKING, PETER (1989). Numerische Berechnung von axialsymmetrischen Propfan–Strömungsfeldern mit einem Stromfunktionsverfahren. Technischer Bericht 88/011, MTU, München.
- MARRA, JOHN (1997). An Expert System Eases Rotor Design. *Mechanical Engineering*, Seiten 70–72.
- MERTENS, B. (1995). Mischungsvorgänge in axialen Verdichtergittern. Forschungsbericht 580, Forschungsvereinigung Verbrennungskraftmaschinen e.V. (FVV).
- MICHALEWICZ, ZBIGNIEW (1996). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer Verlag Berlin Heidelberg, dritte Auflage.

- MILLER IV, PERRY L., JAMES H. OLIVER, DAVID P. MILLER UND DANIEL L. TWEEDT (1997). Using Stream Surfaces for Blade Design. *Mechanical Engineering*, 119(4):66–68.
- MOIN, PARVIZ UND JOHN KIM (1997). Modellieren von Turbulenz. *Spektrum der Wissenschaft*, Seiten 92–101.
- MÜHLENBEIN, HEINZ (1995). The Science of Breeding and its Application to Genetic Algorithms. In WINTER ET AL. (1995), Seiten 59–82.
- MÜLLER-TÖWS, JÖRG (2000). *Aerothermodynamische Auslegung der Meridianströmung mehrstufiger Axialverdichter mit Hilfe von Optimierungsstrategien*. Dissertation, Institut für Thermische Energietechnik, Fachgebiet Strömungsmaschinen, Universität Georgsmuthschule Kassel.
- NADON, LUC J. J. P., STUART C. KRAMER UND PAUL I. KING (1999). Multidisciplinary Optimization in Conceptual Design of Mixed-Stream Turbofan Engines. *Journal of Propulsion and Power*, 15(1):17–22.
- NERURKAR, A.C., T. Q. DANG, E. S. REDDY UND D. R. REDDY (1996). Design Study of Turbomachine Blades by Optimization and Inverse Techniques. In *32nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, Nummer 96-2555 in AIAA Paper, Seiten 1–8. American Institute of Aeronautics and Astronautics.
- OBAYASHI, SHIGERU UND SUSUMU TAKANASHI (1996). Genetic Optimization of Target Pressure Distributions for Inverse Design Methods. *AIAA Journal*, 34(5):881–886.
- OBAYASHI, SHIGERU UND TAKANORI TSUKAHARA (1997). Comparison of Optimization Algorithms for Aerodynamic Shape Design. *AIAA Journal*, 35(8):1413–1415.
- OLIVER, JAMES H., NIRMAL K. NAIR UND DANIEL E. SHANAHAN (1994). Geometric Design of Turbomachinery Blades on General Stream Surfaces. In Rajit Gadh, Herausgeber, *Concurrent Product Design*, Band 74, Seiten 137–144. American Society of Mechanical Engineers (ASME).
- OROZCO, CARLOS E. UND OMAR N. GHATTAS (1996). Infeasible Path Optimal Design Methods with Applications to Aerodynamic Shape Optimization. *AIAA Journal*, 34(2):217–224.
- OUSTERHOUT, JOHN K. (1995). *Tcl und Tk — Entwicklung grafischer Benutzerschnittstellen für das X Window System*. Addison-Wesley, Bonn.
- PENROSE, ROGER (1991). *Computerdenken: die Debatte um künstliche Intelligenz, Bewußtsein und die Gesetze der Physik*. Spektrum der Wissenschaft Verlagsgesellschaft, Heidelberg.
- PÉRIAUX, J., M. SEFRIoui, B. STOUFFLET, B. MANTEL UND E. LAPORTE (1995). Robust Genetic Algorithms for Optimization Problems in Aerodynamic Design. In WINTER ET AL. (1995), Seiten 371–396.

PÉRIAUX, JACQUES, GABRIEL BUGEDA, PANAGIOTIS K. CHAVIAROPOULOS, KYRIAKOS GIANNAKOGLOU, STEPHANE LANTERI UND BERTRAND MANTEL, Herausgeber (1998). *Optimum Aerodynamic Design & Parallel Navier–Stokes Computations (ECARP – European Computational Aerodynamics Research Project)*, Band 61 von *Notes on Numerical Fluid Mechanics (NNFM)*. Vieweg Verlagsgesellschaft Braunschweig.

PÉRIAUX, JACQUES, GABRIEL BUGEDA, PANAGIOTIS K. CHAVIAROPOULOS, THEO E. LABRUJÈRE UND BRUNO STOUFFLET, Herausgeber (1997). *EUROPT: A European Initiative on Optimum Design Methods in Aerodynamics*, Band 55 von *Notes on Numerical Fluid Mechanics (NNFM)*. Vieweg Verlagsgesellschaft Braunschweig.

PFEIL, BERND (1994). *Aerodynamische Optimierung von Turbinenprofilen mit evolutionären Algorithmen*. Diplomarbeit, Institut für Luftfahrtantriebe, Universität Stuttgart.

PIEGL, LES UND WAYNE TILLER (1994). Software–Engineering Approach to Degree Elevation of B–Spline Curves. *Computer–Aided Design*, 26(1):17–28.

PIEGL, LES UND WAYNE TILLER (1995a). Algorithm for Degree Reduction of B–Spline Curves. *Computer–Aided Design*, 27(2):101–110.

PIEGL, LES UND WAYNE TILLER (1995b). *The NURBS Book*. Springer–Verlag Berlin Heidelberg.

PIEGL, LES UND WAYNE TILLER (1997). Symbolic Operators for NURBS. *Computer–Aided Design*, 29(5):361–368.

PIERRET, S. UND R. A. VAN DEN BRAEMBUSSCHE (1999). Turbomachinery Blade Design Using a Navier–Stokes Solver and Artificial Neural Network. *Journal of Turbomachinery*, 121(4):326–332.

QUAGLIARELLA, DOMENICO (1995). Genetic Algorithms Applications in Computational Fluid Dynamics. In WINTER ET AL. (1995), Seiten 417–442.

QUAGLIARELLA, DOMENICO UND ANTONIO DELLA CIOPPA (1995). Genetic Algorithms Applied to the Aerodynamic Design of Transonic Airfoils. *Journal of Aircraft*, 32(4):889–891.

RECHENBERG, INGO (1994). *Evolutionsstrategie '94*. Frommann–Holzboog.

RECHENBERG, INGO (1998). EVOTECH — Einsatz der Evolutionsstrategie in Wissenschaft und Technik, Teilprojekt Theorievollkommen und Objektiventwurf. Abschlußbericht, Fachgebiet Bionik & Evolutionstechnik, Technische Universität Berlin.

REUSCH, BERND, Herausgeber (1997). *Computational Intelligence: Theory and Applications*, Nummer 1226 in Lecture Notes in Computer Science. 5th Fuzzy Days, Dortmund 28.–30. April, Springer–Verlag Berlin.

- ROJAS, RAÚL (1996). *Theorie der neuronalen Netze: eine systematische Einführung*. Springer–Verlag Berlin.
- RÖPER, RALF (1994). *Berechnung mehrstufiger Axialverdichter mit einem Quasi-3D-Through-Flow-Verfahren und Untersuchungen zur Randzonenkorrektur*. Dissertation, RWTH Aachen.
- SADREHAGHIGHI, IDEEN, ROBERT E. SMITH UND SURENDRA N. TIWARI (1995). Grid Sensitivity and Aerodynamic Optimization of Generic Airfoils. *AIAA Journal*.
- SANZ, JOSE M. (1999). A Neural Network Aero Design System for Advanced Turbo-Engines. In *XIV International Symposium of Air Breathing Engines (ISABE)*. ISA-BE 99–7044.
- SCHLICHTING, HERMANN (1979). *Boundary-Layer Theory*. McGraw–Hill, New York, 7. Auflage.
- SCHLICHTING, HERRMANN UND KLAUS GERSTEN (1997). *Grenzschicht-Theorie*. Springer–Verlag, Berlin Heidelberg, 9. Auflage.
- SCHOLZ, NORBERT (1965). *Aerodynamik der Schaufelgitter — Band I*. Verlag G. Braun Karlsruhe.
- SCHÖNE, WOLFGANG (1978). *Differentialgeometrie*. Verlag Harri Deutsch, Frankfurt a.M.
- SCHÖNEBURG, FRANK, FRANK HEINZMANN UND SVEN FEDDERSEN (1994). *Genetische Algorithmen und Evolutionsstrategien*. Addison–Wesley Bonn.
- SCHULZ, VOLKER, THOMAS DREYER, THOMAS SPEER UND HANS GEORG BOCK (1996). Optimum Shape Design of Turbine Blades. In P. Kleinschmidt, A. Bachem, U. Derigs, D. Fischer, U. Leopold-Wildburger und R. Möhring, Herausgeber, *Operations Research Proceedings 1995*, Seiten 190–195. Springer Verlag Heidelberg.
- SCHWARZ, HANS RUDOLF (1993). *Numerische Mathematik*. B.G. Teubner Stuttgart, dritte Auflage.
- SCHWARZ, RAMONA (1992). *Einsatz numerischer Optimierungsverfahren bei der aero-thermodynamischen Auslegung von Radialverdichterstufen*. Dissertation, Fakultät für Maschinenwesen, Technische Universität München.
- SCHWEFEL, HANS-PAUL (1995). *Evolution and Optimum Seeking*. John Wiley & Sons.
- SELIG, MICHAEL S. UND VICTORIA L. COVERSTONE-CARROLL (1995). Application of a Genetic Algorithm to Wind Turbine Design. In Walter E. Musial und Susan M. Hock, Herausgeber, *Wind Energy 1995*, Seiten 13–21. The Energy and Environmental Expo '95, ASME.
- SERVATY, S. UND W. KEPPEL (1998). Turboverdichter–Forschung in der AG TURBO. In *Thermische Strömungsmaschinen: Turbokompressoren im industriellen Einsatz*, Nummer 1425 in VDI–Berichte, Seiten 7–30.

- SMITH, JEFFREY D. (1989). *Design and Analysis of Algorithms*. PWS-KENT Publishing Company Boston.
- SNNS (1998). *Stuttgart Neural Network Simulator (SNNS) User Manual Version 4.2*. Institut für Parallelle und Verteilte Höchstleistungsrechner (IPVR), Universität Stuttgart und Wilhelm-Schickard-Institut für Informatik, Universität Tübingen. <http://www-ra.informatik.uni-tuebingen.de/SNNS/>.
- SONAR, THOMAS (1989). Grid Generation Using Elliptic Partial Differential Equations. Forschungsbericht DFVLR-FB 89-15, Deutsche Forschungsanstalt für Luft- und Raumfahrt (DLR).
- SPEER, THOMAS, MARKUS KUPPE UND JOSEF HOSCHEK (1998). Global Reparameterization for Curve Approximation. *Computer Aided Geometric Design*, 15(9):869–877.
- SPELLUCCI, PETER (1993). *Numerische Verfahren der nichtlinearen Optimierung*. Birkhäuser Verlag.
- STEINERT, E., B. EISENBERG UND H. STARKEN (1991). Design and Testing of a Controlled Diffusion Airfoil Cascade for Industrial Axial Flow Compressor Application. *Journal of Turbomachinery*, 113:583–590.
- STROUSTRUP, BJARNE (1991). *The C++ Programming Language*. Addison-Wesley, Reading, Massachusetts, zweite Auflage.
- SUBEL, CARSTEN (1998). *Inverse Design of Aerodynamic Profiles Using Rational Cubic Bézier Curves and Quasi Newton Optimization Method*. Diplomarbeit, Institut für Thermische Energietechnik, Fachgebiet Strömungsmaschinen.
- TASCHNER, RUDOLF J. (1977). *Differentialgeometrie für Geodäten*. Manzsche Verlags- und Universitätsbuchhandlung, Wien.
- THIEDE, P. (1977). Ein inverses Integralverfahren zur Berechnung abgelöster turbulenter Grenzschichten. Fb 77-16, DLR.
- THOMPSON, JOE F., Z.U.A. WARSI UND C. WAYNE MASTIN (1985). *Numerical Grid Generation*. North-Holland.
- TILLER, WAYNE (1992). Knot-Removal Algorithms for NURBS Curves and Surfaces. *Computer-Aided Design*, 24(8):445–453.
- TONG, S. S., D. POWELL UND S. GOEL (1992). Integration of Artificial Intelligence and Numerical Optimization Techniques for the Design of Complex Aerospace Systems. AIAA Paper 92-1189, AIAA.
- TONG, SIU SHING UND BRENT A. GREGORY (1990). Turbine Preliminary Design Using Artificial Intelligence and Numerical Optimization Techniques. ASME Paper 90-GT-148, ASME.

- TRIGG, M. A., TUBBY G. R. UND A. G. SHEARD (1999). Automatic Genetic Optimization Approach to Two-Dimensional Blade Profile Design for Steam Turbines. *Journal of Turbomachinery*, 121:11–17. ASME Paper 97-GT-392.
- UELSCHEN, MICHAEL, SANDRA JENSEN UND MARTIN LAWERENZ (1998). Dreidimensionaler Entwurf axialer Beschaufelungen und Geometrieoptimierung. Interner Bericht 1/98, Institut für Thermische Energietechnik, Fachgebiet Strömungsmaschinen, Universität Gesamthochschule Kassel.
- UELSCHEN, MICHAEL, JÖRG WEIDENFELLER UND MARTIN LAWERENZ (1999). Einsatz neuronaler Netze in der Strömungsmessung mit Fünfloch-Drucksonden. *Technisches Messen (tm)*, 66(10):372–377.
- VAN DEN BRAEMBUSSCHE, R. A. UND M. MANNA, Herausgeber (1997). *Inverse Design and Optimisation Methods*, Nummer 1997–05 in Lecture Series. von Karman Institute for Fluid Dynamics.
- VAUDEVENDER, W.H. UND K. H. HASKELL (1982). The SLATEC Mathematical Subroutine Library. *SIGNUM Newsletter*, 3(17):16–21. Quelle: *SLATEC — Common Mathematical Library Version 4.1, July 1993*: <http://elib.zib.de/netlib/slatec>.
- VDI (1998). *Computational Intelligence: Neuronale Netze, Evolutionäre Algorithmen, Fuzzy Control im industriellen Einsatz*, Nummer 1381 in VDI-Berichte. Berlin 3./4. März.
- VICINI, ALESSANDRO UND DOMENICO QUAGLIARELLA (1997). Inverse and Direct Airfoil Design Using a Multiobjective Genetic Algorithm. *AIAA Journal*, 35(9):1499–1505.
- VICINI, ALESSANDRO UND DOMENICO QUAGLIARELLA (1999). Airfoil and Wind Design Through Hybrid Optimization Strategies. *AIAA Journal*, 37(5):634–641.
- WALL, MATTHEW (1996). *GAlib: A C++ Library of Genetic Algorithm Components*. Mechanical Engineering Department, Massachusetts Institute of Technology. <http://lancet.mit.edu/ga/>.
- WALZ, ALFRED (1966). *Strömungs- und Temperaturgrenzschichten*. Verlag G.Braun Karlsruhe.
- WANG, TEN-SEE UND BHARAT K. SONI (1991). Goodness of Grid: Quantitative Measures. *Transactions of The Society for Computer Simulation*, 8(2):87–97.
- WATZLAWICK, R. (1991). Bestimmung der wesentlichen Einflußgrößen in der Korrelation der Sekundärströmungsverluste bei Veränderung des Schaufelseitenverhältnisses. Forschungsbericht 482, Forschungsvereinigung Verbrennungskraftmaschinen e.V. (FVV).
- WINTER, G., J. PÉRIAUX, M. GALÁN UND P. CUESTA, Herausgeber (1995). *Genetic Algorithms in Engineering and Computer Science*. John Wiley & Sons.

WU, CHUNG-HUA (1952). A General Theory of Three-Dimensional Flow in Subsonic and Supersonic Turbomachines of Axial-, Radial-, and Mixed-Flow Types. Technical Note 2604, National Advisory Committee for Aeronautics (NACA).

ZELL, ANDREAS (1997). *Simulation neuronaler Netze*. R. Oldenbourg Verlag München.