

Ein Vergleich verschiedener Verfahren
zur Optimierung mehrlagiger Perzeptron-Netze
für die Modellierung dynamischer Systeme
mit chaotischem Verhalten

Von dem Fachbereich für
Elektrotechnik und Informationstechnik
der Universität Hannover

zur Erlangung des akademischen Grades
Doktor-Ingenieur

genehmigte
Dissertation

von

Dipl.-Ing. Siegfried Ergezinger
geboren am 17. November 1959 in Dehmke (jetzt Aerzen)

1998

Referent: Prof. Dr.-Ing. H. P. Kuchenbecker

Korreferent: Prof. Dr.-Ing. W. Nejd

Tag der Promotion: 09. Januar 1998

Abstract

Key Words: Chaotic Time Signals, multilayer Perceptrons, nonlinear Optimization Algorithms

Dynamical systems with chaotic behaviour are characterized by a sensitive dependence on initial conditions. The ability of multilayer perceptrons to model this behaviour and in addition to predict the evolution of the state of the system over time is demonstrated. Synthetic chaotic discrete time signals of the Lorenz system, the Mackey-Glass differential equation, and the Hénon map are used as test signals.

A novel nonlinear optimization technique for multilayer perceptrons - the *Optimization Layer by Layer* (*OLL* algorithm) - is introduced. This algorithm is based on the approach to optimize the weights of the multilayer perceptron independently layer by layer starting with the output layer. In addition the nonlinear optimization problem is reduced to an iterative linear optimization by linearization of the activation functions of the hidden layer perceptrons and by introducing a modified cost function for the optimization problem.

The achievable prediction gain as well as the speed of convergence, the robustness, and the demand for computing time of the new *OLL* algorithm is investigated in a series of experiments aiming at the prediction of chaotic time series. Furthermore these results are compared to those achieved with the widely used variants of the Backpropagation algorithm, several variants of the conjugate gradient algorithms as well as the Quasi-Newton algorithm of Broyden, Fletcher, Goldfarb and Shanno. The results show that multilayer perceptrons are extremely well suited both for the nonlinear prediction of chaotic time series and to model the underlying dynamical systems. Throughout all experiments the *OLL* algorithm proves to be superior to the known optimization techniques by factors between 100 and 1,000,000 looking at the speed of convergence and by factors between 26.5 and 22,000 in reducing computing time. Especially in those areas of the error surface where the known nonlinear optimization techniques either fail due to numerical problems or, show an extremely slow convergence speed the *OLL* algorithm demonstrates significantly improved behaviour.

While the multilayer perceptrons optimized with the *OLL* algorithm approximate the nonlinear mapping of the underlying dynamical systems with a very high accuracy they are also employed for the computation of the Lyapunov-Exponenten of the investigated dynamical systems. Based on the Treppen-Iteration algorithm the Lyapunov exponents of the Lorenz model and the Hénon map are computed with a high accuracy from the respective chaotic time series.

Kurzfassung

Für die Dokumentation: Chaotische Zeitsignale, Mehrlagige Perzeptron Netze, Nichtlineare Optimierungsverfahren

Dynamische Systeme mit chaotischem Verhalten besitzen sehr komplexe nichtlineare Abbildungseigenschaften und sind durch eine *sensitive Abhängigkeit* von den Anfangszuständen charakterisiert. Die Eignung mehrlagiger Perzeptron-Netze zur Modellierung dieser Eigenschaften und darüber hinaus zur Prädiktion der zeitlichen Entwicklung des Systemzustandes wird am Beispiel synthetischer chaotischer Zeitsignale des Lorenz-Systems, der Mackey-Glass-Differentialgleichung sowie der Hénon-Abbildung untersucht.

Zur nichtlinearen Optimierung der Netzgewichte wird das neuartige Verfahren der *Optimierung Lage für Lage* (*OLL-Verfahren*) vorgestellt, welches auf dem Ansatz basiert, die Optimierung der mehrlagigen Perzeptron-Netze nach Lagen getrennt durchzuführen. Zusätzlich wird das nichtlineare Optimierungsproblem durch Linearisierung der Aktivierungsfunktionen in den verdeckten Lagen und durch Einführung eines modifizierten Gütekriteriums auf ein iteratives lineares Problem zurückgeführt. Anhand von Untersuchungen zur Prädiktion chaotischer Zeitsignale wird das neue *OLL-Verfahren* hinsichtlich erzielbarer Prädiktionsgewinne sowie Konvergenzgeschwindigkeit, Robustheit und Rechenzeitbedarf mit den am häufigsten eingesetzten Varianten des Backpropagation-Verfahrens, verschiedenen Verfahren der konjugierten Gradienten und einem Quasi-Newton-Verfahren verglichen.

Die Ergebnisse beweisen, daß mehrlagige Perzeptron-Netze sehr gut zur nichtlinearen Prädiktion chaotischer Zeitsignale und zur Modellierung der zugrundeliegenden dynamischen Systeme geeignet sind. Hierbei erweist sich das *OLL-Verfahren* hinsichtlich der Konvergenzgeschwindigkeit um Faktoren zwischen 100 und 1 000 000 und im direkten Vergleich der insgesamt erforderlichen Rechenzeit um Faktoren zwischen 26,5 und 22 000 den bekannten Verfahren überlegen. Besonders in Bereichen der Fehleroberfläche, in denen die bekannten Verfahren entweder aufgrund numerischer Probleme abbrechen oder wo deren Konvergenzgeschwindigkeit bereits extrem gering geworden ist, ist das *OLL-Verfahren* den bekannten nichtlinearen Optimierungsverfahren weit überlegen. Weiterhin zeigt sich, daß lineare Prädiktoren für diese Aufgabenstellungen ungeeignet sind.

Die mit dem *OLL-Verfahren* ermittelten mehrlagigen Perzeptron-Netze approximieren die Abbildungseigenschaften der untersuchten dynamischen Systeme mit sehr hoher Genauigkeit und werden daher auch zur Bestimmung der Lyapunov-Exponenten der zugrundeliegenden dynamischen Systeme eingesetzt. Unter Verwendung des Treppen-Iterations-Algorithmus werden auf Basis der optimierten mehrlagigen Perzeptron-Netze der positive Lyapunov-Exponent des Lorenz-Systems mit $\pm 0,2\%$ und der negative mit einer Abweichung von $\pm 5\%$ berechnet. Im Falle der Hénon-Abbildung betragen die Abweichungen für den positiven Exponenten ca. $0,5\%$ und für den negativen Exponenten ca. $0,1\%$.

Vorwort

Das Kernstück der vorliegenden Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Allgemeine Nachrichtentechnik der Universität Hannover. Die Arbeit wurde parallel zu meiner Tätigkeit für E-Plus Mobilfunk fertiggestellt.

Mein Dank für die Förderung dieser Arbeit gilt Herrn Prof. Dr.-Ing. H. Kindler und in besonderem Maße Herrn Prof. Dr.-Ing. H. Wehmeier für seine Unterstützung nach dem völlig unerwarteten Tod von Prof. Kindler im Juni 1990.

Ganz besonders danken möchte ich Herrn Prof. Dr.-Ing. H. P. Kuchenbecker für die Übernahme der Betreuung meiner Arbeit, die Erstellung des Hauptreferates sowie die zahlreichen Anregungen und Diskussionen. Herzlich bedanke ich mich bei Herrn Prof. Dr.-Ing. W. Nejdil für die Übernahme des Korreferates.

Darüberhinaus gilt mein besonderer Dank Herrn Dr. U. Parlitz und Herrn Prof. Dr.-Ing. D. Chahabadi für fruchtbare Diskussionen und Herrn Dr.-Ing. P. Meissner für seine motivierende Unterstützung.

Schließlich sei meiner Frau Petra und Herrn Dipl.-Ing. T. Schmitz für die Durchsicht des Manuskripts gedankt.

Mettmann im Januar 1998

Siegfried Ergezinger

Inhaltsverzeichnis

1	Einleitung	1
2	Dynamische Systeme und neuronale Netze	5
2.1	Dynamische Systeme	5
2.2	Stationäre Lösungen dynamischer Systeme, Grenzmengen und Attraktoren	8
2.3	Fraktale Dimensionen seltsamer Attraktoren	11
2.3.1	Kapazität	12
2.3.2	Korrelationsdimension	13
2.4	Lyapunov-Exponenten	14
2.5	Rekonstruktion von Attraktoren	19
2.6	Neuronale Netze zur Modellierung dynamischer Systeme	25
3	Optimierung mehrlagiger Perzeptron-Netze unter Verwendung von Gradientenverfahren	27
3.1	Das mehrlagige Perzeptron-Netz	27
3.2	Abstiegsverfahren	33
3.3	Gradient und Jacobimatrix eines <i>MLP</i> -Netzes	36
3.4	Das Backpropagation-Verfahren und Modifikationen	41
3.5	Verfahren der konjugierten Gradienten	45
3.6	Newton- und Quasi-Newton-Verfahren	48
4	Ein neues Verfahren zur Optimierung mehrlagiger Perzeptron-Netze	54
4.1	Optimierung der Ausgangslage	54
4.2	Optimierung der verdeckten Lagen eines mehrlagigen Perzeptron-Netzes	57
4.2.1	Linearisierung der verdeckten Lagen eines mehrlagigen Perzeptron-Netzes	58
4.2.2	Die modifizierte Gütefunktion	60
4.2.3	Minimierung der modifizierten Gütefunktion	63
4.3	Ablauf des <i>OLL</i> -Optimierungsverfahrens	66

5	Ergebnisse	69
5.1	Ein einfaches Optimierungsproblem	70
5.2	Prädiktionsergebnisse für das chaotische Lorenz-Zeitsignal	74
5.2.1	Lineare Prädiktion des Lorenz-Zeitsignals	75
5.2.2	Nichtlineare Prädiktion des Lorenz-Zeitsignals mit dem <i>OLL</i> - Verfahren	76
5.2.3	Nichtlineare Prädiktion des Lorenz-Signals mit Gradienten-Verfahren	84
5.3	Prädiktionsergebnisse für das chaotische Mackey-Glass-Zeitsignal	91
5.3.1	Mackey-Glass-Zeitsignal für $\Delta T=17$	92
5.3.2	Mackey-Glass-Zeitsignal für $\Delta T=30$	104
5.4	Prädiktionsergebnisse für das chaotische Hénon-Zeitsignal	109
5.5	Weitere Prädiktionsergebnisse	112
5.6	Berechnung der Lyapunov-Exponenten aus dem Zeitsignal	114
5.6.1	Lyapunov-Exponenten des Lorenz-Modells	116
5.6.2	Lyapunov-Exponenten der Hénon-Abbildung	121
6	Zusammenfassung	124
	Literaturverzeichnis	130
A	Untersuchte chaotische Systeme	141
A.1	Das Lorenz-System	141
A.2	Die Mackey-Glass-Differentialgleichung	143
A.3	Die Hénon-Abbildung	146
B	Glossar	149

Symbolverzeichnis

Vektoren, Mengen, Matrizen und vektorwertige Abbildungen sind fett gedruckt. Vektoren werden durch kleine Buchstaben dargestellt, Mengen, Matrizen und Abbildungen durch große.

Es folgen häufig verwendete Formelsymbole:

\mathbf{F}	Vektorfeld eines zeitkontinuierlichen dynamischen Systems
$\mathcal{F}(\mathbf{x}^k)$	Systemabbildung eines zeitdiskreten dynamischen Systems
$\mathcal{G}(\mathbf{x}^k)$	Systemabbildung des rekonstruierten dynamischen Systems
$\tilde{\mathcal{G}}(\mathbf{x}^k, \mathbf{w})$	Abbildung des MLP-Netzes
ϕ_t	Abbildungsgleichung der Lösung eines zeitkontinuierlichen Systems, Fluß des Systems
$\Phi_k(\mathbf{x}^0)$	Übergangsmatrix des linearisierten zeitdiskreten Systems
\mathbf{I}	Einheitsmatrix
λ	Lyapunov-Exponenten $(\lambda_1, \dots, \lambda_{d_S})^T$
\mathbf{M}	Zustandsraum des Systems
\mathbb{R}^n	n -dimensionaler Euklidischer Raum
Ω	Grenzmengen
\mathbf{U}	Einzugsgebiet
\mathcal{A}	Attraktor
$\mathbf{x}(t)$	Zustandsvektor eines zeitkontinuierlichen Systems zum Zeitpunkt t
\mathbf{x}^k	Zustandsvektor eines zeitdiskreten Systems zum Zeitpunkt k
\mathbf{x}_0	Anfangszustand (Zustandsvektor zum Zeitpunkt t_0)
t	Zeit
d_S	Dimension des Zustandsraumes, Ordnung des Systems
d_K	Kapazität
d_C	Korrelationsdimension
d_I	Informationsdimension
d_E	Einbettungsdimension, Rekonstruktionsdimension
$\{v^k\}$	Folge skalarer Meßwerte
$\{\mathbf{x}^k\}$	Folge von (rekonstruierten) Zustandsvektoren eines dynamischen Systems

T_A	Abtastperiode, Periodendauer, Evolutionszeit
τ	Einbettungskonstante
pw	Prädiktionsweite
W	Fensterlänge der Rekonstruktion
L	Anzahl Lagen des MLP-Netzes
H_l	Anzahl der Knoten in der l -ten Lage
M	Anzahl Knoten in der Eingangslage
N	Anzahl Knoten in der Ausgangslage
3:4:5	3-lagiges MLP-Netz mit 3 Knoten in der Eingangs-, 4 in der verdeckten- und 5 Knoten in der Ausgangslage
\mathbf{x}^k	Eingangsvektor ((rekonstruierter) Zustandsvektor) des MLP-Netzes
\mathbf{y}^k	Ausgangsvektor des MLP-Netzes
\mathbf{d}^k	Gewünschter Ausgangsvektor des MLP-Netzes
$\{\mathbf{x}^k, \mathbf{d}^k\}$	Folge der Trainingsmuster
K	Anzahl Trainingsmuster
\mathbf{e}^k	Prädiktionsfehlervektor
\mathbf{s}_l^k	Eingangswerte der Knoten in der l -ten Lage des MLP-Netzes
\mathbf{o}_l^k	Ausgangswerte der Knoten in der l -ten Lage des MLP-Netzes
Q	Anzahl aller Gewichte eines MLP-Netzes, Dimension des Optimierungsproblems
$w_{l,h_l h_{l-1}}$	Gewicht der Verbindung von Knoten h_{l-1} der Lage $(l-1)$ zu Knoten h_l in Lage l
\mathbf{W}_l	Gewichtsmatrizen in den einzelnen Lagen des MLP-Netzes
$\Delta \mathbf{W}_l$	Änderung der Gewichtsmatrizen in den einzelnen Lagen des MLP-Netzes bei Verwendung des <i>OLL</i> -Verfahrens
\mathbf{w}	Gewichtsvektor $(w_{2,10}, w_{2,11}, \dots, w_{2,H_2M}, \dots, w_{L,H_L H_{L-1}})^T$ des MLP-Netzes
\mathbf{w}^*	Vektor der optimalen Gewichte des MLP-Netzes
\mathbf{w}^I	Vektor der Gewichte des MLP-Netzes nach der I -ten Iteration
$\Delta \mathbf{w}^I$	Änderung des Gewichtsvektors des MLP-Netzes in der I -ten Iteration
$E(\mathbf{w})$	Mittlerer quadratischer Prädiktionsfehler, Gütefunktion bei Verwendung der Gradientenverfahren

$E_1^{hid}(\Delta \mathbf{W})$	Modifizierte Gütefunktion bei Verwendung des <i>OLL</i> -Verfahrens
G	Prädiktionsgewinn in dB
G_{tr}	Prädiktionsgewinn in dB für die Trainingsmenge
G_{te}	Prädiktionsgewinn in dB für die Testmenge
$\mathbf{g}(\mathbf{w})$	Gradient $\nabla_{\mathbf{w}}E(\mathbf{w})$ der Zielfunktion $E(\mathbf{w})$
$\mathbf{H}(\mathbf{w})$	Hessematrix $\nabla_{\mathbf{w}}^2E(\mathbf{w})$ der Zielfunktion $E(\mathbf{w})$
η	Schrittweite der Gradientenverfahren
β, γ	Parameter zur Anpassung der Schrittweite der Gradientenverfahren und der Gewichtungsfaktoren des <i>OLL</i> -Verfahrens
α_M	Momentum-Konstante bei Varianten des Backpropagation-Verfahrens
$\boldsymbol{\mu}$	Gewichtungsfaktoren zur Veränderung der modifizierten Gütefunktion des <i>OLL</i> -Verfahrens

Abkürzungsverzeichnis

ADALINE	Adaptives lineares Element
MLP-Netz	Mehrlagiges Perzeptron-Netz
NRMSE	Normalized-Root-Mean-Squared-Error
AKF	Autokorrelationsfunktion
IEEE	<i>Institute of Electrical and Electronics Engineers</i>

Abkürzungen der eingesetzten nichtlinearen Optimierungsverfahren

<i>BP</i> -Verfahren	Varianten des Backpropagation-Verfahrens
<i>BD</i> -Verfahren	Bold-Driver-Heuristik
<i>BFGS</i> -Verfahren	Quasi-Newton-Verfahren nach Broyden, Fletcher, Goldfarb und Shanno
<i>SD</i> -Verfahren	Verfahren des steilsten Abstiegs
<i>PR</i> -Verfahren	Verfahren der konjugierten Gradienten ohne Restart nach Polak-Ribière
<i>PRR</i> -Verfahren	Verfahren der konjugierten Gradienten mit Restart nach Polak-Ribière
<i>FR</i> -Verfahren	Verfahren der konjugierten Gradienten ohne Restart nach Fletcher-Reeves
<i>FRR</i> -Verfahren	Verfahren der konjugierten Gradienten mit Restart nach Fletcher-Reeves
<i>LQN</i> -Verfahren	Limited-Quasi-Newton-Verfahren
<i>QN</i> -Verfahren	Quasi-Newton-Verfahren
<i>SCR</i> -Verfahren	Scaled-Conjugate-Gradient-Verfahren mit Restart
<i>SCG</i> -Verfahren	Scaled-Conjugate-Gradient-Verfahren ohne Restart
<i>OLL</i> -Verfahren	Neues Optimierungsverfahren für <i>MLP</i> -Netze (Optimierung Lage für Lage)

1 Einleitung

Im Rahmen der Signalverarbeitung werden generell zwei Klassen von Zeitsignalen unterschieden. Zum einen sind dies die deterministischen Zeitsignale, deren zeitlicher Verlauf für alle Zeitpunkte t exakt durch ein deterministisches mathematisches Modell beschrieben werden. Die zweite Klasse bilden stochastische Zeitsignale (bzw. stochastischen Prozesse), welche durch Gesetzmäßigkeiten erzeugt werden, die wegen ihrer Komplexität mittels wahrscheinlichkeitstheoretischer mathematischer Modelle beschrieben werden [103]. Beispiele hierfür bilden Sprachsignale und Bildsignale, die als stochastische Prozesse beschrieben werden, um eine möglichst effiziente Verarbeitung (Kodierung, Übertragung etc.) dieser Signale zu gewährleisten [65].

Mit der Veröffentlichung von Lorenz [78] wird diese Klassifizierung jedoch um die Klasse der chaotischen Zeitsignale [24, 126] erweitert. Als solche werden berechnete oder gemessene Größen chaotischer dynamischer Systeme bezeichnet, deren zeitlicher Verlauf in sehr hohem Maße von den gewählten Anfangsbedingungen abhängt. Diese Eigenschaft chaotischer Systeme bewirkt, daß zwei Trajektorien (B.23), ausgehend von zwei beliebig nah beieinander liegenden Anfangszuständen, mit exponentiellen Raten auseinanderstreben, die charakteristisch für das jeweilige System sind. Bereits nach kurzer Zeit erscheinen die zwei Trajektorien unkorreliert. Das Verhalten dieser chaotischen dynamischen Systeme ist, obwohl sie deterministischer Natur sind, über längere Zeiträume nicht berechenbar, da die Anfangsbedingungen nur mit begrenzter Genauigkeit numerisch darstellbar sind und die numerische Berechnung zwangsläufig mit Rundungsfehlern verbunden ist.

Beispiele für solche nichtlinearen dynamischen Systeme mit chaotischem Verhalten sind mathematische Modelle wie die Logistische Abbildung [105, 24], die Hénon-Abbildung [56], das Lorenz-Modell [78], die Duffing-Differentialgleichung [105, 106] oder die Mackey-Glass-Differentialgleichung [81]. Weiterhin werden diese Phänomene aber auch in zunehmendem Maße bei physikalischen [50, 82, 38], biologischen [73, 81], chemischen [11, 84, 122], ökonomischen, elektrischen [88, 147, 48] und signalverarbeitenden [80] Systemen beobachtet.

Auf Grundlage der Theorie chaotischer dynamischer Systeme und der im Rahmen der Chaos-Theorie entwickelten Verfahren eröffnet sich die Möglichkeit, Zeitsignale, die aufgrund ihrer Komplexität zuvor nur als Realisierung stochastischer Prozesse beschrieben werden konnten, nun als Ausgangssignale i. allg. niedrigdimensionaler nichtlinearer dynamischer Systeme zu beschreiben. Dieses, sowie die allgemeine Verfügbarkeit und die enorm gestiegene Leistungsfähigkeit moderner Digitalrechner sind Gründe dafür, daß die Forschungsaktivitäten im Bereich chaotischer Systeme in den letzten zwei bis drei Jahrzehnten stark zugenommen haben.

In der Literatur werden in zunehmendem Maße praktische Anwendungen der Theorie chaotischer dynamischer Systeme aufgezeigt, wobei sich das Interesse im wesentlichen

auf zwei Anwendungsbereiche konzentriert. Der erste verwendet synthetische chaotische Zeitsignale aufgrund ihrer speziellen Eigenschaften für:

- die Maskierung von Zeitsignalen [96, 49]
- den Einsatz als Spreizfolgen für CDMA-Übertragungssysteme [96, 109, 25]
- die Synchronisation zur Übertragung digitaler Nutzsingnale [108, 68, 140]

Der zweite Bereich umfaßt Anwendungen, bei denen die zu verarbeitenden Signale als Ausgangsgrößen chaotischer Systeme beschrieben werden. Hierzu zählen u. a.:

- Detektion von Nutzsingnalen in Rauschsignalen zur Rauschunterdrückung [32, 33, 47, 70, 51]
- Modellierung und Verarbeitung von Sprachsignalen [86, 137], Radarsignalen [50, 51] und anderer Zeitsingnale [96]

In die zweite Kategorie fällt auch die im Rahmen dieser Arbeit betrachtete Anwendung. Basierend auf der Theorie nichtlinearer dynamischer Systeme wird ein Verfahren zur Modellierung sowie zur nichtlinearen Prädiktion von Zeitsingnalen entwickelt und dessen Leistungsfähigkeit am Beispiel von vier synthetischen chaotischen Zeitsingnalen nachgewiesen. Grundlage bildet ein skalares (oder vektorwertiges) Zeitsingnal, welches eine gemessene oder berechnete skalare (oder vektorwertige) Ausgangsgröße eines dynamischen Systems repräsentiert¹. Unter Verwendung des von Packard et al. [100] sowie Takens [136] beschriebenen Verfahrens der zeitverzögerten Komponenten wird aus dem vorliegenden Zeitsingnal eine Folge von Zustandsvektoren gebildet. Bei geeigneter Wahl der Dimension sowie dem zeitlichen Abstand zwischen den Komponenten der rekonstruierten Zustandsvektoren repräsentieren diese alle wesentlichen Eigenschaften des zugrundeliegenden dynamischen Systems. Das Verfahren der zeitverzögerten Komponenten wird zwar sehr häufig eingesetzt [12, 15, 21, 23, 30, 71, 91], über die Auswirkungen einer Variation der Rekonstruktionparameter auf die Qualität der Rekonstruktion liegen jedoch bisher kaum Erkenntnisse vor. Daher ist das Ziel einer Vielzahl im Rahmen dieser Arbeit durchgeführter systematischer Untersuchungen geeignete Kriterien zur optimalen Wahl der Rekonstruktionparameter aus den charakteristischen Kenngrößen der chaotischen Zeitsingnale abzuleiten.

Die Nachbildung der Abbildungseigenschaften des dynamischen Systems aus der Folge rekonstruierter Zustandsvektoren erfolgt durch mehrlagige Perzeptron-Netze (*multilayer*

¹Im Rahmen dieser Arbeit werden ausschließlich chaotische Zeitsingnale untersucht. Die beschriebenen Verfahren können jedoch ebenso auf Ausgangssingnale dynamischer Systeme angewendet werden, die nicht chaotisch oder auch linear sind. Besitzen die zu untersuchenden Zeitsingnale jedoch weniger komplexe Eigenschaften, werden u. U. mit herkömmlichen Verfahren der Signalverarbeitung ebenso gute Ergebnisse erzielt.

Perceptrons), welche einen häufig verwendeten Typ neuronaler Netze repräsentieren. Diese *MLP*-Netze sind geeignet, nahezu beliebige nichtlineare Abbildungen mit sehr hoher Genauigkeit zu approximieren [20, 55]. Die Bestimmung optimaler Netzgewichte stellt ein nichtlineares Optimierungsproblem dar [55], zu dessen Lösung eine Vielzahl unterschiedlicher iterativer Verfahren eingesetzt werden. Diese stellen in der Regel Varianten sogenannter Abstiegsverfahren (*descent methods*) dar. Die optimierten neuronalen Netze werden hier als nichtlineare Prädiktoren zur Bestimmung zukünftiger Zustände des Systems und damit zukünftiger Werte der betrachteten chaotischen Zeitsignale eingesetzt, wobei die Optimierung der Netzgewichte auf die Minimierung des mittleren quadratischen Prädiktionsfehlers ausgerichtet ist.

Das zur Berechnung optimaler Netzgewichte überwiegend eingesetzte Verfahren ist der von Rumelhart, Hinton und Willams [123] beschriebene Backpropagation-Algorithmus, welcher auf dem Verfahren des steilsten Abstiegs [39, 79] beruht. Dieser Algorithmus besitzt jedoch eine sehr geringe Konvergenzgeschwindigkeit, die zudem bei komplexen Optimierungsproblemen und in der Nähe lokaler bzw. globaler Minima weiter abnimmt. Es existiert eine Reihe von Vorschlägen zur Modifikation dieses Verfahrens sowie zum Einsatz anderer Optimierungsverfahren, die eine höhere Konvergenzgeschwindigkeit versprechen.

Ziel der vorliegenden Arbeit bildet die Entwicklung eines nichtlinearen Optimierungsverfahrens, welches speziell auf die Optimierung der Netzgewichte der hier verwendeten mehrlagigen Perzeptron-Netze ausgerichtet ist und im Vergleich zu den in der Literatur eingesetzten Verfahren eine wesentlich höhere Konvergenzgeschwindigkeit und Robustheit aufweist. Das neuartige Verfahren der Optimierung **Lage für Lage** wird als *OLL*-Verfahren bezeichnet. Neben diesem werden mehrere Varianten des Backpropagation-Algorithmus, 7 Varianten des Verfahrens der konjugierten Gradienten sowie eine Version der Quasi-Newton-Verfahren zur Optimierung der mehrlagigen Perzeptron-Netze eingesetzt. Die Beurteilung der Leistungsfähigkeit des vorgestellten *OLL*-Verfahrens im Vergleich zu den bekannten Optimierungsverfahren wird auf Basis der bei einer Vielzahl experimenteller Untersuchungen erzielten Prädiktionsergebnisse durchgeführt. Der Vergleich der unterschiedlichen Optimierungsverfahren bezüglich der Konvergenzgeschwindigkeit und der erzielten Prädiktionsgewinne bildet einen weiteren Schwerpunkt der Arbeit, da für eine Vielzahl der verwendeten Verfahren aus der Literatur nur sehr eingeschränkte Ergebnisse diesbezüglich vorliegen.

Die vorliegende Arbeit ist wie folgt gegliedert:

Nach dieser Einführung gibt das *zweite* Kapitel einen kurzen Überblick über die Theorie dynamischer Systeme mit chaotischem Verhalten. Hierbei liegt der Schwerpunkt darauf, Begriffe und Verfahren einzuführen, die im Rahmen dieser Arbeit zur Modellierung dynamischer Systeme und zur Prädiktion chaotischer Zeitsignale mittels mehrlagiger Perzeptron-Netze Verwendung finden. In diesem Zusammenhang wird auf den Begriff der

fraktalen Dimension und die Definition der Lyapunov-Exponenten chaotischer Systeme eingegangen sowie der Treppen-Iterations-Algorithmus zur Berechnung der Lyapunov-Exponenten vorgestellt. Anschließend werden das Verfahren der zeitverzögerten Komponenten zur Rekonstruktion von Zustandsvektoren aus einem skalaren Zeitsignal und der Einsatz neuronaler Netze zur Modellierung der Abbildungseigenschaften dynamischer Systeme erläutert.

Das *dritte* Kapitel führt die zur Approximation der Abbildung des dynamischen Systems auf Basis der rekonstruierten Zustandsvektoren verwendeten mehrlagigen Perzeptron-Netze ein und beschreibt die im Rahmen der experimentellen Untersuchungen eingesetzten Varianten des Backpropagation-Algorithmus, der Verfahren der konjugierten Gradienten sowie des Quasi-Newton-Verfahrens.

Die Herleitung des neu entwickelten *OLL*-Verfahrens zur nichtlinearen Optimierung mehrlagiger Perzeptron-Netze erfolgt im *vierten* Kapitel, bevor im fünften Kapitel auf die erzielten Ergebnisse eingegangen wird.

Im *fünften* Kapitel steht zunächst der Vergleich des *OLL*-Verfahrens mit verschiedenen Varianten des Backpropagation-Algorithmus am Beispiel eines sehr einfachen mehrlagigen Perzeptron-Netzes im Vordergrund. Diese Untersuchung ermöglicht einen geometrisch anschaulichen Vergleich der Eigenschaften der eingesetzten Verfahren. Im weiteren Verlauf von Kapitel fünf werden die Ergebnisse diskutiert, die für die vier untersuchten chaotischen Zeitsignale unter Verwendung der in Kapitel drei und vier beschriebenen Optimierungsverfahren erzielt wurden. Den Abschluß von Kapitel fünf bildet die Ableitung eines neuen Verfahrens zur Bestimmung der Lyapunov-Exponenten des zugrundeliegenden dynamischen Systems auf Basis der Folge rekonstruierter Zustandsvektoren und der optimierten mehrlagigen Perzeptron-Netze. Die für das Lorenz-System und die Hénon-Abbildung erzielten Ergebnisse werden vorgestellt.

Die Zusammenfassung der Arbeit erfolgt in Kapitel *sechs*.

Im Anhang wird auf die untersuchten chaotischen Systeme und deren Eigenschaften kurz eingegangen. Weiterhin sind im Glossar Definitionen einiger der verwendeten mathematischen Begriffe angegeben.

2 Dynamische Systeme und neuronale Netze

Das von Lorenz im Jahre 1963 [78] beschriebene System aus drei gekoppelten, nicht-linearen gewöhnlichen Differentialgleichungen erster Ordnung weist ein sehr komplexes und über längere Zeiträume nicht vorhersagbares dynamisches Verhalten auf, obwohl es deterministischer Natur ist. Eine besondere Eigenschaft dieses Systems ist, daß das dynamische Verhalten in sehr hohem Maße von den gewählten Anfangsbedingungen und Systemparametern abhängig ist. Diese Eigenschaften, welche auch bei einer Vielzahl anderer niedrigdimensionaler nichtlinearer dynamischer Systeme beobachtet werden, werden als (deterministisch-) chaotisches Verhalten bezeichnet. Für die Beschreibung solcher dynamischer Systeme hat sich ein neuer Zweig der nichtlinearen Systemtheorie entwickelt, für den sich der Begriff *deterministisches Chaos* oder auch einfach *Chaos-Theorie* durchgesetzt hat. Grundsätzlich ist zwischen zeitkontinuierlichen und zeitdiskreten Systemen zu unterscheiden. Die numerische Untersuchung kontinuierlicher Systeme unter Verwendung eines Digitalrechners setzt jedoch eine zeitliche Diskretisierung des dynamischen Verhaltens des beobachteten Systems voraus. Dieses wird durch eine Abtastung von Ausgangsgrößen physikalischer Systeme mit zumeist konstanter Abtastperiode sowie durch die Verwendung einer endlichen Schrittweite bei der Lösung mathematischer Modelle erzielt. Letztlich werden Untersuchungen an zeitkontinuierlichen Systemen durch die Simulation mittels zeitdiskreter Systeme realisiert.

2.1 Dynamische Systeme

Das dynamische Verhalten eines zeitkontinuierlichen Systems ist durch eine vektorwertige funktionale Abhängigkeit von inneren Größen des Systems, den *Zustandsvariablen* $\mathbf{x}(t) \in \mathbb{R}^{d_S}$, und eventuell vorhandenen *Eingangsgrößen* $\mathbf{u}(t) \in \mathbb{R}^{d_V}$ gekennzeichnet. Die minimal notwendige Anzahl an Zustandsvariablen zur Charakterisierung des Systems ist die *Ordnung* d_S des Systems. Ein Vorteil der Beschreibung durch Zustandsvariablen ist, daß sich diesen eine topologische Bedeutung zumessen läßt. Werden die Zustandsvariablen als Koordinaten eines Raumpunktes aufgefaßt, ist zu jedem Zeitpunkt t der Zustand des Systems durch einen Punkt $\mathbf{x}(t)$ in einem d_S -dimensionalen Raum charakterisiert. Der durch die möglichen Werte der Zustandsvariablen aufgespannte Raum wird als Zustandsraum \mathbf{M} des Systems bezeichnet. Dieser kann z.B. ein d_S -dimensionaler euklidischer Raum, eine kompakte (B.13) *Mannigfaltigkeit* (B.16) oder ein ∞ -dimensionaler Hilbertraum² sein. Eine ausführlichere mathematische Beschreibung findet sich bei Eck-

²Als Beispiel sei die Mackey-Glass-Differentialgleichung (A.8) angeführt. Hierbei handelt es sich um eine Differentialgleichung mit retardiertem Argument, die sowohl vom aktuellen Zustand $x(t)$ als auch von dem vergangenen (retardierten) Zustand $x(t - \Delta T)$ abhängig ist. Demzufolge stellt der Verlauf der Zustandsgröße $x(t')$ für $(t - \Delta T) < t' < t$ den Anfangszustand des Systems dar. Der Zustandsraum ist demnach ∞ -dimensional.

mann und Ruelle [24].

Ein *offenes* (B.18) *zeitvariables* kontinuierliches System sei durch Zustandsgleichungen der Form

$$\begin{aligned} \mathbf{F}_\mu &: \mathbb{R}^{ds} \times \mathbb{R}^{dv} \times \mathbb{R} \rightarrow \mathbb{R}^{ds} \\ \frac{d\mathbf{x}(t)}{dt} &= \mathbf{F}_\mu(\mathbf{x}(t), \mathbf{x}(t - \Delta T), \mathbf{u}(t), t) \quad \text{mit} \quad \mathbf{x}(t_0) = \mathbf{x}_0 \in \mathbb{R}^{ds} \end{aligned} \quad (2.1)$$

beschrieben. Dieses System von Differentialgleichungen erster Ordnung kann aus gewöhnlichen oder partiellen Differentialgleichungen, aber auch aus Differentialgleichungen mit retardierten Argumenten² bestehen.

Durch Gleichung (2.1) wird der Zustand des Systems $\mathbf{x}(t)$ bei Kenntnis der Eingangsgrößen $\mathbf{u}(t)$ für $t > t_0$ und des *Anfangszustandes* \mathbf{x}_0 zum *Anfangszeitpunkt* t_0 vollständig beschrieben. Die zeitliche Entwicklung des Systems ist durch das *Vektorfeld* \mathbf{F}_μ (B.26) vorgegeben. Dieses weist im allgemeinen eine nichtlineare Abhängigkeit von \mathbf{x} , \mathbf{u} und t auf. Hierbei kennzeichnet der Index μ Kontrollparameter des Systems, welche die dynamischen Eigenschaften des Systems wesentlich beeinflussen. Ein Beispiel für diese auch als *Bifurkationsparameter* bezeichneten Größen sind die Parameter σ , r und b des Lorenzmodells (A.1), welches im Anhang näher vorgestellt wird. Bei den im Rahmen dieser Arbeit untersuchten Systemen werden die Kontrollparameter als konstant vorausgesetzt, so daß auf deren Einfluß im folgenden nicht weiter eingegangen wird.

Für den Fall verschwindender Eingangsgrößen, $\mathbf{u}(t) \equiv 0$, spricht man von einem *abgeschlossenen* (B.1) zeitvariablen System mit der gegenüber Gleichung (2.1) vereinfachten Zustandsgleichung

$$\mathbf{F}: \mathbb{R}^{ds} \times \mathbb{R} \rightarrow \mathbb{R}^{ds} \quad ; \quad \frac{d\mathbf{x}(t)}{dt} = \mathbf{F}(\mathbf{x}(t), t) \quad \text{mit} \quad \mathbf{x}(t_0) = \mathbf{x}_0 \in \mathbb{R}^{ds} \quad (2.2)$$

Ist darüber hinaus die rechte Seite der Gleichung (2.2), außer bei $\mathbf{x}(t)$, nicht von t abhängig, liegt ein *stationäres*, abgeschlossenes System³ vor. Diese als *autonom* bezeichneten Systeme werden durch Zustandsgleichungen der Form

$$\mathbf{F}: \mathbb{R}^{ds} \rightarrow \mathbb{R}^{ds} \quad ; \quad \frac{d\mathbf{x}(t)}{dt} = \mathbf{F}(\mathbf{x}(t)) \quad \text{mit} \quad \mathbf{x}(t_0) = \mathbf{x}_0 \in \mathbb{R}^{ds} \quad (2.3)$$

beschrieben. Hier kann ohne Einschränkung der Allgemeinheit der Anfangszeitpunkt zu Null gewählt werden.

Im Rahmen dieser Arbeit werden ausschließlich autonome Systeme gemäß Gl.(2.3) untersucht. Da es in vielen Fällen möglich ist, offene zeitvariable Systeme durch Einführung

³In der mathematischen Literatur wird als „dynamisches System“ i. allg. ein stationäres, abgeschlossenes System verstanden

zusätzlicher Zustandsgrößen als autonome Systeme zu beschreiben, stellt die ausschließliche Betrachtung autonomer Systeme keine wesentliche Einschränkung dar [110, 105, 120, 125].

Die Lösung $\mathbf{x}(t)$ der Zustandsgleichungen (2.3) in Abhängigkeit vom Anfangszustand \mathbf{x}_0 bildet eine stetige Kurve und beschreibt den Zustand des Systems für alle Zeitpunkte t . Diese Kurve ist die *Trajektorie* oder *Zustandskurve* des Systems durch den Anfangszustand \mathbf{x}_0 . Sie stellt eine Abbildung des Zustandsraumes auf sich selbst dar.

Die Abbildung ϕ_t des Anfangszustandes $\mathbf{x}(t_0)$ auf den Zustand $\mathbf{x}(t)$ wird als *Fluß* ϕ_t des Systems bezeichnet. Für offene stationäre Systeme gilt

$$\phi_t: \mathbb{R}^{d_S} \rightarrow \mathbb{R}^{d_S} \quad ; \quad \mathbf{x}(t) = \phi_t(\mathbf{x}_0, \mathbf{u}_0, t_0) \quad \text{mit} \quad \mathbf{u}(t_0) = \mathbf{u}_0 \in \mathbb{R}^{d_U} \quad (2.4)$$

Die entsprechende Beschreibung für autonome Systeme lautet

$$\phi_t: \mathbb{R}^{d_S} \rightarrow \mathbb{R}^{d_S} \quad ; \quad \mathbf{x}(t) = \phi_t(\mathbf{x}_0) \quad (2.5)$$

In beiden Fällen besitzt der Fluß die Eigenschaft⁴

$$\phi_{t_1+t_2}(\mathbf{x}_0) = (\phi_{t_1} \circ \phi_{t_2})(\mathbf{x}_0) = (\phi_{t_2} \circ \phi_{t_1})(\mathbf{x}_0) \quad ; \quad \phi_0(\mathbf{x}_0) = \mathbf{x}_0 \quad (2.6)$$

Aus der Tatsache, daß der Fluß $\phi_t(\mathbf{x}_0, t_0)$ bzw. $\phi_t(\mathbf{x}_0)$ im allgemeinen (B.3) eine diffeomorphe Abbildung (B.7) des Zustandsraumes darstellt, ergeben sich einige wichtige Eigenschaften für die Trajektorien des betrachteten Systems [110, 105]

- Die Trajektorien autonomer Systeme sind durch die Anfangszustände vollständig beschrieben und besitzen keine Kreuzungspunkte
 $\phi_t(\mathbf{x}_1) \neq \phi_t(\mathbf{x}_2) \quad \forall \quad t \quad \Leftrightarrow \quad \mathbf{x}_1 \neq \mathbf{x}_2$
- Die Trajektorien offener stationärer Systeme sind bei Kenntnis der Eingangsgröße durch die Anfangszustände sowie die Anfangszeitpunkte eindeutig beschrieben und besitzen bei gleichen Anfangszuständen und -zeitpunkten keine Kreuzungspunkte
 $\phi_t(\mathbf{x}_1, t_0) = \phi_t(\mathbf{x}_2, t_0) \quad \forall \quad t, t_0 \quad \Leftrightarrow \quad \mathbf{x}_1 = \mathbf{x}_2$
- Die Ableitung der Trajektorien des Systems bezüglich des Anfangszustandes existiert. Eine stetige Änderung des Anfangszustandes von \mathbf{x}_0 in \mathbf{x}_1 hat eine stetige Änderung der Lösung des Systems von $\phi_t(\mathbf{x}_0)$ in $\phi_t(\mathbf{x}_1)$ zur Folge. Die Linearisierung der Änderung der Trajektorie bezüglich des Anfangszustandes führt auf die *Variationsgleichung*. Diese Gleichung wird zur Definition der *Lyapunov-Exponenten seltsamer Attraktoren* herangezogen (Kap.2.4).

⁴Für die Produktabbildung gilt: $(\phi_{t_1} \circ \phi_{t_2})(\mathbf{x}_0) = \phi_{t_2}(\phi_{t_1}(\mathbf{x}_0))$

Eine den Gleichungen (2.1), (2.2) und (2.3) entsprechende Einteilung kann auch für zeitdiskrete Systeme erfolgen. Zeitdiskrete Systeme werden auch als *iterierte Karten* bezeichnet. Sie stellen Abbildungen des Zustandsraumes auf sich selbst dar (ähnlich dem Fluß kontinuierlicher Systeme Gl.(2.4), (2.5)) und werden i. allg. durch ein System von Differenzgleichungen erster Ordnung beschrieben.

Ein verschiebungsvariantes zeitdiskretes System ist durch ein Differenzgleichungssystem der Form

$$\mathcal{F} : \mathbb{R}^{d_s} \times \mathbb{Z} \rightarrow \mathbb{R}^{d_s} \quad ; \quad \mathbf{x}^{k+1} = \mathcal{F}(\mathbf{x}^k, k) \quad \text{mit} \quad \mathbf{x}^{k+1}, \mathbf{x}^k \in \mathbb{R}^{d_s} \quad , \quad k \in \mathbb{Z} \quad (2.7)$$

definiert. Die vektorwertige und i. allg. nichtlineare Funktion \mathcal{F} bildet den aktuellen Zustand⁵ \mathbf{x}^k des Systems auf den nächsten Zustand \mathbf{x}^{k+1} ab. Entsprechend ist ein autonomes zeitdiskretes System durch die Abbildung

$$\mathcal{F} : \mathbb{R}^{d_s} \rightarrow \mathbb{R}^{d_s} \quad ; \quad \mathbf{x}^{k+1} = \mathcal{F}(\mathbf{x}^k) \quad \text{mit} \quad \mathbf{x}^{k+1}, \mathbf{x}^k \in \mathbb{R}^{d_s} \quad , \quad k \in \mathbb{Z} \quad (2.8)$$

gekennzeichnet. Bei zeitdiskreten Systemen wird die Folge von Zustandsvektoren $\{\mathbf{x}^k\}$ als *Orbit* bezeichnet.

2.2 Stationäre Lösungen dynamischer Systeme, Grenzmengen und Attraktoren

Die Lösung der Zustandsgleichungen eines dynamischen Systems entspricht der zeitlichen Entwicklung des Systemzustandes im Zustandsraum. Die stationäre Lösung beschreibt das asymptotische Verhalten des Systems für $t \rightarrow \infty$, wobei der Übergangsbereich als *Transiente* bezeichnet wird. Die Existenz einer stationären Lösung setzt ein stabiles System voraus, dessen Trajektorien für $t \rightarrow \infty$ eine Untermenge des Zustandsraumes bilden.

In den weiteren Ausführungen werden ausschließlich dissipative Systeme (B.8) betrachtet, da diese gegenüber klassischen Systemen, sog. *Hamiltonschen-Systemen*, für die Untersuchung realer physikalischer Vorgänge eine wesentlich größere Bedeutung haben [105, 24, 126]. Die besondere Eigenschaft dissipativer dynamischer Systeme besteht darin, daß Volumina im Zustandsraum durch die Wirkung der Vektorfelder \mathbf{F} bzw. \mathcal{F} im Laufe der Zeit komprimiert werden [105, 24, 126]. Zur Veranschaulichung dieses Vorganges sind in Abbildung 2.1 die Lösungen des Lorenz-Systems (A.1), welches einen 3-dimensionalen Zustandsraum besitzt, für acht Anfangszustände über einen kurzen Zeitraum dargestellt. Die Anfangszustände bilden die Ecken eines Würfels, welcher bei abnehmendem Volumen in der dargestellten Weise verformt wird (vergl. auch Abbildung 2.5).

⁵Das k-te Element \mathbf{x}^k einer Folge $\{\mathbf{x}^k\}$ wird durch den hochgestellten Index k gekennzeichnet. Diese Beschreibung wurde gewählt, um bei Variablen mit mehreren Indizes eine größere Übersichtlichkeit zu gewährleisten.

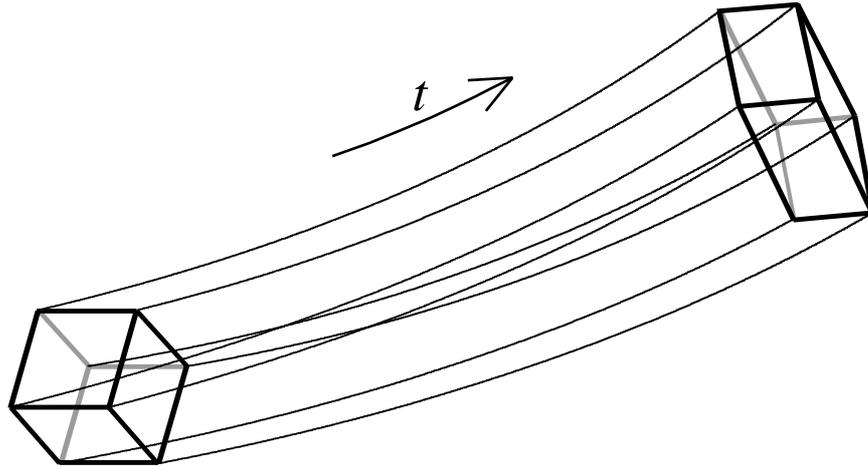


Abbildung 2.1: Darstellung von acht Trajektoriensegmenten des Lorenz-Systems über einen kurzen Zeitraum (0,04 Sekunden). Die Anfangszustände bilden die Ecken eines Würfels. Dieser wird gemäß den Zustandsgleichungen des Systems verformt, wobei sein Volumen abnimmt.

Werden die Lösungen der Zustandsgleichung eines dynamischen Systems für alle möglichen Anfangszustände $\mathbf{x}_0 \in \mathbf{M}$ betrachtet, so entspricht die Dimension dieser Menge zunächst der des Zustandsraumes d_S . Alle Trajektorien dissipativer Systeme werden jedoch auf Grund der Kontraktion von Volumenelementen asymptotisch für $t \rightarrow \infty$ gegen Teilmengen des Zustandsraumes mit i. allg. geringerer Dimension d_A konvergieren [105, 24, 126]. Diese *abgeschlossenen, beschränkten* (B.4) und *zusammenhängenden* Gebiete des Zustandsraumes werden als *Grenzmengen (limit sets)* Ω bezeichnet [105, 24, 126]. Die Menge aller Anfangszustände, deren asymptotische Lösungen einer bestimmten Grenzmenge Ω entsprechen, heißt *Einzugsgebiet U (basin of attraction)* der Grenzmenge Ω . Ist das Einzugsgebiet U größer als die Grenzmenge Ω , so nennt man diese Grenzmenge anziehend (attracting). Aufgrund dieser Eigenschaft wird eine anziehende Grenzmenge Ω üblicherweise als *Attraktor \mathcal{A}* bezeichnet [105, 24, 126].

Bei den möglichen Arten stationären Verhaltens dynamischer Systeme werden seit der Beschreibung chaotischen Verhaltens durch Lorenz [78] vier Typen unterschieden. Diese werden im folgenden vorgestellt und anhand ihrer Trajektorien, Attraktoren und Leistungsdichtespektren verglichen [105, 24, 126].

Die einfachste Form der stationären Lösung einer Zustandsgleichung gemäß Gl.(2.3) ist ein *Gleichgewichtspunkt*, für den gilt

$$\phi_t(\mathbf{x}_{gl}) = \mathbf{x}_{gl} \quad (2.9)$$

Die Grenzmenge ist der Gleichgewichtspunkt \mathbf{x}_{gl} selbst. Die Fourier Transformierte jeder von 0 verschiedenen Komponente von \mathbf{x}_{gl} besitzt einen δ -Impuls bei der Frequenz 0. Die

Dimension (B.22) dieser Grenzmenge ist die eines Punktes, also 0. Alle stabilen linearen Systeme weisen als stationäre Lösung einen Gleichgewichtspunkt auf [105, 24, 126].

Die nächst kompliziertere Form stellt die *periodische Lösung* dar, deren Grenzmenge eine geschlossene Kurve im Zustandsraum ist und folglich die Dimension 1 hat [105, 24, 126]. Der Fluß hat die Eigenschaft

$$\phi_t(\mathbf{x}_0) = \phi_{t+T}(\mathbf{x}_0) \quad (2.10)$$

wobei $T > 0$ die Periodendauer der Lösung darstellt. Die Fourier-Transformierte besteht aus Spektrallinien bei den Frequenzen $f_i = i/T$; $i \in \mathbb{N}$. Nicht alle Frequenzen f_i müssen vertreten sein.

Die *quasiperiodische Lösung* entsteht aus allen ganzzahligen Linearkombinationen von $p \in \mathbb{N}$ periodischen Lösungen, deren Frequenzen in einem irrationalen Verhältnis zueinander stehen. Daher entspricht die Grenzmenge dieser stationären Lösung keiner geschlossenen Kurve, sondern stellt eine diffeomorphe Abbildung eines p -Torus in den Zustandsraum des Systems dar und hat die Dimension p . Die Fourier Transformierte der quasiperiodischen Lösung besteht aus der Überlagerung der Spektren der periodischen Lösungen und ist somit ebenfalls ein Linienspektrum [105, 24, 126].

Die bisher beschriebenen stationären Lösungen dynamischer Systeme haben eine wesentliche Gemeinsamkeit. Der anfängliche Abstand zweier Trajektorien mit Anfangszuständen innerhalb des Einzugsgebietes einer Grenzmenge wird mit zunehmender Zeit immer kleiner. Diese Eigenschaft ist nötig, um das Systemverhalten über längere Zeiträume vorherzusagen, da unvermeidliche Fehler bei der Bestimmung des Anfangszustandes zu vernachlässigbaren Fehlern in der Vorhersage führen.

Neben den aufgeführten klassischen stationären Lösungen ist spätestens mit der Veröffentlichung von Lorenz [78] die Existenz einer weiteren stabilen Grenzmenge nichtlinearer dynamischer Systeme allgemein bekannt. Für diese Grenzmenge wurde von Ruelle und Takens [124] der Begriff *seltsame Grenzmenge* oder entsprechend *seltsamer Attraktor* (*strange attractor*) geprägt. Systeme mit seltsamen Attraktoren nennt man *chaotische Systeme* oder Systeme mit *deterministisch chaotischem Verhalten*. Für eine mathematisch genauere Definition dieser Begriffe sei auf die Literatur [24, 124] und [126] verwiesen.

Das dynamische Verhalten auf seltsamen Attraktoren ist äußerst komplex und durch extreme Empfindlichkeit gegenüber Änderungen der Anfangswerte charakterisiert. Diese sensitive Abhängigkeit von Anfangswerten beruht auf einem exponentiellen Auseinanderstreben zweier anfänglich benachbarter Trajektorien und ist am Beispiel der Hénon-Abbildung (vgl. Anhang A.3) in Abbildung 2.2 dargestellt. Auf Grund dieser Eigenschaft chaotischer Systeme ist eine Prädiktion des Systemzustandes über längere Zeiträume unmöglich, da Fehler bei der Bestimmung von Ausgangsdaten unvermeidlich sind und diese im Laufe der Zeit exponentiell anwachsen. Da jedoch auch seltsame Attraktoren auf einen Bereich des Zustandsraumes begrenzt sind, kann ein exponentielles Auseinander-

streben zweier Trajektorien nur lokal erfolgen und muß von einer globalen „Faltung“ des Attraktors überlagert sein [105, 24, 126].

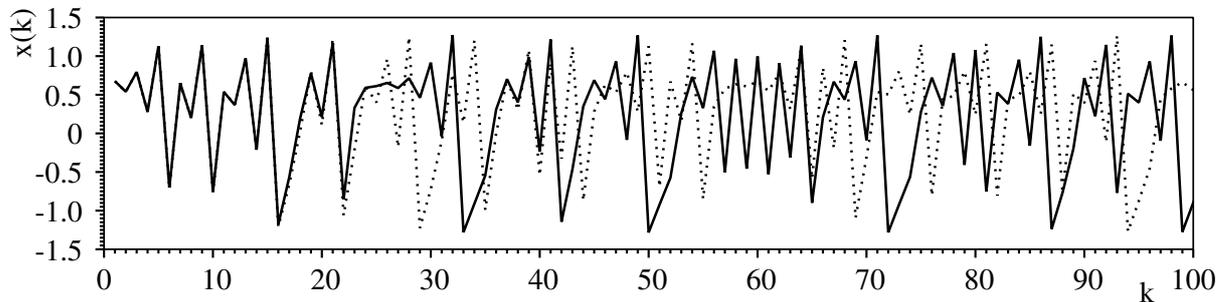


Abbildung 2.2: Darstellung der x-Komponente zweier Trajektoriensegmente der Hénon-Abbildung über 100 Iterationen der Abbildung für zwei Anfangszustände, die sich in der x-Komponente nur um 0,001% unterscheiden.

Expansion und Kontraktion finden gleichzeitig in unterschiedlichen Raumrichtungen statt, wobei der Effekt der Kontraktion auf Grund des dissipativen Charakters der hier betrachteten Systeme insgesamt überwiegt. Die unterschiedlichen Raten, mit denen Abstände benachbarter Trajektorien in einzelnen Raumrichtungen zu- bzw. abnehmen, stellen charakteristische Größen dynamischer Systeme dar und werden mit Hilfe der Lyapunov-Exponenten definiert, auf die weiter unten näher eingegangen wird.

Der angesprochene Prozeß des „Streckens“ und „Faltens“, welcher in der Literatur anschaulich mit dem Kneten eines Teiges verglichen wird, ist einer ständigen Wiederholung unterworfen und hat die sehr komplexe, als fraktal bezeichnete Struktur seltsamer Attraktoren zur Folge. Die Dimension dieser Grenzmengen ist i. allg. nicht ganzzahlig. Man spricht von der *fraktalen Dimension* seltsamer Attraktoren. Diese stellt eine weitere wichtige Kenngröße chaotischer Systeme dar, auf die in Kapitel 2.3 näher eingegangen wird.

2.3 Fraktale Dimensionen seltsamer Attraktoren

Stationäre Lösungen klassischer dynamischer Systeme weisen ausschließlich Grenzmengen mit ganzzahliger Dimension auf. Für diese sind bekannte Begriffe wie die Dimension des Euklidischen Raumes oder die *topologische Dimension* (B.22) einer Mannigfaltigkeit ausreichend. Demgegenüber muß zum Vergleich seltsamer Attraktoren der übliche Dimensionsbegriff um die Definition fraktaler Dimensionen erweitert werden.

Die (fraktale) Dimension (seltsamer) Attraktoren stellt ein Maß für die Komplexität der stationären Lösung des Systems dar. Die Dimension einer Grenzmenge entspricht den Freiheitsgraden des Systems auf dem Attraktor und gibt die minimale Anzahl von Zustandsvariablen an, die zur Beschreibung der stationären Lösung notwendig ist.

Weiterhin ist die Kenntnis der Dimension eines Attraktors erforderlich, um mit dem im Kapitel 2.5 beschriebenen Verfahren aus einem gemessenen skalaren Ausgangssignal eines Systems eine Folge von Zustandsvektoren zu rekonstruieren. Basierend auf dieser Rekonstruktion kann das dynamische Verhalten des Systems mit Hilfe eines Modells nachgebildet und in beschränktem Maße auch prädiziert werden.

Hier wird nur eine kleine Auswahl aus der möglichen Vielzahl fraktaler Dimensionsbegriffe vorgestellt. Diese haben besondere praktische Bedeutung bei der Untersuchung experimentell ermittelter Daten erlangt, da zu ihrer Berechnung leistungsfähige Algorithmen zur Verfügung stehen [105, 24, 43, 44, 110, 120]. Im einzelnen sind dies die *Kapazität* d_K , die *Informationsdimension* [16, 105, 59] d_I und im besonderen Maße die *Korrelationsdimension* d_C . Wobei im Rahmen dieser Arbeit nur die *Kapazität* d_K und die Korrelationsdimension d_C Verwendung finden. Es kann gezeigt werden [105], daß d_K , d_I und d_C einem Teil der sog. *Renyi-Dimensionen* α -ter Ordnung d_α mit $\alpha \in \mathbb{N}$ [118] entsprechen und daher in folgender Relation zueinander stehen

$$d_K = d_{\alpha=0} \geq d_I = d_{\alpha=1} \geq d_C = d_{\alpha=2} \quad (2.11)$$

2.3.1 Kapazität

Die Kapazität ist ein häufig verwendeter Dimensionsbegriff und stellt eine abgewandelte Definition der Hausdorff-Dimension dar. Sie entspricht, wie bereits angegeben, der Renyi-Dimension 0-ter Ordnung.

Es wird ein Attraktor $\mathcal{A} \subset \mathbb{R}^{d_s}$ betrachtet. Gesucht ist eine *Überdeckung* (B.24) von \mathcal{A} mit einer minimalen Anzahl von Volumenelementen (Kugeln, Kuben etc.) $N(\varepsilon)$, die alle die Abmessung ε (Durchmesser, Kantenlänge etc.) besitzen.

Die Kapazität d_K des Attraktors ist dann als Grenzwert⁶ für $\varepsilon \rightarrow 0$

$$d_K := \lim_{\varepsilon \rightarrow 0} \frac{\ln N(\varepsilon)}{\ln(1/\varepsilon)} \quad (2.12)$$

definiert, falls dieser Grenzwert existiert. Andernfalls ist die Kapazität für den Attraktor \mathcal{A} nicht definiert. Die Kapazität von Attraktoren nicht-chaotischer Systeme stimmt bis auf wenige Ausnahmen mit deren topologischer Dimension überein.

Die algorithmische Umsetzung dieses Dimensionsbegriffes ist nicht aufwendig, stößt jedoch für Systeme mit relativ hochdimensionalen Zustandsräumen sehr schnell an Speicherplatzgrenzen [105].

⁶Führen verschiedene Überdeckungen zu unterschiedlichen Werten von d_K , so ist d_K als der minimale Wert definiert, der sich für diese Überdeckungen ergibt.

2.3.2 Korrelationsdimension

Von besonderer praktischer Bedeutung ist die Korrelationsdimension d_C , da zu deren Berechnung von Grassberger und Procaccia [41, 42, 43, 44] ein effizienter und relativ einfach zu implementierender Algorithmus angegeben wurde.

Die Definition der Korrelationsdimension basiert auf der Überdeckung des Attraktors durch Volumenelemente, die bereits zur Definition von d_K betrachtet wurde. Sie lautet

$$d_C := \lim_{\varepsilon \rightarrow 0} \frac{\ln \sum_{i=1}^{N(\varepsilon)} P(i)^2}{\ln \varepsilon} \quad (2.13)$$

wobei P_i wiederum die Wahrscheinlichkeit angibt, mit der sich die Trajektorie im Volumenelement i befindet. Zur Berechnung von d_C wird angenommen, daß eine Folge von N Zustandsvektoren $\{\mathbf{x}^k\}$ gemessen oder berechnet wurde. Liegt nur eine skalare Folge von Meßwerten der Ausgangsgröße eines Systems vor, so kann aus dieser mit dem im Kapitel 2.5 beschriebenen Verfahren zur Rekonstruktion von Attraktoren eine Folge von Zustandsvektoren rekonstruiert werden.

Der Algorithmus von Grassberger und Procaccia beruht auf einem Zusammenhang zwischen der Korrelationsdimension gemäß Gl.(2.13) und dem sog. *Korrelationsintegral*

$$C(\varepsilon) = \lim_{N \rightarrow \infty} \frac{2}{N(N-1)} \left\{ \text{Anzahl von Paaren } (\mathbf{x}^i, \mathbf{x}^j) \text{ mit } \|\mathbf{x}^i - \mathbf{x}^j\| < \varepsilon \right\} \quad \text{mit } i < j \quad (2.14)$$

welches recht einfach aus der Folge von Zustandsvektoren $\{\mathbf{x}^k\}$ geschätzt werden kann. In Gleichung (2.14) stellt $\|\cdot\|$ eine geeignete Norm dar. Üblicherweise werden die Euklidische Norm (B.17)

$$\|\mathbf{x}^i - \mathbf{x}^j\|_2 = \sum_{l=1}^{d_E} (x_l^i - x_l^j)^2 \quad (2.15)$$

oder die Maximumsnorm (B.17)

$$\|\mathbf{x}^i - \mathbf{x}^j\|_\infty = \max_l |x_l^i - x_l^j| \quad \text{mit } 1 < l < d_E \quad (2.16)$$

verwendet.

Die Überdeckung des Attraktors erfolgt im ersten Fall durch Hyperkugeln mit dem Durchmesser ε und im zweiten Fall mit Hyper-Würfeln der Kantenlänge ε . Da die Ergebnisse von der verwendeten Norm unabhängig sind und die Berechnung der Maximumsnorm einen erheblich geringeren Rechenaufwand erfordert, wird diese bevorzugt eingesetzt. Einige Beispiele zur Dimensionsberechnung mit dem Verfahren von Grassberger und Procaccia sind im Anhang A für die im Rahmen dieser Arbeit untersuchten chaotischen Zeitsignale angegeben.

Existiert der Grenzwert

$$\lim_{\varepsilon \rightarrow 0} \frac{\ln C(\varepsilon)}{\ln \varepsilon} = d_C \quad (2.17)$$

so stimmt dieser mit der Korrelationsdimension d_C überein [105, 43].

Wird das Korrelationsintegral $C(\varepsilon)$ in doppelt-logarithmischer Darstellung über dem Abstand ε aufgetragen, so ergibt sich für $\varepsilon \rightarrow 0$ eine Gerade, deren Steigung der Korrelationsdimension entspricht.

Da mit dem Algorithmus von Grassberger und Procaccia ein effizientes Verfahren zur Verfügung steht, mit dem eine fraktale Dimension eines Attraktors auch aus experimentell ermittelten Daten bestimmt werden kann [11, 14, 16, 41, 42, 43, 51, 73, 86], hat sich die Korrelationsdimension in den letzten Jahren durchgesetzt. Wie bei anderen Verfahren muß auch hier die Bewertung der Ergebnisse mit großer Sorgfalt erfolgen, um Fehlinterpretationen aufgrund einer zu geringen Datenmenge oder ungenügender Genauigkeit der Daten zu vermeiden. Hinweise zur algorithmischen Umsetzung und zur Abschätzung der bei den verschiedenen Verfahren entstehenden Fehler finden sich unter anderem bei [105, 43, 59, 60, 110, 120, 127].

2.4 Lyapunov-Exponenten

Einen Hinweis, über welche Zeiträume das Verhalten von dynamischen Systemen vorhersehbar ist, gestatten die Lyapunov-Exponenten. Sie stellen eine weitere charakteristische Größe stationärer Lösungen dynamischer Systeme dar, auf deren Definition im folgenden näher eingegangen wird.

Die Lyapunov-Exponenten beschreiben die Raten, mit denen Abstände benachbarter Trajektorien in verschiedenen Raumrichtungen zu- bzw. abnehmen und werden zur Stabilitätsanalyse aller in Kapitel 2.2 vorgestellten stationären Lösungen dynamischer Systeme eingesetzt. Die zur Stabilitätsanalyse von Gleichgewichtspunkten und periodischen Lösungen verwendeten Eigenwerte der linearisierten Systemfunktion gehen als Sonderfall aus der Definition der Lyapunov-Exponenten hervor.

Zur Definition der Lyapunov-Exponenten wird zunächst der Fluß $\phi_t(\mathbf{x}_0)$ gemäß Gl.(2.5) als Lösung der Zustandsgleichung gemäß Gl.(2.3) betrachtet.

$$\dot{\phi}_t(\mathbf{x}_0) = \mathbf{F}(\phi_t(\mathbf{x}_0)) \quad \text{mit} \quad \dot{\phi}_t(\mathbf{x}_0) = \frac{d\phi_t(\mathbf{x}_0)}{dt} \quad (2.18)$$

Die Auswirkung einer Änderung der Anfangsbedingungen auf den Verlauf der Trajektorie wird in erster Näherung durch den linearisierten Fluß beschrieben. Hierzu wird die partielle Ableitung von Gleichung (2.18) bezüglich des Anfangszustandes \mathbf{x}_0 betrachtet, welche auf die Variationsgleichung [105, 24]

$$\frac{\partial \dot{\phi}_t(\mathbf{x}_0)}{\partial \mathbf{x}_0} = \frac{\partial \mathbf{F}(\phi_t(\mathbf{x}_0))}{\partial \phi_t} \cdot \frac{\partial \phi_t(\mathbf{x}_0)}{\partial \mathbf{x}_0} \quad \text{mit} \quad \frac{\partial \phi_{t_0}(\mathbf{x}_0)}{\partial \mathbf{x}_0} = \mathbf{I} \quad (2.19)$$

führt, wobei \mathbf{I} die $d_S \times d_S$ -Einheitsmatrix darstellt. Mit der abkürzenden Schreibweise

$$\Phi_t(\mathbf{x}_0) := \frac{\partial \phi_t(\mathbf{x}_0)}{\partial \mathbf{x}_0} \quad (2.20)$$

und Gleichung (2.5) erhält man die folgende Form der linearen nichtautonomen Matrix-Variationsgleichung der Trajektorie durch \mathbf{x}_0 :

$$\dot{\Phi}_t(\mathbf{x}_0) = \mathbf{F}_x(\mathbf{x}(t)) \cdot \Phi_t(\mathbf{x}_0) \quad \text{mit} \quad \Phi_{t_0}(\mathbf{x}_0) = \mathbf{I} \quad (2.21)$$

in der

$$\mathbf{F}_x(\mathbf{x}(t)) = \left. \frac{\partial \mathbf{F}(\phi_t(\mathbf{x}_0))}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}(t)} \quad (2.22)$$

die Jacobimatrix (B.12) der partiellen Ableitungen des Vektorfeldes \mathbf{F} an der Stelle $\mathbf{x}(t) = \phi_t(\mathbf{x}_0)$ bezeichnet.

Für zeitkontinuierliche dynamische Systeme ist die zeitabhängige Übergangsmatrix des linearisierten Systems durch die $d_S \times d_S$ -Matrix $\Phi_t(\mathbf{x}_0)$ gegeben. Diese ergibt sich als Lösung der Variationsgleichung (2.21) für die Anfangsbedingung $\Phi_{t_0}(\mathbf{x}_0) = \mathbf{I}$. Im Rahmen dieser Arbeit wird ein neuer Algorithmus zur Bestimmung der Lyapunov-Exponenten dynamischer Systeme unter Verwendung zeitdiskreter rekonstruierter Zustandsvektoren (vergl. Abschnitte 2.5 und 5.6) beschrieben. Daher ist die Definition der Lyapunov-Exponenten zeitdiskreter Systeme hier von besonderem Interesse. Für zeitkontinuierliche Systeme ergeben sich die Lyapunov-Exponenten auf Basis der Matrix $\Phi_t(\mathbf{x}_0)$ analog zu den im folgenden angegebenen Gleichungen (2.26), (2.27) und (2.28).

Im zeitdiskreten Fall wird die das System beschreibende Abbildung $\mathbf{x}^{k+1} = \mathcal{F}(\mathbf{x}^k)$ als Fluß $\mathbf{x}^{k+1} = \phi^1(\mathbf{x}^k)$ für die Evolutionszeit $\Delta T = 1$ aufgefaßt. Damit wird das zeitdiskrete System als ein mit der Periodendauer $\Delta T = 1$ abgetastetes zeitkontinuierliches System behandelt. Die das System charakterisierenden Größen wie die Lyapunov-Exponenten oder die fraktalen Dimensionen werden durch diese Vorgehensweise nicht beeinflusst. Im Umkehrschluß kann demnach die Modellierung kontinuierlicher dynamischer Systeme auf der Basis zeitdiskreter Modellansätze erfolgen [126][S.51] (vgl. Abschnitt 2.5).

Mit dieser Überlegung folgt aus der Zustandsgleichung des zeitdiskreten Systems für \mathbf{x}^k :

$$\mathbf{x}^k = \mathcal{F}(\mathbf{x}^{k-1}) = \mathcal{F}^{(k)}(\mathbf{x}^0) = \phi^k(\mathbf{x}^0) = \phi^1(\phi^1(\dots \phi^1(\mathbf{x}^0) \dots)) \quad (2.23)$$

Die Anwendung der Kettenregel zur Differentiation des Flusses bezüglich des Anfangszustandes ergibt:

$$\begin{aligned} \Phi_k(\mathbf{x}^0) &:= \frac{\partial \phi_k(\mathbf{x}^0)}{\partial \mathbf{x}^0} \\ \Phi_k(\mathbf{x}^0) &= \frac{\partial \phi^1(\mathbf{x}^{k-1})}{\partial \mathbf{x}} \cdot \frac{\partial \phi^1(\mathbf{x}^{k-2})}{\partial \mathbf{x}} \quad \dots \quad \frac{\partial \phi^1(\mathbf{x}^1)}{\partial \mathbf{x}} \cdot \frac{\partial \phi^1(\mathbf{x}^0)}{\partial \mathbf{x}} \end{aligned} \quad (2.24)$$

Die Übergangsmatrix $\Phi_k(\mathbf{x}^0)$ besteht demnach aus dem Produkt der Jacobi-Matrizen der Abbildung \mathcal{F} an den aufeinanderfolgenden Punkten des Orbit

$$\Phi_k(\mathbf{x}^0) = \prod_{j=0}^{k-1} \mathcal{F}_{\mathbf{x}}(\mathbf{x}^j) \quad \text{mit} \quad \mathcal{F}_{\mathbf{x}}(\mathbf{x}^k) := \left. \frac{\partial \mathcal{F}(\mathbf{x}^k)}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}^k} \quad \text{und} \quad \mathcal{F}_{\mathbf{x}}(\mathbf{x}^0) = I \quad (2.25)$$

Für ergodische Systeme ist die Existenz des Grenzwertes:

$$\lim_{k \rightarrow \infty} \left[\Phi_k^*(\mathbf{x}^0) \Phi_k(\mathbf{x}^0) \right]^{1/2k} = \mathbf{A}_{\mathbf{x}^0} \quad (2.26)$$

durch das „Multiplikative Ergodische Theorem“ von Oseledec [98, 24] für nahezu alle \mathbf{x}_0 gesichert, wobei in Gl.(2.26) $\Phi_k^*(\mathbf{x}^0)$ die Adjunkte der Matrix $\Phi_k(\mathbf{x}^0)$ darstellt.

Die Lyapunov-Exponenten,

$$\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_{d_S})^T \quad ; \quad \text{mit} \quad \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{d_S} \quad (2.27)$$

die die mittlere Kontraktions- und Expansionsraten des lokalen Flusses entlang einer Trajektorie beschreiben, sind als die Logarithmen der Eigenwerte $\tilde{\boldsymbol{\lambda}} = (\tilde{\lambda}_1, \dots, \tilde{\lambda}_{d_S})^T$ der positiven Matrix $\mathbf{A}_{\mathbf{x}^0}$ definiert. Für ergodische Systeme sind die Lyapunov-Exponenten unabhängig von der gewählten Trajektorie. Man spricht von den Lyapunov-Exponenten des Attraktors. Werden ausschließlich die unterschiedlichen Lyapunov-Exponenten $\lambda_1 > \lambda_2 > \dots > \lambda_m$ mit $\lambda_i \neq \lambda_j$ betrachtet, so gilt⁷ [105, 24, 126]:

$$\lambda_i = \lim_{k \rightarrow \infty} \frac{1}{k} \ln \|\Phi_k(\mathbf{x}^0) \cdot \mathbf{u}\| \quad ; \quad i = 1, \dots, m \quad (2.28)$$

wobei die Vektoren $\mathbf{u} \in \mathbf{U}_{\mathbf{x}^0}^i \setminus \mathbf{U}_{\mathbf{x}^0}^{i+1} \quad ; \quad \mathbf{U}_{\mathbf{x}^0}^i, \mathbf{U}_{\mathbf{x}^0}^{i+1} \subset \mathbb{R}^{d_S}$

alle Vektoren \mathbf{u} darstellen, die im Unterraum $\mathbf{U}_{\mathbf{x}^0}^i$ und nicht im Unterraum $\mathbf{U}_{\mathbf{x}^0}^{i+1}$ enthalten sind. Die Unterräume $\mathbf{U}_{\mathbf{x}^0}^i \subset \mathbb{R}^m$ mit $i = 1, \dots, m$ werden jeweils von den Eigenvektoren der Matrix $\mathbf{A}_{\mathbf{x}^0}$ aufgespannt, deren korrespondierende Eigenwerte $\leq \exp^{\lambda_i}$ sind, wobei m die Ordnung der Matrix $\mathbf{A}_{\mathbf{x}^0}$ angibt. Weitere Details zur Bedeutung und Definition von Lyapunov-Exponenten finden sich bei [148, 126, 24].

Für Lyapunov-Exponenten von Grenzmengen dynamischer Systeme gelten die folgenden generellen Aussagen [105, 24, 126]:

- Für dissipative Systeme muß eine Kontraktion von Volumina im Zustandsraum erfolgen, daher muß die Summe aller Lyapunov-Exponenten kleiner als 0 sein.
- Die Lyapunov-Exponenten nicht chaotischer Attraktoren sind kleiner oder gleich 0.

⁷Die Lyapunov-Exponenten kontinuierlicher Systeme werden in *bit/Sekunde* und die zeitdiskreter Systeme in *bit/Iteration* berechnet, wenn statt des natürlichen Logarithmus der Logarithmus zur Basis 2 verwendet wird.

- Bei kontinuierlichen Systemen und Grenzmengen, die keine Gleichgewichtspunkte sind, ist immer ein Lyapunov-Exponent gleich 0 [46]. Der zugehörige Eigenvektor entspricht der Ausbreitungsrichtung. Dies spiegelt die Tatsache wider, daß in dieser Richtung keine Ausdehnung oder Kontraktion erfolgt.
- Gibt es Lyapunov-Exponenten größer 0 und ist die untersuchte Grenzmenge stabil, so handelt es sich um eine chaotische Grenzmenge. Da kontinuierliche dissipative Systeme mindestens einen Lyapunov-Exponenten mit dem Wert 0 haben und die Summe aller Lyapunov-Exponenten kleiner 0 sein muß, folgt, daß für dissipative Systeme chaotische Grenzmengen nur in Zustandsräumen mit einer minimalen Dimension von 3 möglich sind. Entsprechend sind bei zeitdiskreten dissipativen Systemen chaotische Grenzmengen nur in Zustandsräumen mit einer minimalen Dimension von 2 möglich.
- Bei einer nicht chaotischen, stabilen Grenzmenge ist die Anzahl der verschwindenden Lyapunov-Exponenten gleich der topologischen Dimension. Ein Gleichgewichtspunkt hat die Dimension 0, ein periodischer Grenzzyklus die Dimension 1 und eine k-quasiperiodische Grenzmenge entsprechend einem k-Torus die Dimension k. Die jeweiligen Grenzmengen haben 0, 1 oder k Lyapunov-Exponenten mit dem Wert 0. Alle anderen Lyapunov-Exponenten sind bei diesen Grenzmengen kleiner als 0.

Zur Berechnung der Lyapunov-Exponenten werden verschiedene Verfahren in der Literatur beschrieben [37, 24, 27, 106, 107, 40, 146, 129]. Ein Vergleich unterschiedlicher Verfahren wird u. a. von Geist et. al. [37] vorgestellt. Basierend auf diesen Ergebnissen wird im Rahmen dieser Arbeit ausschließlich der Algorithmus von Eckmann und Ruelle [24][S.650] zur Bestimmung der Lyapunov-Exponenten dynamischer Systeme aus experimentell ermittelten Daten eingesetzt.

Eckmann und Ruelle schlagen eine sehr effiziente Methode zur Berechnung der Lyapunov-Exponenten aus Gl.(2.28) und Gl.(2.25) vor. Das Verfahren wird als *Treppen-Iterations-Algorithmus* bezeichnet und basiert auf einer **QR**-Faktorisierung [26] von $\Phi_k(\mathbf{x}^0)$. Durch regelmäßige Renormalisierung werden numerische Probleme vermieden, die auf Grund der exponentiellen Kontraktion bzw. Expansion in unterschiedlichen Raumrichtungen auftreten würden.

Die Faktorisierung erfolgt rekursiv entsprechend:

$$\mathbf{Q}^j \mathbf{R}^j = \mathbf{A}^j = \mathcal{F}_{\mathbf{x}}(\mathbf{x}^{j-1}) \mathbf{Q}^{j-1} \quad ; \quad j = 1, \dots, k-1 \quad (2.29)$$

mit $\mathbf{Q}^j, \mathbf{R}^j, \mathbf{A}^j \in \mathbb{R}^{d_s} \times \mathbb{R}^{d_s} ; \mathbf{x}^j \in \mathbb{R}^{d_s}$

Die Rekursion wird mit der Basis $\mathbf{Q}^0 = \mathbf{I}$ gestartet, wobei \mathbf{I} die Einheitsmatrix des \mathbb{R}^{d_s} darstellt. Damit ergibt sich für die Übergangsmatrix:

$$\Phi_k(\mathbf{x}^0) = \mathbf{Q}^{k-1} \cdot \mathbf{R}^{k-1} \cdot \mathbf{R}^{k-2} \cdot \dots \cdot \mathbf{R}^1 \quad (2.30)$$

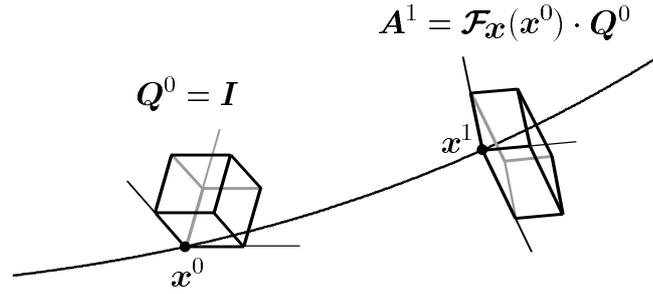


Abbildung 2.3: Geometrische Veranschaulichung der QR -Zerlegung als Teilschritt des Treppen-Iterations-Algorithmus.

Zur Veranschaulichung dieses Verfahrens wird der erste Schritt der Rekursion Gl.(2.29) betrachtet:

$$\mathbf{Q}^1 \cdot \mathbf{R}^1 = \mathbf{A}^1 = \mathcal{F}_{\mathbf{x}}(\mathbf{x}^0) \cdot \mathbf{I} \quad (2.31)$$

Die Einheitsmatrix \mathbf{I} , die in diesem Zusammenhang einen d_S -dimensionalen Einheitswürfel darstellt, wird durch die an der Stelle \mathbf{x}_0 linearisierte Flußabbildung auf das Volumenelement \mathbf{A}^1 abgebildet. Abbildung 2.3 veranschaulicht diese Überlegungen.

Die Matrix \mathbf{A}^1 beschreibt ein Parallelepiped im \mathbb{R}^{d_S} . Die durch die linearisierte Flußabbildung bewirkte Drehung und Deformation wird durch QR -Zerlegung der nichtsingulären Matrix \mathbf{A}^1 in eine orthogonale Drehmatrix \mathbf{Q}^1 und eine obere Dreiecksmatrix \mathbf{R}^1 mit positiven Diagonalelementen $R_{ii}, i = 1, \dots, d_S$ separiert.

Da bei der QR -Zerlegung negative Diagonalelemente auftreten können, werden die nach der Zerlegung berechneten Matrizen $\bar{\mathbf{Q}}$ und $\bar{\mathbf{R}}$ mit der Diagonalmatrix $SIGN(\bar{\mathbf{R}}) := \text{diag}(\text{sign}(\bar{R}_{11}), \dots, \text{sign}(\bar{R}_{d_S d_S}))$ multipliziert:

$$\begin{aligned} \mathbf{Q} &:= \bar{\mathbf{Q}}SIGN(\bar{\mathbf{R}}) \\ \mathbf{R} &:= SIGN(\bar{\mathbf{R}})\bar{\mathbf{R}} \end{aligned} \quad (2.32)$$

Die QR -Zerlegung kann durch Gram-Schmidt-Orthogonalisierung oder durch eine Folge von Householder-Transformationen [115] erfolgen. Im Rahmen dieser Arbeit wird die letztere Methode aufgrund der besseren numerischen Eigenschaften verwendet.

Alle Informationen über die Veränderung des Volumens sind in der Matrix \mathbf{R}^1 enthalten, da die Matrix \mathbf{Q}^1 lediglich eine Basistransformation bewirkt. Die Eigenwerte der Matrix \mathbf{R}^1 beschreiben die Kontraktion ($\tilde{\lambda}_i^1 < 1$) oder Expansion ($\tilde{\lambda}_i^1 > 1$) in den durch die orthonormale Basis \mathbf{Q}^1 beschriebenen Richtungen des Zustandsraumes für die gewählte Evolutionszeit ΔT . Da \mathbf{R}^1 eine Dreiecksmatrix ist, sind ihre Eigenwerte gleich ihren Hauptdiagonalelementen $\tilde{\lambda}_i^1 = R_{ii}^1$. Im folgenden Rekursionsschritt wird das durch die Basis \mathbf{Q}^1 beschriebene Volumenelement durch die im Punkt \mathbf{x}^1 linearisierte Flußabbildung auf \mathbf{A}^2 abgebildet.

Wird diese Vorgehensweise entsprechend Gl.(2.29) und Gl.(2.30) entlang des Orbits bzw. der abgetasteten Trajektorie wiederholt, ergibt sich für die Schätzung der Lyapunov-Exponenten⁸:

$$\lambda_i \approx \lim_{k \rightarrow \infty} \lambda_i^k = \lim_{k \rightarrow \infty} \frac{1}{k \Delta T} \sum_{j=0}^{k-1} \log_2 R_{ii}^j \quad ; \quad i = 1, \dots, d_S \quad (2.33)$$

Auf die mit diesem Verfahren zur Bestimmung von Lyapunov-Exponenten aus experimentellen Daten im Rahmen dieser Arbeit erzielten Ergebnisse wird in Abschnitt 5.6 näher eingegangen.

2.5 Rekonstruktion von Attraktoren

Die bisherigen Ausführungen zur Beschreibung dynamischer Systeme setzen voraus, daß die das System beschreibende Zustandsgleichung bekannt ist. Ein Standardproblem der Systemtheorie besteht darin, das dynamische Verhalten des bekannten Systems vorauszusagen.

Bei der Untersuchung realer physikalischer Systeme sind die Zustandsgleichungen jedoch zumeist unbekannt. Außerdem ist es i. allg. nicht möglich, alle Zustandsgrößen des Systems experimentell zu erfassen. In vielen Fällen kann sogar nur eine skalare Ausgangsgröße eines physikalischen Systems, welches typischerweise einen ∞ -dimensionalen Zustandsraum besitzt, experimentell ermittelt werden.

Ausgehend von einem mit der Periode T_A äquidistant abgetasteten, als stationär angenommenen skalaren Zeitsignal

$$\{v^k\}_{k=1}^N \quad \text{mit} \quad v^k = v(kT_A) \quad \text{und} \quad v^k \in \mathbb{R} \quad ; \quad T_A \in \mathbb{R}^+ \quad ; \quad k \in \mathbb{N} \quad (2.34)$$

stellt sich die Aufgabe, ein dem betrachteten System äquivalentes Modell zu entwickeln. Dieses wird auch als „inverses Problem“ der Systemtheorie bezeichnet. Um Aussagen über das zukünftige Systemverhalten zu erhalten, oder um das zukünftige Ausgangssignal vorherzusagen, muß dieses Modell das beobachtete physikalische System hinreichend genau approximieren.

Vorausgesetzt das dynamische Verhalten eines Systems ist durch Zustandsgleichungen der Form (2.2, 2.3) im kontinuierlichen Fall bzw. (2.7, 2.8) im zeitdiskreten Fall beschreibbar, erfolgt die Lösung dieser Aufgabenstellung in zwei Schritten.

Im ersten Schritt werden aus den skalaren Meßwerten $\{v^k\}$ Zustandsvektoren und damit der Attraktor des beobachteten Systems rekonstruiert. Anschließend werden die dynamischen Eigenschaften des Systems ausgehend von den zuvor rekonstruierten Zustandsvektoren mit Hilfe eines noch zu beschreibenden mathematischen Modells nachgebildet.

⁸Bei der Berechnung der Lyapunov-Exponenten kontinuierlicher Systeme muß die Evolutionszeit ΔT der Abbildung \mathcal{F} berücksichtigt werden. Bei zeitdiskreten Systemen entfällt ΔT in Gl.(2.33).

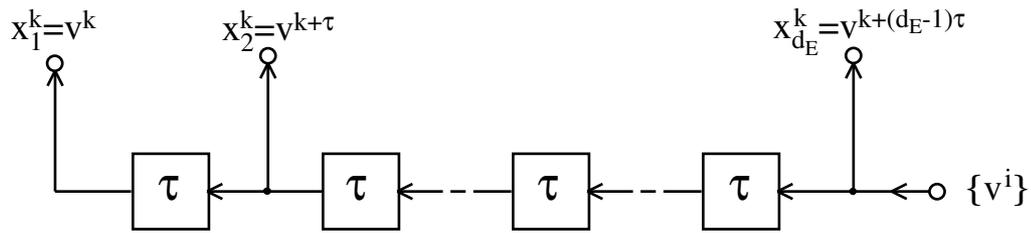


Abbildung 2.4: Ein System, welches die Methode der zeitverzögerten Komponenten realisiert

Das hier verwendete Verfahren zur Rekonstruktion von Attraktoren aus einem skalaren Zeitsignal ist die von Packard et al. [100] sowie Takens [136] vorgeschlagene Methode der *zeitverzögerten Komponenten*. Es gibt auch andere Vorschläge [24, 100] zur Rekonstruktion, diese haben jedoch für die hier betrachtete Anwendung keine Bedeutung erlangt. Ein System zur Realisierung der Methode der zeitverzögerten Komponenten ist in Abbildung 2.4 skizziert. Das Verfahren beruht auf einem Theorem von Whitney [145], welches von Takens [136] und Mañé [85] wie folgt erweitert wurde:

Theorem 1 Takens Einbettungs-Theorem [136]:

Der Attraktor \mathcal{A} eines dynamischen Systems der Ordnung d_S mit dem Fluß $\phi_t(\mathbf{x}_0)$ sei in einer m -dimensionalen kompakten Mannigfaltigkeit $\mathbf{M} \subset \mathbb{R}^{d_S}$ enthalten. Unter der Voraussetzung, daß das Vektorfeld \mathbf{F} des Systems zweimal stetig differenzierbar ist und $f(\phi_t(\mathbf{x}_0)) : \mathbb{R}^{d_S} \rightarrow \mathbb{R}$ eine glatte Funktion auf \mathbf{M} darstellt, ist die Abbildung

$$\Psi_{\mathbf{F},f}(\phi_t(\mathbf{x}_0)) : \mathbf{M} \rightarrow \mathbb{R}^{2m+1}$$

mit $\mathbf{x}(t) = \Psi_{\mathbf{F},f}(\phi_t(\mathbf{x}_0)) := (f(\phi_t(\mathbf{x}_0)), f(\phi_{t+\tau}(\mathbf{x}_0)), \dots, f(\phi_{t+2m\tau}(\mathbf{x}_0)))^T$ (2.35)

im allgemeinen eine Einbettung (B.9).

„Im allgemeinen“ bedeutet, daß $\Psi_{\mathbf{F},f}(\phi_t(\mathbf{x}_0))$ in seltenen Fällen keine Einbettung von \mathbf{M} erzeugt, diese Eigenschaft der Abbildung jedoch durch eine minimale Änderung der Rekonstruktion (z.B. Änderung von τ in Gl.(2.35)) wieder erzielt wird [105]. Der Raum, der das Bild von $\Psi_{\mathbf{F},f}$ enthält, wird als Einbettungs- oder Rekonstruktionsraum bezeichnet. Die Funktion $f(\phi_t(\mathbf{x}_0))$ stellt in dem hier betrachteten Zusammenhang das beobachtete Ausgangssignal $v(t)$ des sich im Zustand $\mathbf{x}(t)$ befindenden Systems dar. Die Einbettungskonstante τ der Rekonstruktion muß demnach der verwendeten Abtastperiode T_A oder einem ganzzahligen Vielfachen derselben entsprechen. Bei zeitdiskreten Systemen ist τ ein Vielfaches der Iteration der Abbildung.

Auf die Folge von Meßwerten $\{v^k\}$ angewendet, bildet Gleichung (2.35) diese skalare Zahlenfolge in eine Folge von rekonstruierten Zustandsvektoren $\{\mathbf{x}^k\}$ der Dimension d_E

ab

$$\begin{aligned}
\mathbf{x}^k &= \Psi_{\mathbf{F},f}(\phi_{kT_A}(\mathbf{x}_0)) \\
&= (f(\phi_{kT_A}(\mathbf{x}_0)), f(\phi_{kT_A+\tau}(\mathbf{x}_0)), \dots, f(\phi_{kT_A+(d_E-1)\tau}(\mathbf{x}_0)))^T \\
&= (v^k, v^{k+\tau}, \dots, v^{k+(d_E-1)\tau})^T
\end{aligned} \tag{2.36}$$

Wird die Einbettungsdimension d_E gemäß Takens [136] zu

$$d_E \geq 2m + 1 \quad \text{mit} \quad m \in \mathbb{N} \tag{2.37}$$

gewählt, so ist Abbildung (2.36) eine Einbettung von $\mathbf{M} \subset \mathbb{R}^{d_s}$ in $\mathbf{M}' \subset \mathbb{R}^{2m+1}$. Die Vektoren $\{\mathbf{x}^k\}$ stellen dann eine diffeomorphe Abbildung der Trajektorie des Systems dar. Daher bleiben die qualitativen Eigenschaften des Urbildes \mathbf{M} erhalten. Dadurch ist es möglich, Aussagen über das dynamische Verhalten sowie charakteristische Größen wie fraktale Dimension und Lyapunov-Exponenten des zugrundeliegenden Systems unter Verwendung der rekonstruierten Zustandsvektoren $\{\mathbf{x}^k\}$ zu gewinnen.

Das Einbettungs-Theorem nach Takens behandelt die Einbettung einer Mannigfaltigkeit mit der ganzzahligen topologischen Dimension m . Für die Rekonstruktion von seltsamen Attraktoren mit im allgemeinen fraktaler Dimension wird die Kapazität d_K zur Bestimmung der Einbettungsdimension d_E des Rekonstruktionsraumes herangezogen. Besitzt die betrachtete Grenzmenge lokal unterschiedliche Dimensionen, stellt die Kapazität d_K eine obere Schranke für die lokale Dimension dar. Entsprechend einem Theorem von Mañé [85] (siehe auch [24]) ergibt sich für Attraktoren mit der Kapazität d_K folgende Ungleichung zur Wahl der Einbettungsdimension d_E

$$d_E \geq 2d_K + 1 \quad \text{mit} \quad d_K \in \mathbb{R}^+ \quad ; \quad d_E \in \mathbb{N} \tag{2.38}$$

Wird d_E entsprechend dieser Gleichung bestimmt, ist sichergestellt, daß die Abbildung Gl.(2.35) bzw. Gl.(2.36) i. allg. eine Einbettung darstellt. Es bleibt anzumerken, daß auch für Einbettungsdimensionen, die kleiner als durch Gl.(2.38) vorgegeben gewählt werden, eine Einbettung erzielt werden kann, solange die Rekonstruktionsdimension nicht kleiner als die fraktale Dimension des Attraktors ist.

Im Falle einer zu klein gewählten Einbettungsdimension ist die Rekonstruktion ungeeignet, da die Eigenschaften des Urbildes nicht erhalten bleiben und die rekonstruierten Trajektorien z. B. Kreuzungspunkte aufweisen können. Eine auf den rekonstruierten Zustandsvektoren basierende Modellierung des Systemverhaltens wird daher unbrauchbare Ergebnisse liefern.

Die genannten Theoreme von Takens und Mañé liefern keinen Hinweis darauf, wie die Abtastperiode T_A und die Einbettungskonstante τ für eine günstige Rekonstruktion einzustellen sind. Dieses Thema wird in einer Reihe von Veröffentlichungen behandelt [12, 35, 76].

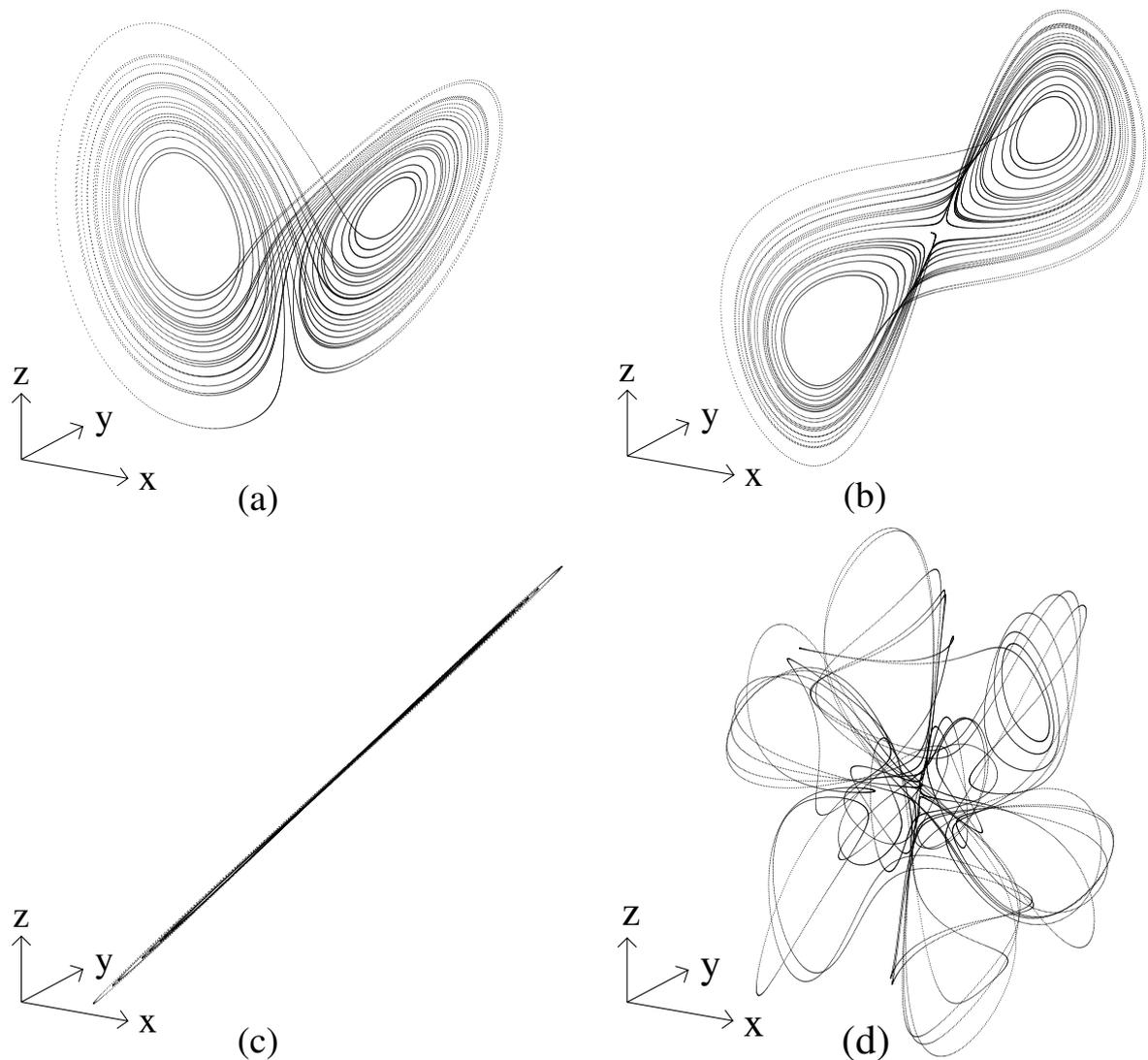


Abbildung 2.5: Vergleich des Lorenz-Attraktors (Bild a) mit drei aus den x-Komponenten rekonstruierten Trajektorien (Bilder b, c und d). In allen Darstellungen sind jeweils 20 000 Punkte skizziert, wobei $T_A=0,001$ und $d_S=3$ bzw. $d_E=3$ gewählt wurde. Die Einbettungskonstanten betragen für (b): $\tau_b=0,05$ (c): $\tau_c=0,001$ und (d): $\tau_d=0,5$.

Im Rahmen dieser Arbeit wird in Kapitel 5 näher auf den Einfluß der Parameter der Rekonstruktion d_E und τ auf die erzielbare Qualität der das Systemverhalten nachbildenden Modelle eingegangen.

An dieser Stelle wird an zwei Beispielen der Einfluß von d_E und τ auf die geometrische Form des rekonstruierten Attraktors graphisch dargestellt. Die zugrundeliegenden chaotischen Systeme werden in Anhang A beschrieben.

Zunächst wird am Beispiel des Lorenz-Systems (Gleichung (A.1)) die Wirkung un-

terschiedlicher Einbettungskonstanten demonstriert. Hierzu dient die Lösung der Zustandsgleichungen für $\mathbf{x}_0=(0,0|0,1|0,0)^T$ unter Verwendung eines Prädiktor-Korrektor-Verfahrens mit einer Schrittweite von $T_A=0,001$. Da das Lorenz-System einen 3-dimensionalen Zustandsraum besitzt, kann die „Original“-Trajektorie in Abbildung 2.5a nur perspektivisch dargestellt werden. Im Vergleich hierzu zeigen die Abbildungen 2.5b, 2.5c und 2.5d drei aus den x -Komponenten der zuvor berechneten Zustandsvektoren gewonnene rekonstruierte Trajektorien, wobei die Einbettungsdimension jeweils $d_E=3$ beträgt. Die Einbettungskonstante variiert dabei von 2.5b: $\tau_b=0,05$ über 2.5c: $\tau_c=0,001$ zu 2.5d: $\tau_d=0,5$.

Offensichtlich wird für $\tau_b=0,05$ eine der „Original“-Trajektorie sehr ähnliche Rekonstruktion allein aus den x -Komponenten der Lösung erzielt. Wird mit $\tau_c=0,001$ die Einbettungskonstante zu klein gewählt, sind die Daten sehr stark korreliert. Die rekonstruierte Trajektorie ist fast ausschließlich auf den Bereich der Hauptdiagonalen des Einbettungsraumes beschränkt und weist keine Ähnlichkeit mit Abbildung (2.5a) auf. Die Rekonstruktion unterscheidet sich auch stark vom „Original“ wenn, wie in Abbildung 2.5d, die Einbettungskonstante zu groß gewählt wird und die Daten dementsprechend nahezu unkorreliert sind.

Im folgenden wird die Mackey-Glass-Differentialgleichung gemäß Gleichung (A.8) betrachtet, welche ebenfalls ein System mit chaotischem Verhalten darstellt. Dieses System besitzt im Gegensatz zum Lorenz-System einen ∞ -dimensionalen Zustandsraum, da die Zustandsgleichung zusätzlich vom vergangenen Zustand $x(t - \Delta T)$ abhängt. Die stationäre Lösung ist jedoch ebenfalls auf einen niedrigdimensionalen Attraktor beschränkt. Gemäß Lapedes und Farber [71] besitzt das System bei $\Delta T=17$ einen Attraktor mit der Dimension $d_A=2,1$. Für $\Delta T=30$ hat die stationäre Lösung die fraktale Dimension $d_A=3,5$.

Die Lösung der Mackey-Glass-Differentialgleichung wurde für $\Delta T=17$ und $\Delta T=30$ mit einem Runge-Kutta-Verfahren vierter Ordnung bei einer Schrittweite von $T_A = 1,0$ berechnet. Die mit den Einbettungsparametern $d_E=3$ und $\tau=4,0$ rekonstruierten Trajektorien sind in Abbildung 2.6 perspektivisch dargestellt.

Bei beiden Rekonstruktionen ist die Einbettungsbedingung Gl.(2.38) nicht erfüllbar, da eine graphische Darstellung auf den \mathbb{R}^3 beschränkt ist. In Abbildung 2.6a ist die fraktale Dimension des Attraktors mit $d_A=2,1$ jedoch zumindest kleiner als die Einbettungsdimension. Obwohl das System einen ∞ -dimensionalen Zustandsraum besitzt, weist die rekonstruierte Trajektorie eine ausgeprägte flächenhafte Struktur auf, die an ein im \mathbb{R}^3 gefaltetes Band erinnert. Dieses entspricht der Erwartung für ein System mit einer fraktalen Dimension, die etwas größer als die einer Fläche ist.

Im Vergleich hierzu wirkt die rekonstruierte Trajektorie für die Mackey-Glass-Differentialgleichung mit $\Delta T=30$ in Abbildung 2.6b wie eine nahezu zufällige Anhäufung von Punkten in einem beschränkten Bereich des \mathbb{R}^3 . Eine entsprechend ausgeprägte Struk-

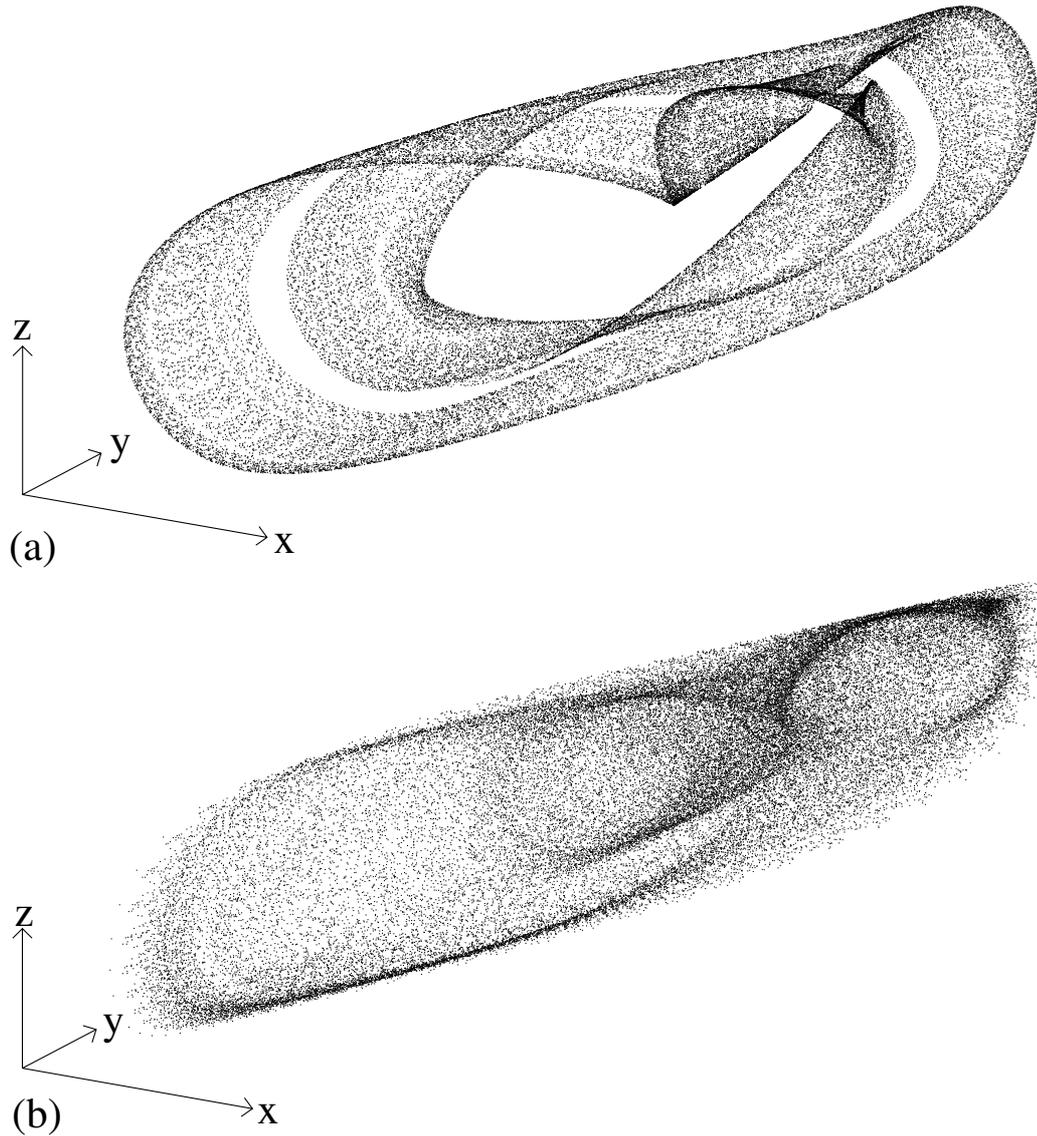


Abbildung 2.6: Mit den Einbettungsparametern $d_E=3$ und $\tau=4,0$ rekonstruierte Trajektorien der Mackey-Glass-Differentialgleichung für
 (a): $\Delta T=17$ und (b): $\Delta T=30$

tur wie in Abbildung 2.6a ist nicht vorhanden. Dieses ist auch nicht verwunderlich, da hier die Rekonstruktionsdimension kleiner als die fraktale Dimension des Attraktors ist.

Aus den beiden Beispielen ist bereits ersichtlich, daß die Parameter d_E und τ die Qualität der Rekonstruktion in entscheidendem Maße beeinflussen. Dieses hat natürlich auch Auswirkungen auf die Leistungsfähigkeit der nachfolgend beschriebenen mathematischen Modelle, deren Aufgabe es ist, basierend auf den rekonstruierten Zustandsvektoren das dynamische Verhalten des zugrundeliegenden Systems möglichst genau zu approximieren.

2.6 Neuronale Netze zur Modellierung dynamischer Systeme

Im folgenden wird auf die Modellierung dynamischer Systeme eingegangen, wobei vorausgesetzt wird, daß diese auf der Basis einer vorgegebenen Folge von Zustandsvektoren $\{\mathbf{x}^k\}_{k=1}^K$ durchgeführt wird. Diese Folge kann aus „Original“-Zustandsvektoren oder aus Zustandsvektoren bestehen, die mit dem in Kapitel (2.5) vorgestellten Verfahren der zeitverzögerten Komponenten rekonstruiert wurden.

Die Zustandsvektoren repräsentieren im Falle eines zeitdiskreten Systems einen Orbit des Systems, so daß die Aufgabe der Modellierung darin besteht, die das System beschreibende Zustandsgleichung bzw. Abbildungsvorschrift zu approximieren. Bei einem zugrundeliegenden kontinuierlichen System stellt die Folge von Zustandsvektoren eine mit der Periodendauer T_A abgetastete Trajektorie des Systems dar. Ein zu entwickelndes Modell muß entsprechend den in Kapitel (2.4) angestellten Überlegungen den Fluß des Systems für eine Evolutionszeit T_A nachbilden. Die Anzahl und Auswahl der Zustandsvektoren $\{\mathbf{x}^k\}$ muß daher sicherstellen, daß diese alle wesentlichen Eigenschaften des Systems widerspiegeln, d. h. alle wesentlichen Teile des Attraktors müssen abgedeckt werden. Die zu ermittelnde Abbildung $\mathcal{G}(\mathbf{x}^k)$ muß die Systemabbildung $\mathcal{F}(\mathbf{x}^k)$ bzw. den Fluß $\phi^{T_A}(\mathbf{x}^k)$ für die Evolutionszeit T_A möglichst genau nachbilden. Im Rahmen dieser Arbeit werden zur Approximation der Abbildung $\mathcal{G}(\mathbf{x}^k)$ neuronale Netze verwendet, die in diesem Zusammenhang Prädiktoren für zukünftige Zustände des Systems darstellen.

Abhängig davon, ob rekonstruierte oder „Original“-Zustandsvektoren verwendet werden und wie weit in die Zukunft der Zustand des Systems vorausgesagt werden soll, finden skalare $\mathcal{G}(\mathbf{x}^k)$ (Gl.(2.40)) oder vektorielle $\mathcal{G}(\mathbf{x}^k)$ (Gl.(2.39)) Prädiktoren Verwendung. Für den Fall rekonstruierter Zustandsvektoren ergibt sich bei gegebener Einbettungsdimension d_E und Einbettungskonstante τ und für eine Prädiktionsweite $pw \leq \tau$ zunächst:

$$\mathcal{G} : \mathbb{R}^{d_E} \rightarrow \mathbb{R}^{d_E} ; \quad \mathbf{x}^{k+pw} = \mathcal{G}(\mathbf{x}^k) \quad \text{mit} \quad \mathbf{x}^{k+pw}, \mathbf{x}^k \in \mathbb{R}^{d_E} \quad k, pw \in \mathbb{Z} \quad (2.39)$$

Werden die Zustandsvektoren \mathbf{x}^k und \mathbf{x}^{k+pw} entsprechend Gl.(2.36) rekonstruiert, erhält man:

$$\begin{aligned} \mathbf{x}^k &= (x_1^k, x_2^k, \dots, x_{d_E}^k)^T = (v^k, v^{k+\tau}, \dots, v^{k+(d_E-1)\tau})^T \\ \mathbf{x}^{k+pw} &= (x_1^{k+pw}, x_2^{k+pw}, \dots, x_{d_E}^{k+pw})^T = (v^{k+pw}, v^{k+pw+\tau}, \dots, v^{k+pw+(d_E-1)\tau})^T \end{aligned} \quad (2.40)$$

Bei näherer Betrachtung dieser Aufgabenstellung fällt auf, daß nur die letzte Komponente des nächsten Zustandsvektors unbekannt ist. Diese kann jedoch mit Hilfe des *skalaren* Prädiktors

$$\mathcal{G} : \mathbb{R}^{d_E} \rightarrow \mathbb{R} ; \quad x_{d_E}^{k+pw} = \mathcal{G}(\mathbf{x}^k) \quad \text{mit} \quad x_{d_E}^{k+pw} \in \mathbb{R} \quad \mathbf{x}^k \in \mathbb{R}^{d_E} \quad k \in \mathbb{Z} \quad (2.41)$$

vorhergesagt werden.

Ist die geforderte Prädiktionsweite pw allerdings größer als τ , ergeben sich im wesentlichen folgende Möglichkeiten zur Bewältigung dieser Aufgabe:

- pw -fache Iteration der für die Evolutionszeit 1 entwickelten Abbildung

$$x_{d_E}^{k+pw} = \mathcal{G}^{(pw)}(\mathbf{x}^k) \quad (2.42)$$

- Bestimmung eines skalaren Prädiktors für die Evolutionszeit pw , der entsprechend der gestellten Prädiktionsaufgabe eingesetzt wird

$$x_{d_E}^{k+pw} = \mathcal{G}(\mathbf{x}^k) \quad (2.43)$$

- Bestimmung eines vektoriellen Prädiktors für die Evolutionszeit pw . Die Dimension der Ausgangsvektoren folgt aus der gestellten Prädiktionsaufgabe.

$$\mathbf{x}^{k+pw} = \mathcal{G}(\mathbf{x}^k) \quad (2.44)$$

Entsprechende Überlegungen sind für die Prädiktion der „Original“-Zustandsvektoren anzustellen. Im Rahmen dieser Arbeit werden in diesem Fall ausschließlich vektorielle Prädiktoren mit

$$\mathcal{G}: \mathbb{R}^{d_S} \rightarrow \mathbb{R}^{d_S} \quad ; \quad \mathbf{x}^{k+1} = \mathcal{G}(\mathbf{x}^k) \quad \text{mit} \quad \mathbf{x}^{k+1}, \mathbf{x}^k \in \mathbb{R}^{d_S} \quad k \in \mathbb{Z} \quad (2.45)$$

untersucht.

Aus der Literatur sind zahlreiche Ansätze zur Prädiktion chaotischer Zeitsignale bekannt. Beiträge zu diesem Thema finden sich u. a. bei Farmer und Sidorowich [30, 31, 32], Broomhead, King und Lowe [12, 13], Lapedes und Farber [71], Moody und Darken [91, 92, 93] sowie bei [2, 15, 19, 21, 23, 64, 66, 113, 128, 143].

In einer Reihe von Untersuchungen wurden verschiedene Typen neuronaler Netze zur Approximation der Abbildung $\mathcal{G}(\mathbf{x}^k)$ und damit zur Prädiktion chaotischer Zeitsignale verglichen. Hierbei wurde festgestellt, daß mit nicht rückgekoppelten mehrlagigen Perzeptron-Netzen [133, 138, 132, 5, 28] im Vergleich zur Verwendung radialer Basisfunktionen [87], rezeptiver Felder [40] oder rückgekoppelter mehrlagiger Perzeptron-Netze [99] die besten Ergebnisse bei akzeptablem Aufwand an Rechenzeit erzielbar sind. Die weiteren Ausführungen beschränken sich daher auf die Verwendung nicht rückgekoppelter mehrlagiger Perzeptron-Netze zur Modellierung dynamischer Systeme.

3 Optimierung mehrlagiger Perzeptron-Netze unter Verwendung von Gradientenverfahren

Mehrlagige Perzeptron-Netze sind in vielen Bereichen der nichtlinearen Signalverarbeitung die zur Zeit am häufigsten eingesetzten neuronalen Netze. Sie werden u. a. bei den folgenden Problemstellungen verwendet:

- Bildverarbeitung-Mustererkennung [18, 52, 102, 119]
- Sprachverarbeitung-Spracherkennung [62, 74, 94, 101, 149]
- Regelungstechnik [95]
- Nichtlineare Prädiktion komplexer Vorgänge [50, 51, 52, 62, 117, 119]
- Verkehrsmanagement in Telekommunikationsnetzen [4, 45]
- CDMA basierte Übertragungsverfahren [1]

Im Rahmen dieser Arbeit wird ihre Eignung zur Modellierung dynamischer Systeme bzw. zur Prädiktion chaotischer Zeitsignale untersucht.

3.1 Das mehrlagige Perzeptron-Netz

Das 1958 von Rosenblatt [121] beschriebene Perzeptron stellt das Grundelement dar, aus dem mehrlagige Perzeptron-Netze (**M**ulti **L**ayer **P**erceptrons) oder kurz *MLP*-Netze aufgebaut sind.

Dieses in Abbildung 3.1 skizzierte Perzeptron addiert zu einem Schwellwert Θ die gewichtete Summe der Komponenten des Eingangsvektors $\mathbf{x} = (x_1, \dots, x_M)^T$. Die Summe wird einer i. allg. nichtlinearen Aktivierungsfunktion zugeführt. Das Ergebnis dieser Abbildung ist der Ausgangswert des Perzeptrons.

Üblicherweise wird der Eingangsvektor \mathbf{x} um eine Komponente mit dem festen Wert $x_0 \equiv 1$ erweitert. Mit diesem Bias-Eingang ist der Schwellwert Θ als Gewicht w_0 der entstehenden Verbindung beschreibbar. Damit ergibt sich die Abbildung eines Perzeptrons zu:

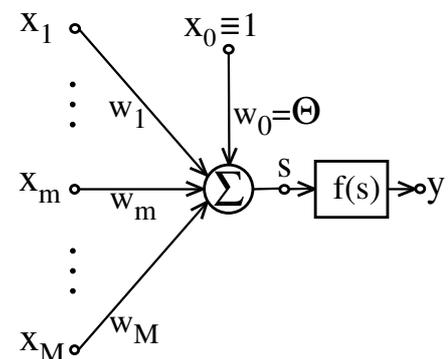


Abbildung 3.1: Das Perzeptron bildet das Grundelement von *MLP*-Netzen.

$$y = f(s) = f\left(\sum_{m=0}^M w_m x_m\right) \quad (3.1)$$

Werden mehrere Perzeptrons zu Lagen zusammengefaßt und diese Lagen hintereinander gruppiert, erhält man den zur Zeit am häufigsten eingesetzten Typ eines neuronalen Netzes: das mehrlagige Perzeptron-Netz oder auch Backpropagation-Netz. In diesem Zusammenhang werden die Perzeptrons als Knoten des neuronalen Netzes bezeichnet, die die Informationsverarbeitung vornehmen. In einem *MLP*-Netz sind alle Knoten einer Lage mit sämtlichen Knoten der vorhergehenden und der nachfolgenden Lage verknüpft. Verbindungen über mehrere Lagen existieren nicht. Der Informationsfluß erfolgt von der Eingangslage ($l = 1$) über die verdeckten Lagen⁹ ($l = 2, \dots, L - 1$) zur Ausgangslage ($l = L$). Da Rückkopplungen nicht auftreten, werden diese *MLP*-Netze auch als *Feed-Forward*-Netze bezeichnet.

Zur Erläuterung der verwendeten Bezeichnungen ist in Abbildung 3.2 die allgemeine Struktur eines *MLP*-Netzes mit L Lagen dargestellt.

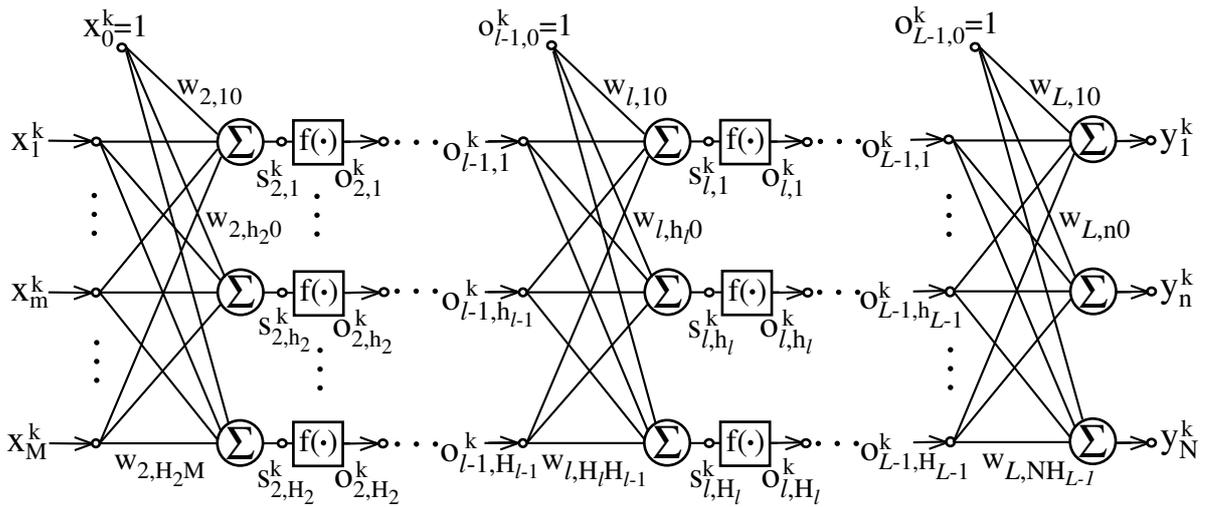


Abbildung 3.2: Allgemeine Struktur eines *MLP*-Netzes bestehend aus L Lagen mit jeweils $H_1=M, \dots, H_l, \dots, H_L=N$ Perzeptrons.

Zunächst werden die Bias-Eingänge aller Knoten einer Lage $l \in \{2, \dots, L\}$ zu einem zusätzlichen Knoten mit dem konstanten Ausgangswert $o_{l-1,0}^k \equiv 1$ in der vorhergehenden Lage zusammengefaßt. In der ersten Lage gilt entsprechend $x_0^k \equiv 1$. Damit besitzt das *MLP*-Netz aus Abbildung 3.2 formal $(H_l + 1)$ Knoten in den Lagen $l = 1, \dots, L - 1$ sowie $H_L = N$ Knoten in der Ausgangslage, wobei ein Knoten durch Angabe der Lage l und der Position h_l innerhalb der Lage eindeutig gekennzeichnet ist.

⁹Die Lagen $l = 2, \dots, L - 1$ werden als verdeckte Lagen bezeichnet, da sie keine direkte Verbindung mit der „Außenwelt“ besitzen.

Die Anzahl der Lagen sowie die Zahl der Knoten innerhalb der einzelnen Lagen bestimmen die Topologie eines *MLP*-Netzes. Die Bias-Eingänge werden in diesem Zusammenhang nicht betrachtet¹⁰, da sie nur zur einfacheren mathematischen Beschreibung eingeführt wurden.

Die $H_1=M$ Knoten der Eingangslage nehmen die Komponenten x_1^k, \dots, x_M^k der Eingangsvektoren \mathbf{x}^k auf und leiten diese zusammen mit dem Bias-Eingang $x_0^k = 1$ an die Knoten der zweiten Lage weiter.

Die Ausgangswerte der verdeckten Knoten des Netzes ergeben sich zu:

$$\begin{aligned} \mathbf{o}_l^k &= (o_{l,0}^k, o_{l,1}^k, \dots, o_{l,h_l}^k, \dots, o_{l,H_l}^k)^T \\ &= (1, f(s_{l,1}^k), \dots, f(s_{l,h_l}^k), \dots, f(s_{l,H_l}^k))^T \quad \text{für } l = 2, \dots, L-1 \end{aligned} \quad (3.2)$$

mit

$$s_{l,h_l}^k = \sum_{h_{l-1}=0}^{H_{l-1}} w_{l,h_l h_{l-1}} o_{l-1,h_{l-1}}^k \quad \text{für } h_l = 1, \dots, H_l \quad (3.3)$$

Die Kennzeichnung der Gewichte des Netzes erfolgt durch drei Indizes, so daß $w_{l,h_l h_{l-1}}$ das Gewicht der Verbindung von Knoten h_{l-1} der Lage $(l-1)$ zu Knoten h_l in Lage l darstellt. Werden zusätzlich die Gewichte in den einzelnen Lagen zu Gewichtsmatrizen

$$\mathbf{W}_l = \begin{pmatrix} w_{l,10} & w_{l,11} & \cdots & w_{l,1H_{l-1}} \\ w_{l,20} & w_{l,21} & \cdots & w_{l,2H_{l-1}} \\ \vdots & \vdots & \ddots & \vdots \\ w_{l,H_l 0} & w_{l,H_l 1} & \cdots & w_{l,H_l H_{l-1}} \end{pmatrix} \quad \text{für } l = 2, \dots, L \quad (3.4)$$

zusammengefaßt, vereinfacht sich Gl.(3.3) zu:

$$\mathbf{s}_l^k = (s_{l,1}^k, \dots, s_{l,H_l}^k)^T = \mathbf{W}_l \mathbf{o}_{l-1}^k \quad \text{für } l = 2, \dots, L-1 \quad (3.5)$$

Die Anzahl aller Gewichte beträgt

$$Q = \sum_{l=2}^L (H_{l-1} + 1) H_l \quad (3.6)$$

Diese bilden den Gewichtsvektor $\mathbf{w} \in \mathbb{R}^Q$

$$\mathbf{w} = (w_{2,10}, w_{2,11}, \dots, w_{2,H_2 M}, \dots, w_{L,H_L H_{L-1}})^T \quad (3.7)$$

eines *MLP*-Netzes.

Nachdem alle L Lagen durchlaufen wurden, steht an den $H_L = N$ Knoten der Ausgangslage der Ausgangsvektor \mathbf{y}^k des *MLP*-Netzes

$$\mathbf{y}^k = (y_1^k, \dots, y_n^k, \dots, y_N^k)^T = \mathbf{W}_L \cdot \mathbf{o}_{L-1}^k \quad (3.8)$$

¹⁰Zum Beispiel wird ein *MLP*-Netz mit vier Lagen, welches in der Eingangslage 12, in der zweiten Lage 10, in der dritten Lage 8 und in der Ausgangslage 1 Knoten besitzt, im folgenden kurz als ein 12:10:8:1 Netz bezeichnet. Gemäß Gl.(3.6) besitzt dieses Netz $Q=227$ Gewichte.

mit

$$y_n^k = w_{L,n0} + \sum_{h_{L-1}=1}^{H_{L-1}} w_{L,nh_{L-1}} \cdot f \left(w_{L-1,h_{L-1}0} + \sum_{h_{L-2}=1}^{H_{L-2}} w_{L-1,h_{L-1}h_{L-2}} \cdot \dots \cdot f \left(\sum_{m=0}^M w_{2,h_2m} \cdot x_m^k \right) \cdot \dots \right) \quad (3.9)$$

zur Verfügung. Auf eine Indizierung der Aktivierungsfunktionen $f(\cdot)$ wird verzichtet, da bei allen verdeckten Knoten der hier untersuchten *MLP*-Netze identische Aktivierungsfunktionen verwendet werden.

Die von Rosenblatt vorgeschlagene nichtlineare Aktivierungsfunktion, die Sprungfunktion,

$$f(s) = \begin{cases} 1 & \text{für } s > 0 \\ 0 & \text{für } s \leq 0 \end{cases} \quad (3.10)$$

kommt bei *MLP*-Netzen i. allg. nicht zum Einsatz, da die für diesen Netztyp existierenden Optimierungsverfahren stetig differenzierbare Aktivierungsfunktionen voraussetzen. Darüber hinaus verlangt der hier untersuchte Einsatz mehrlagiger Perzeptron-Netze zur Modellierung nichtlinearer Abbildungen kontinuierliche Ausgangswerte mit hoher Genauigkeit. Daher werden im folgenden nur Perzeptrons mit sigmoidaler¹¹ bzw. linearer Aktivierungsfunktion betrachtet.

Die neben der Tangens-Hyperbolicus-Funktion

$$f_{tan}(s) = \tanh(\beta s) \quad (3.11)$$

am häufigsten eingesetzte sigmoidale Funktion ist die logistische Funktion,

$$f_{log}(s) = \frac{1}{1 + e^{-\beta s}} \quad (3.12)$$

welche im folgenden als Aktivierungsfunktion aller Knoten der verdeckten Lagen verwendet wird und in Abbildung 3.3 dargestellt ist.

Die logistische Funktion bildet die Eingangswerte $s \in \mathbb{R}$ auf $f_{log}(s) \in \mathbb{R}$ im Intervall $(0, 1)$ ab. Sie besitzt für betragsmäßig große Werte von s ausgesprochene Sättigungsbereiche, also Bereiche sehr kleiner Steigung. Im Gegensatz dazu ist die Steigung im Aktivierungsbereich, bei betragsmäßig kleinen Werten von s , wesentlich größer. Diese Eigenschaft der Aktivierungsfunktion hat große Auswirkungen auf die Leistungsfähigkeit der in den nächsten Abschnitten vorgestellten Optimierungsverfahren.

Im folgenden wird die sog. Steilheit β in Gl.(3.12) ohne Einschränkungen der Abbildungseigenschaften des Netzes¹² zu $\beta=1$ gewählt.

¹¹Als sigmoidale Funktion wird eine reellwertige, stetig differenzierbare, monoton wachsende Funktion $f(s)$ bezeichnet, die die Eingangswerte $s \in \mathbb{R}$ auf $f(s) \in \mathbb{R}$ im Bereich (f_{min}, f_{max}) abbildet [55].

Damit ist $\frac{df(s)}{ds} > 0$ für $-\infty < s < \infty$.

¹²Mit Gleichung (3.1) und (3.12) läßt sich zeigen, daß ein Wert von $\beta \neq 1$ durch geänderte Gewichte $\tilde{w}_m = \beta w_m$ realisiert wird.

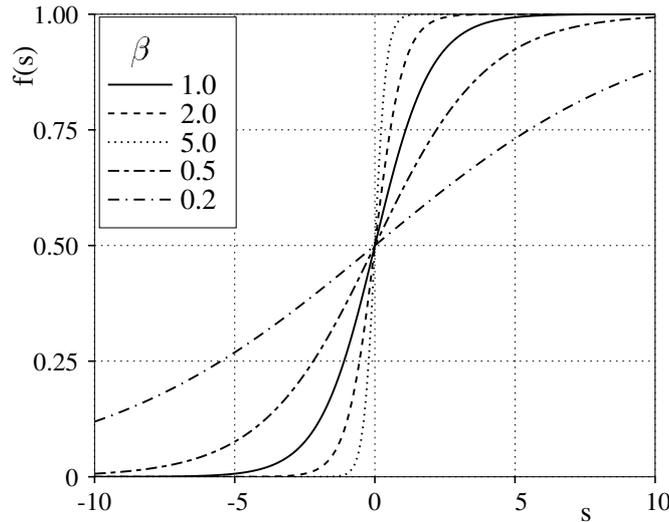


Abbildung 3.3: Die logistische Aktivierungsfunktion für $\beta=0, 2; 0, 5; 1; 2;$ und 5

Die Ableitungen der logistischen Funktion lassen sich in folgender Weise durch die Funktion selbst ausdrücken

$$f'_{log}(s) = \frac{df_{log}(s)}{ds} = \beta f_{log}(s)(1 - f_{log}(s)) \quad (3.13)$$

sowie

$$f''_{log}(s) = \frac{d^2 f_{log}(s)}{ds^2} = \beta^2 f_{log}(s)(1 - f_{log}(s))(1 - 2f_{log}(s)) \quad (3.14)$$

Diese Beziehungen werden bei der Realisierung der in dieser Arbeit untersuchten Optimierungsverfahren genutzt (Kap.3.3, Kap.4.2).

Im Gegensatz zu den verdeckten Lagern besitzen die Knoten der Ausgangslage die lineare Aktivierungsfunktion

$$f_{lin}(s) = s \quad (3.15)$$

Eine weitere Nichtlinearität an dieser Stelle würde hinsichtlich der Abbildungseigenschaften des *MLP*-Netzes keine Vorteile haben, die Optimierung des Netzes jedoch erschweren.

Das vorgestellte *MLP*-Netz bildet den Eingangsvektor $\mathbf{x}^k \in \mathbb{R}^M$ gemäß

$$\tilde{\mathcal{G}} : \mathbb{R}^M \rightarrow \mathbb{R}^N \quad ; \quad \mathbf{y}^k = \tilde{\mathcal{G}}(\mathbf{x}^k) \quad \text{mit} \quad \mathbf{x}^k \in \mathbb{R}^M \quad \mathbf{y}^k \in \mathbb{R}^N \quad k \in \mathbb{Z} \quad (3.16)$$

und Gl.(3.9) auf den Ausgangsvektor $\mathbf{y}^k \in \mathbb{R}^N$ ab. Da sich die Netzabbildung ausschließlich aus affinen Transformationen¹³ und den sigmoidalen Aktivierungsfunktionen zusammensetzt, existieren die partiellen Ableitungen beliebiger Ordnung von $\tilde{\mathcal{G}}(\mathbf{x}, \mathbf{w})$ bezüglich \mathbf{x} und \mathbf{w} ($\tilde{\mathcal{G}}$ ist eine C^∞ -Funktion (B.5)).

Entsprechend den Ausführungen aus Kapitel 2.6 wird das beschriebene Netz zur Modellierung dynamischer Systeme eingesetzt. Abhängig von der Aufgabenstellung muß die

¹³Die Funktion $f(x) = ax + b$ wird als affine Funktion bezeichnet.

Abbildung $\tilde{\mathcal{G}}$ des *MLP*-Netzes eine der Abbildungen \mathcal{G} gemäß Gl.(2.41) \dots Gl.(2.45) hinreichend genau approximieren. Zur Charakterisierung der gewünschten Abbildung \mathcal{G} wird eine repräsentative Teilmenge der (rekonstruierten) Zustandsvektoren $\{\mathbf{x}^k\}$ verwendet, die alle interessierenden Eigenschaften des dynamischen Systems widerspiegelt. Diese Trainingsmenge besteht aus einer Folge von K Eingangs-Ausgangs-Paaren $\{\mathbf{x}^k, \mathbf{d}^k\}_{k=1}^K$, die entsprechend den Anforderungen an den Prädiktor K Beispiele der zu approximierenden Abbildung \mathcal{G} gemäß Kapitel 2.6 darstellen. Hierbei bezeichnet \mathbf{x}^k den k -ten d_E -dimensionalen Zustandsvektor am Eingang und \mathbf{d}^k den *gewünschten Ausgangsvektor* (*desired vector*) des Netzes. Die übrigen Zustandsvektoren $\{\mathbf{x}^k\}_{k=K+1}^{N-\tau(d_E-1)}$ dienen anschließend als Testmenge, um die Verallgemeinerungsfähigkeit des Netzes bei unbekanntem Eingangsdaten zu demonstrieren.

Der vom *MLP*-Netz verursachte Prädiktionsfehlervektor \mathbf{e}^k ist die Differenz

$$\mathbf{e}^k = \mathbf{d}^k - \mathbf{y}^k \quad (3.17)$$

aus gewünschtem (\mathbf{d}^k) und tatsächlichem (\mathbf{y}^k) Ausgangsvektor. Dieser Fehler wird um so geringer, je besser die Abbildung $\tilde{\mathcal{G}}$ des Netzes mit der des dynamischen Systems \mathcal{G} übereinstimmt. In diesem Zusammenhang stellt sich die Frage, welche Arten von Abbildungen mit ausreichender Genauigkeit durch *MLP*-Netze realisierbar sind, und welche Aussagen über die Topologie der Netze gemacht werden können.

Diese Frage war von entscheidender Bedeutung bei der Entwicklung und Erforschung neuronaler Netze [55, 89] und wird in einer Reihe von Veröffentlichungen behandelt [20, 55, 61, 63, 134].

Basierend auf einem 1957 von A. Kolmogorov [69] veröffentlichten Theorem wird in [53, 54, 55] gezeigt, daß *MLP*-Netze mit nur einer verdeckten Lage und beliebigen sigmoidalen Aktivierungsfunktionen in den verdeckten Knoten universelle Approximationseigenschaften besitzen, solange die Anzahl der Knoten in der verdeckten Lage und die Gewichte im Netz keinen Einschränkungen unterliegen.

Theorem 3: Backpropagation-Approximations-Theorem (Hecht-Nielsen):

Für eine beliebige quadratisch integrierbare Funktion¹⁴ $\mathcal{G} : [0, 1]^M \rightarrow \mathbb{R}^N$ und ein beliebiges $E_{max} > 0$ existiert ein dreilagiges Perzeptron-Netz gemäß Abbildung 3.2, welches \mathcal{G} mit einem mittleren quadratischen Fehler $E < E_{max}$ gemäß Gl.(3.20) approximiert.

¹⁴Für eine quadratisch integrierbare Funktion $\mathcal{G} : [0, 1]^M \rightarrow \mathbb{R}$ muß das Integral $\int_{[0, 1]^M} |\mathcal{G}(\mathbf{x})|^2 d\mathbf{x}$ existieren. Eine vektorwertige Funktion $\mathcal{G} : [0, 1]^M \rightarrow \mathbb{R}^N$ heißt quadratisch integrierbar, wenn jede Komponente \mathcal{G}_n ; $n = 1, \dots, N$ quadratisch integrierbar ist. Wobei in Theorem 3 der Definitionsbereich der Funktion \mathcal{G} auf den M -dimensionalen Einheitswürfel $[0, 1]^M$ beschränkt ist.

Dieses Theorem sichert zwar die Existenz eines 3-lagigen Perzeptron-Netzes zu, das die gestellte Approximationsaufgabe löst, läßt jedoch die hierfür benötigte Anzahl verdeckter Knoten sowie Aussagen über die Existenz eines geeigneten Lernverfahrens zur Bestimmung der optimalen Netzgewichte \mathbf{w}^* offen. In vielen Fällen ist es sogar von Vorteil, statt eines 3-lagigen Netzes ein *MLP*-Netz mit 4, 5 oder mehr Lagen zu verwenden, da diese zur Bewältigung der gestellten Aufgabe i. allg. eine insgesamt geringere Anzahl verdeckter Knoten verlangen.

Ein Überblick über die Möglichkeiten zur Approximation kontinuierlicher Funktionen auf der Basis sigmoidaler Funktionen findet sich bei Ito [63]. Weiterhin zeigen Stinchcombe und White [134] sowie Gallant und White [36], daß *MLP*-Netze mit einer verdeckten Lage die universellen Approximationseigenschaften auch für nicht-sigmoidale nichtlineare Aktivierungsfunktionen in den verdeckten Knoten besitzen.

Werden hingegen ausschließlich lineare Aktivierungsfunktionen gemäß Gl.(3.15) verwendet, läßt sich ein L-lagiges Netz in ein äquivalentes zweilagiges Netz umformen, welches einen vektoriellen linearen Prädiktor der Form

$$\tilde{\mathcal{G}} : \mathbb{R}^M \rightarrow \mathbb{R}^N \quad ; \quad \mathbf{y}^k = \mathbf{W}_{res} \cdot \mathbf{x}^k \quad \text{mit} \quad \mathbf{x}^k \in \mathbb{R}^M \quad \mathbf{y}^k \in \mathbb{R}^N \quad k \in \mathbb{Z} \quad (3.18)$$

mit

$$\mathbf{W}_{res} = \mathbf{W}_L \cdot \mathbf{W}_{L-1} \cdot \dots \cdot \mathbf{W}_2 \quad (3.19)$$

beschreibt¹⁵. Hierbei stellt \mathbf{W}_{res} die $(M + 1) \times N$ -Matrix der Prädiktorkoeffizienten dar. Dieses Netz ist naturgemäß nur in der Lage, die linearen Eigenschaften dynamischer Systeme nachzubilden und wird für die späteren Untersuchungen zur Abschätzung der Leistungsfähigkeit linearer Modellansätze verwendet.

3.2 Abstiegsverfahren

Zur Bewertung der Güte der durch das *MLP*-Netz erzielten Approximation $\tilde{\mathcal{G}}$ der Abbildung \mathcal{G} des dynamischen Systems wird der mittlere quadratische Prädiktionsfehler

$$E(\mathbf{w}) = \frac{1}{K} \sum_{k=1}^K E^k(\mathbf{w}) \quad \text{mit} \quad E^k(\mathbf{w}) = \frac{1}{2}(\mathbf{e}^k)^2 = \frac{1}{2}(\mathbf{d}^k - \mathbf{y}^k)^2 \quad (3.20)$$

für die K Eingangs-Ausgangs-Paare der Trainingsmenge herangezogen. Analog zu Gl.(3.20) wird der mittlere quadratische Prädiktionsfehler für die Testmenge berechnet.

Für eine vorgegebene Netztopologie und eine konstante Trainingsmenge führt die Minimierung des mittleren quadratischen Prädiktionsfehlers bezüglich der Netzgewichte \mathbf{w} auf das nichtlineare Optimierungsproblem:

$$E(\mathbf{w}^*) = \min_{\mathbf{w}} E(\mathbf{w}) \quad \text{mit} \quad \mathbf{w} \in \mathbb{R}^Q, E(\mathbf{w}) \in \mathbb{R}^+ \quad (3.21)$$

¹⁵Die Eingangsvektoren \mathbf{x}^k werden wie zuvor formal durch die Bias-Komponenten $x_0^k \equiv 1$ erweitert.

Das Minimum der Zielfunktion $E(\mathbf{w})$ wird für die optimalen Netzgewichte \mathbf{w}^* erreicht. Dieses nichtlineare Optimierungsproblem ohne Restriktionen ist nicht geschlossen lösbar. Daher werden iterative Verfahren eingesetzt, die bei Vorgabe eines Startwertes für die Netzgewichte \mathbf{w}^0 eine Folge von Gewichtsvektoren $\mathbf{w}^1, \mathbf{w}^2, \dots, \mathbf{w}^I, \dots$ erzeugen, welche im Idealfall gegen ein globales Minimum \mathbf{w}^* oder zumindest gegen ein lokales Minimum der Zielfunktion mehrlagiger Perzeptron-Netze Gl.(3.20) konvergiert. Eine besonders günstige Wahl der Startgewichte ist i. allg. nicht bekannt, so daß die Gewichte mit gleichverteilten Zufallszahlen im Bereich $[-w_{max}, w_{max}]$ vorbesetzt werden. Symmetrien in den Gewichten, die zu Problemen bei der Optimierung führen können, werden so ausgeschlossen [123].

Gleichung (3.20) definiert die Fehleroberfläche von *MLP*-Netzen in dem $(Q + 1)$ -dimensionalen Raum, der durch den Vektor der Netzgewichte $\mathbf{w} \in \mathbb{R}^Q$ und den daraus resultierenden Prädiktionsfehler $E(\mathbf{w}) \in \mathbb{R}$ aufgespannt wird. Ein Beispiel für die Fehleroberfläche eines Netzes mit nur zwei Gewichten wird in Abschnitt 5.1 vorgestellt.

Die Beschaffenheit der Fehleroberfläche hat entscheidenden Einfluß auf die Leistungsfähigkeit und hier insbesondere auf die Konvergenzgeschwindigkeit der eingesetzten Optimierungsverfahren. Aufgrund des Sättigungscharakters der sigmoidalen Aktivierungsfunktionen sind ausgedehnte, sehr flache Gebiete vorhanden, in denen der Gradient nahezu Null ist. Daneben existieren aber auch Schluchten mit steilen Wänden und i. allg. geringer Steigung der Talsohlen. Diese Eigenschaft ist einerseits auf die große Steigung der Aktivierungsfunktionen im Übergangsbereich und andererseits auf die Existenz unterschiedlicher globaler Minima zurückzuführen. Die Fehleroberflächen weisen mehrere globale Minima auf, weil *MLP*-Netze bei geeigneter Permutation der Netzgewichte - und damit für unterschiedliche Gewichtskonstellationen - identische Abbildungseigenschaften aufweisen [55, 58].

Diese geometrische Interpretation des Optimierungsproblems impliziert die Verwendung sogenannter *Abstiegsverfahren* (*descent methods*), wie sie aus der Literatur zur nichtlinearen Optimierung bekannt sind [22, 34, 39, 79]. Abstiegsverfahren liefern i. allg. eine Folge von Gewichtsvektoren, die eine Abnahme des Wertes der Zielfunktion in jeder Iteration I ergeben

$$E(\mathbf{w}^{I+1}) < E(\mathbf{w}^I) \quad ; \quad \mathbf{w} \in \mathbb{R}^Q ; E(\mathbf{w}) \in \mathbb{R}^+ ; I \in \mathbb{N}_0 \quad (3.22)$$

Die hier verwendeten schrittweitenbasierten Abstiegsverfahren sind modellgestützte Verfahren, die eine Approximation der lokalen Beschaffenheit der Fehleroberfläche für die jeweilige Gewichtskonstellation \mathbf{w}^I verwenden, um eine Abstiegsrichtung (Suchrichtung) $\mathbf{u}^I \in \mathbb{R}^Q$ zu bestimmen. Eine *Abstiegsrichtung* (*descent direction*) liegt vor, wenn sich der Wert der Zielfunktion $E(\mathbf{w}^I)$ durch geeignete Wahl einer Schrittweite $\eta^I \in \mathbb{R}$ in dieser Richtung verringert.

$$E(\mathbf{w}^I + \eta^I \cdot \mathbf{u}^I) < E(\mathbf{w}^I) \quad (3.23)$$

Weiterhin erfüllt eine Abstiegsrichtung \mathbf{u}^I die Bedingung

$$(\mathbf{g}^I)^T \cdot \mathbf{u}^I < 0 \quad (3.24)$$

In dieser Gleichung wird der Gradient der Zielfunktion $\nabla_w E(\mathbf{w})$ für den Gewichtsvektor \mathbf{w}^I kurz mit $\mathbf{g}^I = \mathbf{g}(\mathbf{w}^I)$ bezeichnet. Dieser besteht aus den partiellen Ableitungen von $E(\mathbf{w})$ bezüglich der Gewichte \mathbf{w}

$$\mathbf{g}(\mathbf{w}) := \nabla_w E(\mathbf{w}) = \left(\frac{\partial E}{\partial w_{2,1,0}}, \dots, \frac{\partial E}{\partial w_{l,h_l,h_{l-1}}}, \dots, \frac{\partial E}{\partial w_{L,H_L,H_{L-1}}} \right)^T. \quad (3.25)$$

Die Änderung der Netzgewichte \mathbf{w}^I wird anschließend entlang der Suchrichtung \mathbf{u}^I vorgenommen

$$\mathbf{w}^{I+1} = \mathbf{w}^I + \eta^I \cdot \mathbf{u}^I \quad \text{mit} \quad \mathbf{w}^I, \mathbf{w}^{I+1}, \mathbf{u}^I \in \mathbb{R}^Q \quad \text{und} \quad \eta^I \in \mathbb{R} \quad (3.26)$$

Das mehrdimensionale Optimierungsproblem Gl.(3.21) wird durch Wahl einer Suchrichtung \mathbf{u}^I zu einer *eindimensionalen Liniensuche (line search)* reduziert. Die Bestimmung einer geeigneten Schrittweite η^I sowie der Suchrichtung \mathbf{u}^I ist vom gewählten Verfahren abhängig.

Die benötigte Modellierung der Zielfunktion (Fehleroberfläche) erfolgt durch Abbruch der Taylorreihe von $E(\mathbf{w})$ im jeweiligen Entwicklungspunkt \mathbf{w}^I nach dem linearen oder quadratischen Glied

$$E(\mathbf{w}) = E(\mathbf{w}^I) + (\mathbf{g}^I)^T (\mathbf{w} - \mathbf{w}^I) + \frac{1}{2} (\mathbf{w} - \mathbf{w}^I)^T \mathbf{H}^I (\mathbf{w} - \mathbf{w}^I) + R_2 \quad (3.27)$$

wobei die Matrix \mathbf{H}^I die Hessematrix $\nabla_w^2 E(\mathbf{w})$ an der Stelle \mathbf{w}^I der Fehleroberfläche kennzeichnet. Die symmetrische $(Q \times Q)$ -Hessematrix ergibt sich aus den zweiten partiellen Ableitungen der Zielfunktion bezüglich der Netzgewichte \mathbf{w} gemäß

$$\mathbf{H}(\mathbf{w}) := \nabla_w^2 E(\mathbf{w}) = \begin{pmatrix} \frac{\partial^2 E(\mathbf{w})}{\partial w_{2,1,0} \partial w_{2,1,0}} & \cdots & \frac{\partial^2 E(\mathbf{w})}{\partial w_{2,1,0} \partial w_{L,H_L,H_{L-1}}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 E(\mathbf{w})}{\partial w_{L,H_L,H_{L-1}} \partial w_{2,1,0}} & \cdots & \frac{\partial^2 E(\mathbf{w})}{\partial w_{L,H_L,H_{L-1}} \partial w_{L,H_L,H_{L-1}}} \end{pmatrix} \quad (3.28)$$

Das Restglied zweiter Ordnung R_2 beinhaltet alle Glieder dritter und höherer Ordnung der Taylorreihe Gl.(3.27).

Werden die beschriebenen Schritte wiederholt durchgeführt, ergibt sich ein iteratives Optimierungsverfahren, welches bei Erreichen eines Minimums \mathbf{w}^\square oder einer maximalen Anzahl an Iterationen abgebrochen wird.

Im Minimum \mathbf{w}^\square der Zielfunktion $E(\mathbf{w})$ ist der Betrag des Gradientenvektors Null

$$\|\nabla_w E(\mathbf{w}^\square)\| = 0 \quad (\text{notwendig}) \quad (3.29)$$

und die Hessematrix erfüllt zusätzlich die Bedingung

$$\mathbf{u}^T \cdot \nabla_w^2 E(\mathbf{w}^\square) \cdot \mathbf{u} > \mathbf{0} \quad \forall \mathbf{u} \in \mathbb{R}^Q \quad \mathbf{u} \neq \mathbf{0} \quad (\text{hinreichend}) \quad (3.30)$$

Bei Vorliegen eines Minimums muß demnach der Gradient verschwinden¹⁶ und die Hessematrix positiv definit sein. Da jedoch die Bedingungen Gl.(3.29) und (3.30) nur lokale Eigenschaften der Fehleroberfläche erfassen und die Zielfunktion keine quadratische Funktion ist, ist nicht sichergestellt, daß ein gefundenes Minimum \mathbf{w}^\square auch globales Minimum \mathbf{w}^* des Optimierungsproblems Gl.(3.21) ist. In der Literatur [55] wird die Konvergenz gegen ein lokales Minimum der Zielfunktion jedoch als unwahrscheinlich angesehen.

Liegt dagegen eine quadratische Funktion vor, ist ein lokales Minimum \mathbf{w}^\square zugleich globales Minimum \mathbf{w}^* , weil quadratische Funktionen konvex sind [34, 79]. Das auf Seite 33 beschriebene *MLP*-Netz mit ausschließlich linearen Aktivierungsfunktionen (vektorieller linearer Prädiktor) ist ein Beispiel für ein solches quadratisches Optimierungsproblem mit einer konvexen Fehleroberfläche.

Alle Verfahren, die im Verlauf dieses Kapitels vorgestellt werden, basieren auf dem folgenden Modellalgorithmus zur mehrdimensionalen Optimierung.

Modellalgorithmus für schrittweisenbasierte Abstiegsverfahren:

0 Initialisierung :

- Startwerte für die Netzgewichte \mathbf{w}^0 wählen
- Iterationszähler I=0 setzen

1 Berechnung der neuen Abstiegsrichtung \mathbf{u}^I :

- für diese gilt $(\mathbf{u}^I)^T \cdot \nabla_w E(\mathbf{w}^I) < 0$

2 Berechnung der neuen Schrittweite η^I :

- abhängig vom gewählten Verfahren ist die Schrittweite η^I konstant, Ergebnis einer Heuristik oder Ergebnis einer Liniensuche

3 Berechnung der neuen Näherungslösung \mathbf{w}^{I+1} :

- $\mathbf{w}^{I+1} = \mathbf{w}^I + \eta^I \cdot \mathbf{u}^I$
- Iterationszähler inkrementieren I:=I+1

4 Wenn Abbruchbedingung nicht erfüllt: Weiter bei 1

5 ENDE

3.3 Gradient und Jacobimatrix eines *MLP*-Netzes

Der Gradient $\nabla_w E(\mathbf{w})$ der Zielfunktion Gl.(3.20) ist für die hier verwendeten Abstiegsverfahren von zentraler Bedeutung und wird daher im folgenden abgeleitet.

¹⁶Punkte, für die der Betrag des Gradient $\|\nabla_w E(\mathbf{w}^\square)\|$ Null ist werden als stationär bezeichnet. Neben Minima stellen Maxima und Sattelpunkte mögliche stationäre Punkte dar.

Aus Gleichung (3.20) folgt bei Anwendung der Kettenregel für die partiellen Ableitungen mit

$$\frac{\partial E}{\partial w_{l,ji}} = \frac{1}{K} \sum_{k=1}^K \frac{\partial E^k}{\partial w_{l,ji}} \quad (3.31)$$

zunächst:

$$\frac{\partial E^k}{\partial w_{l,ji}} = \frac{\partial E^k}{\partial s_{l,j}^k} \cdot \frac{\partial s_{l,j}^k}{\partial w_{l,ji}} \quad \text{mit} \quad \begin{array}{l} j=1, \dots, H_l \\ i=0, \dots, H_{l-1} \end{array} \quad \text{und} \quad l=2, \dots, L \quad (3.32)$$

Die partiellen Ableitungen des Summensignals $s_{l,j}^k$ ergeben sich mit Gl.(3.3) zu

$$\frac{\partial s_{l,j}^k}{\partial w_{l,ji}} = \frac{\partial}{\partial w_{l,ji}} \left(\sum_{r=0}^{H_{l-1}} w_{l,jr} \cdot o_{l-1,r}^k \right) = o_{l-1,i}^k \quad (3.33)$$

Mit Gl.(3.32) und der Definition des Fehlerterms

$$\delta_{l,j}^k := \frac{\partial E^k}{\partial s_{l,j}^k} \quad (3.34)$$

vereinfacht sich die Darstellung der partiellen Ableitungen Gl.(3.32) zu

$$\frac{\partial E^k}{\partial w_{l,ji}} = \delta_{l,j}^k \cdot o_{l-1,i}^k \quad (3.35)$$

Aufgrund der linearen Entscheidungsfunktion Gl.(3.15) in der Ausgangslage gilt $s_{L,n}^k = y_n^k$, und man erhält für die Fehlerterme der Ausgangslage

$$\delta_{L,n}^k = y_n^k - d_n^k \quad ; \quad n = 1, \dots, N \quad (3.36)$$

wobei d_n^k die n -te Komponente des Zielvektors $\mathbf{d}^k = (d_1^k, \dots, d_n^k, \dots, d_N^k)^T$ darstellt.

Die Fehlerterme in den verdeckten Lagen ergeben sich bei Berücksichtigung der nichtlinearen Aktivierungsfunktionen $f(s)$ zu:

$$\delta_{l,j}^k = f'(s_{l,j}^k) \cdot \sum_{r=1}^{H_{l+1}} w_{l+1,rj} \delta_{l+1,r}^k \quad \text{mit} \quad \begin{array}{l} l=2, \dots, L-1 \\ j=1, \dots, H_l \end{array} \quad (3.37)$$

Bei Verwendung der logistischen Aktivierungsfunktion Gl.(3.3) in den verdeckten Knoten ergibt sich hieraus mit Gl.(3.13)

$$\delta_{l,j}^k = o(s_{l,j}^k)(1 - o(s_{l,j}^k)) \cdot \sum_{r=1}^{H_{l+1}} w_{l+1,rj} \delta_{l+1,r}^k \quad \text{mit} \quad \begin{array}{l} l=2, \dots, L-1 \\ j=1, \dots, H_l \end{array} \quad (3.38)$$

Die Berechnung des Gradienten $\nabla_{\mathbf{w}} E(\mathbf{w})$ erfolgt entsprechend den obigen Ausführungen in zwei Schritten. Zunächst wird in einem sog. „Vorwärtsgang“ der Ausgangsvektor \mathbf{y}^k berechnet, wobei die Ausgangswerte der Knoten in den verdeckten Lagen $o_{l,j}^k$ wegen

Gl.(3.35) zwischengespeichert werden müssen. Die partiellen Ableitungen der Einzelfehler gemäß Gl.(3.35) werden daraufhin rekursiv aus den Gleichungen (3.36) und (3.38) bestimmt. Die so ermittelten partiellen Ableitungen der Einzelfehler werden abschließend gemäß Gl.(3.31) über der Trainingsmenge $\{\mathbf{x}^k, \mathbf{d}^k\}_{k=1}^K$ gemittelt.

Bei der Berechnung der partiellen Ableitungen wird das Netz von den Fehlertermen $\delta_{l,j}^k$ in umgekehrter Reihenfolge durchlaufen (*error backpropagation*). Diese rekursive Berechnung des Gradienten wird durch das in Abbildung 3.4 dargestellte Netz realisiert. Es

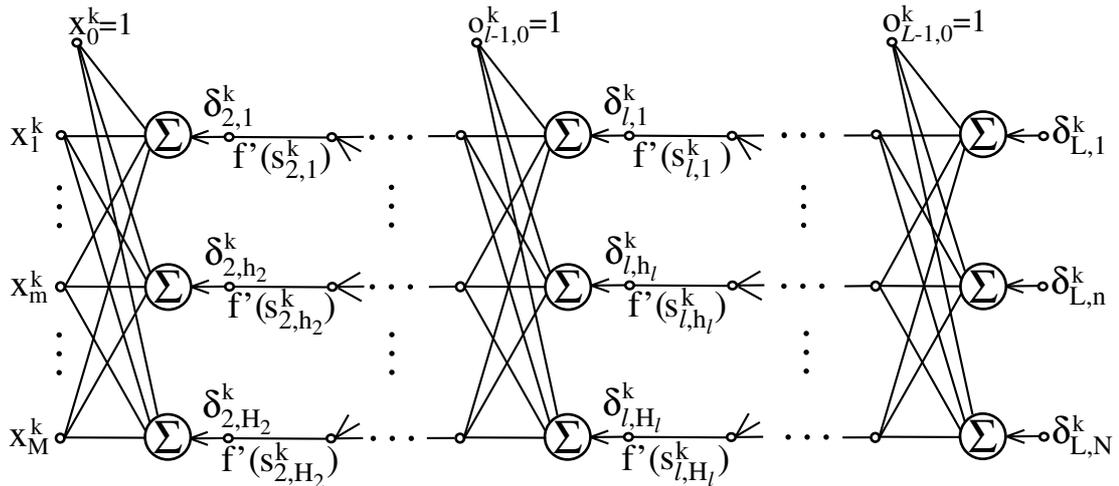


Abbildung 3.4: Netz zur Berechnung des Gradienten von *MLP*-Netzen

entsteht aus dem ursprünglichen *MLP*-Netz (Abb.3.2), wenn statt der nichtlinearen Aktivierungsfunktionen $f(s)$ deren Steigungen $f'(s_{l,j}^k)$ für den aktuellen Eingangsvektor \mathbf{x}^k eingesetzt, und die Ausgangswerte der verdeckten Knoten zwischengespeichert werden.

Der beschriebene Algorithmus zur Berechnung des Gradienten $\nabla_{\mathbf{w}} E(\mathbf{w})$ bietet für die Umsetzung in parallele Hardware zwei Vorteile:

- Alle erforderlichen Berechnungen sind lokal durchführbar, d. h. den informationsverarbeitenden Elementen (den Knoten des Netzes) stehen sämtliche benötigten Größen zur Verfügung. Es sind keine globalen Operationen nötig, und jeder Knoten kann durch einen Prozessor realisiert werden.
- Eine direkte Implementierung der Gradientenberechnung erfordert $O(Q^2)$ Operationen. Demgegenüber liegt der Aufwand bei Verwendung des Backpropagation-Algorithmus (Kap. 3.4) in der Größenordnung von $O(Q)$ Operationen. Damit ist der Aufwand zur Berechnung des Ausgangsvektors \mathbf{y}^k und des Gradienten $\nabla_{\mathbf{w}} E(\mathbf{w})$ in etwa gleich groß.

Auf die Bestimmung der Hessematrix $\nabla_{\mathbf{w}}^2 E(\mathbf{w})$ wird nicht eingegangen, da im Rahmen

dieser Arbeit nur Optimierungsverfahren und Abbruchkriterien eingesetzt werden, die auf eine explizite Berechnung der Hessematrix verzichten.

Zur Schätzung der Lyapunov-Exponenten wird weiterhin die Jacobimatrix der Abbildung $\tilde{\mathcal{G}}$ benötigt, welche die Abbildung des zugrundeliegenden dynamischen Systems approximiert. Zunächst werden die Fälle betrachtet, bei denen die „Original“-Zustandsvektoren vorliegen und das *MLP*-Netz eine Abbildung gemäß Gl.(2.45)

$$\tilde{\mathcal{G}} : \mathbb{R}^{d_S} \rightarrow \mathbb{R}^{d_S} ; \quad \mathbf{y}^k = \tilde{\mathcal{G}}(\mathbf{x}^k) \quad \text{mit} \quad \mathbf{x}^k, \mathbf{y}^k \in \mathbb{R}^{d_S} \quad k \in \mathbb{Z} \quad (3.39)$$

nachbildet oder rekonstruierte Zustandsvektoren vorliegen und das *MLP*-Netz eine Abbildung gemäß Gl.(2.39) mit

$$\tilde{\mathcal{G}} : \mathbb{R}^{d_E} \rightarrow \mathbb{R}^{d_E} ; \quad \mathbf{y}^k = \tilde{\mathcal{G}}(\mathbf{x}^k) \quad \text{mit} \quad \mathbf{x}^k, \mathbf{y}^k \in \mathbb{R}^{d_E} \quad k \in \mathbb{Z} \quad (3.40)$$

nachbildet. Hierbei beschreibt die Jacobimatrix

$$\tilde{\mathcal{G}}_{\mathbf{x}}(\mathbf{x}^k) := \left. \frac{\partial \tilde{\mathcal{G}}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}^k} \quad (3.41)$$

die um \mathbf{x}^k linearisierte Netzabbildung $\tilde{\mathcal{G}}$.

Eine Vorgehensweise analog zur Berechnung des Gradienten führt auf die folgende Gleichung zur Bestimmung der Jacobi-Matrix:

$$\tilde{\mathcal{G}}_{\mathbf{x}}(\mathbf{x}^k) := \mathbf{W}_{lin,2}^k \cdot \dots \cdot \mathbf{W}_{lin,l}^k \cdot \dots \cdot \mathbf{W}_{lin,L-1}^k \cdot \mathbf{W}_L \quad (3.42)$$

mit

$$\mathbf{W}_{lin,l}^k := \begin{pmatrix} w_{l,10} \cdot f'(s_{l,1}^k) & \dots & w_{l,1H_{l-1}} \cdot f'(s_{l,1}^k) \\ \vdots & \ddots & \vdots \\ w_{l,H_l,0} \cdot f'(s_{l,H_l}^k) & \dots & w_{l,H_l,H_{l-1}} \cdot f'(s_{l,H_l}^k) \end{pmatrix} \quad \text{für} \quad l = 2, \dots, L-1 \quad (3.43)$$

Die Berechnung der Jacobi-Matrix gemäß Gl.(3.42) wird durch das in Abbildung 3.5 dargestellte Netz realisiert.

Wird entgegen Gl.(3.39) und (3.40) eine skalare Abbildung gemäß

$$\tilde{\mathcal{G}} : \mathbb{R}^{d_E} \rightarrow \mathbb{R} ; \quad \mathbf{y}^k = \tilde{\mathcal{G}}(\mathbf{x}^k) \quad \text{mit} \quad \mathbf{x}^k \in \mathbb{R}^{d_E} \quad \mathbf{y}^k \in \mathbb{R} \quad k \in \mathbb{Z} \quad (3.44)$$

zur Approximation von Gl.(2.41) und zur Schätzung der Lyapunov-Exponenten eingesetzt, ergibt sich aus Gl.(2.40) unter der Voraussetzung $pw = \tau$ für die Vektoren \mathbf{x}^k und \mathbf{y}^k :

$$\begin{pmatrix} x_1^k \\ x_2^k \\ \vdots \\ x_{d_E-1}^k \\ x_{d_E}^k \end{pmatrix} = \begin{pmatrix} v^k \\ v^{k+\tau} \\ \vdots \\ v^{k+(d_E-2)\tau} \\ v^{k+(d_E-1)\tau} \end{pmatrix} \rightarrow \begin{pmatrix} y_1^k \\ y_2^k \\ \vdots \\ y_{d_E-1}^k \\ y_{d_E}^k \end{pmatrix} = \begin{pmatrix} v^{k+\tau} \\ v^{k+2\tau} \\ \vdots \\ v^{k+(d_E-1)\tau} \\ \tilde{\mathcal{G}}(\mathbf{x}^k) \end{pmatrix} = \begin{pmatrix} x_2^k \\ x_3^k \\ \vdots \\ x_{d_E}^k \\ \tilde{\mathcal{G}}(\mathbf{x}^k) \end{pmatrix}, \quad (3.45)$$

und für die Jacobi-Matrix erhält man:

$$\tilde{\mathcal{G}}_{\mathbf{x}}(\mathbf{x}^k) = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ \frac{\partial \tilde{\mathcal{G}}(\mathbf{x}^k)}{\partial x_1} & \frac{\partial \tilde{\mathcal{G}}(\mathbf{x}^k)}{\partial x_2} & \frac{\partial \tilde{\mathcal{G}}(\mathbf{x}^k)}{\partial x_3} & \dots & \frac{\partial \tilde{\mathcal{G}}(\mathbf{x}^k)}{\partial x_{d_E}} \end{pmatrix}. \quad (3.46)$$

Zur Bestimmung der Jacobi-Matrix gemäß Gl.(3.46) muß nur die letzte Zeile berechnet werden, wozu wiederum Gl.(3.42) verwendet wird, wobei die Matrix der Ausgangsgewichte \mathbf{W}_L zu einem Vektor degeneriert. Diese Berechnung kann ebenfalls mit einem gegenüber Abbildung 3.5 modifizierten Netz erfolgen, welches in der letzten Lage nur einen skalaren „Eingang“ $y^k = \tilde{\mathcal{G}}(\mathbf{x}^k)$ aufweist.

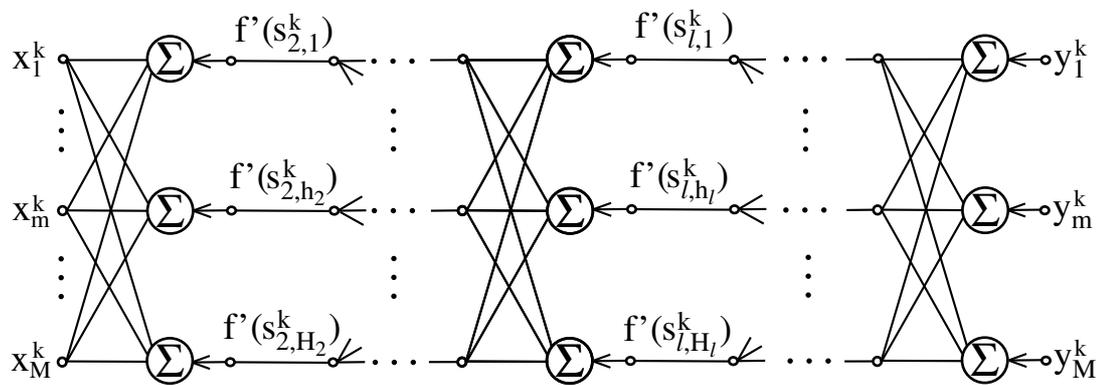


Abbildung 3.5: Netz zur Berechnung der Jacobi-Matrix von *MLP*-Netzen

3.4 Das Backpropagation-Verfahren und Modifikationen

Das einfachste und zugleich am häufigsten verwendete Optimierungsverfahren für *MLP*-Netze ist der **BackPropagation**-Algorithmus¹⁷ (**BP**-Algorithmus). Obwohl zuvor bereits mehrfach beschrieben [3, 17, 104, 144], hat sich dieses Optimierungsverfahren erst mit der Veröffentlichung von Rumelhart, Hinton und Willams [123] zum gebräuchlichsten Lernverfahren für *MLP*-Netze entwickelt. Als Abstiegs- bzw. Suchrichtung wird der negative Gradient $-\nabla_{\mathbf{w}}E(\mathbf{w})$ der Zielfunktion für die jeweiligen Netzgewichte gewählt.

Der Backpropagation-Algorithmus umfaßt zwei Varianten, die sich bezüglich der Berechnung einer neuen Näherungslösung unterscheiden [55, 58].

Bei Verwendung des *Batch-BP*-Algorithmus wird eine Änderung des Gewichtsvektors gemäß der Vorschrift

$$\mathbf{w}^{I+1} = \mathbf{w}^I - \eta_{BP} \cdot \nabla_{\mathbf{w}}E(\mathbf{w}^I) \quad (3.47)$$

erst durchgeführt, nachdem alle K Trainingsmuster verarbeitet wurden. Entsprechend den Ausführungen aus Kapitel 3.3 werden hierbei die Beiträge der Einzelfehler gemäß Gl.(3.31) über der Trainingsmenge gemittelt, um eine genaue Schätzung des Gradienten zu erhalten.

Die zweite Variante, welche als *Online-BP*-Algorithmus bekannt ist [55, 7], führt eine Änderung des Gewichtsvektors

$$\mathbf{w}^{neu} = \mathbf{w}^{alt} - \eta_{BP} \cdot \nabla_{\mathbf{w}}E^k(\mathbf{w}^{alt}) \quad (3.48)$$

nach jedem Trainingsmuster durch. Der Vergleich mit anderen Verfahren wird erleichtert, wenn die einmalige Verarbeitung aller K Trainingsmuster als eine Iteration vereinbart wird. In der Update-Vorschrift Gl.(3.48) wird mit $\nabla_{\mathbf{w}}E^k(\mathbf{w}^{alt})$ eine recht ungenaue Schätzung des Gradienten verwendet. Die Vorteile dieses Verfahrens werden jedoch gerade darin gesehen, daß der eigentlichen Suchrichtung (dem negativen Gradienten) hierdurch eine Art „Rauschen“ überlagert wird, wodurch das Festlaufen des Algorithmus in lokalen Minima weitgehend vermieden werden soll¹⁸.

Die Schrittweite η_{BP} , welche auch als Lernrate des Backpropagation-Algorithmus bezeichnet wird, ist in beiden Fällen konstant und muß vom Anwender vorgegeben werden. Das Minimum entlang der Suchrichtung wird bei dieser Vorgehensweise i. allg. nicht erreicht. Eine günstige Wahl der Schrittweite η_{BP} hat entscheidenden Einfluß auf die Konvergenzgeschwindigkeit des Optimierungsverfahrens. Ein zu kleiner Wert ergibt nur eine sehr geringe Gewichtsänderung, und es werden dementsprechend wesentlich mehr Iterationen benötigt. Wird hingegen η_{BP} zu groß gewählt, kann sogar eine Verschlechterung

¹⁷Das Backpropagation-Lernverfahren ist auch als *verallgemeinerte Delta-Regel (generalized delta rule)* bekannt.

¹⁸Entsprechende Überlegungen bilden die Basis der Verfahren des „simulated annealing“.

des Wertes der Zielfunktion eintreten. Die Konvergenz des Verfahrens ist dann nicht mehr gewährleistet.

Der beschriebene *BP*-Algorithmus mit fester Schrittweite hat den Nachteil, daß neben der eigentlichen Minimierung von Gl.(3.21) der Verfahrensparameter η_{BP} in einer „übergeordneten“ Optimierung bestimmt werden muß. Weiterhin ist die optimale Schrittweite von der lokalen Beschaffenheit der Fehleroberfläche abhängig und damit für jede Iteration eine andere. Besonders in sehr flachen Gebieten mit nahezu verschwindendem Gradienten hat eine konstante Schrittweite η_{BP} nur sehr kleine Gewichtsänderungen und damit eine geringe Konvergenzgeschwindigkeit des *BP*-Algorithmus zur Folge. Zur Vermeidung dieser Nachteile wurde u. a. von Vogel et. al. [141] ein heuristisches Verfahren zur Adaption der Schrittweite vorgestellt, daß ohne nennenswerte Erhöhung des Rechenaufwandes realisierbar ist. Dieses wird als „**Bold-Driver**“-Heuristik (*BD*-Verfahren) bezeichnet und verändert die Schrittweite η_{BD} abhängig vom Fortschritt der Optimierung¹⁹. Verringert sich, wie in Gl.(3.22) für Abstiegsverfahren gefordert, der Wert der Zielfunktion, wird die Schrittweite η_{SD} vergrößert

$$E(\mathbf{w}^I) < E(\mathbf{w}^{I-1}) \iff \eta_{BD}^{I+1} = \eta_{BD}^I \cdot \gamma \quad \text{mit} \quad \gamma > 1 \in \mathbb{R} \quad (3.49)$$

Hat sich der Wert der Zielfunktion hingegen durch die letzte Gewichtsänderung vergrößert, wird die Gewichtsänderung zurückgenommen und die Schrittweite deutlich verringert.

$$E(\mathbf{w}^I) > E(\mathbf{w}^{I-1}) \iff \eta_{BD}^{I+1} = \eta_{BD}^I \cdot \beta \quad \text{mit} \quad 0 < \beta < 1 \in \mathbb{R} \quad (3.50)$$

Bei Verwendung dieser Heuristik²⁰ schwankt die Schrittweite ständig in einem Bereich um die optimale Schrittweite, so daß die Konvergenzgeschwindigkeit des *BD*-Verfahrens, verglichen mit dem *BP*-Algorithmus, wesentlich erhöht wird.

Der *BP*-Algorithmus sowie die *BD*-Heuristik stellen Sonderfälle des bekannten *Verfahrens des steilsten Abstiegs* (*steepest descent method*) dar [79, 34], dessen prinzipieller Ablauf in Abbildung 3.6 veranschaulicht wird. Die Darstellung der Fehleroberfläche erfolgt in Form von Höhenlinien, also Linien mit konstantem Wert der Zielfunktion $E(\mathbf{w})$. Das Minimum der Zielfunktion im „Zentrum“ der Höhenlinien ist mit \mathbf{w}^* gekennzeichnet.

Als Suchrichtung \mathbf{u}^I wird ebenfalls der negative Gradient im aktuellen Punkt \mathbf{w}^I verwendet. Es wird jedoch in jeder Iteration eine optimale Schrittweite η_{SD}^I entlang der Suchrichtung durch Lösung des eindimensionalen Optimierungsproblems

$$E(\mathbf{w}^{I+1}) = E(\mathbf{w}^I - \eta_{SD}^I \cdot \mathbf{g}^I) = \min_{\eta} E(\mathbf{w}^I - \eta \cdot \mathbf{g}^I) \quad \text{mit} \quad \eta, \eta_{SD}^I \in \mathbb{R}^+ \quad (3.51)$$

¹⁹Darüber hinaus wird von Jacobs [64] eine Erweiterung des *BD*-Verfahren vorgeschlagen, bei der jedes einzelne Gewicht w_{l,j_i} eine eigene Schrittweite η_{l,j_i} zugewiesen bekommt.

²⁰Gebräuchliche Werte für die Parameter der *Bold-Driver*-Heuristik sind $\gamma = 1,05$ und $\beta = 0,5$. Diese wurden auch bei sämtlichen mit dem *BD*-Verfahren durchgeführten Untersuchungen aus Kapitel 5 verwendet.

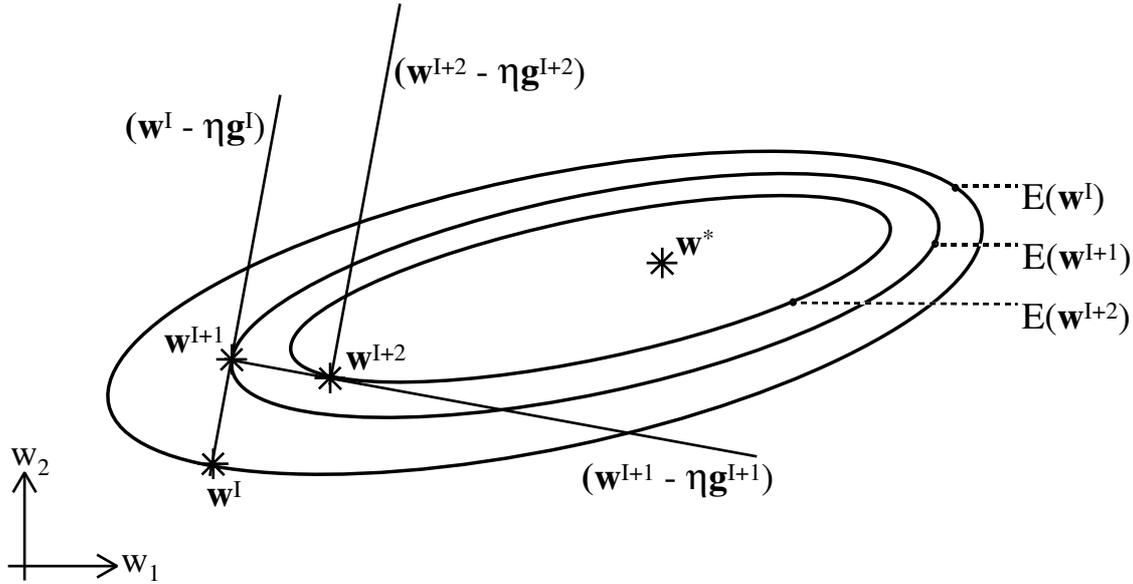


Abbildung 3.6: Verfahren des steilsten Abstiegs

bestimmt. Im Minimum \mathbf{w}^{I+1} der Liniensuche tangiert der Suchstrahl $(\mathbf{w}^I - \eta \cdot \mathbf{g}^I)$ die Höhenlinie der Fehleroberfläche mit dem Wert $E(\mathbf{w}^{I+1})$. Dieses ist eine prinzipielle Eigenschaft des Minimums einer Liniensuche und gilt für jede Suchrichtung \mathbf{u}^I , die Gl.(3.24) erfüllt, also Abstiegsrichtung ist.

$$(\mathbf{g}^{I+1})^T \cdot \mathbf{u}^I = 0 \quad (3.52)$$

Für einen Liniensuch-Algorithmus ist Gl.(3.52) praktisch nicht erfüllbar. Aus diesem Grund muß eine Abbruchbedingung für die eindimensionale Optimierung angegeben werden, die eine Bewertung der erzielten Näherung vornimmt. Die Genauigkeit der Liniensuche kann unter Verwendung der Bedingung

$$|\mathbf{g}(\mathbf{w}^I + \eta \cdot \mathbf{u}^I)^T \cdot \mathbf{u}^I| \leq -\varepsilon \cdot (\mathbf{g}^I)^T \cdot \mathbf{u}^I \quad (3.53)$$

durch den Parameter ε gesteuert werden²¹. Gleichung (3.53) wird im folgenden als Winkelbedingung bezeichnet und dient als Abbruchbedingung der zur Minimierung eingesetzten Liniensuch-Algorithmen.

Das Verfahren erzeugt demnach bei exakten Liniensuchen eine Folge senkrecht aufeinander stehender Suchrichtungen. Diese Besonderheit führt bei schlecht konditionierten Optimierungsproblemen²² zu dem sogenannten *Zick-Zack-Laufen* (*zig-zagging*) [79]. Dieser Effekt kann bereits in Abbildung 3.6 beobachtet werden und tritt in Bereichen mit

²¹Je kleiner der Wert von ε gewählt wird, um so genauer erfolgt die Bestimmung des Minimums entlang der Suchrichtung. Andererseits steigt jedoch dadurch die Anzahl der für die Liniensuche benötigten Funktionswert- und Gradientenberechnungen und damit der Rechenaufwand des Optimierungsverfahrens u. U. erheblich an.

²²Optimierungsprobleme, deren Hessematrix einen stark unterschiedlichen maximalen ($\tilde{\lambda}_{max} > 0$) und

stark unterschiedlichen Eigenwerten der Hessematrix²³ verstärkt auf. Dieses sind die bereits erwähnten „tiefen, schmalen Schluchten“ der Fehleroberfläche mit einer „flachen Talsohle“. Hier ist die Komponente des Gradienten in Richtung des Minimums gering, in anderen Richtungen jedoch sehr groß. Es kommt zu oszillierenden Suchrichtungen, was eine extrem geringe Konvergenzgeschwindigkeit zur Folge hat.

In [79] wird für allgemeine nichtlineare Optimierungsprobleme gezeigt, daß die lokale Konvergenzgeschwindigkeit des *SD*-Verfahrens in Lösungsnahe

$$\frac{E(\mathbf{w}^{I+1}) - E(\mathbf{w}^*)}{E(\mathbf{w}^I) - E(\mathbf{w}^*)} \approx \frac{(\tilde{\lambda}_{max} - \tilde{\lambda}_{min})^2}{(\tilde{\lambda}_{max} + \tilde{\lambda}_{min})^2} \quad (3.54)$$

beträgt, wobei $\tilde{\lambda}_{max}$ und $\tilde{\lambda}_{min}$ der größte bzw. kleinste Eigenwert der als positiv-definit angenommenen Hessematrix $\nabla_w^2 E(\mathbf{w}^*)$ ist. Für den angesprochenen Fall, daß $\tilde{\lambda}_{max} \gg \tilde{\lambda}_{min}$ gilt, wird der Quotient in Gl.(3.54) nahezu eins. Der Wert der Zielfunktion nimmt im Laufe der Iterationen nur minimal ab.

Der Effekt der oszillierenden Suchrichtungen tritt entsprechend auch bei Verwendung des *BP*-Algorithmus und der *BD*-Heuristik auf. Für beide Versionen des *BP*-Algorithmus wird bereits in [112, 123] eine Heuristik zur Modifikation der Update-Vorschrift Gl.(3.47) angegeben, mit der dieser Effekt deutlich reduziert werden kann. Die gleiche Modifikation kann auch für die *BD*-Heuristik (Gl.(3.49), (3.50)) sowie das *SD*-Verfahren (Gl.(3.51)) angewendet werden.

Durch Addition des *Momentum-Terms* $\alpha_M \cdot \Delta \mathbf{w}^I$ werden die Gewichte nicht mehr nur in Richtung des negativen Gradienten geändert.

$$\mathbf{w}^{I+1} = \mathbf{w}^I - \eta_{BP} \cdot \mathbf{g}^I + \alpha_M \cdot \Delta \mathbf{w}^I \quad \text{mit} \quad \alpha_M \in \{0, 1\} \quad (3.55)$$

Die Berücksichtigung der mit der *Momentum-Konstante* α_M bewerteten Gewichtsänderung $\Delta \mathbf{w}^I = \mathbf{w}^I - \mathbf{w}^{I-1}$ der letzten Iteration bewirkt eine Art „Mittelung“ der alten Suchrichtungen. Hierdurch werden das Auftreten oszillierender Richtungen reduziert und die Suchrichtungen in Richtung des Minimums gelenkt. Weiterhin kann gezeigt werden [142], daß in Gebieten der Fehleroberfläche mit nahezu konstantem Gradienten der Momentum-Term zu einer um den Faktor $1/(1 - \alpha_M)$ vergrößerten Schrittweite führt, wodurch die Konvergenzgeschwindigkeit in diesen Gebieten steigt.

Die Verwendung des Momentum-Terms hat jedoch den Nachteil, daß der optimale Wert der Momentum-Konstante α_M experimentell ermittelt werden muß²⁴. In Verbindung mit

minimalen ($\tilde{\lambda}_{min} > 0$) Eigenwert besitzen, werden als schlecht konditionierte (steife) Probleme mit der Konditionszahl $\kappa = \tilde{\lambda}_{max}/\tilde{\lambda}_{min}$ bezeichnet. Zu diesen schlecht konditionierten Problemen gehört auch die Optimierung von *MLP*-Netzen, weil deren Fehleroberflächen i. allg. die bereits erwähnte komplexe Struktur aufweisen.

²³Sind hingegen die Eigenwerte der Hessematrix alle identisch, entsprechen die Höhenlinien der Fehleroberfläche konzentrischen Kreisen, und der negative Gradient zeigt in Richtung des Minimums.

²⁴Gebräuchliche Werte der Momentum-Konstante bewegen sich im Bereich $0,7 < \alpha_M < 0,95$ [55].

dem *BP*-Algorithmus sind also bereits zwei Verfahrensparameter vom Anwender vorzugeben. Die Lokalität der Berechnungen bleibt bei dieser Kombination allerdings gewährleistet.

Dem gegenüber verletzen alle Verfahren, die eine Liniensuche erfordern, das für neuronale Netze geforderte Prinzip der strikten Lokalität aller Berechnungen, weil der Line-Search-Algorithmus einen globalen Informationsaustausch erforderlich macht. Hierzu zählt im Prinzip auch die *BD*-Heuristik. Diese Verfahren werden jedoch wegen ihrer um Größenordnungen besseren Konvergenzgeschwindigkeiten vermehrt eingesetzt. Eine effiziente Hardware-Umsetzung ist trotz der globalen Berechnungen in den meisten Fällen möglich.

3.5 Verfahren der konjugierten Gradienten

Die *Verfahren der konjugierten Gradienten* (*Conjugate Gradient methods*) (*CG*-Verfahren) sind bezüglich ihrer Ansätze sowie ihrer Konvergenzverhalten zwischen dem *SD*-Verfahren und dem Newton-Verfahren anzuordnen. Im Vergleich zum *SD*-Verfahren ist der Rechenaufwand bei wesentlich verbesserten Konvergenzeigenschaften nur geringfügig höher. Da der Speicheraufwand ebenfalls nur linear mit der Ordnung des Problems ansteigt, gehören diese Verfahren derzeit zu den effizientesten für die Optimierung sehr großer *MLP*-Netze.

Die geringe Konvergenzgeschwindigkeit der im letzten Abschnitt behandelten Verfahren beruht im wesentlichen darauf, daß aufeinanderfolgende eindimensionale Optimierungen entlang der Suchrichtungen $\mathbf{u}^I, \mathbf{u}^{I-1}$ nicht gleichbedeutend mit der Minimierung der Zielfunktion bezüglich des durch $\mathbf{u}^I \times \mathbf{u}^{I-1}$ aufgespannten Unterraumes sind. Die Verfahren der konjugierten Gradienten verhindern durch die Erzeugung konjugierter Suchrichtungen, daß sich aufeinanderfolgende Liniensuchen negativ beeinflussen.

Die Ableitung dieser Verfahren erfolgt auf der Grundlage eines quadratischen Optimierungsproblems mit positiv-definiter Hessematrix. Ausgehend von der Taylorreihenentwicklung der Zielfunktion Gl.(3.27) erhält man die quadratische Approximation der lokalen Beschaffenheit der Fehleroberfläche gemäß

$$E(\mathbf{w} + \Delta\mathbf{w}) \approx \tilde{E}(\mathbf{w} + \Delta\mathbf{w}) \stackrel{!}{=} E(\mathbf{w}) + \nabla_{\mathbf{w}}E(\mathbf{w})^T \cdot \Delta\mathbf{w} + \frac{1}{2}(\Delta\mathbf{w})^T \cdot \mathbf{H} \cdot \Delta\mathbf{w} \quad (3.56)$$

Die Ableitung des quadratischen Modells bezüglich des Differenzvektors der Netzgewichte lautet

$$\nabla_{\mathbf{w}}\tilde{E}(\mathbf{w} + \Delta\mathbf{w}) = \nabla_{\mathbf{w}}E(\mathbf{w}) + \mathbf{H} \cdot \Delta\mathbf{w} \quad (3.57)$$

Die *CG*-Verfahren bilden die Suchrichtungen \mathbf{u}^I , ähnlich wie die im letzten Abschnitt behandelten Gradientenverfahren mit Momentum-Term, aus der Überlagerung des negativen Gradienten mit vergangenen Suchrichtungen. Als erste Suchrichtung wird der

negative Gradient im Startpunkt verwendet.

$$\begin{aligned}\mathbf{u}^0 &= -\nabla_w \tilde{E}(\mathbf{w}^0) \\ \mathbf{u}^I &= -\nabla_w \tilde{E}(\mathbf{w}^I) + \beta^{I-1} \cdot \mathbf{u}^{I-1} \quad \text{mit } I = 1, \dots, Q-1; \beta^{I-1} \in \mathbb{R}\end{aligned}\quad (3.58)$$

Im Gegensatz zur Momentum-Konstanten α_M ergeben sich die Werte der Faktoren β^{I-1} aus der Forderung, daß eine Liniensuche entlang \mathbf{u}^I die Ergebnisse vorangegangener Liniensuchen nicht beeinträchtigt.

Gemäß Gl.(3.52) gilt für den Gradienten im Minimum der Liniensuche entlang \mathbf{u}^{I-1}

$$(\mathbf{u}^{I-1})^T \cdot \nabla_w \tilde{E}(\mathbf{w}^I) = 0 \quad (3.59)$$

Die nächste Suchrichtung \mathbf{u}^I ist demnach so zu wählen, daß der sich ergebende Gradient $\nabla_w \tilde{E}(\mathbf{w}^{I+1})$ in erster Näherung keine Komponente in Richtung der alten Suchrichtung \mathbf{u}^{I-1} aufweist

$$(\mathbf{u}^{I-1})^T \cdot \nabla_w \tilde{E}(\mathbf{w}^I + \eta_{CG} \mathbf{u}^I) = 0 \quad (3.60)$$

Einsetzen der aus dem quadratischen Modell abgeleiteten Gleichung (3.57) ergibt unter Berücksichtigung von $\eta_{CG} \neq 0$ sowie Gl.(3.52) folgende Bedingung für die aufeinanderfolgenden Suchrichtungen

$$(\mathbf{u}^{I-1})^T \cdot \mathbf{H} \cdot \mathbf{u}^I = 0 \quad (3.61)$$

Wird die Eigenschaft Gl.(3.60) für alle vorangegangenen Liniensuchen gefordert, sind im Verlauf des *CG*-Verfahrens Suchrichtungen $\{\mathbf{u}^i\}_{i=0}^{Q-1}$ zu erzeugen, die bezüglich der Hessematrix \mathbf{H} des quadratischen Optimierungsproblems konjugiert (B.15) sind

$$(\mathbf{u}^i)^T \cdot \mathbf{H} \cdot \mathbf{u}^j = 0 \quad \forall \quad i \neq j \quad i, j = 0, \dots, Q-1 \quad (3.62)$$

Ausgehend von der rekursiven Vorschrift Gl.(3.58) zur Erzeugung der Suchrichtungen sowie unter Ausnutzung der Eigenschaft konjugierter Richtungen Gl.(3.62), ergeben sich verschiedene Möglichkeiten zur Berechnung der Faktoren η_{CG}^{I-1} . Bei den klassischen Verfahren der konjugierten Gradienten werden die Bestimmungsgleichungen nach Polak-Ribière und Fletcher-Reeves unterschieden

$$\beta^{I-1} = \frac{(\nabla_w \tilde{E}(\mathbf{w}^I) - \nabla_w \tilde{E}(\mathbf{w}^{I-1}))^T \nabla_w \tilde{E}(\mathbf{w}^I)}{\nabla_w \tilde{E}(\mathbf{w}^{I-1})^T \cdot \nabla_w \tilde{E}(\mathbf{w}^{I-1})} \quad (\text{Polak-Ribière}) \quad (3.63)$$

Wird weiterhin $(\nabla_w \tilde{E}(\mathbf{w}^{I-1}))^T \nabla_w \tilde{E}(\mathbf{w}^I) = 0$ eingesetzt, folgt aus Gl.(3.63) das Verfahren von Fletcher-Reeves

$$\beta^{I-1} = \frac{\nabla_w \tilde{E}(\mathbf{w}^I)^T \cdot \nabla_w \tilde{E}(\mathbf{w}^I)}{\nabla_w \tilde{E}(\mathbf{w}^{I-1})^T \cdot \nabla_w \tilde{E}(\mathbf{w}^{I-1})} \quad (\text{Fletcher-Reeves}) \quad (3.64)$$

Die Implementierung der *CG*-Verfahren erfolgt auf der Basis des auf Seite 36 beschriebenen Modellalgorithmus. Wie bereits angesprochen, ist der Speicheraufwand mit den

vier Vektoren $\mathbf{w}^I, \mathbf{u}^I$ sowie $\mathbf{g}^I, \mathbf{g}^{I-1}$ kaum größer als bei den in Kapitel 3.4 vorgestellten Verfahren. Der Vorteil der *CG*-Verfahren liegt jedoch darin, daß zusätzlich Informationen zweiter Ordnung über die Beschaffenheit der Fehleroberfläche durch die Liniensuchen und die Erzeugung neuer Suchrichtungen bei der Optimierung berücksichtigt werden, ohne die Hessematrix berechnen oder speichern zu müssen.

Auf quadratische Optimierungsprobleme angewendet, besitzen die Verfahren der konjugierten Gradienten die nachfolgend beschriebenen Eigenschaften [79, 34]

Theorem 4: Konvergenz der Verfahren konjugierter Gradienten (Fletcher [34]):

Gegeben sei ein quadratisches Optimierungsproblem in Q Variablen mit positiv-definiter Hessematrix \mathbf{H} , eine Menge $\mathbf{u}^0, \dots, \mathbf{u}^{Q-1}$ nichttrivialer \mathbf{H} -orthogonaler Vektoren, sowie ein Startvektor \mathbf{w}^0 .

Ein Verfahren der konjugierten Gradienten löst das quadratische Optimierungsproblem mit maximal Q exakten Liniensuchen²⁵ entlang der Suchrichtungen $\mathbf{u}^0, \dots, \mathbf{u}^{Q-1}$. Dabei stellt jeder Iterationspunkt \mathbf{w}^{I+1} das Minimum der quadratischen Zielfunktion bezüglich des durch die Vektoren $\mathbf{u}^0, \dots, \mathbf{u}^I$ aufgespannten Unterraumes dar.

Für quadratische Optimierungsprobleme besitzen die *CG*-Verfahren Q -Schritt-superlineare Konvergenz (B.14). Werden diese Verfahren für allgemeine nichtlineare Probleme eingesetzt, wird ein Q -Schritt-quadratisches Konvergenzverhalten erwartet [79]. Bei der Optimierung von *MLP*-Netzen wird gegenüber dem *BP*-Algorithmus und der *BD*-Heuristik eine Steigerung der Konvergenzgeschwindigkeit um mindestens eine Zehnerpotenz erzielt [5, 71]. Auf allgemeine nichtlineare Probleme angewendet, konvergieren die *CG*-Verfahren i. allg. nicht in Q Iterationen. Da jedoch maximal Q konjugierte Richtungen bei einem Problem in Q Variablen existieren, werden bei den klassischen Verfahren im wesentlichen zwei Restart-Strategien unterschieden.

Einerseits können die Suchrichtungen unverändert in der beschriebenen Weise berechnet werden. Andererseits besteht die Möglichkeit, nach jeweils Q Iterationen einen Restart durchzuführen. In diesem Fall wird als Suchrichtung erneut der negative Gradient verwendet. Die in den vergangenen Suchrichtungen enthaltenen Informationen zweiter Ordnung gehen in diesem Fall allerdings verloren. Von Luenberger [79] wird trotzdem ein Restart nach jeweils Q Iterationen empfohlen.

²⁵Die Schrittweiten η_{CG}^I ; $I = 0, \dots, Q - 1$ berechnen sich bei quadratischen Optimierungsproblemen zu $\eta_{CG}^I = -(\nabla_w \tilde{E}(\mathbf{w}^I)^T \mathbf{u}^I) / ((\mathbf{u}^I)^T \mathbf{H} \mathbf{u}^I)$ [39]. Soll die Berechnung und Speicherung der Hessematrix vermieden werden, sind die Schrittweiten mittels exakter Liniensuchen zu bestimmen.

Im folgenden werden entsprechend den obigen Ausführungen vier Varianten der klassischen *CG*-Verfahren unterschieden:

- Polak-Ribière ohne Restart (PR-Verfahren)
- Polak-Ribière mit **R**estart (PRR-Verfahren)
- Fletcher-Reeves ohne Restart (FR-Verfahren)
- Fletcher-Reeves mit **R**estart (FRR-Verfahren)

Die Berechnung der Suchrichtung gemäß Gl.(3.58) in Verbindung mit nicht exakten Liniensuchen kann unter Umständen zu Aufstiegsrichtungen führen. Multipliziert man Gl.(3.58) von links mit dem Gradienten, ergibt sich für die Richtungsableitung in der *I*-ten Iteration:

$$\nabla_w \tilde{E}(\mathbf{w}^I)^T \cdot \mathbf{u}^I = -\nabla_w \tilde{E}(\mathbf{w}^I)^T \cdot \nabla_w \tilde{E}(\mathbf{w}^I) + \beta^{I-1} \cdot \nabla_w \tilde{E}(\mathbf{w}^I)^T \cdot \mathbf{u}^{I-1} \quad (3.65)$$

Der erste Summand auf der rechten Seite ist definitionsgemäß negativ, in flachen Bereichen der Fehleroberfläche jedoch sehr klein. Daher kann die Richtungsableitung durch den zweiten Term positiv werden, der bei exakter Liniensuche gemäß Gl.(3.52) null ist. Es läge eine Aufstiegsrichtung vor, wobei die anschließende eindimensionale Optimierung versagt. Um diese Schwierigkeit zu vermeiden, sind bei Verwendung der klassischen *CG*-Verfahren nach Fletcher-Reeves und Polak-Ribière exakte Liniensuchen besonders in flachen Bereichen der Fehleroberfläche unerlässlich.

Diese Forderung bewirkt, daß der Rechenaufwand pro Iteration je nach Qualität des implementierten Liniensuch-Algorithmus gegenüber dem *BP*-Algorithmus oder der *BD*-Heuristik wesentlich höher ausfallen kann. Die letzteren Verfahren benötigen je Iteration eine Funktionswertberechnung $E(\mathbf{w}^I)$ und eine Gradientenberechnung $\nabla_w E(\mathbf{w}^I)$. Aus der Literatur ist bekannt, daß für eine exakte Liniensuche bei der Optimierung von *MLP*-Netzen im Mittel 3 - 15 Funktions- und Gradientenberechnungen nötig sind [90], der Rechenaufwand also um den Faktor 3 - 15 höher ist. Aus diesem Grund ist bei der Verwendung der Verfahren konjugierter Gradienten ein effizienter Liniensuch-Algorithmus überaus wichtig.

3.6 Newton- und Quasi-Newton-Verfahren

Das Newton-Verfahren basiert ebenso wie die *CG*-Verfahren auf der quadratischen Approximation der Zielfunktion $E(\mathbf{w})$ im jeweiligen Startpunkt \mathbf{w}^I

$$\tilde{E}(\mathbf{w}^I + \Delta \mathbf{w}) \stackrel{!}{=} E(\mathbf{w}^I) + \nabla_w E(\mathbf{w}^I)^T \cdot \Delta \mathbf{w} + \frac{1}{2} (\Delta \mathbf{w})^T \cdot \nabla_w^2 E(\mathbf{w}^I) \cdot \Delta \mathbf{w} \quad (3.66)$$

Eine neue Näherungslösung $\mathbf{w}^{I+1} = \mathbf{w}^I + \Delta \mathbf{w}^I$ des allgemeinen nichtlinearen Optimierungsproblems Gl.(3.21) wird mit dem das quadratische Modell minimierenden Differenzvektor

$$\Delta \mathbf{w}^I = -\nabla_w^2 E(\mathbf{w}^I)^{-1} \cdot \nabla_w E(\mathbf{w}^I) \quad (3.67)$$

berechnet. Damit ergibt sich die neue Näherungslösung zu

$$\mathbf{w}^{I+1} = \mathbf{w}^I - \nabla_w^2 E(\mathbf{w}^I)^{-1} \cdot \nabla_w E(\mathbf{w}^I) \quad (3.68)$$

Das durch diese Gleichung definierte iterative Verfahren ist das klassische Newton-Verfahren. Der Vergleich mit Gl.(3.26) zeigt, daß

$$\mathbf{u}_N^I = -\nabla_w^2 E(\mathbf{w}^I)^{-1} \cdot \nabla_w E(\mathbf{w}^I) \quad (3.69)$$

der Suchrichtung des Newton-Verfahrens entspricht, die als Newton-Richtung bezeichnet wird. Die Lösung eines quadratischen Problems mit positiv-definiter Hessematrix wird durch Gleichung (3.68) in einem Schritt bestimmt. Auf allgemeine nichtlineare Probleme angewendet, besitzt das Newton-Verfahren mindestens quadratische Konvergenz, wenn der Startpunkt \mathbf{w}^0 in ausreichender Nähe des Minimums \mathbf{w}^* gewählt wird und die Hessematrix $\nabla_w^2 E(\mathbf{w}^I)$ positiv-definit ist [79, 34].

Die Optimierung von *MLP*-Netzen ist mit dem Verfahren in der beschriebenen Form nicht durchführbar, da die Existenz einer positiv-definiten Hessematrix nicht garantiert ist und die Newton-Richtung daher nicht immer eine Abstiegsrichtung ist. Auch bei positiv-definiter Hessematrix ist durch Gl.(3.68) die Konvergenz des Verfahrens nicht gesichert. Der Wert der Zielfunktion kann sogar ansteigen, wenn das quadratische Modell die Zielfunktion nicht ausreichend genau approximiert, weil der Startvektor nicht in der Nähe des gesuchten Minimums gewählt wurde.

Das Newton-Verfahren wird robuster, wenn statt der festen Schrittweite $\eta = 1$ (vgl. Gl.(3.68)) die in jeder Iteration optimale Schrittweite η_N^I mit Hilfe einer Liniensuche entlang der Newton-Richtung \mathbf{u}_N^I bestimmt wird²⁶.

Eine neue Näherungslösung ergibt sich bei Verwendung des Newton-Verfahrens mit Liniensuche zu

$$\mathbf{w}^{I+1} = \mathbf{w}^I - \eta_N^I \cdot \nabla_w^2 E(\mathbf{w}^I)^{-1} \cdot \nabla_w E(\mathbf{w}^I) \quad (3.70)$$

Die Probleme bei negativ-definiter oder indefiniter Hessematrix werden jedoch auch durch die Verwendung einer Liniensuche nicht gelöst. Probleme ergeben sich bei der Implementierung des Newton-Verfahrens durch den hohen Speicherplatzbedarf und den Aufwand zur Berechnung ($O(Q^2)$) und Invertierung ($O(Q^3)$) der Hessematrix.

²⁶In Lösungsnähe wird die Zielfunktion durch das quadratische Modell immer genauer approximiert. Daher muß die Folge der Schrittweiten $\{\eta_N^I\}$ gegen 1 konvergieren.

Die beschriebenen Nachteile des Newton-Verfahrens werden durch die Klasse der Quasi-Newton-Verfahren (oder auch Secant-Verfahren) weitgehend vermieden. Diese basieren auf dem Newton-Verfahren mit Liniensuche Gl.(3.70), vermeiden jedoch die Berechnung und Invertierung der Hessematrix $\nabla_w^2 E(\mathbf{w})$. Zur Bestimmung der Suchrichtungen η_{QN}^I wird die inverse Hessematrix $\nabla_w^2 E(\mathbf{w})^{-1}$ durch eine symmetrische, positiv-definite Matrix $\widetilde{\mathbf{B}}^I$ approximiert

$$\mathbf{u}_{QN}^I = -\widetilde{\mathbf{B}}^I \cdot \nabla_w E(\mathbf{w}^I)^T \quad (3.71)$$

Diese Matrix wird nach jeder Iteration aktualisiert, so daß bei der Optimierung einer quadratischen Funktion nach Q Iterationen im Idealfall $\widetilde{\mathbf{B}}^0 = \nabla_w^2 E(\mathbf{w})^{-1}$ gilt. Als Startwert für diese Matrix wird i. allg. die Einheitsmatrix gewählt ($\widetilde{\mathbf{B}}^0 = \mathbf{I}$).

Damit stellt sich der prinzipielle Ablauf der Quasi-Newton-Verfahren folgendermaßen dar:

Modellalgorithmus für Quasi-Newton-Verfahren:

0 Initialisierung :

Startwerte \mathbf{w}^0 wählen und $\widetilde{\mathbf{B}}^0 = \mathbf{I}$ setzen

Iterationszähler I=0 setzen

1 Bestimmung der Suchrichtung :

$$\mathbf{u}_{QN}^I = -\widetilde{\mathbf{B}}^I \cdot \nabla_w E(\mathbf{w}^I)^T$$

2 Liniensuche :

$$\eta_{QN}^I = \min_{\eta} E(\mathbf{w}^I + \eta \cdot \mathbf{u}_{QN}^I)$$

3 Berechnung der neuen Näherungslösung :

$$\mathbf{w}^{I+1} = \mathbf{w}^I + \eta_{QN}^I \cdot \mathbf{u}_{QN}^I$$

4 Approximation der inversen Hessematrix aktualisieren :

Abhängig vom gewählten Verfahren wird

$\widetilde{\mathbf{B}}^{I+1}$ aus $\widetilde{\mathbf{B}}^I$ berechnet

Iterationszähler inkrementieren I:=I+1

5 Wenn Abbruchbedingung nicht erfüllt: Weiter bei 1

6 ENDE

Die QN-Verfahren haben gegenüber dem Newton-Verfahren die Vorteile, daß einerseits nur der Gradient und nicht die Hessematrix benötigt wird, die definitionsgemäß positiv-definite Matrix $\widetilde{\mathbf{B}}^I$ Abstiegsrichtungen garantiert und andererseits der Rechenaufwand nur proportional zu Q^2 und nicht proportional zu Q^3 steigt. Der Speicheraufwand steigt weiterhin mit Q^2 , da die Matrix $\widetilde{\mathbf{B}}^I$ gespeichert werden muß.

Es gibt verschiedene Möglichkeiten, eine Approximation der inversen Hessematrix mit den geforderten Eigenschaften aufzubauen. Von diesen gehört das *BFGS*-Verfahren von *B*royden, *F*letcher, *G*oldfab und *S*hanno [34, 79, 39] zu den derzeit besten und hat sich zur Optimierung allgemeiner nichtlinearer Probleme als besonders geeignet erwiesen [39, 79, 142].

Die Aktualisierungsvorschrift für die Approximation der inversen Hessematrix lautet für das *BFGS*-Verfahren

$$\begin{aligned} \widetilde{\mathbf{B}}_{BFGS}^{I+1} = \widetilde{\mathbf{B}}^I &+ \left(1 + \frac{(\Delta \mathbf{g}^I)^T \cdot \widetilde{\mathbf{B}}^I \cdot \Delta \mathbf{g}^I}{(\Delta \mathbf{g}^I)^T \cdot \Delta \mathbf{w}^I} \right) \frac{\Delta \mathbf{w}^I \cdot (\Delta \mathbf{w}^I)^T}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} \\ &- \frac{\Delta \mathbf{w}^I \cdot (\Delta \mathbf{g}^I)^T \cdot \widetilde{\mathbf{B}}^I + \widetilde{\mathbf{B}}^I \cdot \Delta \mathbf{g}^I \cdot (\Delta \mathbf{w}^I)^T}{(\Delta \mathbf{g}^I)^T \cdot \Delta \mathbf{w}^I} \end{aligned} \quad (3.72)$$

wobei die Abkürzungen $\Delta \mathbf{w}^I = \mathbf{w}^{I+1} - \mathbf{w}^I$ für die Änderung der Netzgewichte und $\Delta \mathbf{g}^I = \nabla_{\mathbf{w}} E(\mathbf{w}^{I+1}) - \nabla_{\mathbf{w}} E(\mathbf{w}^I)$ für die Änderung des Gradienten verwendet werden.

Ein wesentlicher Vorteil des *BFGS*-Verfahrens besteht darin, daß auch mit ungenauen Liniensuchen sehr gute Konvergenzeigenschaften erzielt werden. Diese Eigenschaft hat zur Folge, daß der Liniensuch-Algorithmus weniger Funktionswert- und Gradientenberechnungen benötigt. Da diese Berechnungen bei der Optimierung von *MLP*-Netzen mit einem hohen Aufwand verbunden sind, kann der Gesamtbedarf an Berechnungen trotz der aufwendigen Berechnung der Matrix $\widetilde{\mathbf{B}}_{BFGS}^{I+1}$ durch ungenaue Liniensuchen vermindert werden.

Von Nachteil ist jedoch, daß der Speicheraufwand mit $O(Q^2)$ steigt und dieses Verfahren damit für Netze mit einer großen Anzahl an Gewichten²⁷ schlecht geeignet ist. Mit dem Ziel, den Speicher- und Rechenaufwand auf $O(Q)$ zu reduzieren, wird daher in der Update-Vorschrift des *BFGS*-Verfahrens Gl.(3.72) die alte Näherung der Matrix $\widetilde{\mathbf{B}}^I$ durch die Einheitsmatrix \mathbf{I} ersetzt. Dies entspricht dem ersten Schritt des *BFGS*-Verfahrens. Wird die so berechnete Matrix $\widetilde{\mathbf{B}}_{LQN}^{I+1}$ in Gleichung (3.71) zur Bestimmung der Suchrichtung eingesetzt, erübrigt sich die Speicherung der Matrix $\widetilde{\mathbf{B}}^I$ für die nächste Iteration, und man erhält

$$\begin{aligned} \mathbf{u}_{LQN}^{I+1} = & - \nabla_{\mathbf{w}} E(\mathbf{w}^{I+1}) \\ & + \left(\frac{(\Delta \mathbf{g}^I)^T \cdot \nabla_{\mathbf{w}} E(\mathbf{w}^{I+1})}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} - \left(1 + \frac{(\Delta \mathbf{g}^I)^T \cdot \Delta \mathbf{g}^I}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} \right) \frac{(\Delta \mathbf{w}^I)^T \cdot \nabla_{\mathbf{w}} E(\mathbf{w}^{I+1})}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} \right) \cdot \Delta \mathbf{w}^I \\ & + \frac{(\Delta \mathbf{w}^I)^T \cdot \nabla_{\mathbf{w}} E(\mathbf{w}^{I+1})}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} \cdot \Delta \mathbf{g}^I \end{aligned} \quad (3.73)$$

Da der Update der Matrix $\widetilde{\mathbf{B}}_{LQN}^{I+1}$ ohne Verwendung der vorhergehenden Matrix $\widetilde{\mathbf{B}}_{LQN}^I$ durchgeführt wird, ist diese Aktualisierungsvorschrift gedächtnislos. Informationen über die Krümmung der Fehleroberfläche, die im Falle des *BFGS*-Verfahrens in der Approximation der inversen Hessematrix enthalten sind, werden nicht berücksichtigt. Entsprechend wird der durch Gl.(3.73) definierte Algorithmus als gedächtnisloses *BFGS*-Verfahren (*Memoryless Quasi Newton, Limited memory Quasi Newton*) bezeichnet, und im folgenden mit *LQN*-Verfahren abgekürzt.

²⁷unabhängig von der Topologie des Netzes

Ebenso wie bei den bereits vorgestellten *CG*-Verfahren wird das *LQN*-Verfahren mit dem negativen Gradienten im Startpunkt

$$\mathbf{u}^0 = -\nabla_{\mathbf{w}}E(\mathbf{w}^0) \quad (3.74)$$

initialisiert. Weiterhin wird ein zyklischer Restart der Suchrichtungen gemäß

$$\mathbf{u}^I = -\nabla_{\mathbf{w}}E(\mathbf{w}^I) \quad \text{für } I = Q, 2Q, \dots \quad (3.75)$$

nach jeweils Q Iterationen durchgeführt.

Das Verfahren ist eng verwandt mit den klassischen *CG*-Verfahren. Bei exakter Liniensuche reduziert sich Gl.(3.73) auf die Suchrichtungsbestimmung nach Polak-Ribière Gl.(3.63) bzw. Fletcher-Reeves Gl.(3.64). Es wird jedoch erwartet [6, 7, 79], daß sich das *LQN*-Verfahren im Gegensatz zu den klassischen *CG*-Verfahren auch bei ungenauen Liniensuchen als ähnlich robust erweist wie das *BFGS*-Verfahren.

Das *LQN*-Verfahren wurde bereits von Battiti [6] zur Optimierung von *MLP*-Netzen eingesetzt. In [7] wird dafür die Bezeichnung *OSS*-Verfahren (*One-Step Secant*) geprägt.

Von Shanno [131] werden zwei weitere auf dem *LQN*-Verfahren basierende Optimierungsverfahren vorgestellt, welche bei ungenauen Liniensuchen bessere Ergebnisse versprechen als die vorgestellten *CG*-Verfahren.

Die erste Modifikation besteht in der Anwendung der Skalierungsstrategie nach Oren-Spedicato²⁸ [97] auf die dem *LQN*-Verfahren zugrundeliegende Aktualisierungsvorschrift zur Approximation der inversen Hessematrix. Mit dem Skalierungsfaktor

$$\gamma^I = \frac{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I}{(\Delta \mathbf{g}^I)^T \cdot \Delta \mathbf{g}^I} \quad (3.76)$$

ergibt sich aus dieser Vorgehensweise nachfolgende Gleichung zur Berechnung der Suchrichtungen des *Scaled Conjugate Gradient*-Verfahrens (*SCG*-Verfahren):

$$\begin{aligned} \mathbf{u}_{SCG}^{I+1} = & - \gamma^I \nabla_{\mathbf{w}}E(\mathbf{w}^{I+1}) + \gamma^I \frac{(\Delta \mathbf{w}^I)^T \cdot \nabla_{\mathbf{w}}E(\mathbf{w}^{I+1})}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} \cdot \Delta \mathbf{g}^I \\ & + \left(\gamma^I \frac{(\Delta \mathbf{g}^I)^T \cdot \nabla_{\mathbf{w}}E(\mathbf{w}^{I+1})}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} - \left(1 + \gamma^I \frac{(\Delta \mathbf{g}^I)^T \cdot \Delta \mathbf{g}^I}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} \right) \frac{(\Delta \mathbf{w}^I)^T \cdot \nabla_{\mathbf{w}}E(\mathbf{w}^{I+1})}{(\Delta \mathbf{w}^I)^T \cdot \Delta \mathbf{g}^I} \right) \cdot \Delta \mathbf{w}^I \end{aligned} \quad (3.77)$$

Diese Gleichung entspricht für den Sonderfall $\gamma^I = 1$ der Gleichung (3.73) des *LQN*-Verfahrens. Die Initialisierung und der Restart der Suchrichtungen \mathbf{u}_{SCG}^I erfolgt in gleicher Weise wie bei dem *LQN*-Verfahren (Gl.(3.74) und Gl.(3.75)).

²⁸Oren und Spedicato erreichen durch Skalierung der Aktualisierungsvorschrift zur Berechnung der Approximation der Inversen Hessematrix $\tilde{\mathbf{B}}^I$ mit einem Faktor γ^I für *Quasi-Newton*-Verfahren, daß die Konditionszahl der Matrix $(\tilde{\mathbf{B}}^I)^{-1} \cdot \tilde{\mathbf{B}}^{I+1}$ minimiert wird.

Das zweite von Shanno [131], [Abschnitt IV] vorgeschlagene modifizierte *LQN*-Verfahren basiert ebenfalls auf Gl.(3.73) sowie Gl.(3.77). Es werden jedoch zusätzlich Ansätze aus einem Restart-Algorithmus von Beale [8] zur Modifikation der Such- und Restartrichtungen herangezogen. Außerdem wird, neben dem zyklischen Restart nach jeweils Q Iterationen, ein Restart durchgeführt, wenn Powells Kriterium [114]

$$|(\nabla_w E(\mathbf{w}^{I-1}))^T \cdot (\mathbf{g}^I)^T| \geq 0.2 \cdot (\nabla_w E(\mathbf{w}^I))^T \cdot \nabla_w E(\mathbf{w}^I) \quad (3.78)$$

erfüllt ist. Dieses Verfahren der konjugierten Gradienten nach Shanno hat sich bei den durchgeführten Untersuchungen [5] zwar als sehr unempfindlich gegenüber ungenauen Liniensuchen erwiesen, die Prädiktionsergebnisse waren jedoch i. allg. schlechter als diejenigen, die mit anderen Verfahren der konjugierten Gradienten erzielt wurden. Aus diesem Grunde wird auf dieses Verfahren nicht weiter eingegangen.

In Kapitel 5 werden entsprechend den obigen Ausführungen vier Varianten von *QN*-Verfahren untersucht:

Broyden-Fletcher-Goldfab-Shanno	(BFGS -Verfahren)
Limited-Quasi-Newton	(LQN -Verfahren)
Scaled-Conjugate-Gradient mit Restart	(SCR -Verfahren)
Scaled-Conjugate-Gradient ohne Restart	(SCG -Verfahren)

4 Ein neues Verfahren zur Optimierung mehrlagiger Perzeptron-Netze

Die im letzten Kapitel vorgestellten Gradientenverfahren sind für quadratische Optimierungsprobleme entwickelt worden. Sie werden jedoch in verstärktem Maße auch für die Optimierung von neuronalen Netzen eingesetzt [5, 6, 7, 55, 71, 72]. Aus der Literatur [5, 7, 71] ist bekannt, daß die Verfahren, die auf dem Ansatz der konjugierten Richtungen basieren, bereits eine deutlich bessere Konvergenzgeschwindigkeit besitzen, verglichen mit dem klassischen Backpropagation-Algorithmus. Weiterhin besteht bei Verwendung von Gradientenverfahren mit zunehmender Dimension des Optimierungsproblems die Gefahr, daß der Gradient betragsmäßig sehr klein wird. Dieser Fall tritt in der Umgebung von Sattelpunkten und lokalen Minima auf. Weist die Fehleroberfläche darüber hinaus einen sehr flachen Verlauf auf, ist die Konvergenzgeschwindigkeit der Verfahren, welche ohne eine eindimensionale Optimierung arbeiten, extrem niedrig. Bei den übrigen Verfahren wird der Einschachtelungsalgorithmus oder auch die anschließende eindimensionale Optimierung versagen. Wenn die implementierten Ausnahmeregeln in einem solchen Fall nicht mehr greifen, wird das Verfahren abgebrochen.

Zur Beseitigung dieser Nachteile wird im folgenden ein Verfahren vorgestellt, welches speziell auf die Optimierung von *MLP*-Netzen ausgerichtet ist und als **OLL**-Verfahren (**O**ptimierung **L**age für **L**age) bezeichnet wird.

Die wesentlichen Ansätze dieses Verfahrens sind:

- die Optimierung der Gewichte erfolgt getrennt nach Lagen
- innerhalb einer Lage werden die Gewichte abhängig voneinander optimiert
- die Optimierung der Gewichte \mathbf{W}_L der Ausgangslage ist für diesen Ansatz ein lineares Problem
- die Optimierung der Gewichte \mathbf{W}_l ; $l=2, \dots, L-1$ der verdeckten Lagen wird durch die Linearisierung der sigmoidalen Aktivierungsfunktionen Gl.(3.3) auf ein lineares Problem reduziert

Im folgenden werden die verschiedenen Abschnitte des *OLL*-Verfahrens für das in Bild 3.2 dargestellte mehrlagige Perzeptron-Netz abgeleitet.

4.1 Optimierung der Ausgangslage

Der Grundgedanke des neuen Verfahrens besteht darin, die Optimierung des mehrlagigen Perzeptron-Netzes getrennt nach Lagen durchzuführen. Es wird erneut die

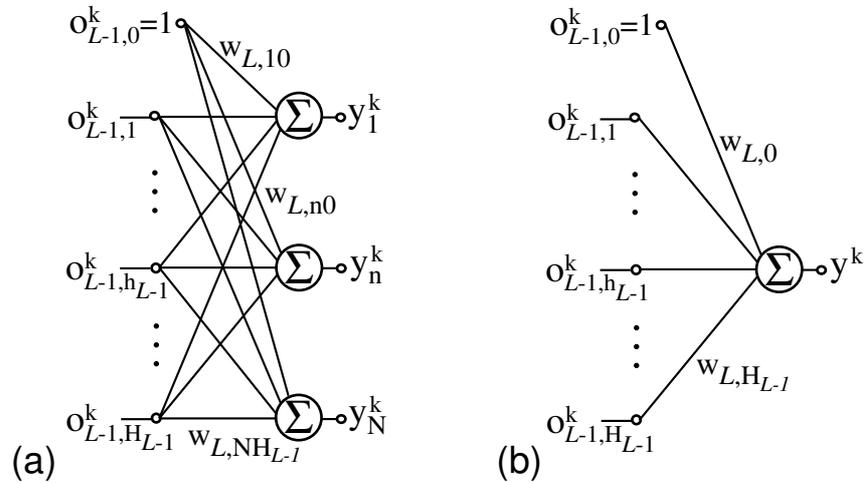


Abbildung 4.1: Die Ausgangslage eines mehrlagigen Perzeptron-Netzes

(a) mit vektoriellem Ausgang

(b) mit skalarem Ausgang

K Eingangs-Ausgangs-Paare umfassende Trainingsmenge $\{\mathbf{x}^k, \mathbf{d}^k\}_{k=1}^K$ verwendet. Unter dieser Annahme sind die Gewichte \mathbf{W}_l ; $l=2, \dots, L-1$ der verdeckten Lagen bezüglich der Optimierung der Gewichte \mathbf{W}_L der Ausgangslage Konstanten. Folglich stellen auch die Ausgangswerte der Knoten in der letzten verdeckten Lage $\mathbf{o}_{L-1}^k = (o_{L-1,0}^k, \dots, o_{L-1,h}^k, \dots, o_{L-1,H_{L-1}}^k)^T$ für $k = 1, \dots, K$ bezüglich dieses Ansatzes konstante Werte dar.

Zur besseren Übersicht ist die letzte Lage des in Bild 3.2 dargestellten mehrlagigen Perzeptron-Netzes sowohl für einen vektoriellen als auch für einen skalaren Netzausgang nochmals in Abbildung 4.1 wiedergegeben.

Wird nur ein skalarer Netzausgang verwendet, so entspricht die letzte Lage einem Adaptiven Linearen Element (ADALINE) [55]. Bei vektorieller Netzausgangslage stellt die letzte Lage ein lineares assoziatives Netz mit einem Bias-Eingang $o_{L-1,0}^k = 1$ dar, welches wiederum aus N ADALINE-Netzen besteht. Für diese Art von Netzen wurden spezielle Lernverfahren entwickelt, die auf eine Matrixinversion verzichten. Hier sei nur auf das Widdrow'sche Lernverfahren für ADALINE-Netze und die Hebb'sche Lernregel für das lineare assoziative Netz verwiesen [55].

Das *OLL*-Verfahren verzichtet jedoch auf ein iteratives Verfahren zur Optimierung der Gewichte der letzten Lage. Die in Abbildung 4.1 dargestellten Netze sind in ihrer Struktur einem skalaren bzw. vektoriellen linearen Prädiktor sehr ähnlich. Im Gegensatz zur linearen Prädiktion sind die hier betrachteten Eingangssignale $o_{L-1,1}^k, \dots, o_{L-1,H_{L-1}}^k$ gleichanteilsbehaftet. Dies ist auch der Grund für die Verwendung des Bias-Eingangs $o_{L-1,0}^k = 1$.

Die Gewichte \mathbf{W}_L werden mit dem linearen Prädiktionsansatz optimiert [83]. Das Gütekriterium ist der bereits in Gleichung (3.20) eingeführte mittlere quadratische Prädiktionsfehler.

Der Ausgangsvektor \mathbf{y}^k ergibt sich zu:

$$y_n^k = \sum_{h=0}^{H_{L-1}} w_{L,nh} o_{L-1,h}^k \quad \text{für } n = 1, \dots, N \quad (4.1)$$

oder in Matrixschreibweise

$$\mathbf{y}^k = \mathbf{W}_L^T \mathbf{o}_{L-1}^k \quad (4.2)$$

Mit den Gleichungen (3.20) und (4.1) stellt sich die Optimierung der Ausgangslage folgendermaßen dar:

$$E(\mathbf{W}_L^{opt}) = \min_{\mathbf{W}_L} E(\mathbf{W}_L) \quad \text{mit} \quad E(\mathbf{W}_L) = \frac{1}{K} \sum_{k=1}^K \frac{1}{2} (\mathbf{d}^k - \mathbf{W}_L^T \mathbf{o}_{L-1}^k)^2 \quad (4.3)$$

Zur Bestimmung der optimalen Gewichte \mathbf{W}_L^{opt} werden die partiellen Ableitungen der Kostenfunktion bezüglich $w_{L,nh}$ berechnet und zu Null gesetzt:

$$\frac{\partial E(w_{L,nh})}{\partial w_{L,nh}} = \frac{1}{K} \sum_{k=1}^K \frac{1}{2} \sum_{j=1}^N \frac{\partial}{\partial w_{L,nh}} (d_j^k - \sum_{i=0}^{H_{L-1}} w_{L,ji} o_{L-1,i}^k)^2 = 0 \quad (4.4)$$

Die weitere Berechnung führt auf ein System von $(H_{L-1} + 1) \cdot (N)$ linearen Gleichungen

$$\sum_{i=0}^{H_{L-1}} w_{L,ni}^{opt} \frac{1}{K} \sum_{k=1}^K o_{L-1,i}^k o_{L-1,h}^k = \frac{1}{K} \sum_{k=1}^K d_n^k o_{L-1,h}^k \quad \text{für } \begin{array}{l} n = 1, \dots, N \\ h = 1, \dots, H_{L-1} \end{array} \quad (4.5)$$

welches in N unabhängige Gleichungssysteme mit jeweils $(H_{L-1} + 1)$ Gleichungen zerlegt werden kann.

Werden die über der Trainingsmenge gemittelten Werte

$$\mathbf{A} = \frac{1}{K} \sum_{k=1}^K \mathbf{o}_{L-1}^k (\mathbf{o}_{L-1}^k)^T \quad \mathbf{B} = \frac{1}{K} \sum_{k=1}^K \mathbf{o}_{L-1}^k (\mathbf{d}^k)^T \quad (4.6)$$

in Matrixschreibweise dargestellt, ergibt sich für Gleichung (4.5)

$$\mathbf{A}(\mathbf{W}_L^{opt})^T = \mathbf{B} \quad (4.7)$$

Hierin stellt die $(H_{L-1} + 1) \cdot (H_{L-1} + 1)$ -Matrix \mathbf{A} die symmetrische Korrelationsmatrix der Ausgangsvektoren \mathbf{o}_{L-1}^k der letzten verdeckten Lage dar, die i. allg. positiv-definit ist. Zur effizienten Berechnung der Inversen \mathbf{A}^{-1} können diese Eigenschaften ausgenutzt werden [135].

Die optimalen Gewichte der N ADALINE-Netze ergeben sich aus den folgenden Beziehungen

$$\mathbf{w}_{L,n}^{opt} = \mathbf{A}^{-1} \mathbf{b}_n \quad \text{für } n = 1, \dots, N \quad (4.8)$$

In dieser Gleichung bezeichnen der Vektor $\mathbf{w}_{L,n}^{opt}$ die n -te Zeile der Gewichtsmatrix \mathbf{W}_L^{opt} und der Vektor \mathbf{b}_n die n -te Spalte der Matrix \mathbf{B} . Bei skalarem Netzausgang ist in Gleichung (4.8) $N = 1$ zu setzen.

Es kann gezeigt werden [55], daß die Optimierung der einzelnen ADALINE-Netze quadratische Optimierungsprobleme mit einer entsprechend parabolischen Fehleroberfläche darstellen. Diese Fehleroberflächen besitzen im nicht-degenerierten Fall nur ein globales Minimum, wovon allgemein ausgegangen werden kann.

An dieser Stelle sei nochmals darauf hingewiesen, daß die Matrix \mathbf{W}_L^{opt} die optimalen Gewichte der Ausgangslage bezüglich der als konstant angenommenen Gewichte der verdeckten Lagen darstellt. Zur Bestimmung der die Gleichung (3.20) minimierenden Gewichte \mathbf{w}^* ist ein iteratives Verfahren notwendig, welches in den nächsten Abschnitten abgeleitet wird.

4.2 Optimierung der verdeckten Lagen eines mehrlagigen Perzeptron-Netzes

Wie bereits in Abschnitt 4.1 gezeigt wurde, läßt sich die Optimierung der Ausgangslage als lineares Problem formulieren.

Im folgenden wird, basierend auf einer Linearisierung der sigmoidalen Aktivierungsfunktionen der Neuronen in den verdeckten Lagen $l \in \{2, \dots, L-1\}$, die Optimierung der Gewichte \mathbf{W}_l dieser Lagen auf ein lineares Problem reduziert. Auf diese Weise ist es möglich, analog zur Vorgehensweise bei der Optimierung der Ausgangslage, ein den Gleichungen (4.7) und (4.8) entsprechendes lineares Gleichungssystem für die Bestimmung der optimalen Gewichte in den verdeckten Lagen aufzustellen.

Da die Optimierung des Netzes getrennt nach Lagen erfolgt und durch die Linearisierung der Aktivierungsfunktionen ein zusätzlicher Fehler unvermeidbar ist, kann diese Vorgehensweise nicht nach einmaliger Optimierung der Gewichte aller Lagen zu den die Gütefunktion Gl.(3.20) minimierenden Gewichten \mathbf{w}^* führen.

Diese Aufgabe kann jedoch von einem iterativen Algorithmus gelöst werden, welcher die Gewichtsmatrizen \mathbf{W}_l der einzelnen Lagen wiederholt in der Reihenfolge $\mathbf{W}_L, \dots, \mathbf{W}_1, \dots, \mathbf{W}_2$ optimiert.

Auf eine Indizierung der jeweils vorliegenden Iteration soll hier aus Gründen der Übersichtlichkeit verzichtet werden. Für die aktuelle Iteration werden die in der vorherigen Iteration ermittelten Gewichte mit dem Index *alt* und die bei der jeweiligen Iteration zu

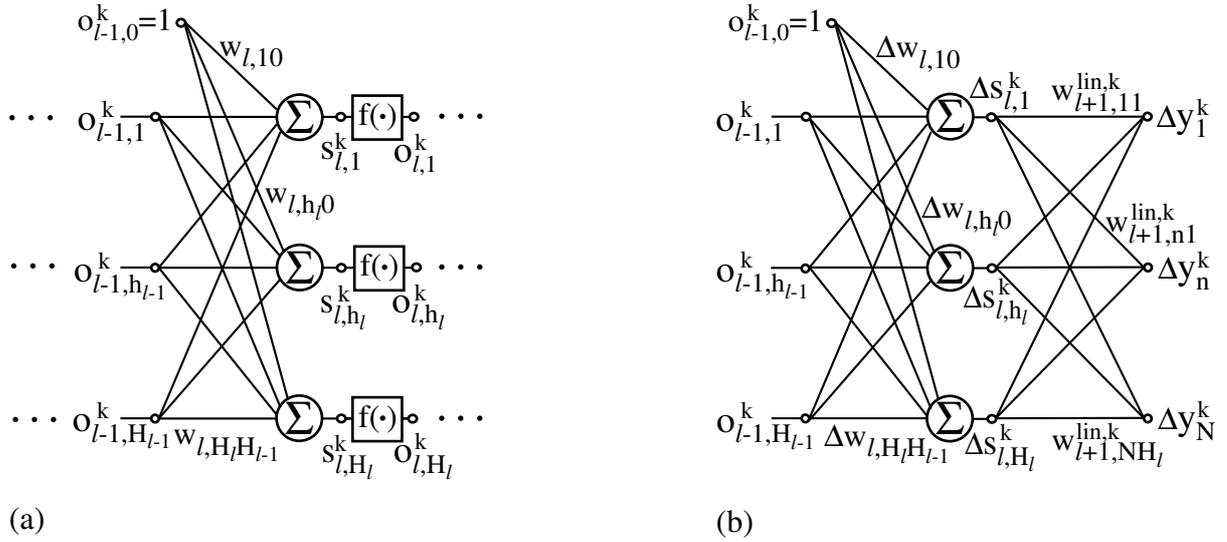


Abbildung 4.2: (a) Die l-te Lage eines mehrlagigen Perzeptron-Netzes

(b) Ein ab der l-ten Lage linearisiertes mehrlagiges Perzeptron-Netz

berechnenden Gewichte mit dem Index *neu* gekennzeichnet. Zum besseren Verständnis ist die l-te Lage des *MLP*-Netzes nochmals in Abbildung (4.2a) dargestellt.

4.2.1 Linearisierung der verdeckten Lagen eines mehrlagigen Perzeptron-Netzes

Ausgehend von den aktuellen Gewichten $w_{l,hm}^{\text{alt}}$ der verdeckten Lage $l \in \{2, \dots, L-1\}$ hat eine Optimierung die Gewichtsänderung

$$\Delta W_l = \begin{pmatrix} \Delta w_{l,10} & \Delta w_{l,11} & \cdots & \Delta w_{l,1H_{l-1}} \\ \Delta w_{l,20} & \Delta w_{l,21} & \cdots & \Delta w_{l,2H_{l-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \Delta w_{l,H_l0} & \Delta w_{l,H_l1} & \cdots & \Delta w_{l,H_lH_{l-1}} \end{pmatrix} \quad (4.9)$$

zur Folge. Damit ergibt sich für die optimierten Gewichte dieser Lage

$$w_{l,hm}^{\text{neu}} = w_{l,hm}^{\text{alt}} + \Delta w_{l,hm} \quad \text{für} \quad \begin{array}{l} m = 0, \dots, H_{l-1} \\ h = 1, \dots, H_l \end{array} \quad (4.10)$$

Die Gewichte \mathbf{W}_i mit $i \in \{2, \dots, L | i \neq l\}$ der übrigen Lagen werden wiederum als konstant angenommen.

Diese Gewichtsänderung gemäß Gl.(4.10) bewirkt zwangsläufig eine Änderung der Eingangssignale der Neuronen in der betrachteten Lage. Für das k-te Trainingsmuster ergibt

sich somit

$$\begin{aligned}
s_{l,h}^{neu,k} &= \sum_{m=0}^{H_{l-1}} w_{l,hm}^{neu} o_{l-1,m}^k \\
&= \sum_{m=0}^{H_{l-1}} w_{l,hm}^{old} o_{l-1,m}^k + \sum_{m=0}^{H_{l-1}} \Delta w_{l,hm} o_{l-1,m}^k \quad \text{für } h = 1, \dots, H_l \\
&= s_{l,h}^{alt,k} + \Delta s_{l,h}^k
\end{aligned} \tag{4.11}$$

Bei der Optimierung der ersten verdeckten Lage ($l=2$) ist in dieser sowie in den folgenden Gleichungen zu beachten, daß die Ausgangswerte \mathbf{o}_{l-1}^k der Knoten der vorhergehenden Lage gerade den Eingangsvektoren \mathbf{x}^k des Netzes entsprechen.

Eine Gewichtsänderung $\Delta w_{l,hm}$ hat gemäß Gl.(4.11) eine Änderung $\Delta s_{l,h}^k$ der Eingangswerte der Neuronen in der l -ten Lage zur Folge. Zur Berechnung der Auswirkungen auf den Netzausgang $\mathbf{y}^{neu,k}$ muß zunächst der Einfluß der geänderten Eingangswerte $s_{l,h}^{neu,k}$ auf die Ausgangswerte $\mathbf{o}_l^{neu,k}$ der sigmoidalen Aktivierungsfunktionen ermittelt werden.

Die angestrebte lineare Abhängigkeit einer Gewichtsänderung auf den Netzausgang kann durch lineare Approximation der sigmoidalen Aktivierungsfunktionen basierend auf deren Taylorreihendarstellung erzielt werden.

Die Taylorreihenentwicklung der Aktivierungsfunktion $f(s)$ im Punkt $s_{l,h}^{alt,k}$ ist gegeben durch

$$\begin{aligned}
f(s_{l,h}^{neu,k}) &= f(s_{l,h}^{alt,k} + \Delta s_{l,h}^k) \\
&= f(s_{l,h}^{alt,k}) + \frac{\partial f(s_{l,h}^{alt,k})}{\partial s_{l,h}^{alt,k}} \Delta s_{l,h}^k + \frac{\partial^2 f(s_{l,h}^{alt,k})}{2 \partial^2 s_{l,h}^{alt,k}} (\Delta s_{l,h}^k)^2 + R_2(\Delta s_{l,h}^k)
\end{aligned} \tag{4.12}$$

Hierin stellt $R_2(\Delta s_{l,h}^k)$ alle Terme höherer Ordnung dar und wird als Restglied zweiter Ordnung bezeichnet.

Für kleine Änderungen $\Delta s_{l,h}^k$ werden im folgenden die nichtlinearen Aktivierungsfunktionen gemäß Gl.(4.12) durch eine Taylorreihe erster Ordnung approximiert.²⁹ Die Ausgangswerte der Neuronen werden durch

$$o_{l,h}^{neu,k} = o_{l,h}^{alt,k} + \Delta o_{l,h}^k \approx \begin{cases} f(s_{l,h}^{alt,k}) + f'(s_{l,h}^{alt,k}) \Delta s_{l,h}^k & \text{für } h = 1, \dots, H_l \\ o_{l,h}^{alt,k} \equiv 1 & \text{für } h = 0 \end{cases} \tag{4.13}$$

angenähert. Damit ergeben sich die Eingangswerte der Neuronen in der $(l+1)$ -ten Lage zu

$$\begin{aligned}
s_{l+1,h}^{neu,k} &= \sum_{m=0}^{H_l} w_{l+1,hm} o_{l,m}^{neu,k} \\
&= \sum_{m=0}^{H_l} w_{l+1,hm} o_{l,m}^{alt,k} + \sum_{m=1}^{H_l} w_{l+1,hm} f'(s_{l,m}^{alt,k}) \Delta s_{l,m}^k \quad \text{für } h = 1, \dots, H_{l+1} \\
&= s_{l+1,h}^{alt,k} + \Delta s_{l+1,h}^k
\end{aligned} \tag{4.14}$$

²⁹Zur Begrenzung des Linearisierungsfehlers wird später (vergl. Seite 62) ein Strafterm eingeführt und eine modifizierte Gütefunktion abgeleitet.

Bei der Optimierung der letzten verdeckten Lage ($l=L-1$) gilt $s_{l+1,h}^{neu,k} = y_n^{neu,k}$.

Mit der abkürzenden Schreibweise

$$\tilde{w}_{l+1,hm}^k = w_{l+1,hm} f'(s_{l,m}^{alt,k}) \quad \text{für} \quad \begin{array}{l} m = 1, \dots, H_l \\ h = 1, \dots, H_{l+1} \end{array} \quad (4.15)$$

erhält man das folgende Ergebnis

$$\Delta s_{l+1,h}^k = \sum_{m=1}^{H_l} \tilde{w}_{l+1,hm}^k \Delta s_{l,m}^k \quad (4.16)$$

oder in Matrixschreibweise

$$\Delta \mathbf{s}_{l+1}^k = \tilde{\mathbf{W}}_{l+1}^k \Delta \mathbf{s}_l^k \quad (4.17)$$

Wird diese Vorgehensweise für alle weiteren verdeckten Lagen bis zur Ausgangslage durchgeführt, ergibt sich der geforderte lineare Zusammenhang zwischen einer Änderung $\Delta \mathbf{s}_l^k$ der Eingangswerte der Knoten in der l -ten verdeckten Lage und der Änderung des Netzausgangs $\Delta \mathbf{y}^k$ zu

$$\Delta \mathbf{y}^k = \prod_{i=L}^{l+1} \tilde{\mathbf{W}}_i^k \Delta \mathbf{s}_l^k = \mathbf{W}_{l+1}^{lin,k} \Delta \mathbf{s}_l^k \quad (4.18)$$

Den Einfluß einer Gewichtsänderung in der l -ten Lage auf den Netzausgang erhält man mit Gleichung (4.11)

$$\Delta \mathbf{y}^k = \mathbf{W}_{l+1}^{lin,k} \Delta \mathbf{W}_l \mathbf{o}_{l-1}^k \quad (4.19)$$

Diese Gleichung beschreibt das in Abbildung (4.2b) dargestellte, ab der l -ten Lage linearisierte Netz. Zu bemerken bleibt, daß die Gewichte $\mathbf{W}_{l+1}^{lin,k}$ des linearisierten Netzes vom aktuellen Trainingsmuster $\{\mathbf{x}^k, \mathbf{d}^k\}$ abhängen, da die Linearisierung vom jeweiligen Arbeitspunkt $s_{l,h}^{alt,k}$ abhängig ist.

4.2.2 Die modifizierte Gütefunktion

Das Ziel einer Gewichtsänderung $\Delta \mathbf{W}_l$ ist es, den mit den aktuellen Gewichten des Netzes erzielten Prädiktionsfehler

$$E(\mathbf{w}^{alt}) = \frac{1}{K} \sum_{k=1}^K \frac{1}{2} (\mathbf{d}^k - \mathbf{y}^{alt,k})^2 = \frac{1}{K} \sum_{k=1}^K \frac{1}{2} (\mathbf{e}^{alt,k})^2 \quad (4.20)$$

weiter zu verringern.

Wird versucht, die Gewichtsänderungen $\Delta \mathbf{W}_l$ so zu wählen, daß der Differenzvektor $\Delta \mathbf{y}^k$ gemäß Gleichung (4.19) gerade dem aktuellen Fehlervektor entspricht, ergibt sich als zu minimierende Zielfunktion für das in Abbildung 4.2.b dargestellte linearisierte Netz

$$E_l^{lin}(\Delta \mathbf{W}_l) = \frac{1}{K} \sum_{k=1}^K \frac{1}{2} (\mathbf{e}^{alt,k} - \Delta \mathbf{y}^k)^2 = \frac{1}{K} \sum_{k=1}^K \frac{1}{2} \sum_{n=1}^N (\mathbf{e}_n^{alt,k} - \Delta y_n^k)^2 \quad (4.21)$$

Auf Grund dieser Überlegungen wird die Optimierung der verdeckten Lage und damit des linearisierten Netzes mit der modifizierten Trainingsmenge $\{\mathbf{x}^k, \mathbf{e}^{alt,k}\}$ durchgeführt. Die mit Hilfe von Gl.(4.21) zu ermittelnden optimalen Gewichtsänderungen $\Delta \mathbf{W}_l^{opt}$ werden allerdings nur zu einer weiteren Verbesserung der ursprünglichen Zielfunktion Gl.(3.20) führen, wenn der durch Gl.(4.14) verursachte Linearisierungsfehler nicht zu groß ist. Gleichung (4.19) und somit das linearisierte Netz (Abb.4.2b) stellt nur für hinreichend kleine Linearisierungsfehler eine ausreichend gute Näherung für den Einfluß einer Gewichtsänderung $\Delta \mathbf{W}_l$ auf den Netzausgang $\mathbf{y}^{neu,k}$ dar. Die aus der Optimierung von Gl.(4.21) hervorgehenden Gewichtsänderungen sind deshalb zu begrenzen.

Zur Linearisierung des Netzes wurden die nichtlinearen Aktivierungsfunktionen durch eine Taylorreihe erster Ordnung angenähert. Der Approximationsfehler setzt sich demnach aus dem quadratischen Term und dem Restglied zweiter Ordnung zusammen. Im folgenden wird der quadratische Term der Taylorreihe gemäß Gl.(4.13)

$$\epsilon_{l,h}^k = \frac{1}{2} f''(s_{l,h}^{alt,k}) (\Delta s_{l,h}^k)^2 \quad \text{für } h = 1, \dots, H_l \quad (4.22)$$

zur Abschätzung des durch die Linearisierung (Gl.(4.14)) verursachten Approximationsfehlers herangezogen.

Eine Begrenzung dieses Fehlers auf den Maximalwert ϵ_{max} ist durch Lösung des folgenden restringierten Optimierungsproblems [79, 39]

$$\begin{aligned} E_l^{lin}(\Delta \mathbf{W}_l^{opt}) &= \min_{\Delta \mathbf{W}_l} E_l^{lin}(\Delta \mathbf{W}_l) \\ \text{mit} \quad \epsilon_{l,h}^k &\leq \epsilon_{max} \quad \text{für} \quad \begin{array}{l} h = 1, \dots, H_l \\ k = 1, \dots, K \end{array} \end{aligned} \quad (4.23)$$

prinzipiell möglich.

Die Lösung dieser Optimierung unter Nebenbedingungen ist jedoch auf Grund der sehr hohen Anzahl³⁰ von $H_l \cdot K$ Nebenbedingungen praktisch nicht durchführbar. Daher wird im folgenden die alternative Möglichkeit der Begrenzung des Linearisierungsfehlers durch die Addition eines Strafterms [79] zur Gütefunktion Gl.(4.21) des linearisierten Netzes betrachtet.

Aufgabe des zu formulierenden Strafterms $E_l^{pen}(\Delta \mathbf{W}_l)$ ist, dafür Rechnung zu tragen, daß das linearisierte Netz eine im Rahmen der Optimierung zulässige Beschreibung der Abhängigkeit des Netzausgangs von den Gewichtsänderungen $\Delta \mathbf{W}_l$ darstellt. Daher muß $E_l^{pen}(\Delta \mathbf{W}_l)$ die Wirkung der Linearisierungsfehler Gl.(4.22) auf den Netzausgang erfassen und gegebenenfalls begrenzen. Dieses kann durch Multiplikation der $\epsilon_{l,h}^k$ mit den in Abschnitt 4.2.1 abgeleiteten linearisierten Ausgangsgewichten $\mathbf{W}_{l+1}^{lin,k}$ erzielt werden. Dabei ist jedoch zu beachten, daß in der Berechnung der linearisierten Ausgangsgewichte

³⁰Bereits für ein recht kleines Problem mit einem Umfang von 1 000 Trainingsmustern und einer Anzahl von nur 50 Gewichten in einer verdeckten Lage ergeben sich 50 000 Nebenbedingungen.

$w_{l+1,nh}^{lin,k}$ entsprechend den Gleichungen (4.15) und (4.18) bereits die Terme $f'(s_{l,h}^{alt,k})$ als Faktoren enthalten sind. Da der Strafterm jedoch ausschließlich den Einfluß des quadratischen Terms der Taylorreihe auf den Netzausgang annähern soll, sind die linearisierten Ausgangsgewichte durch $f'(s_{l,h}^{alt,k})$ zu dividieren. Diesen Überlegungen entsprechend wird der Strafterm wie folgt definiert:

$$E_l^{pen}(\Delta \mathbf{W}_l) = \frac{1}{K} \sum_{k=1}^K \frac{1}{N} \sum_{n=1}^N \frac{1}{H_l} \sum_{h=1}^{H_l} \left| \epsilon_{l,h}^k \frac{w_{l+1,nh}^{lin,k}}{f'(s_{l,h}^{alt,k})} \right| \quad (4.24)$$

Dieser Strafterm bewertet die Qualität der linearen Approximation durch Mittelung der Absolutfehler über die Netzaugänge sowie die Trainingsmenge. Auf ein quadratisches Fehlermaß wurde bewußt verzichtet, um ein lineares Gleichungssystem analog zu Gl.(4.8) zur Bestimmung der optimalen Gewichtsänderungen $\Delta \mathbf{W}_l$ zu erhalten.

Die modifizierte Gütefunktion für die Optimierung der l-ten verdeckten Lage basierend auf dem ab der l-ten Lage linearisierten Netzes lautet dann

$$E_l^{hid}(\Delta \mathbf{W}_l) = E_l^{lin}(\Delta \mathbf{W}_l) + \mu_l E_l^{pen}(\Delta \mathbf{W}_l) \quad (4.25)$$

mit $E_l^{lin}(\Delta \mathbf{W}_l)$ und $E_l^{pen}(\Delta \mathbf{W}_l)$ entsprechend Gl.(4.21) und (4.24).

Der Gewichtungsfaktor μ_l bestimmt den Einfluß des Strafterms gegenüber der Gütefunktion des linearisierten Netzes $E_l^{lin}(\Delta \mathbf{W}_l)$. Eine günstige Wahl der Faktoren $\boldsymbol{\mu} = (\mu_2, \dots, \mu_l, \dots, \mu_{L-1})^T$ ist von entscheidender Bedeutung für die Konvergenzgeschwindigkeit des *OLL*-Verfahrens.

Ein zu kleiner Wert von μ_l ermöglicht auf Grund der fehlenden Begrenzung durch den Strafterm große Gewichtsänderungen $\Delta \mathbf{W}_l$, welche erhebliche Änderungen $\Delta \mathbf{s}_l^k$ der Eingangswerte der linearisierten Aktivierungsfunktionen zur Folge haben können. Da die Linearisierung in einem solchen Fall nicht mehr gültig ist, wird zwar die Gütefunktion Gl.(4.25) des linearisierten Netzes minimiert, der Prädiktionsgewinn des ursprünglichen *MLP*-Netzes wird jedoch i. allg. abnehmen.

Andererseits bewirkt ein zu großer Wert des Gewichtungsfaktors μ_l , daß nur sehr kleine Gewichtsänderungen erzielt werden können. Die Linearisierung ist in diesem Fall zwar gültig, und der Prädiktionsgewinn des *MLP*-Netzes wird i. allg. bei jeder Iteration ansteigen, jedoch ist die erzielbare Konvergenzgeschwindigkeit dann sehr gering.

Die Bestimmung und Adaption eines günstigen Wertes für μ_l muß, ähnlich dem Bold-Driver-Verfahren, mit Hilfe einer Heuristik durchgeführt werden, da dieser nicht aus dem Optimierungsansatz selbst abgeleitet werden kann. Der im Rahmen des *OLL*-Verfahrens implementierte Algorithmus zur Adaption der μ_l wird auf Seite 67 erläutert. Für die folgenden Betrachtungen wird der Gewichtungsfaktor μ_l als konstant angenommen.

4.2.3 Minimierung der modifizierten Gütefunktion

Die Minimierung der Gütefunktion der verdeckten Lagen wird analog zur Optimierung der Ausgangslage durchgeführt. Zunächst werden die partiellen Ableitungen von $E_l^{lin}(\Delta \mathbf{W}_l)$ gemäß Gleichung (4.21) bezüglich $\Delta w_{l,hm}$ berechnet.

$$\frac{\partial E_l^{lin}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = \frac{1}{K} \sum_{k=1}^K \sum_{n=1}^N (\Delta y_n^k - e_n^{alt,k}) \frac{\partial \Delta y_n^k}{\partial w_{l,hm}} \quad (4.26)$$

Die Ersetzung von Δy_n^k mit Hilfe der Gleichungen (4.18) und (4.11) führt auf

$$\frac{\partial E_l^{lin}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = \frac{1}{K} \sum_{k=1}^K \sum_{n=1}^N \left[\left(\sum_{j=1}^{H_l} w_{l+1,nj}^{lin,k} \sum_{i=0}^{H_{l-1}} \Delta w_{l,ji} o_{l-1,i}^k \right) - e_n^{alt,k} \right] \cdot w_{l+1,nh}^{lin,k} o_{l-1,m}^k \quad (4.27)$$

Wird die Summationsreihenfolge geändert, erhält man

$$\begin{aligned} \frac{\partial E_l^{lin}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} &= \frac{1}{K} \sum_{i=0}^{H_{l-1}} \sum_{j=1}^{H_l} \Delta w_{l,ji} \sum_{k=1}^K \sum_{n=1}^N w_{l+1,nh}^{lin,k} o_{l-1,m}^k w_{l+1,nj}^{lin,k} o_{l-1,i}^k \\ &\quad - \frac{1}{K} \sum_{k=1}^K \sum_{n=1}^N w_{l+1,nh}^{lin,k} o_{l-1,m}^k e_n^{alt,k} \end{aligned} \quad (4.28)$$

Mit den Substitutionen

$$\begin{aligned} a_{hm,ji} &= \sum_{k=1}^K \sum_{n=1}^N w_{l+1,nh}^{lin,k} o_{l-1,m}^k w_{l+1,nj}^{lin,k} o_{l-1,i}^k \\ \tilde{b}_{hm} &= \sum_{k=1}^K \sum_{n=1}^N w_{l+1,nh}^{lin,k} o_{l-1,m}^k e_n^{alt,k} \end{aligned} \quad (4.29)$$

ergeben sich schließlich die partiellen Ableitungen von $E_l^{lin}(\Delta \mathbf{W}_l)$ zu

$$\frac{\partial E_l^{lin}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = \frac{1}{K} \left[\sum_{i=0}^{H_{l-1}} \sum_{j=1}^{H_l} \Delta w_{l,ji} \tilde{a}_{hm,ji} - \tilde{b}_{hm} \right] \quad \text{für } \begin{array}{l} m = 0, \dots, H_{l-1} \\ h = 1, \dots, H_l \end{array} \quad (4.30)$$

Für die partiellen Ableitungen des Strafterms $E_l^{pen}(\Delta \mathbf{W}_l)$ gemäß Gl.(4.24) ergibt sich mit Gl.(4.22) zunächst

$$\frac{\partial E_l^{pen}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = \frac{1}{K} \sum_{k=1}^K \frac{1}{N} \sum_{n=1}^N \frac{1}{H_l} \sum_{j=1}^{H_l} \frac{1}{2} \frac{\partial}{\partial w_{l,hm}} \left[\left| f''(s_{l,j}^{alt,k}) \frac{w_{l+1,nj}^{lin,k}}{f'(s_{l,j}^{alt,k})} \right| (\Delta s_{l,j}^k)^2 \right] \quad (4.31)$$

Nach Ersetzen der $\Delta s_{l,j}^k$ mit Hilfe von Gl.(4.11) und nach der Differentiation erhält man

$$\frac{\partial E_l^{pen}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = \frac{1}{KH_lN} \sum_{k=1}^K \sum_{n=1}^N \sum_{i=0}^{H_{l-1}} \Delta w_{l,hi}^k o_{l-1,i}^k o_{l-1,m}^k \left| f''(s_{l,h}^{alt,k}) \frac{w_{l+1,nh}^{lin,k}}{f'(s_{l,h}^{alt,k})} \right| \quad (4.32)$$

Nach Vertauschen der Summationsreihenfolge ergibt sich

$$\frac{\partial E_l^{pen}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = \frac{1}{KH_l N} \sum_{i=0}^{H_{l-1}} \Delta w_{l,hi}^k \sum_{k=1}^K \sum_{n=1}^N \left| f''(s_{l,h}^{alt,k}) \frac{w_{l+1,nh}^{lin,k}}{f'(s_{l,h}^{alt,k})} \right| o_{l-1,m}^k o_{l-1,i}^k \quad (4.33)$$

Eine erneute Substitution

$$c_{hm,i} := \frac{1}{N} \sum_{k=1}^K \sum_{n=1}^N \left| f''(s_{l,h}^{alt,k}) \frac{w_{l+1,nh}^{lin,k}}{f'(s_{l,h}^{alt,k})} \right| o_{l-1,m}^k o_{l-1,i}^k \quad (4.34)$$

vereinfacht die partiellen Ableitungen des Strafterms zu

$$\frac{\partial E_l^{pen}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = \frac{1}{KH_l} \sum_{i=0}^{H_{l-1}} \Delta w_{l,hi}^k c_{hm,i} \quad \text{für} \quad \begin{array}{l} m = 0, \dots, H_{l-1} \\ h = 1, \dots, H_l \end{array} \quad (4.35)$$

Zur Berechnung der Gewichtsänderungen für die Optimierung der l-ten verdeckten Lage werden die partiellen Ableitungen der modifizierten Gütefunktion Gl.(4.25) zu Null gesetzt. Mit den Gleichungen (4.30) und (4.35) folgt

$$\frac{\partial E_l^{hid}(\Delta w_{l,hm})}{\partial \Delta w_{l,hm}} = 0 = \sum_{i=0}^{H_{l-1}} \sum_{j=1}^{H_l} a_{hm,ji} \Delta w_{l,ji} + \frac{\mu_l}{H_l} \sum_{i=0}^{H_{l-1}} c_{hm,i} \Delta w_{l,hi}^k - \tilde{b}_{hm} \quad (4.36)$$

für $\begin{array}{l} m = 0, \dots, H_{l-1} \\ h = 1, \dots, H_l \end{array}$

Aus Gleichung (4.36) ergibt sich mit

$$\tilde{a}_{hm,ji} := \begin{cases} a_{hm,ji} & \text{für } j \neq h \\ a_{hm,ji} + \frac{\mu_l}{H_l} c_{hm,i} & \text{für } j = h \end{cases} \quad (4.37)$$

analog zur Optimierung der Ausgangslage ein aus $(H_{l-1}+1) \cdot (H_l)$ Gleichungen bestehendes lineares Gleichungssystem

$$\begin{aligned} \tilde{a}_{10,10} \Delta w_{l,10}^{opt} + \dots + \tilde{a}_{10,ji} \Delta w_{l,ji}^{opt} + \dots + \tilde{a}_{10,H_l H_{l-1}} \Delta w_{l,H_l H_{l-1}}^{opt} &= \tilde{b}_{10} \\ \vdots & \\ \tilde{a}_{hm,10} \Delta w_{l,10}^{opt} + \dots + \tilde{a}_{hm,ji} \Delta w_{l,ji}^{opt} + \dots + \tilde{a}_{hm,H_l H_{l-1}} \Delta w_{l,H_l H_{l-1}}^{opt} &= \tilde{b}_{hm} \\ \vdots & \\ \tilde{a}_{H_l H_{l-1},10} \Delta w_{l,10}^{opt} + \dots + \tilde{a}_{H_l H_{l-1},ji} \Delta w_{l,ji}^{opt} + \dots + \tilde{a}_{H_l H_{l-1},H_l H_{l-1}} \Delta w_{l,H_l H_{l-1}}^{opt} &= \tilde{b}_{H_l H_{l-1}} \end{aligned} \quad (4.38)$$

zur Berechnung der gesuchten optimalen Gewichtsänderungen $\Delta w_{l,hm}^{opt}$.

Wird die Matrix der Gewichtsänderungen $\Delta \mathbf{W}_l^{opt}$ als $(H_{l-1} + 1) \cdot (H_l)$ -dimensionaler Vektor $\Delta \mathbf{w}_l^{opt} = (\Delta w_{l,10}^{opt}, \dots, \Delta w_{l,1H_{l-1}}^{opt}, \dots, \Delta w_{l,H_l 0}^{opt}, \dots, \Delta w_{l,H_l H_{l-1}}^{opt})$ aufgefaßt, so läßt sich das Gleichungssystem (4.38) in Matrixschreibweise formulieren

$$\tilde{\mathbf{A}} \cdot \Delta \mathbf{w}_l^{opt} = \tilde{\mathbf{b}} \quad (4.39)$$

Die Lösung dieses Gleichungssystems liefert die gesuchten optimalen Gewichtsänderungen $\Delta \mathbf{w}_l^{opt}$ zur Optimierung der Gewichte der l-ten verdeckten Lage auf Basis des hier vorgestellten *OLL*-Verfahrens.

$$\Delta \mathbf{w}_l^{opt} = \tilde{\mathbf{A}}^{-1} \tilde{\mathbf{b}} \quad (4.40)$$

Die Matrix $\tilde{\mathbf{A}}$ ist symmetrisch und positiv-semidefinit. Ihre Dimension entspricht der Anzahl Gewichte in der zu optimierenden verdeckten Lage. Besonders bei größeren Netzen mit einer hohen Anzahl von Gewichten in den einzelnen verdeckten Lagen ist es wichtig, ein zugleich effektives und robustes Verfahren zur Lösung des Gleichungssystems (4.39) einzusetzen. Der im Rahmen der vorliegenden Arbeit implementierte Algorithmus basiert auf einer Cholesky-Zerlegung der Matrix $\tilde{\mathbf{A}}$. Dieses Verfahren hat sich bei den durchgeführten Untersuchungen bewährt.

Neben der Lösung des Gleichungssystems bedeutet die Berechnung der Koeffizienten entsprechend den Gleichungen (4.29), (4.34) und (4.37) einen erheblichen Rechenaufwand. Dieser kann jedoch durch konsequente Ausnutzung von Symmetrien in den Koeffizienten erheblich reduziert werden.

Abschließend sei nochmals darauf hingewiesen, daß die mit Gleichung (4.40) ermittelten Gewichtsänderungen $\Delta \mathbf{w}_l^{opt}$ nur bezüglich der modifizierten Gütefunktion (4.25) und bei Konstanz der übrigen Netzgewichte ein Optimum darstellen.

Zur Bestimmung der die ursprüngliche Zielfunktion (den mittleren quadratischen Prädiktionsfehler Gl.(3.20)) minimierenden Netzgewichte \mathbf{w}^* ist es nötig, die Gewichte der einzelnen Lagen wiederholt zu optimieren, bis die Anforderung an den Prädiktionsgewinn erfüllt ist oder die maximale Anzahl an Iterationen überschritten wird.

Der Ablauf des *OLL*-Verfahrens zur Optimierung mehrlagiger Perzeptron-Netze ist in der folgenden Übersicht zusammengefaßt:

4.3 Ablauf des OLL-Optimierungsverfahrens

0 Initialisierung:

- Startwerte für die Netzgewichte $\mathbf{w} = \mathbf{w}^{start} = (\mathbf{W}_2^{start}, \dots, \mathbf{W}_L^{start})$ wählen
- Gewichtungsfaktoren für die modifizierten Gütefunktionen (4.25)
 $\boldsymbol{\mu} = \boldsymbol{\mu}^{start} = (\mu_2^{start}, \dots, \mu_{L-1}^{start})^T$ setzen
- maximal zulässige Anzahl Iterationen IC_{max} vorgeben
- angestrebten maximalen Prädiktionsfehler E^{des} des *MLP*-Netzes vorgeben
- Iterationszähler zu Null setzen $IC = 0$

I Optimierung der Gewichte der Ausgangslage (Kap. 4.1)

- Iterationszähler inkrementieren $IC := IC + 1$
- Berechnung der optimalen Gewichte der Ausgangslage \mathbf{W}_L^{opt} gemäß Gleichung (4.7) für die aktuellen Gewichte der verdeckten Lagen \mathbf{W}_l mit $l \in \{2, \dots, L-1\}$
- Änderung der Gewichte der Ausgangslage $\mathbf{W}_L = \mathbf{W}_L^{opt}$
- Berechnung des aktuellen Prädiktionsfehlers $E^{act} = E(\mathbf{w})$ gemäß Gleichung (3.20) für das *MLP*-Netz mit den Gewichten \mathbf{w}

II Optimierung der Gewichte der verdeckten Lagen (Kap. 4.2)

DO $l = L - 1, 2, STEP - 1$

- II.1 - Berechnung der optimalen Gewichtsänderungen $\Delta \mathbf{W}_l^{opt}$ aus Gleichung (4.25) mit dem aktuellen Gewichtungsfaktor der $l - ten$ Lage μ_l
- Berechnung des Prädiktionsfehlers $E^{test} = E(\mathbf{w})$ gemäß Gleichung (3.20) für das *MLP*-Netz mit den Gewichten $\mathbf{W}_l^{test} = \mathbf{W}_l + \Delta \mathbf{W}_l^{opt}$

II.2 IF $E^{test} < E^{act}$ THEN

II.2.A:

- Änderung der Gewichte der $l - ten$ verdeckten Lage $\mathbf{W}_l := \mathbf{W}_l + \Delta \mathbf{W}_l^{opt}$
- Verminderung des Einflusses des Strafterms durch Verkleinerung von μ_l
- weiter mit Schritt II.3

ELSE

II.2.B:

- Verstärkung des Einflusses des Strafterms durch Vergrößerung von μ_l
- weiter mit Schritt II.1

ENDIF

II.3 - Setzen des erzielten Prädiktionsfehlers des *MLP*-Netzes zu $E^{act} = E^{test}$

ENDDO

III Überprüfung der Abbruchbedingungen

IF $(IC < IC_{max})$ AND $(E^{act} > E^{des})$ THEN

III.A:

- weiter mit Schritt I

ELSE

III.B:

- Abspeichern der endgültigen Netzgewichte $\mathbf{W}_l^* := \mathbf{W}_l$ mit $l = 2, \dots, L$
- Ende des OLL-Optimierungsverfahrens

ENDIF

Wie auf Seite 62 bereits festgestellt wurde, läßt sich für die Adaption der zur Berechnung der modifizierten Gütefunktion benötigten Gewichtungsfaktoren μ kein Verfahren aus dem Optimierungsansatz ableiten. Daher wurden, basierend auf einer Vielzahl von Untersuchungen zum Konvergenzverhalten des *OLL*-Verfahrens, folgende empirische Regeln zur Adaption der Gewichtungsfaktoren μ an den Iterationsverlauf aufgestellt:

II.2.A: Verminderung des Einflusses des Strafterms durch Verkleinerung von μ_l

Wird mit ic die Anzahl der direkt aufeinanderfolgenden Iterationen bezeichnet, für die keine Verschlechterung des Prädiktionsfehlers bei der Optimierung der Lage l trotz ständiger Verringerung des Gewichtungsfaktors μ_l eingetreten ist, so berechnet sich der neue Wert des Gewichtungsfaktors μ_l wie folgt:

$$\mu_l := \mu_l \beta^{ic} \quad \text{mit} \quad 0 < \beta < 1 \quad (4.41)$$

II.2.B: Verstärkung des Einflusses des Strafterms durch Vergrößerung von μ_l

Hier werden 3 Fälle, abhängig vom Wert der Gewichtungsfaktoren μ_l , unterschieden

II.2.B.1: $\mu_l^{min} < \mu_l < \mu_l^{max}$

Wird mit jc die Anzahl der Subiterationen bezeichnet, welche angibt, wie oft bei Optimierung der verdeckten Lage l während der Iteration IC der Fall II.2.B aktuell durchlaufen wurde, so berechnet sich der neue Wert des Gewichtungsfaktors μ_l entsprechend

$$\mu_l := \mu_l \gamma^{jc} \quad \text{mit} \quad \gamma > 1 \quad (4.42)$$

II.2.B.2: $\mu_l \leq \mu_l^{min}$

Bei Untersuchungen hat sich gezeigt, daß der Wert des Gewichtungsfaktors um Größenordnungen zu erhöhen ist, wenn bei Auftreten von II.2.B der Gewichtungsfaktor bereits einen sehr kleinen Wert aufweist. Daher wird in diesem Falle zur Verringerung der Anzahl von Subiterationen der Gewichtungsfaktor μ_l auf den Startwert zurückgesetzt

$$\mu_l := \mu_l^{start} \quad (4.43)$$

II.2.B.3: $\mu_l^{max} \leq \mu_l$

Bei Erreichen eines absoluten oder lokalen Minimums, welches nicht den geforderten maximalen Prädiktionsfehler unterschreitet, kann die Bedingung $E^{test} < E$ nicht erfüllt werden, und II.2.B wird ständig durchlaufen. In diesem Fall wird das *OLL*-Verfahren ebenfalls wie unter III.B beschrieben beendet, wenn die folgende Bedingung erfüllt ist

$$\mu_l > \mu_l^{max} \quad \text{für} \quad l \in 2, \dots, L - 1 \quad . \quad (4.44)$$

Bei allen in Kapitel 5 beschriebenen Untersuchungen mit dem *OLL*-Verfahren wurden die Parameter, welche die Wahl und Adaption der Gewichtungsfaktoren $\boldsymbol{\mu}$ wie oben beschrieben beeinflussen, folgendermaßen gewählt

$$\begin{aligned} \mu_l^{min} = 10^{-20} \quad ; \quad \mu_l^{start} = 10^{-4} \quad ; \quad \mu_l^{max} = 10^{37} \quad \text{für } l = 2, \dots, L-1 \quad (4.45) \\ \gamma = 1, 2 \quad \text{und} \quad \beta = 0, 98 \end{aligned}$$

Wenn nicht anders angegeben, wurden die Startwerte der Netzgewichte $\boldsymbol{w} = \boldsymbol{w}^{start} = (\boldsymbol{W}_2^{start}, \dots, \boldsymbol{W}_L^{start})$ sowohl bei Untersuchungen mit dem *OLL*-Verfahren als auch bei allen anderen Verfahren als gleichverteilte Zufallszahlen aus dem Bereich $[-1, 1]$ vorgegeben.

5 Ergebnisse

Nachdem in den vorangegangenen Kapiteln die theoretischen Grundlagen zur Durchführung der Prädiktion chaotischer Zeitsignale beschrieben wurden, soll nun auf die praktische Eignung der in Kapitel 3 und 4 beschriebenen Optimierungsmethoden eingegangen werden.

Im Vordergrund der folgenden Untersuchungen steht der Vergleich des hier vorgestellten *OLL*-Verfahrens mit den aus der Literatur bekannten Optimierungsverfahren für *MLP*-Netze. Ein Vergleich der *BP*-, *CG*- und *QN*-Verfahren miteinander sowie eine Untersuchung der die verschiedenen Verfahren beeinflussenden Parameter wie Schrittweiten, Momentum-Term, Suchrichtungen, Restartbedingungen oder die optimale Einstellung der Parameter des Liniensuch-Algorithmus finden sich z.B. bei [5].

Einen weiteren Schwerpunkt bilden Untersuchungen, bei denen unter Verwendung des *OLL*-Verfahrens analysiert wird, welchen Einfluß die Rekonstruktionsparameter d_E und τ , die Netztopologie und die Anzahl Q der Netzgewichte sowie die Wahl und der Umfang der Trainingsmuster auf die Qualität der Prädiktionsergebnisse und damit auf die Güte der Nachbildung der Systemabbildung $\mathcal{F}(\mathbf{x}^k)$ bzw. des Flusses $\phi^{TA}(\mathbf{x}^k)$ durch die Abbildung $\mathcal{G}(\mathbf{x}^k)$ des *MLP*-Netzes haben. In diesem Zusammenhang wird auch darauf eingegangen, inwieweit sich Rückschlüsse aus der Kenntnis charakteristischer Kenngrößen des zugrundeliegenden dynamischen Systems wie Dimension, Lyapunov-Exponenten und Autokorrelationsfunktion auf eine optimale Dimensionierung des zur Modellierung verwendeten *MLP*-Netzes ziehen lassen.

Zur Beurteilung der Qualität des Prädiktors ist das der Optimierung zugrundeliegende Fehlermaß (Gl. (3.20)) nur bedingt geeignet. Stattdessen wird als Gütekriterium der Prädiktionsgewinn G verwendet. Dieser ist definiert als Quotient aus Signalleistung und Fehlerleistung, wobei die Erwartungswerte durch Zeitmittelwerte angenähert werden. Für die Trainingsmenge ergibt sich der Prädiktionsgewinn G_{tr} in [dB] zu:

$$\frac{G_{tr}}{[dB]} = 10 \log_{10} \frac{\sum_{k=1}^K \|\mathbf{d}^k\|_2^2}{\sum_{k=1}^K \|\mathbf{d}^k - \mathbf{y}^k\|_2^2} = - \frac{NRMSE}{[dB]} \quad (5.1)$$

Analog wird der Prädiktionsgewinn G_{te} in [dB] für die Testmenge ermittelt. Der negative Prädiktionsgewinn wird in der Literatur [128, 21, 71] auch als **N**ormalized **R**oot-**M**ean **S**quared **E**rror (NRMSE) bezeichnet.

Es werden 3 unterschiedliche Arten von dynamischen Systemen untersucht. Hierbei handelt es sich um die Hénon-Abbildung, das durch 3 Differentialgleichungen beschriebene Lorenz-System sowie zwei Varianten der Mackey-Glass-Differentialgleichung. Alle Signale liegen in doppelt genauer Fließkommadarstellung gem. IEEE 745 vor und werden so auf den Bereich $[0, 1]$ normiert, daß der Gleichanteil über das Gesamtsignal 0,5 beträgt.

Weitere Angaben zu den Eigenschaften und Kenngrößen dieser Systeme finden sich im Anhang A.

5.1 Ein einfaches Optimierungsproblem

Für einen ersten Vergleich des in Kapitel 4 vorgestellten *OLL*-Verfahrens mit dem in der Literatur am häufigsten verwendeten BP-Algorithmus und dessen Varianten (siehe Kapitel 3.4) wird das in Bild 5.1 dargestellte 2:1:1 *MLP*-Netz zur Prädiktion der mit $T_A = 0,01$ abgetasteten x -Komponente $\mathbf{L}_{x,0,01}$ des Lorenz-Attraktors verwendet (siehe Anhang A).

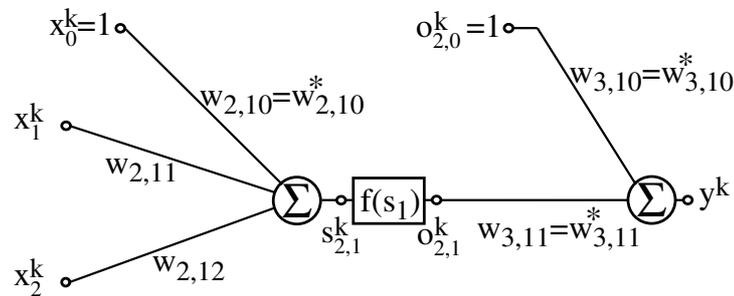


Abbildung 5.1: Ein *MLP*-Netz mit der Topologie 2:1:1, welches zur Visualisierung der prinzipiellen Konvergenzeigenschaften des *OLL*- und verschiedener Varianten des BP-Optimierungsverfahrens eingesetzt wird.

Die Rekonstruktion der Folge $\{\mathbf{x}^k\}$ von Zustandsvektoren wurde gem. Gl.(2.36) mit einer Einbettungsdimension $d_E = 2$ und einer Einbettungskonstanten $\tau=0,01$ durchgeführt. Die Prädiktionsweite wurde mit $pw=0,01$ (1 Abtastwert) vorgegeben. Die Trainingsmenge umfaßte 1 000 Paare $\{\mathbf{x}^k, d^k\} = \{(L_{x,0,01}^k, L_{x,0,01}^{k+1})^T, L_{x,0,01}^{k+2}\}$, welche aus den ersten 1002 Werten des $\mathbf{L}_{x,0,01}$ -Zeitsignals gebildet wurden.

Zur Bestimmung der optimalen Netzgewichte wurde das *OLL*- und das *BD*-Verfahren mit Momentum-Term ($\alpha_M=0,0$) eingesetzt. Die Iteration wurde so lange fortgesetzt, bis die optimalen Gewichte $\mathbf{w}^* = \{w_{2,10}^*, w_{2,11}^*, w_{2,12}^*, w_{3,10}^*, w_{3,11}^*\}$ bestimmt waren. Bei diesem nur 5-dimensionalen Optimierungsproblem stimmen die erzielten Ergebnisse überein. Der Vergleich beider Verfahren bezüglich Konvergenzgeschwindigkeit und Komplexität erbrachte einen Geschwindigkeitsvorteil um den Faktor 44 zugunsten des *OLL*-Verfahrens. Auf diese Art von Vergleichen wird in den folgenden Abschnitten bei dann wesentlich komplexeren Optimierungsproblemen noch näher eingegangen.

Es wurde bewußt eine Einbettungsdimension von $d_E = 2$ gewählt, um den Konvergenzverlauf beider Verfahren graphisch darzustellen. Da für diese Wahl der Rekonstruktions-

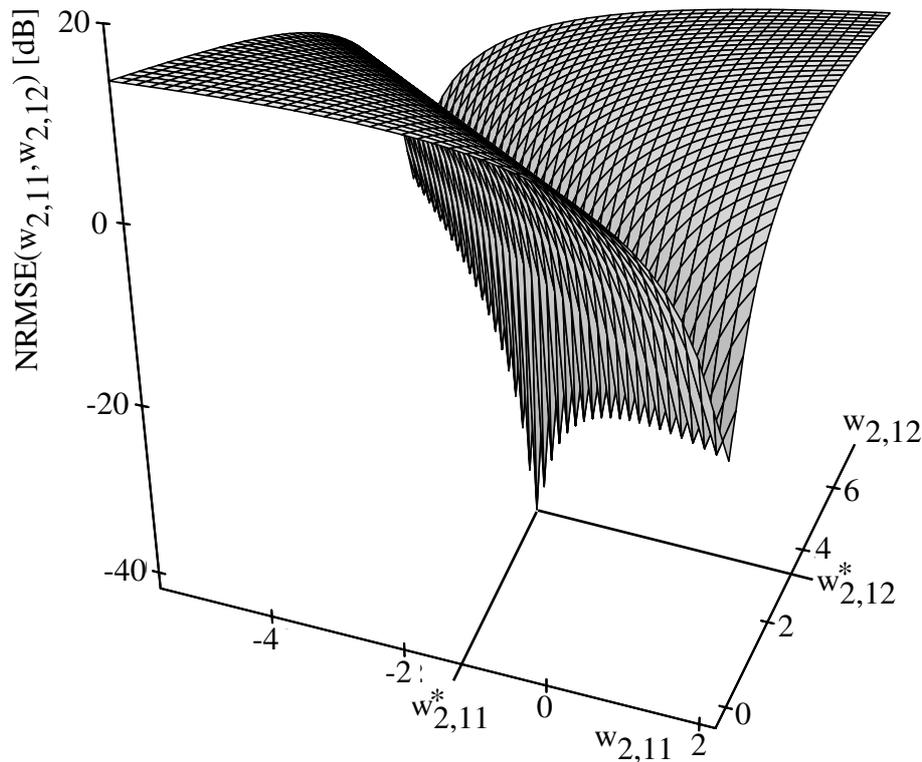


Abbildung 5.2: Die Werte des NRMSE spannen über den Gewichten $(w_{2,11}, w_{2,12})$ der verdeckten Lage bei konstanten Gewichten $\{w_{2,10}^*, w_{3,10}^*, w_{3,11}^*\}$ eine Fehleroberfläche des in Bild 6.1 dargestellten *MLP*-Netzes auf.

onsdimension keine Einbettung vorliegt, wurde mit $G=41,83 \text{ dB}$ auch kein sehr hoher Prädiktionsgewinn erzielt.

Zur Visualisierung und zum Vergleich des Konvergenzverhaltens verschiedener Optimierungsverfahren wurden die Gewichte $\{w_{2,10}, w_{3,10}, w_{3,11}\}$ gleich ihren zuvor bestimmten optimalen Werten $\{w_{2,10}^*, w_{3,10}^*, w_{3,11}^*\}$ gesetzt. Die gemäß Gl.(5.1) normierte Fehleroberfläche³¹ wird dann von den Gewichten $\{w_{2,11}, w_{2,12}\}$ der verdeckten Lage aufgespannt und ist in Abbildung (5.2) dargestellt.

Abbildung 5.3a zeigt die bei Einsatz des *OLL*-Verfahrens ermittelten Gewichte $w_{2,11}^i, w_{2,12}^i$ nach jeder Iteration i für die Wahl der Startwerte $\{0, 0; 0, 0\}$. Bei allen hier beschrie-

³¹Hierzu ist zu bemerken, daß die Darstellung der *NRMSE*-Werte in logarithmischem Maßstab erfolgte. Die nicht normierte Fehleroberfläche weist demnach noch wesentlich ausgeprägtere *Schluchten* bzw. Gradientenunterschiede auf. Dieses gibt bereits einen Hinweis darauf, wie komplex die Fehleroberflächen und damit die Optimierung der Netzgewichte bei den in den Abschnitten 5.4, 5.2 und 5.3 untersuchten wesentlich größeren *MLP*-Netzen ist.

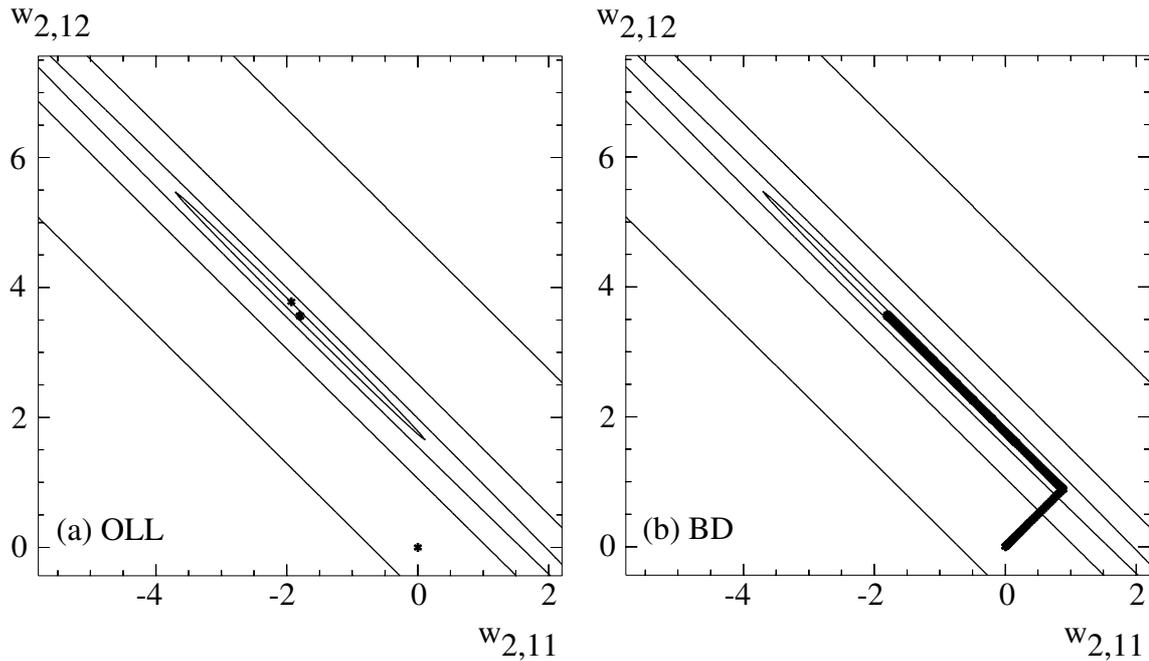


Abbildung 5.3: Darstellung der Höhenlinien für die Werte -40 dB , -20 dB , 0 dB und 20 dB der in Bild 6.2 gezeigten normierten Fehleroberfläche. Die nach jeder Iteration des (a) *OLL*-Verfahrens und (b) des *BD*-Verfahrens mit $\alpha=0,99$ ermittelten Gewichte für den Startwert $\{w_{2,11}=0,0; w_{2,12}=0,0\}$ sind durch * gekennzeichnet.

benen Untersuchungen wurde die Optimierung abgebrochen, wenn der erzielte Prädiktionsgewinn um weniger als $0,0001\text{ dB}$ vom zuvor bestimmten Optimum abwich. Das *OLL*-Verfahren benötigte 3 Iterationen um die optimalen Gewichte $\{w_{2,11}^*, w_{2,12}^*\}$ mit der geforderten Genauigkeit zu ermitteln.

Zum Vergleich wurde der BP-Algorithmus in der Batch-Version ohne Momentum-Term gemäß Gl.(3.47) und mit Momentum-Term gemäß Gl.(3.55) mit unterschiedlichen Werten der Momentum-Konstante α_M eingesetzt. Weiterhin wurde die Schrittweite im Bereich $10^{-2} \leq \eta_{BP} \leq 10^2$ variiert. Die Ergebnisse für unterschiedliche Werte von α_M und η_{BP} sind in den ersten 4 Spalten von Tabelle (5.1) zusammengefaßt.

Die Untersuchungen für Schrittweiten $\eta_{BP} \leq 1$ sind nicht erfaßt, da die Konvergenzgeschwindigkeit hier noch geringer war. Für Werte von $\eta_{BP} \geq 20$ konvergierte das *BP*-Verfahren nicht.

Zusätzlich sind in den letzten 4 Spalten von Tabelle (5.1) die Ergebnisse wiedergegeben, die mit dem *BD*- und dem *OLL*-Verfahren bei unterschiedlicher Wahl der Startwerte $\{w_{2,11}, w_{2,12}\}$ erzielt wurden.

Die Auswertung der Ergebnisse zeigt, daß bei Verwendung des *BD*-Verfahrens und einer Momentum-Konstante $\alpha_M = 0,99$ für die Startgewichte $\{0,0; 0,0\}$ mit 1 250 Iterationen

	Startwerte für Gewichte $\{w_{2,11}, w_{2,12}\}$ der verdeckten Lage							
	{0;0}				{1;0,75}	{-5,5;2}	{-5,5;0}	
	Varianten des Backpropagation-Verfahrens							
	feste Schrittweite η_{BP}				Bold-Driver			
	1,0	2,0	5,0	10,0	variable Schrittweite η_{BD}			
$\alpha_M=0,0$	194862	97429	38969	19483	14846	14955	13634	9869
$\alpha_M=0,1$	175374	87685	35071	17533	11918	12005	10917	7918
$\alpha_M=0,3$	136398	68195	27273	13633	9007	9070	8243	5996
$\alpha_M=0,5$	97420	48703	19473	9730	6542	6549	5981	4339
$\alpha_M=0,7$	58434	29204	11666	5820	4879	4794	4406	3265
$\alpha_M=0,9$	19397	9654	3806	1855	2435	2461	2285	1719
$\alpha_M=0,99$	12433	5751	2125	1381	1250	1249	1172	896
	<i>OLL</i> -Verfahren							
	3				5	6	3	

Tabelle 5.1: Anzahl der Iterationen, die zur Bestimmung der optimalen Netzgewichte ($w_{2,11}^*, w_{2,11}^*$) bei unterschiedlicher Wahl der Startgewichte ($w_{2,11}, w_{2,11}$) und der Optimierungsparameter η_{BP} und α_M von den *BP*- und *BD*-Verfahren im Vergleich zum *OLL*-Verfahren benötigt werden.

das beste Ergebnis erzielt wird. Demgegenüber bestimmt das *OLL*-Verfahren bereits nach 3 Iterationen das Optimum.

Entsprechend Bild (5.3a) sind in Bild (5.3b) die Netzgewichte nach jeder der 1250 Iterationen des *BD*-Verfahrens in die Höhenliniendarstellung der normierten Fehleroberfläche eingetragen. Die genauere Auswertung des Konvergenzverhaltens aus Bild (5.3b) ergibt, daß bereits nach ca. 80 Iterationen die *Talsole* bei $\{0,9;0,9\}$ erreicht wird. Anschließend benötigt das Gradientenverfahren mehr als 1 100 Iterationen, um entlang der *Talsole* das Minimum zu erreichen. Es arbeitet demnach in Zielnähe sehr ineffektiv. Dieses Verhalten, daß auf starken Oszillationen (Zick-Zack-Laufen) im Lernverlauf beruht, ist typisch für Gradientenverfahren und tritt besonders bei steifen Optimierungsproblemen in Gebieten

auf, wo die Fehleroberfläche stark gekrümmt ist oder in *Tälern* der Fehleroberfläche, die von steilen *Wänden* umgeben sind.

Wie Tabelle (5.1) weiterhin zu entnehmen ist, ändert auch die Variation der Startwerte $\{w_{2,11}, w_{2,12}\}$ an dem Vergleich zwischen *BD*- und *OLL*-Verfahren wenig. Auch für die im Rahmen dieser Untersuchung günstigste Wahl der Startgewichte benötigt das *BD*-Verfahren immer noch ca. 900 Iterationen, um das Minimum zu bestimmen. Demgegenüber erreicht das *OLL*-Verfahren das Optimum immer in 3 bis 6 Iterationen.

Aufgrund der speziellen Form der Fehleroberfläche werden bei den obigen Untersuchungen mit dem *BP*- und *BD*-Verfahren die besten Ergebnisse für recht große Werte von $\alpha_M = 0,99$ und $\eta_{BP} = 10$ erzielt. Aus der Literatur [55] ist jedoch bekannt, daß bei größeren *MLP*-Netzen und damit bei komplexeren Optimierungsproblemen für η_{BP} und α_M Werte in den Bereichen $0,6 \leq \alpha_M \leq 0,9$ und $10^{-3} \leq \eta_{BP} \leq 10^{-1}$ optimal sind. Alle im weiteren vorgestellten Untersuchungen mit dem *BP*- bzw. *BD*-Verfahren wurden mit Werten von $\eta_{BP} = 0,01$ und $\alpha_M = 0,8$ durchgeführt.

Dieses einfache Beispiel zeigt bereits einige wesentliche Vorteile des *OLL*-Verfahrens gegenüber den häufig eingesetzten *BP*- und *BD*-Verfahren auf:

- die Konvergenzgeschwindigkeit ist sehr hoch
- das Verfahren bedarf keiner Einstellung von Parametern, die vom jeweiligen Optimierungsproblem abhängen (die Einbettungsparameter, die Netztopologie und die Trainingsmuster müssen natürlich vorgegeben werden)
- die Optimierung basiert auf keiner speziellen Annahme bezüglich der Form der Fehleroberfläche
- der Einfluß der gewählten Startgewichte auf die Optimierungsergebnisse ist gering

5.2 Prädiktionsergebnisse für das chaotische Lorenz-Zeitsignal

Die Diskussion der Prädiktionsergebnisse, die für das chaotische Lorenz-Zeitsignal $\mathbf{L}_{x,0.1}$ erzielt wurden, ist Inhalt der folgenden Ausführungen. Im Unterschied zu den Untersuchungen des vorherigen Abschnittes werden die rekonstruierten Zustandsvektoren gemäß Abschnitt 2.5 jetzt aus dem in Anhang A.1 vorgestellten $\mathbf{L}_{x,0.1}$ -Signal und nicht mehr aus dem $\mathbf{L}_{x,0.01}$ -Signal gebildet. Damit erhöht sich die Abtastzeit um den Faktor 10, womit sich die Bindungen zwischen aufeinanderfolgenden Abtastwerten wesentlich verringern.

Wenn nicht anders angegeben, wird bei allen Untersuchungen eine Trainingsmenge von 2 000 Mustern $\{\mathbf{x}^k, d^k\}_{k=1}^K$ und eine Testmenge von 100 000 Mustern $\{\mathbf{x}^k, d^k\}_{k=K+1}^{N-\tau(d_E-1)}$ verwendet. Alle nachfolgend vorgestellten Prädiktionsergebnisse beruhen auf einer Prädik-

tionsweite von $pw = 0,1$ und einem skalaren Prädiktionsansatz. Dieses entspricht der Vorhersage des jeweils nächsten Abtastwertes des $\mathbf{L}_{x,0,1}$ -Signals.

In den folgenden Abschnitten werden die Vergleiche auch auf die in den Abschnitten 3.5 und 3.6 vorgestellten *CG*- und *QN*-Verfahren ausgedehnt. Dabei wird insbesondere auf die Eignung der Verfahren für wesentlich komplexere *MLP*-Netze eingegangen.

5.2.1 Lineare Prädiktion des Lorenz-Zeitsignals

Sowohl für das Lorenz-Signal als auch für die in den Abschnitten 5.3 und 5.4 untersuchten Mackey-Glass- und Hénon-Signale werden zunächst die Prädiktionsergebnisse präsentiert, die mit linearen Prädiktoren erzielbar sind. Anschließend werden die Resultate, die mit *MLP*-Netzen für die nichtlineare Prädiktion dieser Signale erzielt werden, sowohl untereinander hinsichtlich der Eigenschaften der eingesetzten Optimierungsverfahren, als auch mit den Ergebnissen der linearen Prädiktoren verglichen.

Verfahren zur Berechnung der optimalen Prädiktorkoeffizienten finden sich bei Makhoul [83], Jayant und Noll [65] bzw. [67, 139]. Diese Verfahren wurden dahingehend erweitert, daß auch eine Prädiktion weiter in der Zukunft liegender Werte möglich ist ($pw > 1$), und daß auch nicht direkt aufeinanderfolgende Abtastwerte zur Berechnung der Koeffizienten der linearen Prädiktoren verwendet werden können ($\tau > 1$ Abtastwert).

Zur linearen Prädiktion der Signale wurden Prädiktoren der Ordnung $P=1$ bis $P=100$ verwendet. Mit der hier gewählten Bezeichnungsweise entspricht die Ordnung P des Prädiktors der Einbettungsdimension d_E . Abbildung 5.4 zeigt die für die Testmenge erzielten Prädiktionsgewinne in Abhängigkeit von der Einbettungsdimension d_E und der Einbettungskonstanten τ für das $\mathbf{L}_{x,0,1}$ -Signal, wobei die Resultate für die Trainingsmenge nicht mehr als um ca. $0,1 \text{ dB}$ über den dargestellten liegen. Die Berechnung der Prädiktorkoeffizienten erfolgte aus der Trainingsmenge.

Die erzielten Prädiktionsergebnisse liegen für $\tau=1$ bereits bei Prädiktorordnungen von $P=20$ in der Größenordnung von 4 dB und steigen auch für größere Prädiktorordnungen nicht mehr wesentlich an. Wie erwartet werden für größere Werte von τ schlechtere Ergebnisse im Bereich von $3,2 \text{ dB}$ bis $3,3 \text{ dB}$ erzielt, da nur die linearen Bindungen im Signal zur Prädiktion berücksichtigt werden und diese um so geringer sind, je größer der Wert von τ ist. Insgesamt liefert die lineare Prädiktion für das Lorenz-Signal sehr geringe Prädiktionsgewinne, da die linearen Bindungen sehr schwach ausgeprägt sind, was auch durch die auf Seite 143 dargestellte Autokorrelationsfunktion und das Leistungsdichtespektrum des $\mathbf{L}_{x,0,1}$ -Signals unterstrichen wird. Lineare Ansätze sind daher zur Prädiktion des Lorenz-Signals ungeeignet.

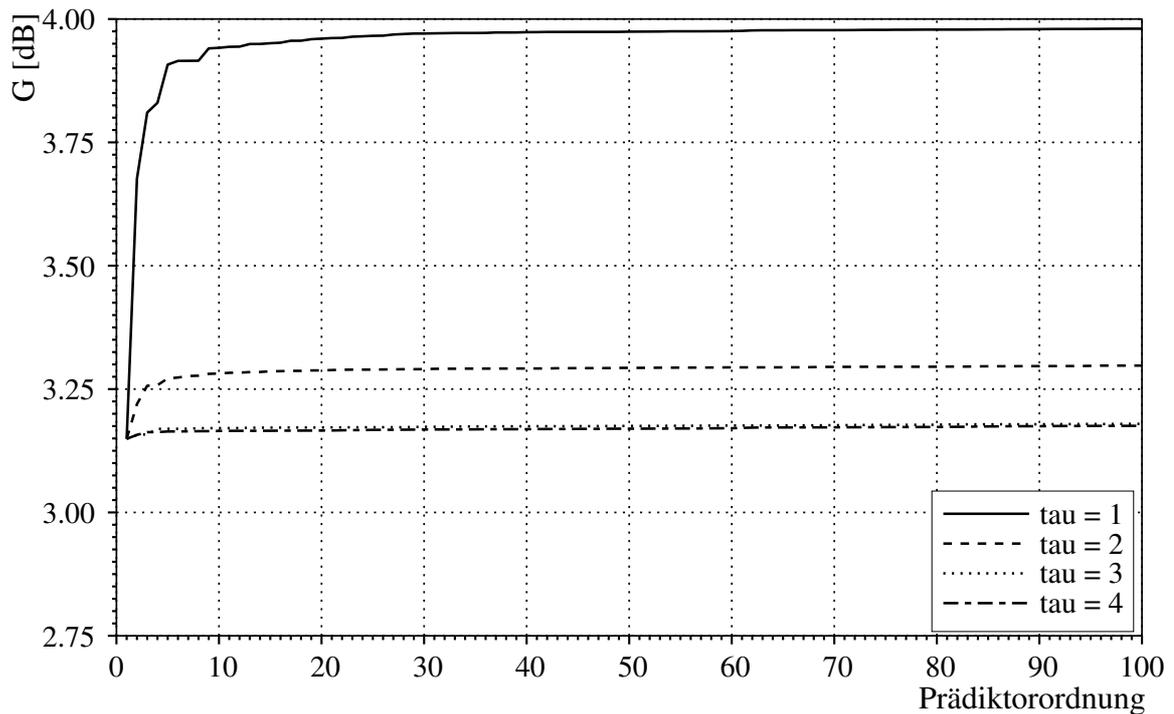


Abbildung 5.4: Ergebnis der linearen Prädiktion für das Lorenz-Signal bei Verwendung von Prädiktoren der Ordnung $P=1$ bis $P=100$ und Variation der Einbettungskonstanten $\tau = 1, \dots, 4$.

5.2.2 Nichtlineare Prädiktion des Lorenz-Zeitsignals mit dem *OLL*-Verfahren

Der Lorenz-Attraktor besitzt eine Kapazität und eine Korrelationsdimension von $d_K = 2,07$ bzw. $d_C = 2,06$, womit zur Rekonstruktion des Attraktors aus dem skalaren Zeitsignal $\mathbf{L}_{x,0,1}$ eine minimale Dimension der Zustandsvektoren von $d_E = 3$ erforderlich ist. Wird die Dimension der rekonstruierten Zustandsvektoren entsprechend den Bedingungen Gl.(2.37, 2.38) von Takens und Mañé zu $d_E \geq 6$ gewählt, so ist sichergestellt, daß die Rekonstruktion gemäß Gl.(2.36) eine Einbettung darstellt.

Werden *MLP*-Netze zur nichtlinearen Prädiktion des $\mathbf{L}_{x,0,1}$ Signals eingesetzt, so muß neben den Einbettungsparametern d_E und τ zusätzlich die Topologie des Netzes festgelegt werden. Da durch d_E die Anzahl der Knoten in der Eingangslage und durch das Prädiktionsproblem die Anzahl der Knoten in der Ausgangslage definiert sind, ist noch die Anzahl der verdeckten Lagen sowie die Anzahl der Knoten in diesen Lagen vorzugeben. Die Anzahl Q der Netzgewichte kann dann mit Gl.(3.6) ermittelt werden und ist ein Maß für die Komplexität des Netzes.

Zunächst wird untersucht, welchen Einfluß die Parameter d_E und τ der Rekonstruktion auf die Prädiktionsergebnisse haben. Dazu werden 3-lagige *MLP*-Netze unterschiedlicher Topologie, aber mit annähernd konstanter Anzahl von $Q \approx 350$ Netzgewichten, mit

dem *OLL*-Verfahren optimiert, wobei unabhängig vom erzielten Prädiktionsgewinn immer 1 000 Iterationen durchgeführt werden.

Die Topologie der eingesetzten *MLP*-Netze, die Anzahl der Gewichte in den einzelnen Lagen und des gesamten Netzes sowie der relative Rechenaufwand T_{IC} für eine Iteration des *OLL*-Verfahrens sind in Tabelle 5.2 zusammengestellt. Als Bezug wurde der Berechnungsaufwand für das 3:70:1 *MLP*-Netz gewählt.

Topol.	3:70:1	4:59:1	5:50:1	6:44:1	7:39:1	8:35:1	9:32:1	10:29:1	11:27:1
d_E	3	4	5	6	7	8	9	10	11
Q	351	355	350	353	352	351	353	349	352
T_{IC}	1,00	0,99	0,94	0,91	0,92	0,92	0,91	0,90	0,89
Topol.	12:25:1	13:23:1	14:22:1	15:21:1	16:19:1	17:18:1	18:17:1	19:17:1	20:16:1
d_E	12	13	14	15	16	17	18	19	20
Q	351	346	353	358	343	343	341	358	353
T_{IC}	0,92	0,89	0,93	0,94	0,89	0,90	0,86	0,95	0,94

Tabelle 5.2: Topologie ausgewählter 3-lagiger *MLP*-Netze mit einer Anzahl von Q Netzgewichten und einem im Vergleich mit dem 3:70:1 Netz relativen Rechenaufwand von T_{IC} je Iteration des *OLL*-Verfahrens.

Zusätzlich zu den hier verwendeten Netzen sind diese Daten auch für Netze mit $13 \leq d_E \leq 20$ Knoten in der Eingangslage angegeben. Auf diese wird in Abschnitt 5.3.2 näher eingegangen.

Die Abbildungen 5.5 (a) und 5.5 (b) zeigen die Prädiktionsgewinne für das $\mathbf{L}_{x,0.1}$ -Signal, die mit den aufgeführten *MLP*-Netzen für Einbettungsdimensionen im Bereich $3 \leq d_E \leq 12$ und Einbettungskonstanten im Bereich $1 \leq \tau \leq 4$ erzielt werden, wobei Abbildung 5.5 (a) die Prädiktionsgewinne $G_{tr}[dB]$ für die Trainingsmenge und Abbildung 5.5 (b) die Prädiktionsgewinne $G_{te}[dB]$ für die Testmenge wiedergibt.

Der Vergleich mit den Ergebnissen der linearen Prädiktion zeigt, daß mit der nichtlinearen Prädiktion auf Basis der *MLP*-Netze um Größenordnungen bessere Prädiktionsgewinne von maximal 76,8 dB gegenüber 4,1 dB für die Trainingsmenge und von maximal 47,4 dB gegenüber 4,0 dB für die Testmenge erzielt werden. Der auffällig hohe Unterschied von ca. 20 dB bis 30 dB zwischen den Prädiktionsgewinnen für die Trainings- und die Testmenge bei Verwendung von Einbettungsdimensionen im Bereich von $5 \leq d_E \leq 12$ ist im wesentlichen auf zwei Phänomene zurückzuführen. Zum einen kann die Trainingsmenge zu klein oder auch ungünstig gewählt, zum anderen die Optimierung der *MLP*-Netze nicht rechtzeitig beendet worden sein.

Die Rekonstruktion mit 3- und 4-dimensionalen Zustandsvektoren liefert insgesamt wesentlich geringere Prädiktionsgewinne als mit 5- und höher-dimensionalen Einbettungen.

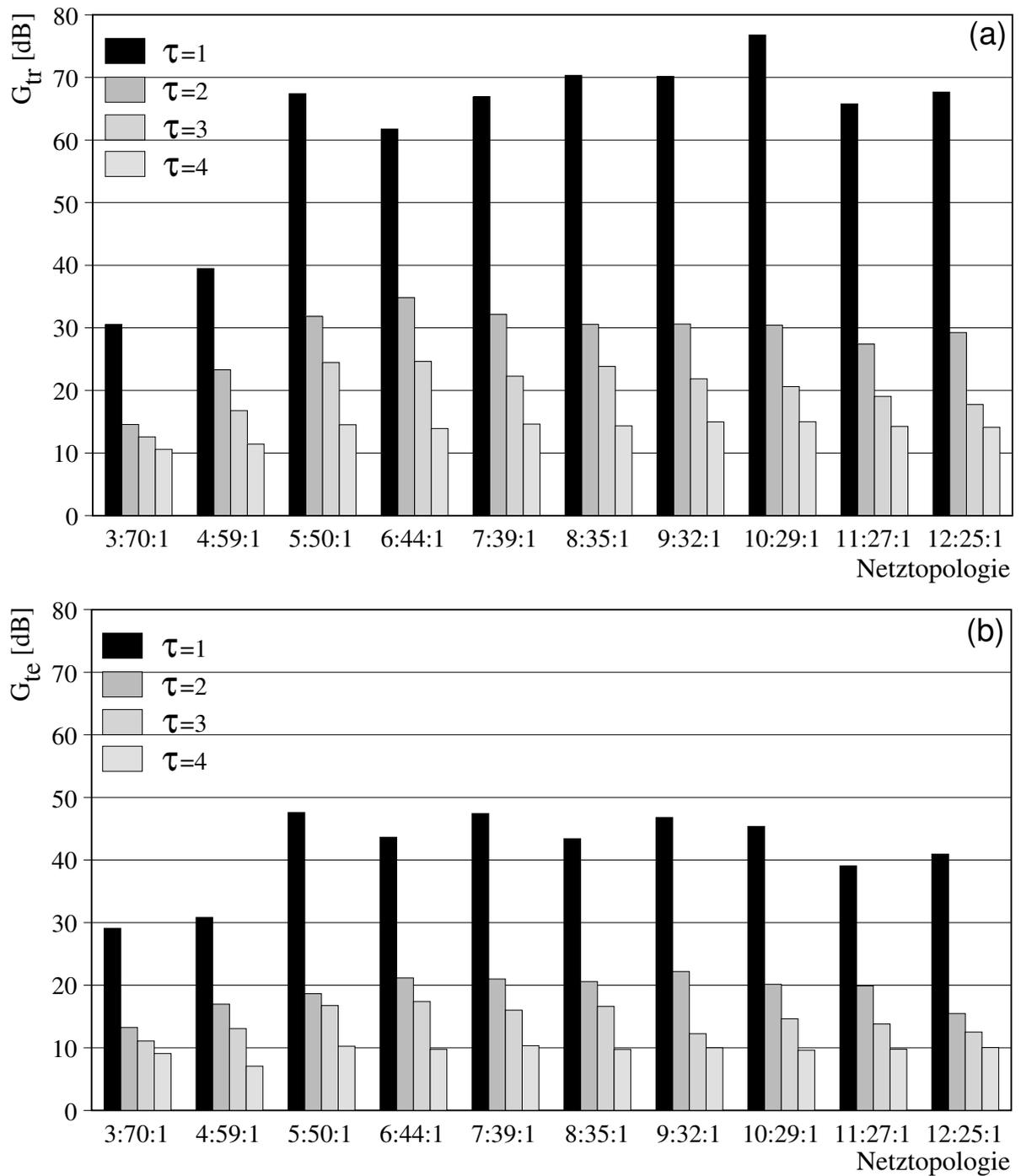


Abbildung 5.5: Vergleich der Prädiktionsergebnisse, die mit ausgewählten 3-lagigen *MLP*-Netzen mit annähernd konstanter Anzahl von 350 Netzgewichten für das chaotische Lorenz-Signal mit dem *OLL*-Verfahren erzielt werden. (a): Ergebnisse für die 2 000 Muster umfassende Trainingsmenge (b): Ergebnisse für die 100 000 Muster umfassende Testmenge

Daher fallen die Unterschiede der Ergebnisse für die Trainings- und Testmenge hier

auch deutlich niedriger aus. Offensichtlich werden bei Rekonstruktionen mit mehr als 4-dimensionalen Zustandsvektoren die wesentlichen Abbildungseigenschaften des Lorenz-Systems erfaßt und durch die *MLP*-Netze nachgebildet.

Vergleicht man die Ergebnisse bezüglich der Einbettungskonstanten τ , so ist festzustellen, daß wie zuvor bei den linearen Prädiktoren die Wahl von $\tau = 1$ die mit Abstand besten Resultate liefert. Diese Eigenschaft, die für lineare Prädiktoren theoretisch begründet ist, läßt sich jedoch auf nichtlineare Prädiktoren nicht verallgemeinern, wie die Untersuchungen anderer chaotischer Zeitsignale in Abschnitt 5.3 zeigen werden.

Die Auswertung der Abbildungen 5.5 (a) und (b) bezüglich der Einbettungsparameter zeigt, daß die Wahl von $d_E = 5$ und $\tau = 1$ im Vergleich zu anderen Kombinationen die besten Ergebnisse liefert. Daher werden für die verbleibenden Untersuchungen des $L_{x,0,1}$ -Signals diese Einbettungsbedingungen beibehalten.

Zur Beurteilung der Auswirkungen einer Veränderung der Netztopologie auf die erzielbaren Prädiktionsgewinne, bei gleichzeitig annähernd konstanter Anzahl von $Q \approx 350$ Netzgewichten, werden die Ergebnisse der in Tabelle 5.3 aufgeführten 4- und 5-lagigen *MLP*-Netze mit denen des bereits beschriebenen 5:50:1 Netzes verglichen. Obwohl gemäß dem Backpropagation-Approximations-Theorem von Hecht-Nielsen [53, 54, 55] (siehe auch Seite 32) und anderer Veröffentlichungen [63, 134, 36] bereits 3-lagige *MLP*-Netze universelle Approximationseigenschaften besitzen, erweist sich die Aufteilung der verdeckten Knoten auf mehr als eine verdeckte Lage in verschiedener Hinsicht als vorteilhaft, wie im folgenden gezeigt wird.

Topol.	5:7:34:1	5:11:12:10:1	5:11:22:1	5:13:18:1	5:15:15:1	5:16:10:7:1	5:18:12:1
Q	349	351	353	349	346	351	349
T_{IC}	0,81	0,53	0,73	0,69	0,66	0,55	0,65
Topol.	5:18:6:16:1	5:21:10:1	5:22:9:1	5:23:7:5:1	5:24:8:1	5:28:6:1	5:31:5:1
Q	351	357	349	352	353	349	352
T_{IC}	0,55	0,66	0,64	0,61	0,65	0,67	0,76

Tabelle 5.3: Topologie ausgewählter 4- und 5-lagiger *MLP*-Netze mit einer Anzahl von Q Netzgewichten und einem im Vergleich mit dem 3:70:1 Netz relativen Rechenaufwand von T_{IC} je Iteration des *OLL*-Verfahrens.

Neben der Topologie dieser *MLP*-Netze ist in Tabelle 5.3 die Anzahl Q der Netzgewichte sowie der Rechenaufwand pro Iteration des *OLL*-Verfahrens im Vergleich zu einem 3:70:1 Netz angegeben. Bei Betrachtung des Rechenaufwandes fällt auf, daß dieser besonders für 5-lagige Netze um bis zu 47 % unterhalb des für ein 3:70:1 Netz erforderlichen liegt. Dieses Verhalten ist durch Gleichung (4.40) in Verbindung mit den Gleichungen (4.29, 4.34, 4.36, 4.37 und 4.38) des *OLL*-Verfahrens begründet. Einen wesentlichen Anteil am

Rechenaufwand je Iteration hat die Aufstellung und Invertierung der Matrix $\tilde{\mathbf{A}}$, wobei der Aufwand zur Aufstellung für die hier untersuchten Netztopologien gegenüber dem Aufwand zur Invertierung bei weitem überwiegt.

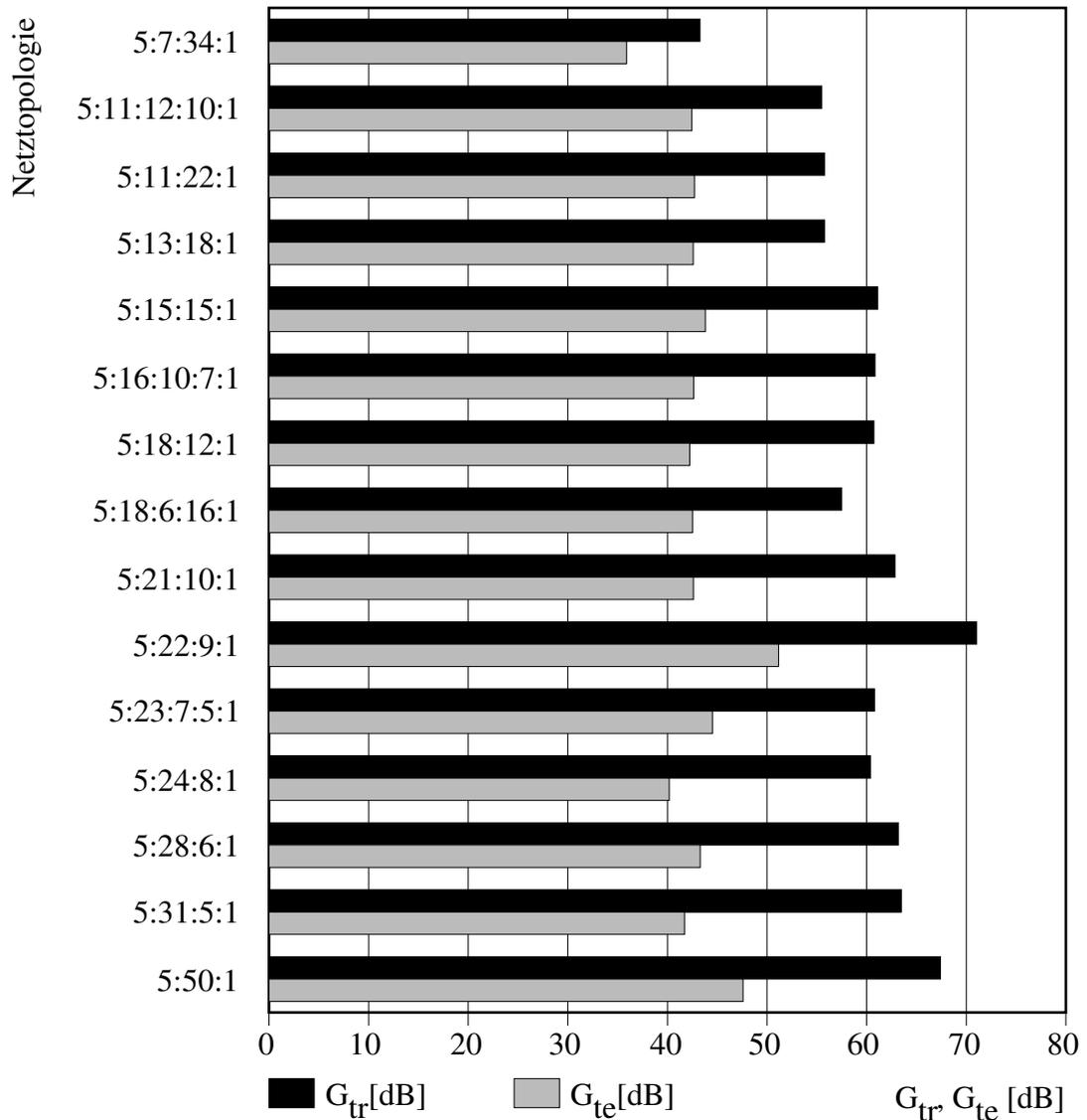


Abbildung 5.6: Vergleich der Prädiktionsgewinne für die Trainings- und Testmenge, die mit 3-, 4- und 5-lagigen *MLP*-Netzen mit annähernd konstanter Anzahl von 350 Netzgewichten und den Einbettungsparametern $d_E = 5, \tau = 1$ für das chaotische Lorenz-Signal mit dem *OLL*-Verfahren erzielt werden.

Dieser Aufwand steigt linear mit dem Umfang der Trainingsmenge aber quadratisch mit der maximalen Anzahl an Gewichten der verdeckten Lagen an. Bei annähernd konstanter Gesamtanzahl hat daher eine gleichmäßige Verteilung dieser Gewichte auf fünf statt wie im Falle des 3:70:1 Netzes auf nur drei Lagen eine erhebliche Reduktion des Rechen-

aufwandes zur Folge. So hat das 5:18:6:16:1 *MLP*-Netz bei einem um 45 % reduzierten Rechenaufwand 108 Gewichte in der zweiten, 114 in der dritten, 112 in der vierten und 17 Gewichte in der Ausgangslage. Im Vergleich hierzu besitzt das 3:70:1 Netz 280 Gewichte in der zweiten und in der Ausgangslage 71 Gewichte.

Die in Abbildung 5.6 dargestellten Prädiktionsergebnisse für die Trainings- und Testmenge zeigen, daß 4- und 5-lagige *MLP*-Netze bei geringerem Rechenaufwand den 3-lagigen Netzen vergleichbare und im Fall des 5:22:9:1 Netzes sogar bessere Ergebnisse liefern. Dabei liegen die Prädiktionsgewinne für die Trainingsmenge, die wie zuvor 2 000 Muster umfaßt, zwischen 55 *dB* und 71 *dB*. Allerdings fallen die erzielten Resultate für die Testmenge um 15 *dB* bis 20 *dB* niedriger aus.

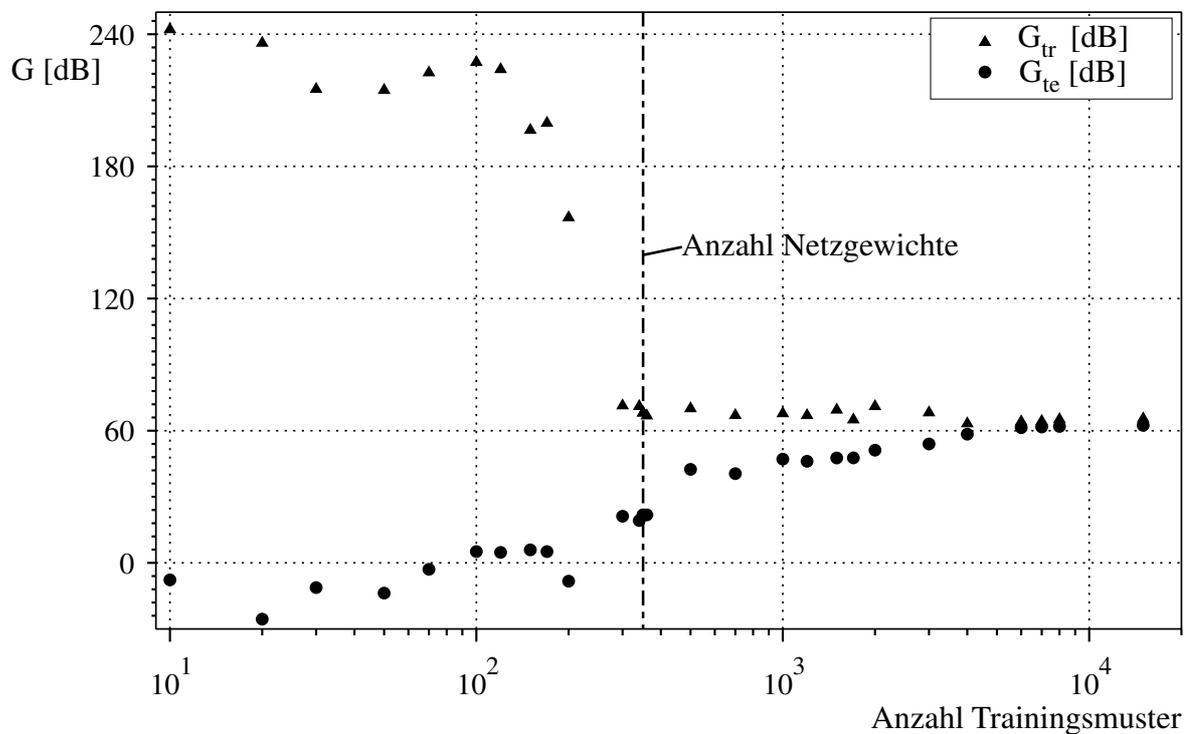


Abbildung 5.7: Vergleich der Prädiktionsgewinne für Trainings- und Testmenge, die mit einem 5:22:9:1 *MLP*-Netz für das Lorenz-Signal erzielt werden, wenn die Anzahl der Trainingsmuster zwischen 10 und 15 000 variiert wird.

Zur Erklärung dieses Phänomens, welches in Abbildung 5.5 in noch stärkerem Maße beobachtet wurde, wird im folgendem am Beispiel des 5:22:9:1 *MLP*-Netzes untersucht, welchen Einfluß der Umfang der Trainingsmenge auf die erzielbaren Prädiktionsgewinne für die Testmenge hat. Bei diesen Untersuchungen wurde die aus 100 000 Mustern bestehende Testmenge $\{\mathbf{x}^k, d^k\}_{k=15001}^{115000}$ nicht verändert. Der Umfang der Trainingsmengen wurde jedoch zwischen $K=10$ und $K=15\,000$ variiert. Die Muster der verschiedenen Trainingsmengen $\{\mathbf{x}^k, d^k\}_{k=15000-K}^{15000}$ wurden der Folge von rekonstruierten Zustandsvektoren direkt vor denen der Testmenge entnommen.

Für die Optimierung der 5:22:9:1-Netze mit verschiedenen Trainingsmengen wurde eine maximale Anzahl von 1000 Iterationen vorgegeben. Diese Anzahl wurde jedoch nur für $K > 120$ erreicht. Aufgrund der sehr hohen Konvergenzgeschwindigkeit beendet das *OLL*-Verfahren für $K=10, 20, 30, 50, 70, 100$ und 120 bereits nach 2, 3, 12, 107, 317, 160 bzw. 480 Iterationen mit Fall II.2.B.3 (siehe Seite 67) die Optimierung der Netzgewichte, da im Rahmen der Rechengenauigkeit keine Verbesserung des Prädiktionsgewinns für die Trainingsmenge mehr erreicht wurde. Die erzielten Prädiktionsgewinne für die Trainings- und die Testmenge sind in Abbildung 5.7 dargestellt. Die Anzahl der Netzgewichte beträgt für dieses Netz $Q=349$ und ist durch eine gestrichelte senkrechte Linie markiert. Diese Gegenüberstellung der Prädiktionsgewinne zeigt sehr deutlich, daß der Umfang der Trainingsmenge einen entscheidenden Einfluß auf die Qualität der Nachbildung der Abbildungseigenschaften des dynamischen Systems durch die *MLP*-Netze hat. Hier sind im wesentlichen drei Bereiche identifizierbar, in denen unterschiedliche Effekte auftreten.

Für den ersten Bereich ist aus Abbildung 5.7 ersichtlich, daß die Zielwerte $\{d^k\}$ der Trainingsmenge praktisch fehlerfrei vorhergesagt werden, die Prädiktionseigenschaften des *MLP*-Netzes für die Testmenge jedoch unzureichend sind, wenn die Anzahl K der Trainingsmuster geringer als die Anzahl Q der Netzgewichte ist. Dieser Fall tritt, wie Abbildung 5.7 zu entnehmen ist, für Trainingsmengen mit $K=10, 20, 30, 50, 70, 100, 120, 150, 170$ sowie 200 Mustern auf und hat zur Folge, daß die entsprechenden *MLP*-Netze die Zielwerte $\{d^k\}_{k=15000-K}^{15000}$ der Trainingsmengen nur speichern und reproduzieren, ohne die Abbildungseigenschaften des dynamischen Systems auch nur annähernd nachzubilden. Dieses Verhalten des *OLL*-Verfahrens ist dadurch begründet, daß für diese Beispiele die Ordnung des Optimierungsproblems mit $Q=349$ größer als die Anzahl zur Verfügung gestellter Daten ist. In diesem Bereich sind die Prädiktionsgewinne für die Trainingsmenge größer als 160 dB für die Testmenge, jedoch weit unter 10 dB und in einigen Fällen sogar negativ.

Im zweiten Bereich für $300 \leq K \leq 3000$ treten zwei Effekte auf. Zum einen wird die Optimierung der *MLP*-Netze nicht rechtzeitig beendet und es findet das sogenannte *Overlearning* oder *Overtraining* statt (vgl. Hecht-Nielsen [55], Seite 115), zum anderen ist die Trainingsmenge zu klein oder ungünstig gewählt, so daß die Abbildungseigenschaften des zugrundeliegenden dynamischen Systems nicht in ausreichendem Maße durch die rekonstruierten Zustandsvektoren der Trainingsmenge repräsentiert werden.

Der letzte Bereich mit $K > 3000$ ist kennzeichnend für Trainingsmengen mit einem ausreichenden Datenumfang, die die Eigenschaften des zugrundeliegenden dynamischen Systems gut repräsentieren. Mit den auf Basis dieser Trainingsmengen optimierten *MLP*-Netzen werden Prädiktionsgewinne von 64 dB bis 66 dB für die jeweilige Trainingsmenge erzielt. Im Gegensatz zu den zuvor beschriebenen Untersuchungen fallen die Prädiktionsergebnisse für die Testmengen allerdings nur um ca. $2 - 3 \text{ dB}$ geringer aus als für die jeweilige Trainingsmenge. Im Rahmen der Untersuchungen zur Prädiktion des $\mathbf{L}_{x,0.1}$ -

Signals wurden mit diesen *MLP*-Netzen die besten Prädiktionsgewinne für die Testmenge mit annähernd 63 dB erzielt.

Zur besseren Darstellung der bei Veränderung des Umfangs K der Trainingsmenge auftretenden Effekte sind in Abbildung 5.8 die Prädiktionsgewinne für die Trainings- und Testmengen mit $K=120, 349, 1\,000$ sowie $15\,000$ Mustern über dem Iterationsverlauf aufgetragen.

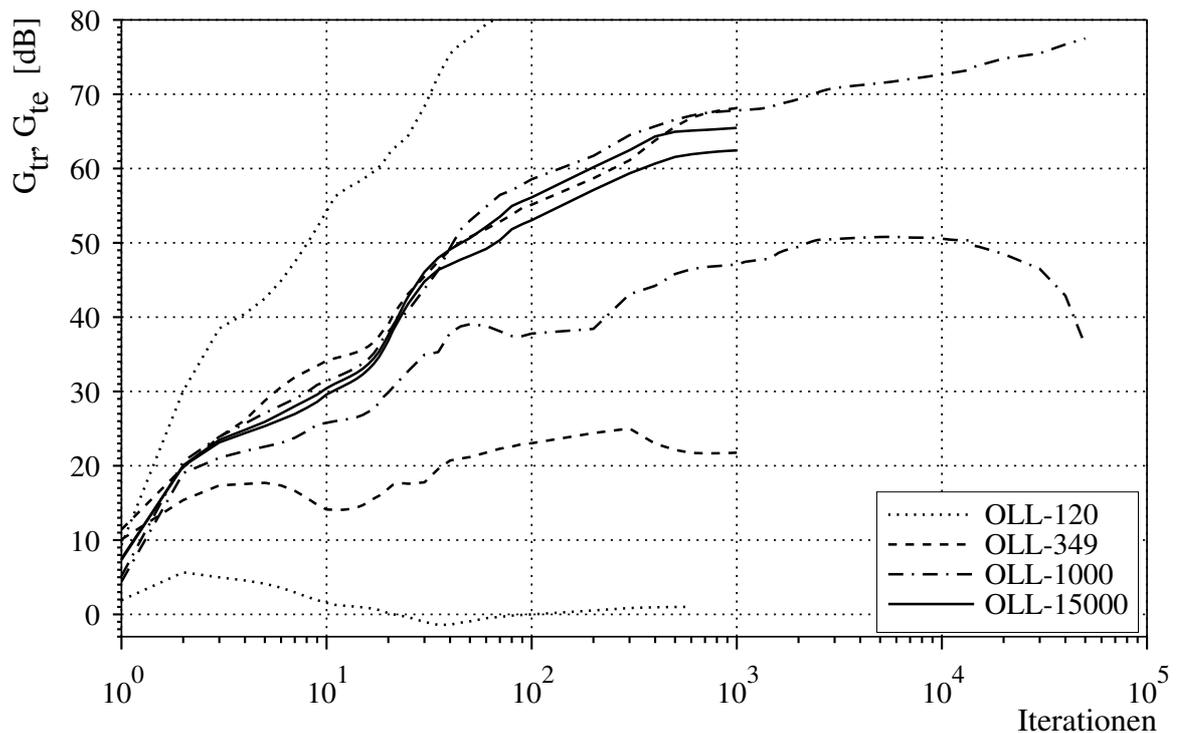


Abbildung 5.8: Gegenüberstellung der Prädiktionsgewinne für Trainings- (jeweils obere) und Testmenge (jeweils untere Kurve) über der logarithmierten Anzahl von Iterationen im Verlauf der Optimierung eines $5:22:9:1$ *MLP*-Netzes mit dem *OLL*-Verfahren bei Verwendung unterschiedlich großer Testmengen.

Bei $K=120$ bzw. $K=349$ Mustern sind die Trainingsmengen deutlich zu klein gewählt, so daß die Abbildungseigenschaften des dynamischen Systems durch die *MLP*-Netze nicht oder nur unzureichend nachgebildet werden, und die Prädiktionsgewinne für die Testmenge mit $4,7 \text{ dB}$ bzw. $21,8 \text{ dB}$ sehr gering ausfallen. Bei Verwendung von nur 120 Trainingsmustern werden die vorgegebenen 1000 Iterationen nicht erreicht. Das *OLL*-Verfahren wird hier bereits nach 480 Iterationen beendet, da bei einem Prädiktionsgewinn von 224 dB im Rahmen der Rechengenauigkeit ein lokales Minimum der Fehleroberfläche erreicht wird.

Der Overlearning-Effekt tritt bei der Wahl von $K=1\,000$ Trainingsmustern deutlich auf. Hier erreicht der Prädiktionsgewinn für die Testmenge nach ca. $5\,500$ Iterationen mit

annähernd 51 dB sein Maximum und nimmt im weiteren Iterationsverlauf wieder ab, wobei der Prädiktionsgewinn für die Testmenge natürlich weiterhin steigt. In diesem Fall werden die Abbildungseigenschaften des *MLP*-Netzes zu stark an die lokalen Eigenschaften der Trainingsmenge angepaßt, und die für die Prädiktion der Testmenge erforderliche Nachbildung der globalen Abbildungseigenschaften des zugrundeliegenden dynamischen Systems erfolgt nur unzureichend, so daß die Verallgemeinerungseigenschaften der neuronalen Netze nicht ausgenutzt werden. Zur Vermeidung dieses Effektes ist entweder das Verfahren rechtzeitig abzubrechen oder die Trainingsmenge zu vergrößern.

Wird der Umfang der Trainingsmenge auf mehr als $K=3\,000$ oder wie in dem vorliegenden Beispiel auf $K=15\,000$ Muster erhöht, treten die zuvor beschriebenen Effekte nicht mehr auf. Die Trainingsmenge repräsentiert dann alle relevanten Abbildungseigenschaften des Systems, so daß die Prädiktionsgewinne für die Trainings- und die Testmenge in nahezu gleichem Verhältnis steigen und nach 1 000 Iterationen 65,5 bzw. 62,5 dB betragen. Da der erforderliche Rechenzeitaufwand in erster Näherung linear mit dem Umfang der Trainingsmenge steigt, sollte diese so klein wie möglich, aber ausreichend groß gewählt werden, so daß die beschriebenen Effekte nicht auftreten.

Aus den durchgeführten Untersuchungen lassen sich folgende Dimensionierungsregeln ableiten:

- Die Anzahl der Trainingsmuster K sollte größer sein als die Ordnung Q des Optimierungsproblems, damit die Zielwerte $\{d^k\}$ nicht nur gespeichert und reproduziert werden.
- Zur Vermeidung des Overlearning-Effektes sollte K im Bereich $3 \cdot Q$ bis $5 \cdot Q$ gewählt werden.
- Sind die Prädiktionsgewinne für die Testmenge deutlich geringer als für die Trainingsmenge, muß die Trainingsmenge so lange vergrößert werden, bis die Abbildungseigenschaften des dynamischen Systems in ausreichendem Maße repräsentiert werden.

5.2.3 Nichtlineare Prädiktion des Lorenz-Signals mit Gradienten-Verfahren

Der Vergleich der Leistungsfähigkeit des *OLL*-Verfahrens mit den in Kapitel 3 vorgestellten Gradientenverfahren wird ebenfalls am Beispiel des bereits mehrfach verwendeten 5:22:9:1 *MLP*-Netzes durchgeführt, wobei die Ausgangsdaten die aus der Trainings- und Testmenge sowie den Startwerten für die Netzgewichte bestehen, identisch zu denen des *OLL*-Verfahrens gewählt wurden.

Alle Verfahren mit Ausnahme des *OLL*-Verfahrens basieren auf dem in Abschnitt 3.2 (Seite 36) vorgestellten Modellalgorithmus für schrittweitenbasierte Optimierungsverfahren. Sie unterscheiden sich lediglich hinsichtlich der Bestimmung einer optimalen Suchrichtung und Schrittweite. Die in der Literatur am häufigsten eingesetzten Varianten des Verfahrens des steilsten Abstiegs verwenden jeweils den negativen Gradienten Gl.(3.25) als Suchrichtung und verzichten auf die eindimensionale Optimierung (Liniensuche) entlang der Suchrichtung. Das *BP*-Verfahren wurde in der Variante gemäß Gl.(3.55) mit einer festen Schrittweite von $\eta_{BP} = 0,01$ und einer Momentum-Konstanten $\alpha_M = 0,8$ verwendet. Auf einen Momentum-Term wurde bei dem *BD*-Verfahren hier verzichtet und dieses entsprechend den Gleichungen (3.49) und (3.50) implementiert, wobei für die Heuristik zur Anpassung der Schrittweite die in der Literatur [55] üblichen Werte von $\gamma = 1,05$ und $\beta = 0,5$ gewählt wurden.

Alle eingesetzten *CG*- und *QN*-Verfahren verwenden in der ersten Hauptiteration als Suchrichtung den negativen Gradienten und bedienen sich zur Bestimmung der optimalen Schrittweite einer eindimensionalen Minimierung (Liniensuche) entlang der jeweiligen Suchrichtung analog zu Gl.(3.51). Detaillierte Ausführungen zur Art der Realisierung, zu den Konvergenzeigenschaften und zur Leistungsfähigkeit finden sich in der Arbeit von Arz [5]. Basierend auf den dort erzielten Erkenntnissen wurde bei sämtlichen im Rahmen dieser Arbeit vorgestellten Untersuchungen der Parameter, welcher die Abbruchbedingung des Liniensuch-Algorithmus gemäß Gl.(3.53) steuert, zu $\varepsilon=0,001$ gewählt.

Zur Optimierung der 5:22:9:1 *MLP*-Netze wurden neben dem *BP*- und *BD*-Verfahren 4 Varianten der klassischen *CG*-Verfahren von Polak-Ribière und Fletcher-Reeves, 3 Varianten modernerer *CG*-Verfahren und als Variante der *QN*-Verfahren das von Broyden, Fletcher, Goldfarb und Shanno angegebene *BFGS*-Verfahren im Vergleich zum *OLL*-Verfahren untersucht.

Die Verfahren von Polak-Ribière und Fletcher-Reeves ergeben sich aus Gl.(3.58) und unterscheiden sich lediglich hinsichtlich der Bestimmung der Koeffizienten β^{I-1} (vgl. Gl.(3.63) und Gl.(3.64)). Diese Verfahren wurden sowohl ohne Restart-Strategie (*PR*, *FR*) als auch mit Restart-Strategie (*PRR*, *FRR*) eingesetzt, wobei als Restart-Richtung jeweils nach Q Hauptiterationen der negative Gradient als neue Suchrichtung vorgegeben wurde.

Als Beispiele für modernere *CG*-Verfahren kamen das gedächtnislose *BFGS*-Verfahren (*LQN*) entsprechend den Gleichungen (3.73, 3.74 und 3.75) sowie das Verfahren der skalierten konjugierten Gradienten nach Shanno [131] gemäß Gl.(3.77) sowohl mit Restart (*SCR*) als auch ohne Restart (*SCG*) zum Einsatz.

Die mit den verschiedenen Verfahren erzielten Ergebnisse sind in Abbildung 5.9 (a) und (b) wiedergegeben. Zur Reduzierung des enormen Rechenzeitbedarfs wurde bei diesen Untersuchungen eine Testmenge mit nur 1 000 Mustern verwendet, da hier der Vergleich

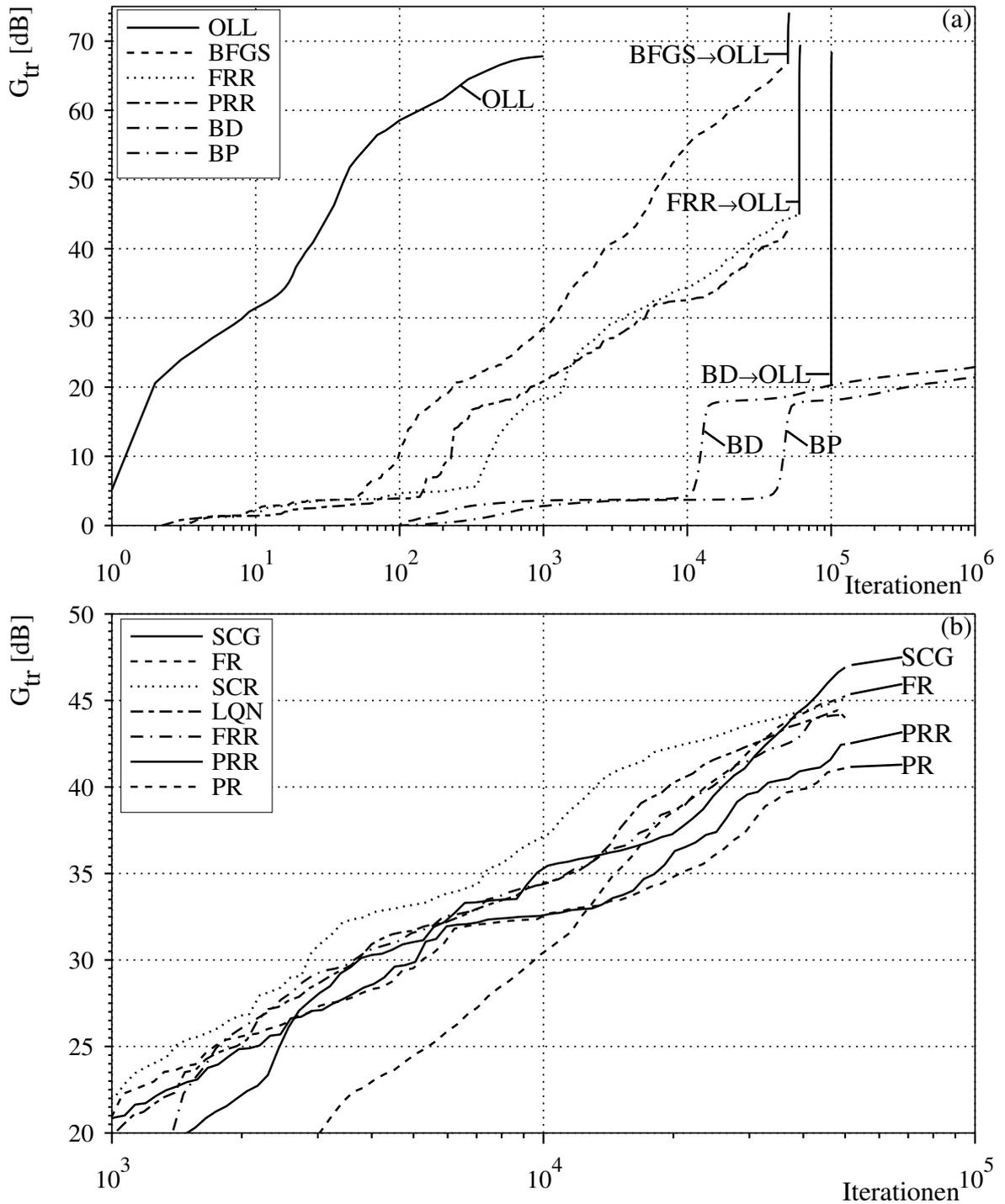


Abbildung 5.9: Gegenüberstellung der Prädiktionsgewinne für die Trainingsmenge über dem Iterationsverlauf für die Optimierung eines 5:22:9:1 *MLP*-Netzes zur Prädiktion des Lorenz-Signals mit verschiedenen Optimierungsverfahren.

(a): Vergleich zwischen *OLL*-, *BFGS*-, *CG*- und *BP*-Verfahren

(b): Ausschnitt aus (a) zum Vergleich aller betrachteten *CG*-Verfahren

der unterschiedlichen Optimierungsverfahren und nicht die Optimierung der Abbildungs-

eigenschaften der *MLP*-Netze für die Testmenge im Vordergrund stand. Die Darstellung wurde im doppelt logarithmischen Maßstab vorgenommen³², da ansonsten ein Vergleich des Iterationsverlaufes für die verschiedenen Optimierungsverfahren in einem Diagramm nicht sinnvoll darstellbar war.

In Abbildung 5.9 (a) werden zunächst die Prädiktionsergebnisse des *OLL*-Verfahrens mit denen des *BFGS*-, *FRR*-, *PRR*-, *BD*- und des *BP*-Verfahrens verglichen. Wie bereits in Abschnitt 5.1 festgestellt wurde, weist das *OLL*-Verfahren eine um mehrere Größenordnungen bessere Konvergenzgeschwindigkeit im Vergleich zu dem *BD*- und *BP*-Verfahren auf. Mit diesen wird erst nach 122 000 bzw. 478 000 Iterationen ein Prädiktionsgewinn erzielt, welcher mit dem *OLL*-Verfahren bereits nach nur 2 Iterationen erreicht wird. Auch nach 10^6 Iterationen weisen beide Verfahren einen deutlich niedrigeren Prädiktionsgewinn als das *OLL*-Verfahren nach 3 Iterationen auf. Wie erwartet ist das *BD*-Verfahren aufgrund der Adaption der Schrittweite dem *BP*-Verfahren etwas überlegen.

Mit dem *BP*- und *BD*-Verfahren werden jedoch im Vergleich zu den klassischen *CG*-Verfahren von Polak-Ribière und Fletcher-Reeves mit Restart-Strategie deutlich geringere Prädiktionsgewinne erreicht. Diese *CG*-Verfahren erzielen nach 60 000 bzw. 50 000 Iterationen Prädiktionsgewinne von 44,9 *dB* bzw. 42,5 *dB*, welche allerdings immer noch um mehrere Größenordnungen unterhalb derer liegen, die mit dem *OLL*-Verfahren erzielt werden.

Die Prädiktionsgewinne, die unter Verwendung aller 7 untersuchten *CG*-Verfahren erzielt wurden, sind zur genaueren Differenzierung als Ausschnitt in Abbildung 5.9 (b) wiedergegeben. Diese Darstellung zeigt, daß alle *CG*-Verfahren vergleichbare Prädiktionsgewinne im Bereich von 41 *dB* bis 47 *dB* liefern. Die genaueren Zahlenwerte sind in Tabelle 5.4 angegeben. Hier schneidet als bestes das *SCG*-Verfahren vor dem *FR*-, *SCR*-, *LQN*-, *FRR*-, *PRR*- und dem *PR*-Verfahren ab. Die Unterschiede sind jedoch nur gering, und die hier erzielte Reihenfolge konnte bei anderen Untersuchungen nicht bestätigt werden, sondern variierte von Untersuchung zu Untersuchung.

Zusammenfassend ist festzustellen, daß sich das *BFGS*-Verfahren mit einem Prädiktionsgewinn von 66,9 *dB* nach 50 000 Iterationen im Vergleich zu den *CG*-Verfahren als deutlich überlegen erweist und auch als einziges der untersuchten Verfahren einen Prädiktionsgewinn erzielt, der in der Größenordnung des mit dem *OLL*-Verfahren nach 1 000 Iterationen erreichten von 67,8 *dB* liegt. Das *BFGS*-Verfahren liefert bei allen durchgeführten Untersuchungen im Vergleich zu den anderen in Kapitel 3 vorgestellten Verfahren ca. 20 *dB* höhere Prädiktionsgewinne. Nur mit diesem Verfahren gelingt es auch bei anderen Untersuchungen die Prädiktionsgewinne, die mit dem *OLL*-Verfahren erzielt werden, wenigstens annähernd zu erreichen. Allerdings wurde hierzu immer eine um einen Faktor von ca. 100 höhere Anzahl von Iterationen benötigt.

³²Der Prädiktionsgewinn in [dB] ist über der logarithmierten Anzahl an Iterationen aufgetragen.

Aufgrund des enormen Rechenaufwandes war es nicht möglich, mit den *CG*-, *BD*- oder *BP*-Verfahren eine ausreichende Anzahl an Iterationen durchzuführen, um nachzuweisen, daß es auch mit diesen Verfahren prinzipiell möglich ist, ähnlich hohe Prädiktionsgewinne wie mit dem *OLL*-Verfahren zu erzielen. Weiterhin traten bei einigen Untersuchungen sowohl mit dem *BFGS*- als auch mit den *CG*-Verfahren numerische Probleme auf, die zum vorzeitigen Abbruch der Iteration führten, wobei die dann erreichten Prädiktionsgewinne noch wesentlich unter denen des *OLL*-Verfahrens lagen. Diese Abbrüche traten immer während der Liniensuche auf, wenn die Bestimmung einer U-Konfiguration zum Start des Einschachtelungsprozesses im Rahmen der Rechengenauigkeit nicht mehr möglich war (vgl. *Arz* [5]). Diese Schwierigkeiten bei der Liniensuche weisen auf lokale Minima oder zumindest extrem flache Gebiete der Fehleroberfläche hin und traten auch bei den in Abbildung 5.9 (a) vorgestellten Untersuchungen mit dem *BFGS*-Verfahren nach 50 635 Iterationen bei einem Prädiktionsgewinn von 66,9 dB auf.

Das *BFGS*-Verfahren führt, wie die anderen Gradientenverfahren auch, in sehr flachen Regionen der Fehleroberfläche, in denen in Suchrichtung nur extrem kleine Änderungen der Gütefunktion auftreten, verfahrensbedingt nur sehr kleine Änderungen der Netzgewichte durch, oder bricht ab, wenn, wie beschrieben, keine U-Konfiguration zum Start der Liniensuche mehr gebildet werden kann. Aus diesem Grund ist die Konvergenzgeschwindigkeit der Gradientenverfahren in diesen Gebieten und damit auch in der Nähe lokaler oder absoluter Minima extrem gering. Das *OLL*-Verfahren ist besser an die spezielle Struktur der hier vorliegenden Optimierungsprobleme angepaßt, da es auf einem grundsätzlich unterschiedlichen Ansatz beruht und nicht von der lokalen Beschaffenheit der Fehleroberfläche abhängig ist, wie dies bei den *BFGS*-, *CG*- oder *BD*-Verfahren der Fall ist. Dieses Verhalten des *OLL*-Verfahrens wurde bereits in Abschnitt 5.1 beschrieben und in Abbildung 5.3 graphisch veranschaulicht. Um die Eigenschaften des *OLL*-Verfahrens in einem solchen Fall näher zu untersuchen, wurden die Netzgewichte, die nach 50 635 Iterationen des *BFGS*-Verfahrens berechnet wurden und zum Abbruch des Verfahrens führten, als Startwerte für das *OLL*-Verfahren verwendet und 1 000 Iterationen durchgeführt. Der Iterationsverlauf ist in Abbildung 5.9 (a) als durchgezogene Linie im Anschluß an den Verlauf des *BFGS*-Verfahrens wiedergegeben und mit *BFGS*→*OLL* bezeichnet. Obwohl das *BFGS*-Verfahren bei dieser Kombination der Netzgewichte abbricht, kann mit 1 000 Iterationen des *OLL*-Verfahrens der Prädiktionsgewinn sogar noch deutlich um 7,2 dB auf 74,1 dB gesteigert werden.

Von besonderem Interesse hierbei ist, daß dieses für die Gradientenverfahren extrem ungünstige Gebiet der Fehleroberfläche zu keinen numerischen Problemen bei Einsatz des *OLL*-Verfahrens führt, und daß es gelingt, dieses Gebiet zu verlassen und den Prädiktionsgewinn dabei noch erheblich zu steigern. Um zu untersuchen, wie groß die Änderungen der Netzgewichte im Verlauf der Optimierung mit dem *OLL*-Verfahren im Vergleich zum *BFGS*-Verfahren sind, werden im folgenden der Betrag des Vektors der Netzgewichte

sowie dessen Änderung im Laufe der Iteration betrachtet. Beide Verfahren wurden mit dem gleichen Gewichtsvektor \mathbf{w}^0 mit $|\mathbf{w}^0|=10,9$ gestartet, dessen einzelne Komponenten als gleichverteilte Zufallszahlen aus dem Bereich $[-1, 0 < 1, 0]$ vorgegeben wurden.

Bei Verwendung des *BFGS*-Verfahrens ändert sich der Betrag des Vektors der Netzgewichte nach der 50 000 Iteration auf $|\mathbf{w}_{BFGS}^{50000}|=83,41$ und nach der 50 635 Iteration auf $|\mathbf{w}_{BFGS}^{50635}|=83,43$. Der Differenzvektor ist ein Maß für die Größe der Änderungen der Netzgewichte für hier 635 Iterationen des *BFGS*-Verfahrens und hat den Betrag $|\mathbf{w}_{BFGS}^{50635} - \mathbf{w}_{BFGS}^{50000}|=0,6$. Die letzten Iterationen vor Abbruch des *BFGS*-Verfahrens lieferten sogar nur Beträge der Differenzvektoren in der Größenordnung von 10^{-4} bis 10^{-10} . Das *BFGS*-Verfahren führt hier nur sehr geringe Änderungen der Netzgewichte durch. Im Vergleich hierzu hat der Vektor der Netzgewichte bei gleichem Startvektor \mathbf{w}^0 nach 1 000 Iterationen des *OLL*-Verfahrens einen Betrag von $|\mathbf{w}_{OLL}^{1000}|=9,228$. Dieser ändert sich nach weiteren 1 000 Iterationen auf $|\mathbf{w}_{OLL}^{2000}|=12,058$, wobei der Differenzvektor einen Betrag von $|\mathbf{w}_{OLL}^{2000} - \mathbf{w}_{OLL}^{1000}|=5,296$ hat.

Bei Start des *OLL*-Verfahrens mit den nach 50 635 Iterationen des *BFGS*-Verfahrens ermittelten Netzgewichten ändert sich der Betrag des Vektors bereits nach einer Iteration des *OLL*-Verfahrens auf $|\mathbf{w}_{BFGS,OLL}^{50635,1}|=94,2$, wobei das *OLL*-Verfahren die einzelnen Netzgewichte bereits erheblich verändert, so daß der Differenzvektor bereits einen Betrag von $|\mathbf{w}_{BFGS,OLL}^{50635,1} - \mathbf{w}_{BFGS}^{50635}|=42,0$ aufweist. Nach 1 000 Iterationen des *OLL*-Verfahrens hat der Vektor der Netzgewichte einen Betrag von $|\mathbf{w}_{BFGS,OLL}^{50635,1000}|=22,048$.

Die Eigenschaft, bei der Änderung der Netzgewichte von der lokalen Beschaffenheit der Fehleroberfläche unabhängig zu sein, ermöglicht es dem *OLL*-Verfahren im Gegensatz zu den Gradientenverfahren, erheblich größere Änderungen der Netzgewichte vorzunehmen und trägt zu der sehr hohen Konvergenzgeschwindigkeit des *OLL*-Verfahrens bei. Weiterhin wird hierdurch die Gefahr des Abbruchs des Verfahrens aufgrund numerischer Probleme sowie das Festlaufen an den stationären Punkten der Fehleroberfläche mit relativ geringem Prädiktionsgewinn erheblich verringert. Die Konvergenz des *OLL*-Verfahrens ist dennoch gesichert, da in Lösungsnähe stets linearisiert wird.

Neben dem mit dem *BFGS*-Verfahren ermittelten *MLP*-Netz wurden auch die mit dem *FRR*-Verfahren nach 60 000 und dem *BD*-Verfahren nach 100 000 Iterationen ermittelten *MLP*-Netze als Startwerte für das *OLL*-Verfahren vorgegeben und jeweils 1 000 Iterationen durchgeführt. Die sich hieraus ergebenden Iterationsverläufe sind ebenfalls in Abbildung 5.9 (a) dargestellt und mit *FRR*→*OLL* bzw. *BD*→*OLL* bezeichnet. Bei Fortführung des *FRR*-Verfahrens ergibt sich ein Prädiktionsgewinn von 69,5 dB und bei Fortführung des *BD*-Verfahrens von 68,4 dB.

Diese drei Untersuchungen zeigen deutlich die überlegenen Eigenschaften des *OLL*-Verfahrens auch und besonders in Bereichen der Fehleroberfläche, wo die betrachteten Gradientenverfahren entweder aufgrund numerischer Probleme abbrechen oder wo deren

Konvergenzgeschwindigkeit aufgrund der lokalen Beschaffenheit der Fehleroberfläche bereits extrem gering geworden ist.

Verfahren	$G_{tr}[dB]$	Iterationen	Iterationen mit OLL	Iterations-Faktor	CPU-Faktor	Gesamt-Faktor
OLL	67,8	1000	1000	1,0	15,1	1,0
BFGS	66,7	50000	518	97	4,15	26,5
SCG	46,9	50000	36	1389	2,94	270
FR	45,3	50000	33	1515	2,68	269
SCR	45,1	50000	32	1563	2,85	295
LQN	44,6	50000	32	1563	2,90	300
FRR	44,9	60000	32	1875	2,67	332
PRR	42,5	50000	28	1786	2,65	313
PR	41,1	50000	26	1923	2,66	339
BD	20,6	122000	2	~61000	1,05	~4200
BP	20,6	478000	2	~239000	1,0	~16000

Tabelle 5.4: Vergleich des Rechenzeitaufwandes zur Optimierung eines 5:22:9:1 *MLP*-Netzes zur Prädiktion des Lorenz-Signals für verschiedene Optimierungsverfahren.

Die bisher durchgeführten Vergleiche der Leistungsfähigkeit der einzelnen Optimierungsverfahren basierten auf einem Vergleich der nach einer bestimmten Anzahl von Iterationen mit dem jeweiligen Verfahren erzielbaren Prädiktionsgewinne mittels Abbildung 5.9, lassen aber keinen direkten Vergleich der Verfahren bezüglich des erforderlichen Berechnungsaufwandes und damit des erzielbaren Geschwindigkeitsvorteils zu.

Da bei allen untersuchten Verfahren der Aufwand je Iteration nicht konstant ist³³ und auch nicht hinreichend genau abgeschätzt werden kann, wird hier als Näherung ein direkter Vergleich der im Mittel je Iteration benötigten Rechenzeit verwendet. Dieser wird als *CPU-Faktor* bezeichnet und ist relativ zum mittleren Rechenzeitaufwand je Iteration des *BP*-Verfahrens in Spalte 6 von Tabelle 5.4 angegeben. Weiterhin enthält die Tabelle für jedes Verfahren in Spalte 2 den erzielten Prädiktionsgewinn und in Spalte 3 die dafür benötigte Anzahl von Iterationen.

Die mit dem *OLL*-Verfahren erforderliche Anzahl an Iterationen zur Erzielung des gleichen Prädiktionsgewinns enthält die 4te Spalte. In Spalte 5 (*Iterations-Faktor*) ist das

³³Bei den Verfahren, die eine Liniensuche voraussetzen, ist der Aufwand je nach Anzahl der notwendigen Funktionswert- und Gradientenberechnungen für die einzelne Liniensuche u. U. sehr unterschiedlich. Das *OLL*-Verfahren führt Subiterationen durch, wenn der Wert der Gütefunktion nach Berechnung der neuen Netzgewichte nicht verbessert wird. Ein ähnlicher Ansatz wird beim *BD*-Verfahren verfolgt.

Verhältnis der benötigten Iterationen im Vergleich mit dem *OLL*-Verfahren angegeben. Die sich bei Verwendung des *OLL*-Verfahrens gegenüber dem jeweils betrachteten Optimierungsverfahren ergebende Rechenzeitreduktion ist in Spalte 7 (*Gesamt-Faktor*) ausgewiesen. Dieser Rechenzeitvorteil ergibt sich aus dem Produkt der Spalten 5 und 6, wobei der Wert für das *OLL*-Verfahren auf 1.0 normiert ist.

Der Vergleich dieser Ergebnisse bestätigt die bereits bei Auswertung der Abbildung 5.9, bezüglich der Konvergenzgeschwindigkeit, erhaltene Einteilung der untersuchten Verfahren in vier Gruppen. Das leistungsfähigste Verfahren ist das *OLL*-Verfahren, welches um den Faktor 26,5 schneller ist als das *BFGS*-Verfahren. Die sieben untersuchten Varianten der *CG*-Verfahren bilden die dritte Gruppe. Gegenüber diesen weist das *OLL*-Verfahren einen Geschwindigkeitsvorteil um den Faktor 269 bis 339 auf. Hier ist auch das *BFGS*-Verfahren mit Faktoren zwischen 10 und 13 deutlich überlegen. Gerade die in der Literatur am häufigsten zur Optimierung von *MLP*-Netzen eingesetzten *BD*- und *BP*-Verfahren erweisen sich sowohl bezüglich der Konvergenzgeschwindigkeit als auch im Hinblick auf die erzielte Qualität der Nachbildung der Abbildungseigenschaften des zugrundeliegenden Lorenz-Systems am ungeeignetsten.

5.3 Prädiktionsergebnisse für das chaotische Mackey-Glass-Zeitsignal

Die in den vorhergehenden Abschnitten 5.1 und 5.2 vorgestellten Untersuchungen beschränkten sich auf die Prädiktion der Lorenz-Signale $\mathbf{L}_{x,0.01}$ und $\mathbf{L}_{x,0.1}$. In diesem Abschnitt werden die experimentellen Ergebnisse beschrieben, die bei der Untersuchung der Signale der im Anhang A.2 vorgestellten chaotischen Mackey-Glass-Differentialgleichung ermittelt wurden. Um die Ergebnisse mit anderen Veröffentlichungen vergleichbar zu gestalten, wurden die Parameter, welche die Eigenschaften des Systems maßgeblich beeinflussen, wie in Anhang A.2 beschrieben, gewählt und zwei Varianten der Mackey-Glass-Differentialgleichung mit $\Delta T=17$ und $\Delta T=30$ betrachtet. Bei allen Untersuchungen der chaotischen Mackey-Glass-Signale \mathbf{MG}_{17} gemäß Gl.(A.9) und \mathbf{MG}_{30} gemäß Gl.(A.10) wurde, wenn nicht besonders angegeben, eine Trainingsmenge von 2000 und eine Testmenge von 100 000 Mustern verwendet. Aus den Zeitsignalen rekonstruierte Trajektorien dieser beiden Systeme wurden bereits in Abbildung 2.6 für die Einbettungsbedingung $d_E = 3$ und $\tau = 4$ dargestellt.

Entsprechend den Betrachtungen anderer Veröffentlichungen [21, 132, 5] wurden skalare Prädiktoren für die Prädiktionsweite $pw=6$ optimiert. Damit wird der jeweils 6 Abtastzeitpunkte in der Zukunft liegende Abtastwert der Signale vorhergesagt, und die Prädiktionsaufgabe stellt sich wieder entsprechend den Gleichungen (2.41) und (2.43) dar.

5.3.1 Mackey-Glass-Zeitsignal für $\Delta T=17$

Der zeitliche Verlauf sowie einige charakteristische Kenngrößen des Mackey-Glass-Signals \mathbf{MG}_{17} sind im Anhang in Abbildung A.2 wiedergegeben. Im Vergleich zu dem bisher untersuchten Lorenz-Signal $\mathbf{L}_{x,0.1}$ weist dieses Signal eine deutlichere Periodizität³⁴ sowie erheblich stärkere lineare Bindungen zwischen den Abtastwerten auf, was insbesondere aus dem Verlauf der normierten Autokorrelationsfunktion hervorgeht.

5.3.1.1 Lineare Prädiktion des \mathbf{MG}_{17} -Signals

Zunächst werden auch für das \mathbf{MG}_{17} -Signal die Prädiktionsgewinne linearer Prädiktoren der Ordnung $P=1$ bis $P=100$ bei Variation der Einbettungskonstanten zwischen $\tau=1$ und $\tau=4$ ermittelt. Wie erwartet steigt in allen Fällen der Prädiktionsgewinn für die Trainingsmenge ab einer Prädiktorordnung von $P=10$ bis $P=20$ nicht mehr wesentlich an und erreicht einen maximalen Wert von $G_{tr}=20,7 \text{ dB}$ bei $P=100$ und $\tau=1$. Auch hier werden für die Einbettungskonstante $\tau=1$ deutlich bessere Prädiktionsgewinne als mit größeren Werten von τ erzielt. Diese liegen bei $15,9 \text{ dB}$ für $\tau=2$, bei $14,8 \text{ dB}$ für $\tau=3$ und bei $14,1 \text{ dB}$ für $\tau=4$. Die Prädiktionsgewinne für die Testmenge sind nur unwesentlich schlechter.

Aufgrund der stärkeren linearen Bindungen des \mathbf{MG}_{17} -Signals werden hier mit linearen Prädiktoren erheblich höhere Prädiktionsgewinne von maximal $G_{tr}=20,7 \text{ dB}$ im Vergleich zu $G_{tr}=3,9 \text{ dB}$ für das $\mathbf{L}_{x,0.1}$ -Signal erzielt (vgl. Abbildung 5.4).

5.3.1.2 Nichtlineare Prädiktion des \mathbf{MG}_{17} -Signals mit dem *OLL*-Verfahren

Im folgenden wird untersucht, inwieweit sich diese Ergebnisse durch den Einsatz nichtlinearer Prädiktoren auf Basis von *MLP*-Netzen verbessern lassen. Für die Untersuchungen des \mathbf{MG}_{17} -Signals wird der gleiche Ansatz verwendet wie zuvor für das $\mathbf{L}_{x,0.1}$ -Signal. Zur Bestimmung der optimalen Einbettungsparameter d_E und τ werden die in Tabelle 5.2 aufgeführten *MLP*-Netze eingesetzt und die Einbettungsdimension im Bereich $3 \leq d_E \leq 12$ sowie die Einbettungskonstante im Bereich $1 \leq \tau \leq 4$ variiert. Die hierbei für die Prädiktion des \mathbf{MG}_{17} -Signals erzielten Ergebnisse sind in den Abbildungen 5.10 (a) für die Trainingsmenge und in Abbildung 5.10 (b) für die Testmenge dargestellt.

Der Vergleich beider Abbildungen liefert ein bemerkenswertes Ergebnis. Im Gegensatz zu den Untersuchungen des Lorenz-Signals (vgl. Abbildung 5.5) sind hier die Prädiktionsgewinne G_{tr} für die Trainingsmenge nur unwesentlich höher als die Prädiktionsgewinne G_{te} für die Testmenge. Eine genauere Betrachtung der Zahlenwerte zeigt, daß für $\tau=1$ die

³⁴Der Verlauf des normierten Leistungsdichtespektrums des \mathbf{MG}_{17} -Signals beweist jedoch, daß es sich bei der betrachteten Lösung der Mackey-Glass-Differentialgleichung keinesfalls um eine periodische oder quasi-periodische Lösung handelt.

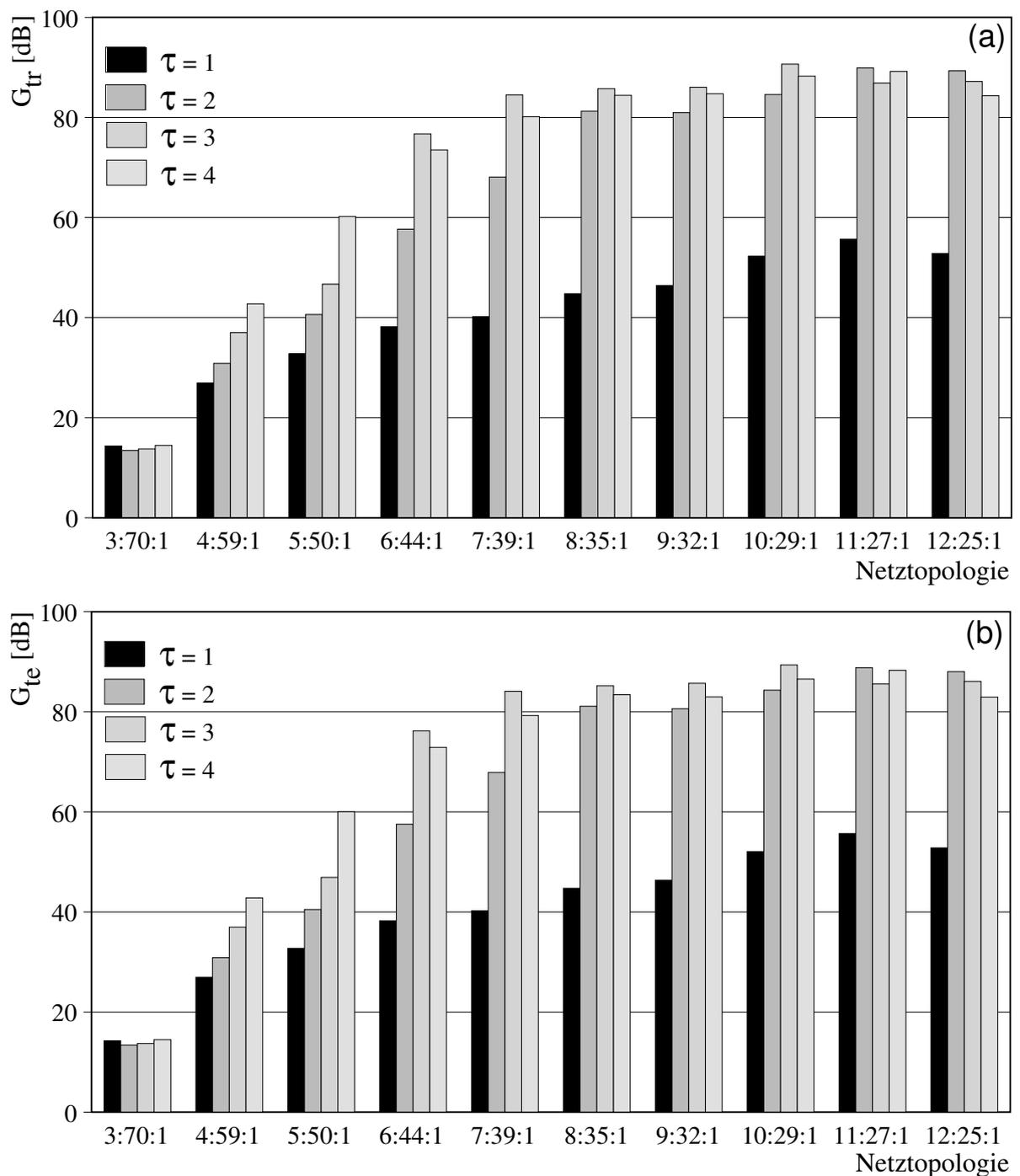


Abbildung 5.10: Vergleich der Prädiktionsergebnisse, die mit ausgewählten 3-lagigen *MLP*-Netzen mit annähernd konstanter Anzahl von 350 Netzgewichten für das chaotische Mackey-Glass-Signal \mathbf{MG}_{17} erzielt werden.

(a): Ergebnisse für die 2000 Muster umfassende Trainingsmenge

(b): Ergebnisse für die 100 000 Muster umfassende Testmenge

maximale Abweichung nur 0,2 dB beträgt, für größere Werte von τ im wesentlichen unter

1 dB liegt und nur für wenige Ausnahmen maximal 1,8 dB erreicht.

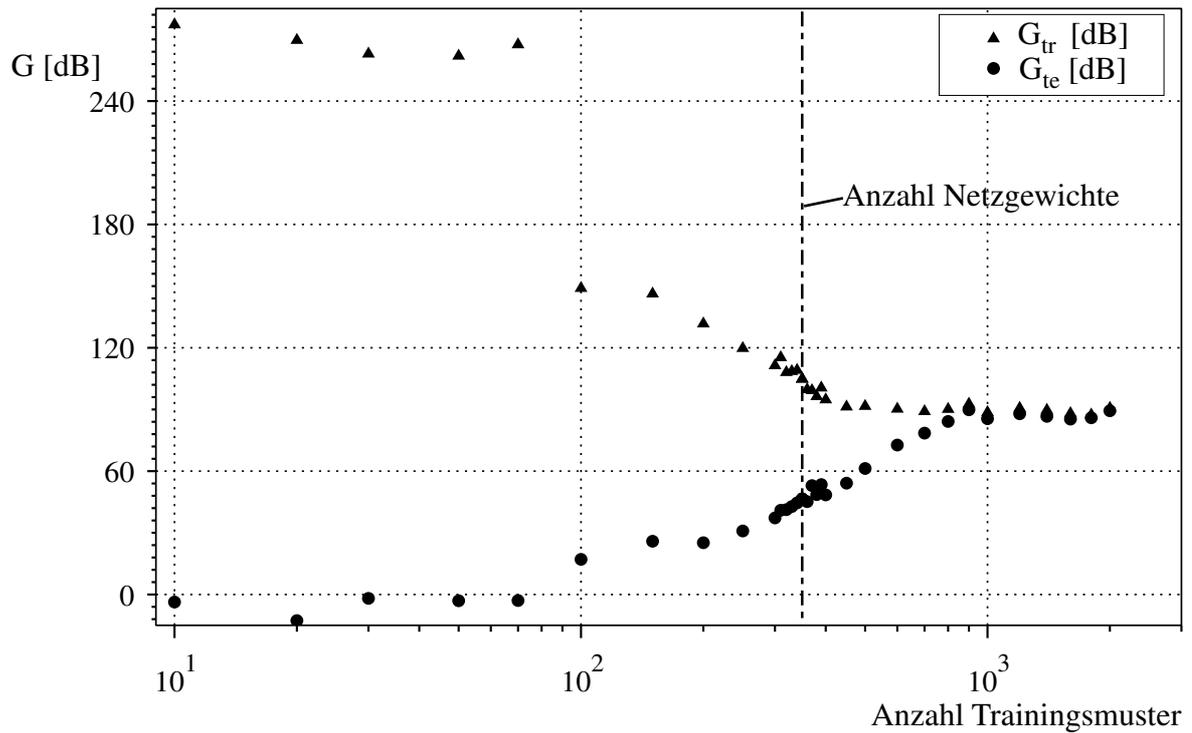


Abbildung 5.11: Vergleich der Prädiktionsgewinne für die Trainings- und Testmenge bei Variation der Anzahl an Trainingsmustern zwischen $K=10$ und $K=2\,000$ für die Prädiktion des MG_{17} -Signals mit einem 10:29:1 *MLP*-Netz.

Demzufolge repräsentieren die 2 000 Muster umfassenden Trainingsmengen bereits alle wesentlichen Abbildungseigenschaften der Mackey-Glass-Differentialgleichung, und die *MLP*-Netze bilden diese Eigenschaften sehr umfassend und genau nach. Der maximale Prädiktionsgewinn wird für das 10:29:1 Netz bei $\tau = 3$ erzielt und beträgt 90,7 dB für die Testmenge bzw. 89,4 dB für die Trainingsmenge und liegt damit wesentlich über den Werten, die für die Prädiktion des Lorenz-Signals erzielt wurden. Der notwendige Umfang der Trainingsmenge hängt demnach in erheblichem Maße von dem zu modellierenden System ab. Für das Lorenz-Signal $\mathbf{L}_{x,0.1}$ wurden erst durch mehr als 4 000 Trainingsmuster die Abbildungseigenschaften des Systems ausreichend repräsentiert. Welcher Umfang der Trainingsmenge für die Prädiktion des MG_{17} -Signals minimal notwendig ist wurde durch Variation der Anzahl der Trainingsmuster im Bereich zwischen $K=10$ und $K=2\,000$ unter Verwendung des 10:29:1 *MLP*-Netzes und Wahl von $\tau = 3$ untersucht. Die hierbei erzielten Prädiktionsgewinne für Trainings- und Testmenge nach i. allg. 1 000 Iterationen des *OLL*-Verfahrens sind in Abbildung 5.11 dargestellt, wobei die 100 000 Muster umfassende Testmenge nicht verändert wurde.

Auch bei dieser Untersuchung treten für den Fall, daß die Anzahl der für die Optimierung zur Verfügung stehender Daten (Trainingsmenge) geringer ist als die Ordnung des

Optimierungsproblems, welche für die eingesetzten 10:29:1 *MLP*-Netze $Q=349$ beträgt, die gleichen Effekte auf, wie sie bereits bei der Untersuchung des Lorenz-Signals (Abbildungen 5.7 und 5.8) beobachtet wurden. Die *MLP*-Netze speichern und reproduzieren die Trainingsdaten und erlangen nicht die Verallgemeinerungsfähigkeit, die zur Prädiktion nicht in der Trainingsmenge enthaltener Daten notwendig ist. Die Prädiktionsgewinne für die Trainingsmengen sind in diesen Fällen extrem hoch, die Werte für die Testmenge hingegen sehr gering und in einigen Fällen sogar negativ. Die vorgegebenen 1 000 Iterationen wurden für $K=10, 20, 30, 50$ und 70 wiederum nicht erreicht, da das *OLL*-Verfahren nach 2,12, 300, 400 bzw. 500 Iterationen mit Fall II.2.B.3 (siehe Seite 67) beendet wurde. Der Overlearning Effekt tritt bei diesen Untersuchungen für Trainingsmengen mit $K=350$ bis $K=600$ Mustern auf. Eine genügend große Trainingsmenge wird bereits mit 900 Trainingsmustern erreicht. Mit zunehmendem Umfang der Trainingsmenge ändern sich dann die Prädiktionsgewinne nur noch unwesentlich und sind für die Trainings- und Testmenge annähernd gleich. Zusammenfassend ist festzustellen, daß für das \mathbf{MG}_{17} -Signal bereits 900 Trainingsmuster die wesentlichen Abbildungseigenschaften der Mackey-Glass-Differentialgleichung in ausreichendem Maße repräsentieren.

Die Auswertung der Abbildung 5.10 bezüglich der optimalen Wahl für die Einbettungsparameter zeigt zunächst, daß die Prädiktionsgewinne ab einer Einbettungsdimension von $d_E=6$ stark gegenüber den mit $d_E=3$ bis $d_E=5$ erzielten Werten ansteigen. Dieses Ergebnis bestätigt erneut die Einbettungsbedingungen von Takens (Gl.(2.37) und Mañé (Gl.2.38)), wonach erst bei einer Rekonstruktionsdimension von $d_E > 2d_C + 1$ eine Einbettung des Attraktors garantiert ist und somit alle wesentlichen Eigenschaften des Systems repräsentiert sind. Diese Bedingung ist für den Attraktor der Mackey-Glass-Differentialgleichung mit $\Delta T=17$ und $d_C(\mathbf{MG}_{17})=2,1$ gemäß Gl.(A.11) bei $d_E \geq 6$ erfüllt.

Im Gegensatz zu den Untersuchungen des Lorenz-Systems zeigt sich bezüglich der optimalen Wahl der Einbettungskonstanten τ , daß für das \mathbf{MG}_{17} -Signal nicht der Wert von $\tau=1$ zu den besten Ergebnissen führt, sondern für verschiedene Einbettungsdimensionen unterschiedliche Werte von τ die höchsten Prädiktionsergebnisse liefern, und $\tau=1$ in keinem der Fälle optimal ist. Dieses Verhalten wird auf die Periodizität des Mackey-Glass-Signals zurückgeführt, die aus der Darstellung des Zeitsignals und dem Verlauf der normierten Autokorrelationsfunktion aus Abbildung A.2 deutlich wird. Diese weist im Darstellungsbereich bei Verschiebungen von $\Delta t=24$ und $\Delta t=75$ Minima, bei $\Delta t=49$ und $\Delta t=100$ Maxima sowie bei $\Delta t=12, \Delta t=38, \Delta t=62$ und $\Delta t=88$ Nullstellen auf.

Werden die in Abbildung 5.10 dargestellten Ergebnisse und die Periodizität des \mathbf{MG}_{17} -Signals über die zeitliche Ausdehnung der rekonstruierten Zustandsvektoren in Verbindung mit der Prädiktionsweite $pw=6$ in Zusammenhang gebracht, so lassen sich zusätzlich zur Einbettungsbedingung von Takens weitere Bedingungen für die Wahl vorteilhafter Einbettungsparameter angeben.

Zunächst sind in Tabelle 5.5 für die Untersuchungen aus Abbildung 5.10 die Einbettungsparameter d_E und τ sowie der jeweils erzielte Prädiktionsgewinn für die Trainingsmenge zusammengestellt. Zusätzlich enthält die Tabelle die Zeitspanne, die durch die jeweiligen rekonstruierten Zustandsvektoren und die Prädiktionsweite pw erfaßt wird. Diese *Fensterlänge* ermittelt sich zu $W=(d_E - 1)\tau + pw$.

d_E	4				5				6			
τ	1	2	3	4	1	2	3	4	1	2	3	4
W	9	12	15	18	10	14	18	22	11	16	21	26
$G_{tr}[dB]$	26,9	30,9	37,0	42,7	32,8	40,6	46,7	60,2	38,2	57,7	76,7	73,5
d_E	7				8				9			
τ	1	2	3	4	1	2	3	4	1	2	3	4
W	12	18	24	30	13	20	27	34	14	22	30	38
$G_{tr}[dB]$	40,2	68,1	84,5	80,1	44,8	81,3	85,7	84,4	46,4	80,9	86,0	84,7
d_E	10				11				12			
τ	1	2	3	4	1	2	3	4	1	2	3	4
W	15	24	33	42	16	26	36	46	17	28	39	50
$G_{tr}[dB]$	52,3	84,6	90,7	88,3	55,7	89,9	86,9	92,1	52,8	89,3	87,2	84,3

Tabelle 5.5: Vergleich der Prädiktionsgewinne für die Trainingsmenge bei Variation der Einbettungsparameter d_E und τ und der sich mit der Prädiktionsweite ergebenden erfaßten Zeitspanne W des **MG₁₇**-Signals.

Bis zu einer Einbettungsdimension von $d_E=5$ werden bei Variation von τ mit $\tau=4$ die besten Ergebnisse erzielt, wobei die Prädiktionsgewinne mit zunehmender Fensterlänge und Einbettungsdimension steigen, aber noch nicht sehr hohe Werte annehmen. Die Fensterlänge erreicht mit maximal $W=22$ für $d_E=5$ und $\tau=4$ noch nicht den Wert des ersten Minimums der normierten AKF. Für Werte von $6 \leq d_E \leq 10$ werden für $\tau=3$ und bei $d_E=11$ und $d_E=12$ für $\tau=2$ die höchsten Prädiktionsgewinne erzielt. Die Ergebnisse für $\tau=2$, $\tau=3$ und $\tau=4$ unterscheiden sich jedoch ab einer Einbettungsdimension $d_E=8$ nur noch wenig. Im Vergleich hierzu sind die Prädiktionsgewinne für $\tau=1$ erheblich geringer, wobei die Fensterlänge zwischen $W=13$ und $W=17$ variiert und damit deutlich unter dem Wert des ersten Minimums der normierten AKF von $\Delta t=24$ bleibt.

Aus diesen Betrachtungen in Verbindung mit der jeweils angegebenen Fensterlänge W ergibt sich folgende Hypothese für die Wahl günstiger Einbettungsparameter:

Um eine Rekonstruktion des Attraktors eines dynamischen Systems aus einem skalaren Zeitsignal zu erzeugen, welche die Abbildungseigenschaften des Systems in ausreichendem Maße repräsentiert und damit hohe Prädiktionsgewinne ermöglicht, sind die Einbettungsparameter d_E und τ so zu wählen, daß die durch die rekonstruierten Zustandsvektoren in Verbindung mit der Prädiktionsweite erfaßte Zeitspanne (Fensterlänge) in der Größenordnung des ersten Minimums der normierten AKF des skalaren Zeitsignals liegt. Die Einbettungsbedingungen von Takens Gl.(2.37) und Mañé (2.38) sind zusätzlich zu erfüllen.

Zur Erweiterung der Untersuchungen auf größere Bereiche der Fensterlänge von bis zu $W=150$, was in etwa dem dritten Maximum der normierten AKF entspricht, werden im folgenden die Ergebnisse vorgestellt, die mit 4-lagigen *MLP*-Netzen der Topologie $d_E:10:10:1$ erzielt wurden. Dabei wurde die Einbettungsdimension, und damit die Anzahl der Knoten in der Eingangslage, im Bereich $2 \leq d_E \leq 19$ variiert. Die Anzahl der Knoten in den 2 verdeckten Lagen und in der Ausgangslage wurde konstant gehalten. Damit wächst die Anzahl Q der Netzgewichte von $Q=151$ im Falle des $2:10:10:1$ auf $Q=322$ für das $19:10:10:1$ *MLP*-Netz an. Zusätzlich wurden für die Einbettungskonstante Werte im Bereich $1 \leq \tau \leq 8$ vorgegeben. Die bei diesen Untersuchungen erzielten Prädiktionsgewinne für die Trainingsmengen sind in Abbildung 5.12 zusammengefaßt. Die Prädiktionsgewinne für die Testmenge liegen um maximal 3 dB unter diesen Werten.

Die Ergebnisse der Auswertung von Abbildung 5.10 werden auch für diese Untersuchungen bestätigt. Für Einbettungsdimensionen $d_E \geq 6$ werden hohe Prädiktionsgewinne erzielt, wenn gleichzeitig die Fensterlänge W in der Größenordnung des ersten Minimums der normierten AKF des \mathbf{MG}_{17} -Signals liegt.

Um dieses Verhalten genauer zu untersuchen, sind in Tabelle 5.6 für die Einbettungsdimensionen $d_E \geq 6$ die Fensterlängen W für die Werte von τ zusammengestellt, mit denen die drei höchsten Werte des Prädiktionsgewinns erzielt werden.

Bis zu $d_E=13$ werden die maximalen Prädiktionsgewinne bei der jeweiligen Einbettungsdimension für das τ erzielt, welches ein W zur Folge hat, das in der Größenordnung des ersten Minimums der normierten AKF liegt. Deutlich schlechtere Prädiktionsgewinne werden für Werte von τ erzielt, die zu einer Fensterlänge führen, die stärker von $\Delta t=24$ abweicht oder deutlich kleiner als $\Delta t=24$ ist.

Für Einbettungsdimensionen $d_E \geq 14$ nähern sich die Fensterlängen für $\tau=1$ dem Wert von $\Delta t=24$ an und sind für $\tau > 1$ deutlich größer. Dennoch werden für $\tau=1$ deutlich schlechtere Ergebnisse als mit anderen Werten von τ erzielt. Ein Vergleich des Iterationsverhaltens des

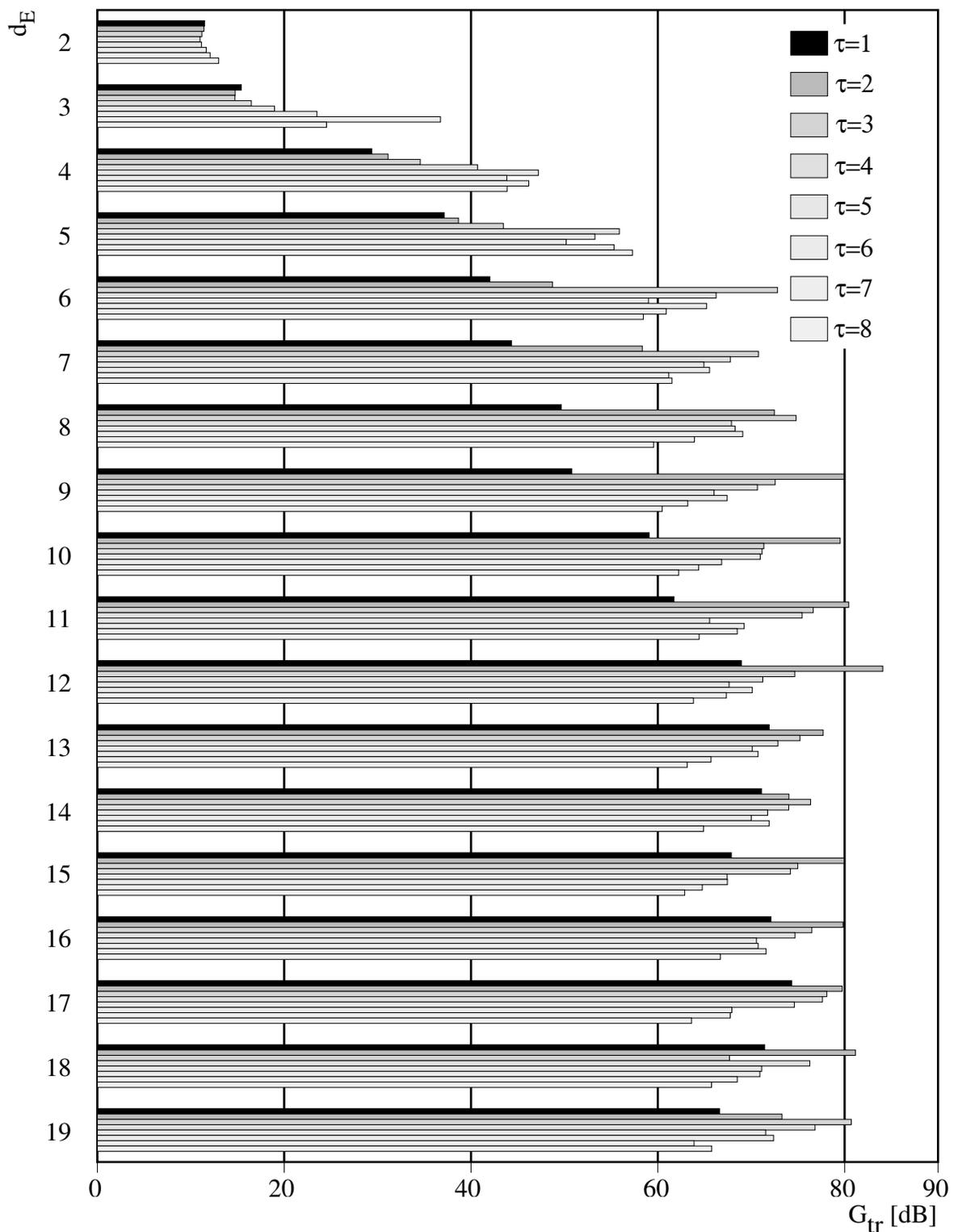


Abbildung 5.12: Vergleich der Prädiktionsgewinne, die mit *MLP*-Netzen der Topologie $d_E:10:10:1$ für die Prädiktion des \mathbf{MG}_{17} -Signals bei Variation der Einbettungsparameter d_E und τ erzielt wurden.

OLL-Verfahrens bei unterschiedlichen Werten von τ für Einbettungsdimensionen $d_E \geq 6$

d_E	$G_{tr,1} [dB]$	τ_1	W_1	$G_{tr,2} [dB]$	τ_2	W_2	$G_{tr,3} [dB]$	τ_3	W_3
6	72,8	3	21	66,2	4	26	65,2	6	36
7	70,8	3	24	67,8	4	30	65,5	6	42
8	74,8	3	27	72,5	2	20	69,1	6	48
9	79,9	2	22	72,6	3	30	70,7	4	38
10	79,5	2	24	71,4	3	33	71,2	4	42
11	80,4	2	26	76,6	3	36	75,4	4	46
12	84,1	2	28	74,7	3	39	71,2	4	50
13	77,7	2	30	75,2	3	42	72,9	4	54
14	76,4	3	45	74,0	2	32	74,0	4	58
15	79,9	2	34	75,0	3	48	74,2	4	62
16	79,8	2	36	76,5	3	51	74,7	4	66
17	79,7	2	38	78,1	3	54	77,6	4	70
18	81,2	2	40	76,3	4	74	71,4	1	23
19	80,7	3	60	76,8	4	78	73,3	2	42

Tabelle 5.6: Vergleich der Werte der Einbettungskonstante und Fensterlänge, die bei der jeweiligen Einbettungsdimension mit *MLP*-Netzen der Topologie $d_E:10:10:1$ für das **MG**₁₇-Signal die drei höchsten Werte des Prädiktionsgewinns für die Trainingsmenge ergeben.

liefert das unerwartete Ergebnis, daß das *OLL*-Verfahren für $\tau=1$ eine wesentlich höhere Anzahl an Subiterationen durchführt als bei allen anderen Werten von τ . Im Mittel führt das *OLL*-Verfahren 5 % bis 10 % Subiterationen durch, d. h. bei Optimierung der verdeckten Lagen wird der Fall II.2.B (vgl. Seite 66) in 5 % bis 10 % der Fälle durchlaufen. Für die Wahl von $\tau=1$ fallen hier jedoch im Mittel 20 % bis 30 % Subiterationen an, was einer Steigerung um den Faktor 3 bis 5 entspricht. Dieses Verhalten führt zu den schlechten Ergebnissen, die bei $\tau=1$ im Vergleich zu anderen Werten von τ erzielt werden. Außer für $d_E=14$ und $d_E=19$ führt die Wahl von $\tau=2$ und damit Fensterlängen, die die kleinste Abweichung von $\Delta t=24$ aufweisen, zu den besten Ergebnissen. Für $d_E=14$ bzw. $d_E=19$ liegt die Fensterlänge in der Größenordnung der zweiten ($\Delta t=38$) bzw. dritten ($\Delta t=62$) Nullstelle der normierten AKF.

Ähnliche Betrachtungen wurden auch von Caputo, Malraison und Atten [14] für die optimale Wahl der Einbettungsparameter zur Bestimmung der Korrelationsdimension chaotischer Zeitsignale mit dem Verfahren von Grassberger und Procaccia angestellt. Dort wird eine Fensterlänge vorgeschlagen, die in der Größenordnung des Produktes aus fraktaler Dimension und Periodizität des Signals liegt. Im Fall des **MG**₁₇-Signals müßte die Fensterlänge demzufolge ca. 100 betragen, was jedoch für den hier beschriebenen Anwendungsfall nicht bestätigt werden kann.

5.3.1.3 Nichtlineare Prädiktion des MG_{17} -Signals mit Gradienten-Verfahren

Wie zuvor für das Lorenz-Signal wird im folgenden auch für das MG_{17} -Signal ein Vergleich der Leistungsfähigkeit des OLL -Verfahrens mit den im Rahmen dieser Arbeit betrachteten Gradientenverfahren am Beispiel des 10:29:1 MLP -Netzes bei Wahl der Einbettungskonstanten $\tau = 3$ durchgeführt, da hierfür die besten Ergebnisse erzielt werden konnten. Die Untersuchungen mit den Gradientenverfahren wurden analog zu den in Abschnitt 5.2.3 vorgestellten durchgeführt. Insbesondere wurden für die Optimierungsverfahren die gleichen Parameter verwendet. Lediglich der Umfang der Trainingsmenge wurde bei 2 000 Mustern belassen und nicht wie zuvor auf 1 000 Muster reduziert. In Abbildung 5.13 sind die mit den verschiedenen Verfahren erzielten Prädiktionsgewinne für die Trainingsmenge über dem Iterationsverlauf dargestellt. Abbildung 5.13 (b) gibt wiederum nur einen Ausschnitt zum besseren Vergleich der untersuchten CG -Verfahren wieder.

Der in Abbildung 5.13 (a) durchgeführte Vergleich der Ergebnisse des OLL -Verfahrens mit denen des $BFGS$ -, FRR -, PRR -, BD - und BP -Verfahrens bestätigt die bereits in Abschnitt 5.2.3 vorgenommene Beurteilung der Verfahren. Das OLL -Verfahren erweist sich dem $BFGS$ -Verfahren um zwei und den CG -Verfahren um drei Zehnerpotenzen in Bezug auf die Konvergenzgeschwindigkeit überlegen. Weiterhin wird deutlich, daß die BD - und BP -Verfahren für diese Art Optimierungsprobleme ungeeignet sind, da sie erst nach 10^6 Iterationen die Ergebnisse des OLL -Verfahrens nach zwei bzw. einer Iteration erreichen. Die nach jeweils 50 000 Iterationen mit den CG -Verfahren erzielten und in Abbildung 5.13 (b) dargestellten Ergebnisse sind wieder alle in der gleichen Größenordnung und liegen zwischen $55,3 \text{ dB}$ für das SCG -Verfahren und $61,1 \text{ dB}$ für das FRR -Verfahren.

Eine Aussage darüber, ob mit den Gradientenverfahren die gleichen maximalen Prädiktionsgewinne wie mit dem OLL -Verfahren erzielbar sind, kann nicht erfolgen, da aufgrund des extrem hohen Rechenzeitbedarfs die dafür erforderliche Anzahl an Iterationen mit diesen Verfahren nicht durchgeführt werden konnte. Bei den für das MG_{17} -Signal durchgeführten Untersuchungen war es daher nicht möglich, den Prädiktionsgewinn von $90,7 \text{ dB}$ zu erreichen, der nach 1 000 Iterationen des OLL -Verfahrens erzielt wurde. Das $BFGS$ -Verfahren lieferte nach 50 000 Iterationen mit $80,1 \text{ dB}$ das beste Ergebnis, wobei jedoch ein gegenüber den Untersuchungen des Lorenz-Signals gravierend langsameres Konvergenzverhalten des Verfahrens bei der Liniensuche festgestellt wurde. Bei der Optimierung des 5:22:9:1 MLP -Netzes zur Prädiktion des $L_{x,0.1}$ -Signals waren für die 50 000 durchgeführten Iterationen ca. 170 000 Berechnungen des Funktionswerts und ca. 78 000 Berechnungen des Gradienten notwendig. Dieses entspricht 3,4 Funktionswert- sowie 1,6 Gradientenberechnungen je Iteration. Im hier betrachteten Fall der Optimierung des 10:29:1 MLP -Netzes zur Prädiktion des MG_{17} -Signals waren jedoch für 50 000 Iterationen ca. 732 000 Funktionswert- und ca. 217 000 Gradientenberechnungen erforderlich. Dies entspricht mit 14,6 Funktionswertberechnungen je Iteration einer Steigerung um

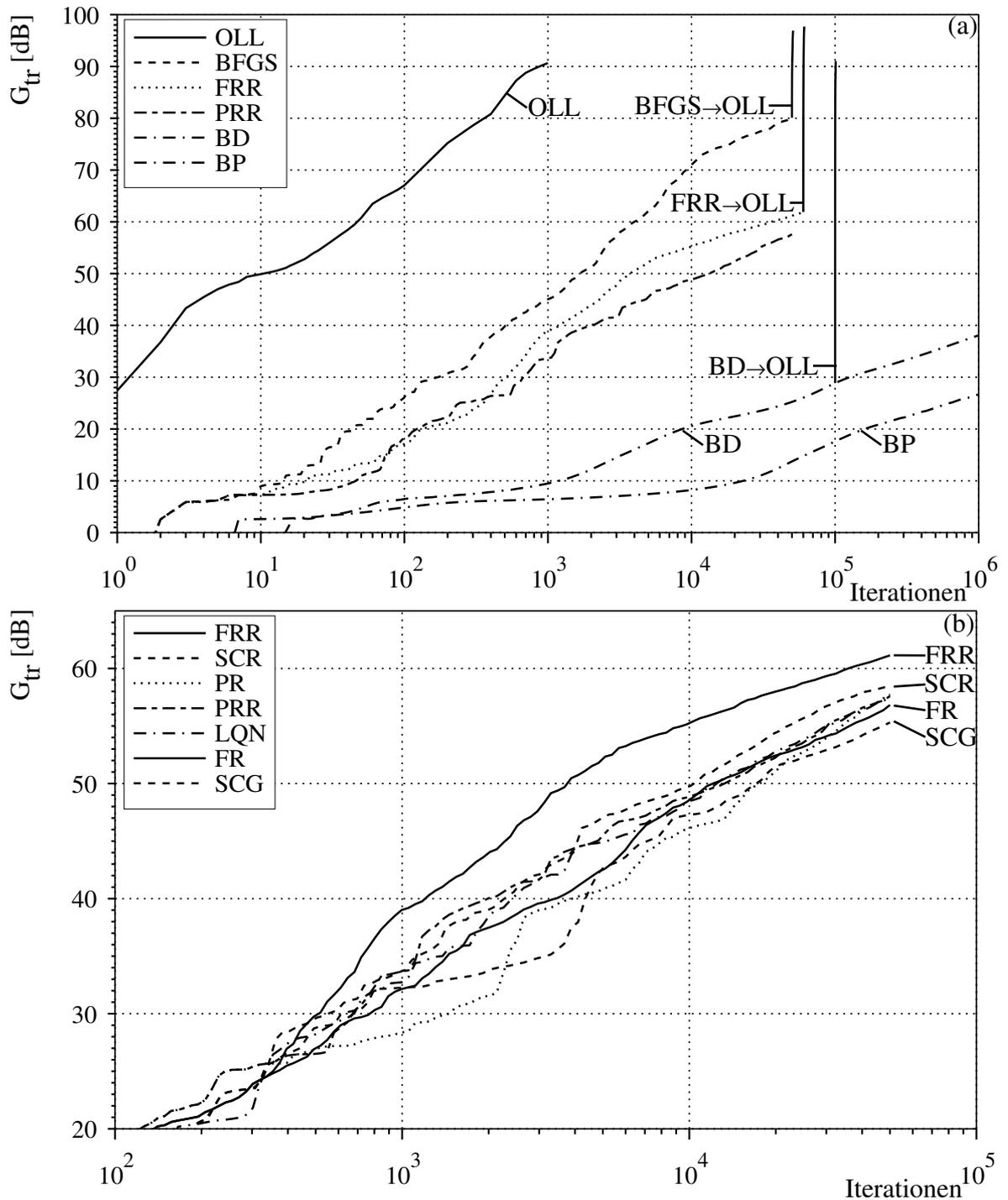


Abbildung 5.13: Gegenüberstellung der Prädiktionsgewinne für die Trainingsmenge über dem Iterationsverlauf für die Optimierung eines 10:29:1 MLP -Netzes zur Prädiktion des Mackey-Glass-Signals MG_{17} mit verschiedenen Optimierungsverfahren.

(a): Vergleich zwischen OLL -, $BFGS$ -, CG - und BP -Verfahren

(b): Ausschnitt aus (a) zum Vergleich aller betrachteten CG -Verfahren

den Faktor 4,3 und mit 4,3 Gradientenberechnungen je Iteration einer Steigerung um

den Faktor 2,7. Von den 732 000 Funktionswertberechnungen waren ca. 142 000 für die Bestimmung einer U-Konfiguration und ca. 590 000 für die Liniensuche selbst notwendig. Ein ähnliches Verhalten wurde auch bei allen *CG*-Verfahren beobachtet und weist auf eine im Vergleich zu den in Abschnitt 5.2.3 durchgeführten Optimierungen wesentlich komplexere Beschaffenheit der Fehleroberfläche hin, was zu dem in Tabelle 5.7 (Spalte 6) angegebenen, im Vergleich zu den *BD*- und *BP*-Verfahren und den Werten aus Tabelle 5.4 wesentlich höheren Berechnungsaufwand je Iteration führt.

Verfahren	$G_{tr}[dB]$	Iterationen	Iterationen mit OLL	Iterations-Faktor	CPU-Faktor	Gesamt-Faktor
OLL	90,7	1000	1000	1,0	44,7	1,0
BFGS	80,1	50000	373	134	24,2	72,5
FRR	61,1	50000	51	980	5,82	128
SCR	58,5	50000	41	1220	5,85	160
PR	57,7	50000	38	1316	5,9	174
PRR	57,6	50000	37	1351	5,73	173
LQN	57,4	50000	36	1389	5,81	181
FR	56,8	50000	33	1515	5,69	193
SCG	55,3	50000	28	1786	6,04	241
BD	38,0	1000000	2	~500000	1,05	~12000
BP	26,7	1000000	1	~1000000	1,0	~22000

Tabelle 5.7: Vergleich des Rechenzeitaufwandes zur Optimierung eines 10:29:1 *MLP*-Netzes zur Prädiktion des Mackey-Glass-Signals mit $\Delta T=17$ für verschiedene Optimierungsverfahren.

Auch bei der Optimierung mit dem *OLL*-Verfahren fällt eine auf ca. 26 % gestiegene Anzahl an Subiterationen auf. Dieses führt neben der geänderten Topologie des Netzes ebenfalls für das *OLL*-Verfahren zu einem im Vergleich zu Tabelle 5.4 deutlich gestiegenen Berechnungsaufwand je Iteration.

Die beschriebenen Schwierigkeiten des *BFGS*- und der *CG*-Verfahren bei der Liniensuche traten zu Beginn der Optimierung nicht auf, wurden aber gravierend bei höheren Prädiktionsgewinnen und damit zunehmender Komplexität der Fehleroberfläche in der Nähe stationärer Punkte. Wie bereits für das $L_{x,0.1}$ -Signal wurden auch hier die mit dem *BFGS*-, *FRR*- und *BD*-Verfahren nach 50 000, 60 000 bzw. 10^5 Iterationen berechneten *MLP*-Netze als Startwerte für das *OLL*-Verfahren vorgegeben und 1 000 Iteration durchgeführt. Die hierbei erzielten Ergebnisse sind als durchgezogene Linien im Anschluß an die Ergebnisse des jeweiligen Gradientenverfahrens in Abbildung 5.13 (a) dargestellt und mit *BFGS*→*OLL*, *FRR*→*OLL* bzw. *BD*→*OLL* gekennzeichnet. Das beste Ergebnis wird bei

der weiteren Optimierung des mit dem *FRR*-Verfahrens berechneten *MLP*-Netzes mit einem Prädiktionsgewinn von $97,7 \text{ dB}$ für die Trainingsmenge erreicht, der sogar noch um $7,0 \text{ dB}$ über dem liegt, welcher zuvor mit dem *OLL*-Verfahren erzielt wurde. Auch die weitere Optimierung der mit dem *BFGS*- und *BD*-Verfahren bestimmten Netzgewichte durch 1000 Iterationen des *OLL*-Verfahrens führt mit $96,9 \text{ dB}$ und $91,0 \text{ dB}$ zu sehr guten Ergebnissen, die ebenfalls noch oberhalb jener Werte liegen, die mit dem *OLL*-Verfahren bei Start mit zufällig gewählten Netzgewichten erzielt werden. Die Prädiktionsgewinne für die Testmenge liegen um maximal $2,8 \text{ dB}$ unter diesen Werten. Diese Untersuchung zeigt ein weiteres Mal die überlegenen Konvergenzeigenschaften des *OLL*-Verfahrens auch in sehr komplexen Bereichen der Fehleroberfläche, in denen bei Verwendung der Gradientenverfahren numerische Schwierigkeiten auftreten und daher deren Konvergenzgeschwindigkeit sehr gering ist.

Für einen abschließenden Vergleich sind in Tabelle 5.7 die mit den einzelnen Optimierungsverfahren erzielten Ergebnisse nochmals als Zahlenwerte zusammengefaßt. Diese werden ergänzt durch den relativ zum *BP*-Verfahren notwendigen Rechenaufwand je Iteration sowie die insgesamt mit dem *OLL*-Verfahren gegenüber den anderen Verfahren erzielte Reduktion des Rechenaufwandes.

Dieses Ergebnis unterscheidet sich in einigen Aspekten deutlich von dem Vergleich, der für das $\mathbf{L}_{x,0.1}$ -Signal angestellt wurden. Relativ zum *OLL*-Verfahren ist das *BFGS*-Verfahren mit einem um den Faktor 72,5 höheren Rechenaufwand deutlich langsamer geworden. Auch im Vergleich zu den *CG*-Verfahren ist der Vorteil des *BFGS*-Verfahren deutlich geringer geworden. Von diesen erweist sich das *FRR*-Verfahren mit einem Faktor von 128 als mit Abstand bestes und das *SCG*-Verfahren mit dem Faktor 241 als mit Abstand schlechtestes. Relativ zum *OLL*-Verfahren ist der Nachteil der *CG*-Verfahren gegenüber den Untersuchungen des Lorenz-Signals, bei denen die *CG*-Verfahren einen um Faktoren zwischen 269 und 339 höheren Rechenaufwand erforderten, geringer geworden. Im Gegensatz zu Feststellungen anderer Veröffentlichungen [131, 90, 5] sind generelle Aussagen über generelle Vor- bzw. Nachteile einzelner *CG*-Verfahren gegenüber anderen *CG*-Verfahren auf Basis der hier durchgeführten Untersuchungen nicht möglich.

Das hier verwendete chaotische Zeitsignal \mathbf{MG}_{17} wurde auch von Lapedes und Farber [71, 72], Farmer und Sidorowich [30, 31] sowie Day und Davenport [21] untersucht und als Referenzsignal verwendet. Ein direkter Vergleich der oben beschriebenen Prädiktionsergebnisse ist jedoch nur mit den von Day und Davenport angegebenen Ergebnissen möglich, da in [71, 72] und [30, 31] andere Prädiktionsweiten als $pw = 6$ untersucht wurden. Von Day und Davenport wird eine neue Netzstruktur vorgestellt, die neben den zu optimierenden Netzgewichten für jedes Gewicht zusätzlich eine zu optimierende Zeitverzögerung einführt. Das zur Prädiktion des \mathbf{MG}_{17} -Signals eingesetzte Netz verfügte über 150 Netzverbindungen mit adaptiven Gewichten und Zeitverzögerungen. Die nach $5 \cdot 10^7$ Iterationen des im *Online*-Modus (vgl. Seite 41) betriebenen Verfahrens erzielten

Prädiktionsgewinne liegen bei den zwei dort untersuchten Varianten bei ca. $42,5 \text{ dB}$ und $46,0 \text{ dB}$. Demgegenüber wird mit dem *OLL*-Verfahren und den hier verwendeten Einbettungsbedingungen und Netztopologien ein Prädiktionsgewinn von maximal $90,7 \text{ dB}$ erreicht. Die wesentlichen Gründe für diese extrem unterschiedlichen Ergebnisse sind:

- Eine ungünstige Wahl der Einbettungsdimension d_E
- Eine ungünstige Wahl der Einbettungskonstanten τ
- Eine ungünstige Wahl der Netztopologie
- Aufgrund der geringen Konvergenzgeschwindigkeit und des daraus resultierenden enormen Rechenzeitbedarfs kann eine genügend große Anzahl Iterationen nicht durchgeführt werden.
- Die geringe Konvergenzgeschwindigkeit hat zur Folge, daß mit steigender Anzahl an Iterationen die Steigerung des Prädiktionsgewinns in jeder Iteration extrem gering wird und fälschlicherweise auf Konvergenz des Verfahrens geschlossen wird. Es sei daran erinnert, daß die in Abbildung 5.13 dargestellten Prädiktionsgewinne über der logarithmierten Anzahl an Iterationen aufgetragen sind. Wäre dieses nicht der Fall, wäre der Konvergenzverlauf der Gradientenverfahren bei hoher Anzahl an Iterationen extrem flach und es läge nahe auf Konvergenz der Verfahren zu schließen, wenn die mit dem *OLL*-Verfahren erzielbaren Prädiktionsgewinne dieses nicht bereits ausschließen würden!

5.3.2 Mackey-Glass-Zeitsignal für $\Delta T=30$

Ein weiteres, oft in der Literatur [71, 72, 30, 31, 91, 92, 93, 128] zu Vergleichszwecken herangezogenes chaotisches Zeitsignal stellt die Lösung der Mackey-Glass-Differentialgleichung (Gl.(A.8)) für den Parameter $\Delta T=30$ dar. Dieses dynamische System besitzt mit $d_C(\mathbf{MG}_{30})=3,54$ im Vergleich zur Mackey-Glass-Differentialgleichung bei $\Delta T=17$ und zum Lorenz-System eine deutlich höhere fraktale Dimension und wird aus diesem Grunde hier als drittes Testsignal verwendet.

5.3.2.1 Lineare Prädiktion des \mathbf{MG}_{30} -Signals

Ähnlich wie das \mathbf{MG}_{17} -Signal weist auch das \mathbf{MG}_{30} -Signal eine Periodizität und relativ starke lineare Bindungen auf, wie aus der Darstellung des Zeitsignals und der normierten AKF in Abbildung A.3 deutlich wird. In Folge dieser übereinstimmenden Eigenschaften unterscheiden sich die Prädiktionsgewinne, die mit linearen Prädiktoren der Ordnung $P=1$ bis $P=100$ bei Variation der Einbettungskonstanten zwischen $\tau=1$ und $\tau=4$ für dieses Signal erzielt werden, nur unwesentlich von den Ergebnissen der Untersuchung des

MG_{17} -Signals. Mit $G_{tr}=20,5\text{ dB}$ wird für eine Prädiktorordnung von $P=100$ bei $\tau=1$ der maximale Prädiktionsgewinn erzielt. Für Werte von $\tau=2$, $\tau=3$ und $\tau=4$ werden ebenfalls für $P=100$ Prädiktionsgewinne von $15,3\text{ dB}$, $13,0\text{ dB}$, $11,7\text{ dB}$ für die Trainingsmenge erreicht.

5.3.2.2 Nichtlineare Prädiktion des MG_{30} -Signals mit dem *OLL*-Verfahren

Aufgrund der höheren fraktalen Dimension des MG_{30} -Signals wird die Untersuchung nichtlinearer Prädiktoren auf Basis von *MLP*-Netzen mit annähernd konstanter Anzahl von $Q\approx 350$ Netzgewichten auf einen größeren Variationsbereich der Einbettungsdimension von $3\leq d_E\leq 20$ erweitert.

Hier kamen ausschließlich 3-lagige *MLP*-Netze zur Anwendung. Die Topologie der einzelnen Netze sowie die jeweilige Gesamtanzahl der Netzgewichte und der relativ zu einem 3:70:1 Netz erforderliche Rechenaufwand pro Iteration ist in Tabelle 5.2 angegeben. Zur Optimierung der Netzgewichte wurden jeweils 1 000 Iterationen mit dem *OLL*-Verfahren durchgeführt. Die mit den optimierten *MLP*-Netzen für das MG_{30} -Signal bei einer Prädiktionsweite von $pw=6$ erzielten Prädiktionsgewinne für die jeweils 2 000 Muster umfassenden Trainingsmengen sind in Abbildung 5.14 und für die 100 000 Muster der Testmenge in Abbildung 5.15 graphisch dargestellt.

Besonders bei hohen Prädiktionsgewinnen treten deutliche Unterschiede zwischen den Ergebnissen für die Trainingsmenge und denen für die Testmenge auf. Diese sind zwar nicht in der Größenordnung wie es für das Lorenz-Signal der Fall war, jedoch mit ca. 3 bis 11 dB deutlich größer als für das MG_{17} -Signal. Dieses Verhalten weist darauf hin, daß die Trainingsmengen mit jeweils 2 000 Mustern einen zu geringen Umfang besaßen. Hier stand jedoch die Wahl der Einbettungsbedingungen und die generelle Eignung der *MLP*-Netze zur Prädiktion des MG_{30} -Signals im Vordergrund.

Gemäß den Einbettungsbedingungen von Takens und Mañé wird für das MG_{30} -Signal ab einer Rekonstruktionsdimension von $d_E=9$ eine Einbettung sichergestellt. Bei den in Abbildung 5.14 dargestellten Ergebnissen wird jedoch bereits ab $d_E=8$ eine erhebliche Verbesserung im Vergleich zu kleineren Werten von $3\leq d_E\leq 7$ verzeichnet. Dieser Sachverhalt wird zum einen darauf zurückgeführt, daß gemäß Gl.(2.37) und Gl.(2.37) die Rekonstruktionsdimension nur geringfügig größer als 8 sein muß (aber ganzzahlig), zum anderen darauf, daß auch für kleinere Werte der Rekonstruktionsdimension eine Einbettung vorliegen kann, solange diese größer als die Dimension des Systems ist.

Eine weitere erhebliche Verbesserung der Prädiktionsgewinne wird für Einbettungsdimensionen $d_E\geq 11$ erzielt, wobei wiederum unterschiedliche Werte von τ zu den höchsten Prädiktionsgewinnen führen. Zur Beurteilung dieses Verhaltens wurden die in Abbildung 5.14 dargestellten Prädiktionsgewinne für Einbettungsdimensionen $d_E\geq 8$ nach Größe sortiert

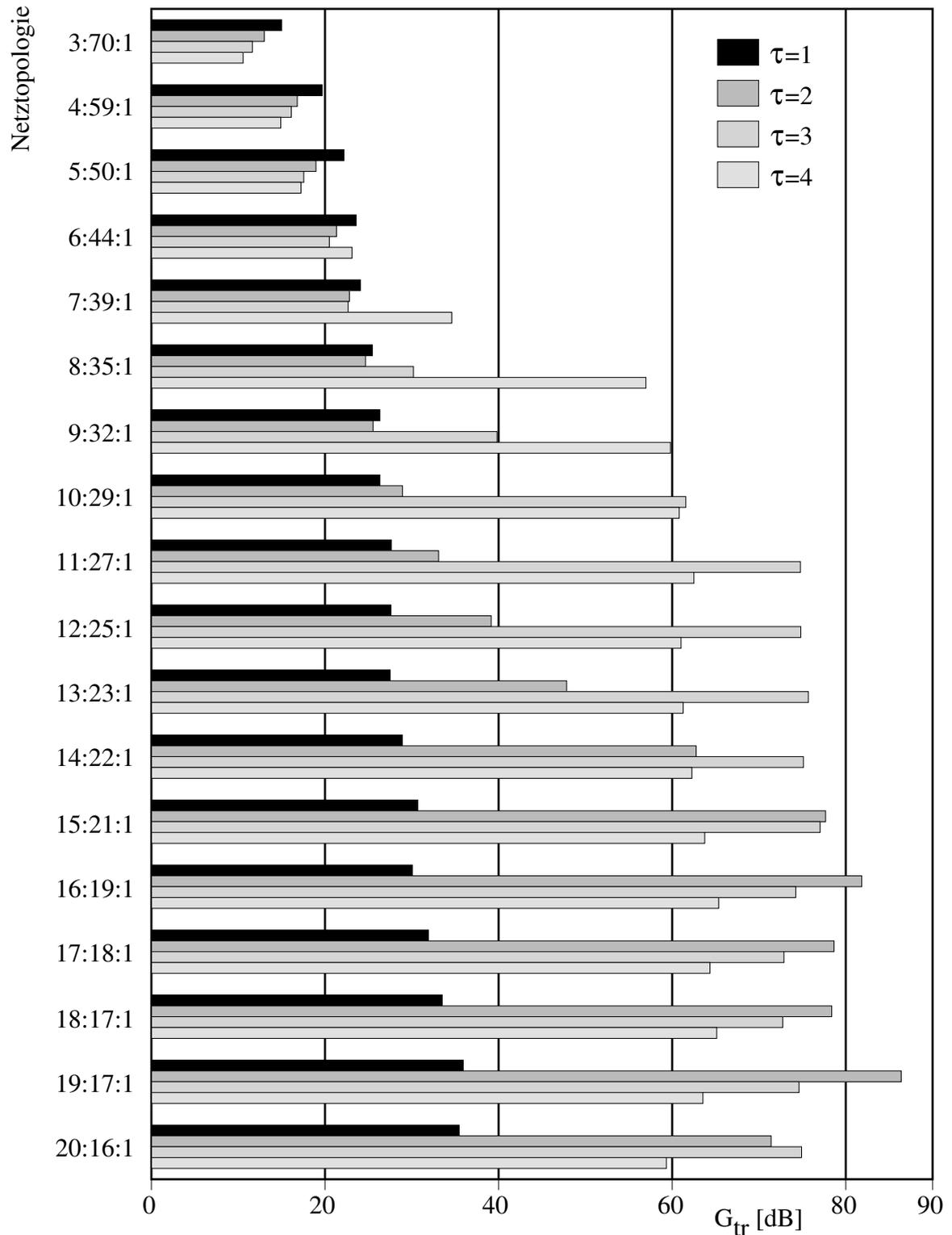


Abbildung 5.14: Vergleich der Prädiktionsgewinne G_{tr} , die mit 3-lagigen *MLP*-Netzen mit annähernd konstanter Anzahl von $Q \approx 350$ Netzgewichten für die Prädiktion des **MG**₃₀-Signals bei Variation der Einbettungsparameter d_E und τ für die jeweilige Trainingsmenge erzielt wurden.

und mit den zugehörigen Werten der Einbettungskonstanten τ und der daraus resultie-

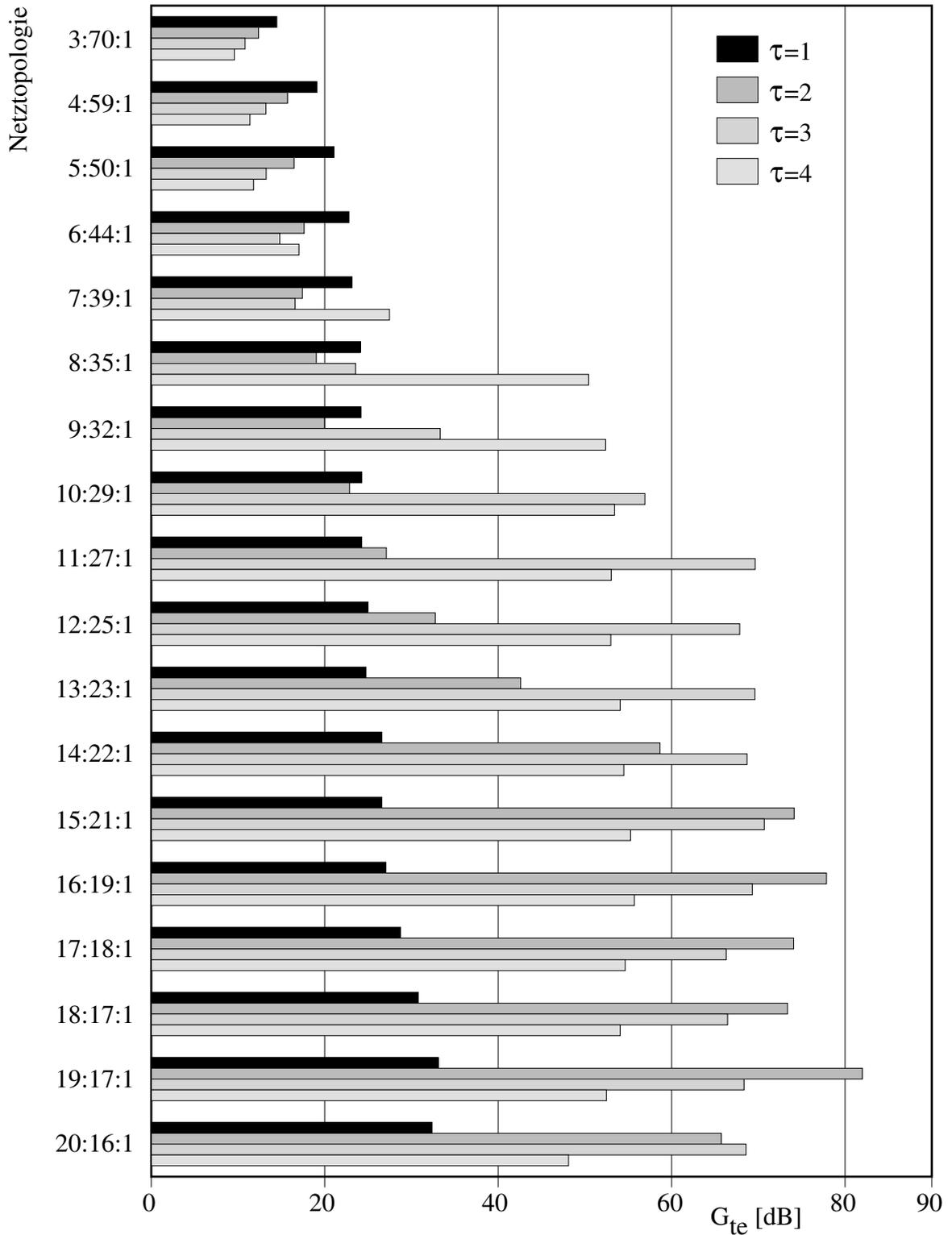


Abbildung 5.15: Vergleich der Prädiktionsgewinne G_{te} , die mit 3-lagigen *MLP*-Netzen mit annähernd konstanter Anzahl von $Q \approx 350$ Netzgewichten für die Prädiktion des **MG**₃₀-Signals bei Variation der Einbettungsparameter d_E und τ für die jeweilige Testmenge erzielt wurden.

renden Fensterlänge W in Tabelle 5.8 zusammengefaßt.

d_E	$G_{tr,1} [dB]$	τ_1	W_1	$G_{tr,2} [dB]$	τ_2	W_2	$G_{tr,3} [dB]$	τ_3	W_3	$G_{tr,4} [dB]$	τ_4	W_4
8	60,0	4	34	30,2	3	27	25,5	1	13	24,7	2	20
9	59,8	4	38	39,8	3	30	26,3	1	14	25,5	2	22
10	61,6	3	33	60,8	4	42	28,9	2	24	26,3	1	15
11	74,8	3	36	62,5	4	46	33,1	2	26	27,7	1	16
12	74,8	3	39	61,0	4	50	39,2	2	28	27,6	1	17
13	75,7	3	42	61,3	4	54	47,8	2	30	27,5	1	18
14	75,1	3	45	62,7	2	32	62,3	4	58	28,9	1	19
15	77,7	2	34	77,0	3	48	63,7	4	62	30,7	1	20
16	81,8	2	36	74,2	3	51	65,4	4	66	30,1	1	21
17	78,6	2	38	72,9	3	54	64,4	4	70	31,9	1	22
18	78,4	2	40	72,7	3	57	65,1	4	74	33,5	1	23
19	86,4	2	42	74,6	3	60	63,5	4	78	35,9	1	24
20	74,9	3	63	71,4	2	44	59,3	4	82	35,4	1	25

Tabelle 5.8: Vergleich der Werte der Einbettungskonstanten und Fensterlänge, die bei der jeweiligen Einbettungsdimension mit *MLP*-Netzen für das **MG**₃₀-Signal die vier höchsten Werte des Prädiktionsgewinns für die Trainingsmenge ergeben.

Für die Einbettungsdimensionen $d_E=8$ und $d_E=9$ stellt $\tau=4$ die günstigste Wahl dar, da diese Fensterlängen von $W=34$ bzw. $W=38$ zur Folge hat, die in der Größenordnung des ersten Minimums von $\Delta t=37$ der normierten AKF des **MG**₃₀-Signals liegen (vgl. Abbildung A.3). Für andere Werte von τ sind die Fensterlängen deutlich kleiner und die Prädiktionsgewinne wesentlich niedriger, obwohl die Einbettungsbedingungen von Takens und Mañé erfüllt sind.

Für Einbettungsdimensionen im Bereich $10 \leq d_E \leq 14$ werden mit $\tau=3$ die höchsten Prädiktionsgewinne bei Fensterlängen von $W=33$ bis $W=45$ erzielt. Der sprunghafte Anstieg des Prädiktionsgewinns bei $d_E=11$ mit $\tau=3$ ist verbunden mit einer Fensterlänge von $W=36$, die nahezu dem ersten Minimum der AKF bei $\Delta t=37$ entspricht, und bei Werten von $d_E=12$, $d_E=13$ und $d_E=14$ für $\tau=3$ die geringste Abweichung aufweist. Dieses Verhalten setzt sich auch für Einbettungsdimensionen im Bereich $15 \leq d_E \leq 19$ fort, wo $\tau=2$ die günstigste Wahl darstellt, da für $\tau=1$ die Fensterlängen noch deutlich kleiner und für $\tau=3$ bzw. $\tau=4$ bereits wesentlich größer als $\Delta t=37$ sind. Lediglich die Ergebnisse für $d_E=20$ weichen von diesem Verhalten ab, wobei allerdings beachtet werden muß, daß sich die Prädiktionsgewinne nicht wesentlich unterscheiden und die Fensterlänge mit Werten von $W=44$ und $W=63$ deutlich größer als $\Delta t=37$ sind.

Zusammenfassend ist festzustellen, daß die in Abschnitt 5.3.1 aufgestellten Bedingungen zur Wahl günstiger Einbettungsparameter auch durch die Ergebnisse dieser Untersuchungen bestätigt werden.

Ein direkter Vergleich der hier erzielten Prädiktionsergebnisse für das **MG**₃₀-Signal bei

der Prädiktionsweite $pw=6$ kann nur mit den von Lapedes und Farber [71, 72] sowie Sanger [128] vorgestellten Untersuchungsergebnissen erfolgen, da in anderen Veröffentlichungen [30, 31, 91, 92, 93] entweder abweichende Prädiktionsweiten verwendet oder die Parameter der Untersuchung nicht vollständig angegeben wurden.

Von Lapedes und Farber wird ein 6:10:10:1 *MLP*-Netz und die Einbettungskonstante $\tau=6$ zur Prädiktion des \mathbf{MG}_{30} -Signals eingesetzt. Der für die Prädiktionsweite $pw=6$ erzielte Gewinn kann mit ca. 30 dB bis 32 dB einer Abbildung der Veröffentlichung [71] entnommen werden. Sanger stellt in [128] einen neuen Typ neuronaler Netze vor, welcher eine baumartige Topologie aufweist, die sich in der Lernphase dynamisch an die zu modellierenden Daten anpaßt. Als ein Anwendungsbeispiel wird die Prädiktion des \mathbf{MG}_{30} -Signals untersucht, und ein Prädiktionsgewinn von ca. 32 dB für die Prädiktionsweite $pw=6$ angegeben.

Im Vergleich zu diesen Ergebnissen werden bei den hier durchgeführten Untersuchungen für Einbettungsdimensionen im Bereich von $11 \leq d_E \leq 20$ Prädiktionsgewinne zwischen 75 dB und 86 dB erzielt. Der maximal erzielte Gewinn ist mit 86 dB um ca. 50 dB höher als die in der Literatur angegebenen Werte. Dieses entspricht einem Verhältnis der Prädiktionsfehlerleistungen von 1 zu 10^5 . Auf die Gründe, die hier zu diesen wesentlich besseren Ergebnissen führen, wurde bereits in Abschnitt 5.3.1 eingehend eingegangen. Betrachtet man die Abbildungen 5.14 und 5.15, so wird deutlich, daß im Falle des \mathbf{MG}_{30} -Signals die Wahl der Einbettungsdimension zu $d_E=6$ ungeeignet ist.

5.4 Prädiktionsergebnisse für das chaotische Hénon-Zeitsignal

Die bisher untersuchten chaotischen Zeitsignale waren entweder Lösungen oder eine einzelne Komponente der Lösungen der Zustandsgleichungen zeitkontinuierlicher dynamischer Systeme. Als Beispiel für ein zeitdiskretes dynamisches System mit chaotischem Verhalten wird im folgenden die 2-dimensionale Hénon-Abbildung betrachtet, welche entsprechend Gl.(A.13) durch zwei Differenzgleichungen beschrieben wird. Das im folgenden betrachtete Hénon-Signal \mathbf{H}_x (Gl.A.16) entspricht der x-Komponente der Lösung dieses Gleichungssystems. Wie dem Zeitsignal und der normierten AKF des \mathbf{H}_x -Signals aus Abbildung A.4 zu entnehmen ist, weist dieses chaotische Signal nahezu keine Korrelation zwischen aufeinanderfolgenden Werten auf. Das Leistungsdichtespektrum ist sehr breitbandig, nahezu konstant und erinnert eher an ein Rauschsignal.

Wie der Verlauf der normierten AKF vermuten läßt, sind die Prädiktionsgewinne, die mit linearen Prädiktoren erzielbar sind, äußerst gering. Es wurden wiederum lineare Prädiktoren der Ordnung $P=1$ bis $P=100$ bei Variation der Einbettungskonstanten im Bereich $1 \leq \tau \leq 4$ berechnet und die Prädiktionsgewinne für Trainings- und Testmenge bestimmt. Der maximale Wert in Höhe von $G_{tr}=1,40$ dB wurde für $P=100$ bei $\tau=1$ erreicht und ist damit sehr gering im Vergleich zu den Ergebnissen, die für das \mathbf{MG}_{17} - und \mathbf{MG}_{30} -Signal

erzielt wurden. Für größere Werte von τ nimmt der Prädiktionsgewinn weiter ab und beträgt für einen Prädiktor der Ordnung $P=100$ bei $\tau=4$ nur noch $G_{tr}=0,53 \text{ dB}$. Lineare Prädiktoren sind demzufolge zur Prädiktion des Hénon-Zeitsignals absolut ungeeignet.

Entgegen dem MG_{17} - und MG_{30} -Signal weist das H_x -Signal keine Periodizität und nahezu keine linearen Bindungen zwischen aufeinanderfolgenden Werten auf. Weiterhin entspricht die y -Komponente bis auf einen Faktor der um einen Iterationsschritt verschobenen x -Komponente. Aus diesen Eigenschaften läßt sich bereits im Vorgriff auf die anschließend diskutierten Untersuchungsergebnisse schließen, daß für das H_x -Signal bei Wahl der Einbettungsbedingungen zu $d_E=2$ und $\tau=1$ die besten Prädiktionsergebnisse erzielbar sind, da der so rekonstruierte Attraktor bis auf die Ausdehnung in y -Richtung dem *Original-Attraktor* des Hénon-Systems entspricht.

Die in Abbildung 5.16 in Teil (a) für die Trainingsmengen und in Teil (b) für die Testmenge dargestellten Prädiktionsgewinne für 3-lagige *MLP*-Netze mit erneut annähernd konstanter Anzahl von $Q \approx 350$ Netzgewichten bestätigt diese Vermutung.

Aufgrund der speziellen Eigenschaften des H_x -Signals wird ein 2:88:1 *MLP*-Netz mit insgesamt $Q=353$ Gewichten zusätzlich zu den bereits zur Prädiktion des $L_{x,0.1}$ - und MG_{17} -Signals eingesetzten *MLP*-Netzen mit den Topologien 3:70:1 bis 12:25:1 (vgl. in Tabelle 5.2) eingesetzt. Die Auswertung der Ergebnisse bestätigt die Annahme, daß wegen der fehlenden Periodizität und des Verlaufs der normierten AKF des H_x -Signals für alle Einbettungsdimensionen im untersuchten Bereich von $2 \leq d_E \leq 12$ bei Variation von τ für $\tau=1$ der mit Abstand beste Prädiktionsgewinn erreicht wird.

Der Vergleich der Ergebnisse, die bei Variation der Einbettungsdimension erzielt werden, zeigt, daß der mit $G_{tr}=139,8 \text{ dB}$ für die Trainings- bzw. $G_{te}=139,2 \text{ dB}$ für die Testmenge maximale Prädiktionsgewinn bei Verwendung des 2:88:1 Netzes für $\tau=1$ erreicht wird. Da für diese Einbettungsbedingungen der „Original“-Attraktor des Systems rekonstruiert wird, liefern größere Einbettungsdimensionen niedrigere Prädiktionsgewinne, wobei der sich nach Takens und Mañé aufgrund der fraktalen Dimension von $d_K=1,26$ des Hénon-Systems ergebende Wert der Einbettungsdimension von $d_E=4$ das zweitbeste Ergebnis liefert.

Zusammenfassend ist festzustellen, daß die *MLP*-Netze bei geeigneter Wahl der Einbettungsbedingungen die zugrundeliegende recht einfache nichtlineare Hénon-Abbildung sehr exakt nachbilden, lineare Prädiktoren jedoch aufgrund der nichtlinearen Abhängigkeiten für diese Aufgabenstellung ungeeignet sind. Bemerkenswert ist, daß auch mit *MLP*-Netzen gleicher Topologie bei ungünstiger Wahl der Einbettungskonstanten von $\tau \neq 1$ wesentlich schlechtere Ergebnisse erzielt werden, und damit die Variation der Einbettungskonstanten τ wesentlich stärkere Auswirkungen auf die erzielbaren Ergebnisse hat als die Anzahl der Netzgewichte, die Topologie des Netzes oder die Einbettungsdimension, solange diese die Bedingungen von Takens und Mañé erfüllt.

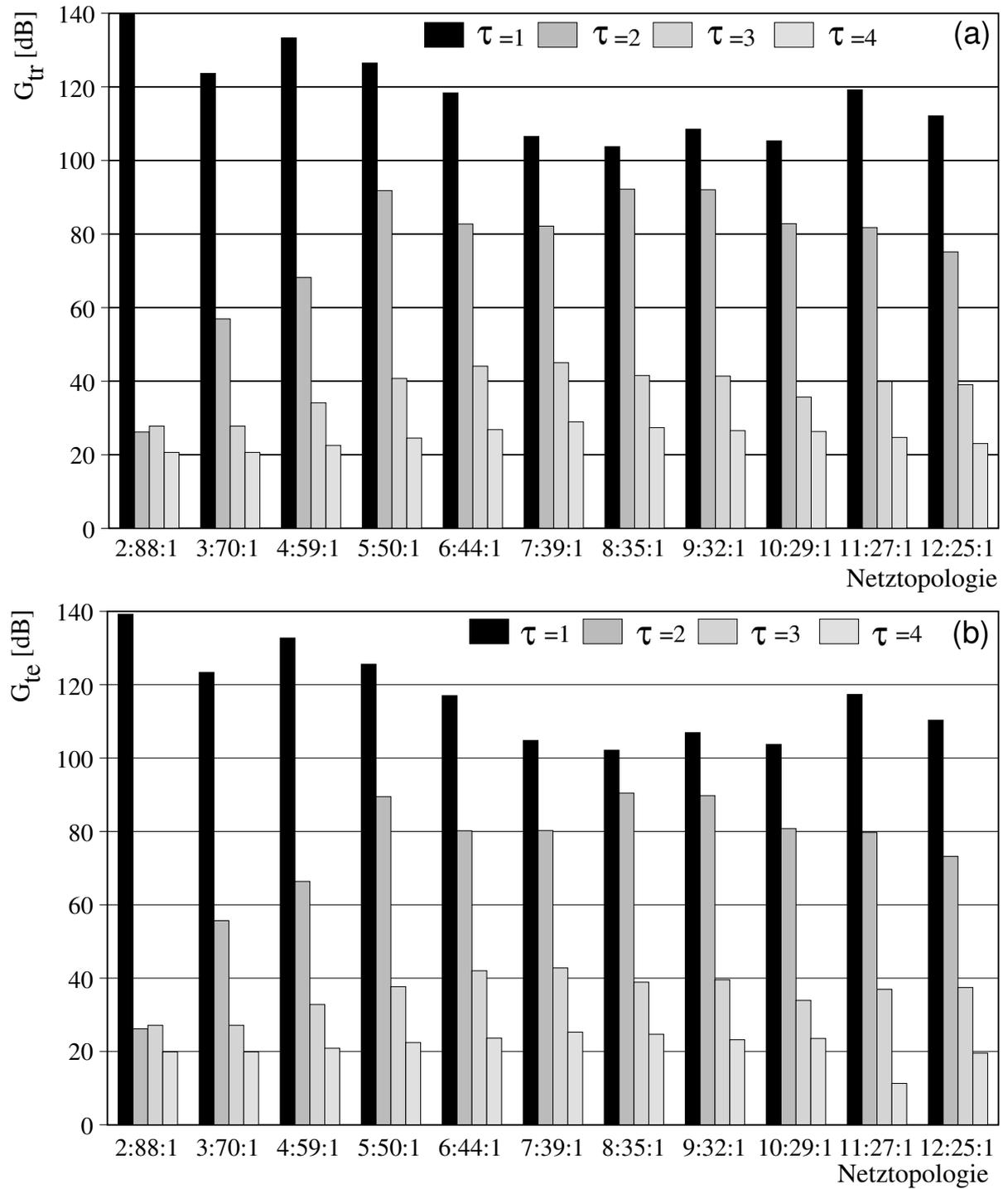


Abbildung 5.16: Vergleich der Prädiktionsergebnisse, die mit ausgewählten 3-lagigen *MLP*-Netzen mit annähernd konstanter Anzahl von 350 Netzgewichten für das chaotische Hénon-Signal erzielt werden.

(a): Ergebnisse für die 2 000 Muster umfassende Trainingsmenge

(b): Ergebnisse für die 100 000 Muster umfassende Testmenge

5.5 Weitere Prädiktionsergebnisse

In den vorangegangenen Abschnitten (5.1) bis (5.4) wurden neben einem 2:1:1 *MLP*-Netz ausschließlich *MLP*-Netze mit einer Anzahl von ca. 350 Gewichten aber unterschiedlicher Topologie zur skalaren Prädiktion chaotischer Zeitsignale eingesetzt. In einer Vielzahl weiterer Untersuchungen [138, 132, 5, 28] wurden neben diesen auch *MLP*-Netze mit unterschiedlicher Anzahl an Netzgewichten zur Prädiktion der vier vorgestellten chaotischen Zeitsignale verwendet. Das *OLL*-Verfahren erweist sich auch bei diesen Untersuchungen den anderen nichtlinearen Optimierungsverfahren hinsichtlich Konvergenzgeschwindigkeit, Rechenzeitbedarf und Robustheit weit überlegen. Die Prädiktionsgewinne fallen bei Verwendung kleinerer *MLP*-Netze mit einer um den Faktor 2 bis 4 geringeren Anzahl an Gewichten bei optimaler Wahl der Rekonstruktionsparameter nur um ca. 5 *dB* bis 15 *dB* geringer aus. Diese Ergebnisse zeigen, daß auch *MLP*-Netze mit wesentlich weniger als 350 Gewichten die nichtlinearen Abbildungseigenschaften der untersuchten chaotischen Systeme sehr gut nachbilden. Ein Zusammenhang zwischen der Komplexität der zu approximierenden Abbildung und der bei vorgegebener Güte der Approximation hierzu minimal notwendigen Anzahl an Gewichten und der optimalen Topologie des *MLP*-Netzes kann hier jedoch nicht angegeben werden und muß Ziel weitergehender Untersuchungen sein.

Die Eignung von *MLP*-Netzen zur Approximation einer anderen Klasse von Abbildungen wird am Beispiel der von Biegler-König und Bärmann [9, 10] angegebenen nichtlinearen vektoriellen Abbildung 8-dimensionaler Eingangsvektoren \mathbf{x}^k auf 3-dimensionale Ausgangsvektoren \mathbf{d}^k gemäß:

$$\mathbf{f} : \mathbb{R}^8 \rightarrow \mathbb{R}^3 \quad ; \quad \mathbf{d}^k = \mathbf{f}(\mathbf{x}^k) \quad (5.2)$$

mit

$$\mathbf{x}^k = (x_1^k, \dots, x_8^k)^T \in \mathbb{R}^8 ; \quad \mathbf{d}^k = (d_1^k, \dots, d_3^k)^T \in \mathbb{R}^3$$

und

$$\begin{aligned} d_1 &= \frac{x_1 \cdot x_2 + x_3 \cdot x_4 + x_5 \cdot x_6 + x_7 \cdot x_8}{4} \\ d_2 &= \frac{x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8}{8} \\ d_3 &= \frac{1}{2} \sqrt{4 - x_1 \cdot x_2 + x_3 \cdot x_4 + x_5 \cdot x_6 + x_7 \cdot x_8} \end{aligned}$$

untersucht.

Die Optimierung der Netzgewichte mit dem *OLL*-Verfahren erfolgte unter Verwendung einer $K=50$ Eingangs-Ausgangs-Paare $\{\mathbf{x}^k, \mathbf{d}^k\}_{k=1}^K$ umfassenden Trainingsmenge. Die einzelnen Komponenten der Eingangsvektoren \mathbf{x}^k waren im Bereich $[0,1]$ gleichverteilte Zufallszahlen.

Im Unterschied zu dem zu Beginn dieses Kapitels eingeführten Prädiktionsgewinn G in $[dB]$ verwenden Biegler-König und Bärman den Erwartungswert $E_{abs}(\mathbf{w})$ gemäß:

$$E_{abs}(\mathbf{w}) = \frac{1}{K} \sum_{k=1}^K E_{abs}^k \quad \text{mit} \quad E_{abs}^k(\mathbf{w}) = \frac{1}{8} \sum_{n=1}^8 |d_n^k - y_n^k| \quad (5.3)$$

Die mit dem *OLL*-Verfahren erzielten Werte $E_{abs}(\mathbf{w})$ sowie die aus [10] übernommenen Ergebnisse der von Biegler-König und Bärman vorgestellten Optimierungsverfahren *BV1*, *BV2* und des *BP*-Verfahrens sind in Tabelle 5.9 für *MLP*-Netze mit den Topologien 8:8:8:3 und 8:14:3 dargestellt.

Iteration	8:8:8:3 <i>MLP</i> -Netz				8:14:3 <i>MLP</i> -Netz
	BV1	BV2	BP	OLL	OLL
0	0.2231	0.2231	0.2231	0.861	1.32
1	0.0167	0.0204	0.0738	0.0188	0.0140
2	-	-	-	0.00721	0.00613
3	-	-	-	0.00503	0.00451
10	0.0163	0.0232	0.0704	0.00169	0.000679
50	-	0.0237	0.0241	0.00079	0.000203
500	-	0.0087	0.0165	0.00034	0.000103
2000	-	-	0.0073	0.00020	0.000058

Tabelle 5.9: Vergleich der Genauigkeit der Approximation der in Gleichung (5.2) angegebenen nichtlinearen Abbildung, die mit *MLP*-Netzen der Topologie 8:8:8:3 und 8:14:3 erzielt werden. Die Optimierung der Netzgewichte erfolgte mit dem *OLL*-Verfahren, den von Biegler-König und Bärman angegebenen Verfahren *BV1* und *BV2* sowie dem *BP*-Verfahren.

Für das 8:8:8:3 *MLP*-Netz wird mit dem *OLL*-Verfahren bereits nach 2 Iterationen ein Wert für $E_{abs}(\mathbf{w})$ erzielt, welcher kleiner ist als die Werte die mit dem *BV2*-Verfahren nach 500 und dem *BP*-Verfahren nach 2 000 Iterationen erreicht werden.

Entsprechend den Angaben in [10] und der ermittelten Rechenzeit je Iteration des *OLL*-Verfahrens verglichen mit dem *BP*-Verfahren ergibt sich für die verwendeten Optimierungsverfahren ein relativer Rechenaufwand je Iteration von $OLL/BV1/BV2/BP=8/2/4/1$ für das 8:8:8:3 *MLP*-Netz. Unter Berücksichtigung dieser Werte und der Ergebnisse aus Tabelle 5.9 erweist sich das *OLL*-Verfahren für diese Aufgabenstellung hinsichtlich des insgesamt erforderlichen Rechenzeitbedarfs dem Verfahren *BV2* um einen Faktor von ca. 125 und gegenüber dem *BP*-Verfahren um einen Faktor von ca. 500 überlegen. Weiterhin ist der Tabelle zu entnehmen, daß mit dem 8:14:3 *MLP*-Netz, welches ebenso wie das 8:8:8:3 *MLP*-Netz über 171 Gewichte verfügt, eine wesentlich höhere Güte der Approximation erzielt wird.

Dieses Ergebnis zeigt, daß die optimale Topologie des *MLP*-Netzes von dem zugrundeliegenden Approximationsproblem abhängig ist, und daß mit *MLP*-Netzen, die nur eine verdeckte Lage aufweisen, nicht in jedem Falle schlechtere Ergebnisse erzielt werden als mit *MLP*-Netzen mit mehreren verdeckten Lagen.

5.6 Berechnung der Lyapunov-Exponenten aus dem Zeitsignal

Die Ausführungen der Kapitel 5.2, 5.3 und 5.4 haben gezeigt, daß *MLP*-Netze sehr gut zur Prädiktion chaotischer Zeitsignale geeignet sind. Demzufolge approximiert die Abbildung $\tilde{\mathcal{G}}$ des *MLP*-Netzes die zugrundeliegende Abbildung \mathcal{G} des rekonstruierten Attraktors sehr genau. Wenn die Rekonstruktion zusätzlich die Bedingungen aus Kapitel 2.5 für eine Einbettung erfüllt, bleiben auch die charakteristischen Eigenschaften des durch $\mathcal{F}(\mathbf{x}^k)$ bzw. $\phi^{T_A}(\mathbf{x}^k)$ beschriebenen Systems erhalten und können aus der Folge von rekonstruierten Zustandsvektoren $\{\mathbf{x}^k\}_{k=0}^K$ und der Abbildung $\tilde{\mathcal{G}}$ des *MLP*-Netzes ermittelt werden.

Zur Bestimmung der Lyapunov-Exponenten wird die Jacobimatrix $\tilde{\mathcal{G}}_{\mathbf{x}}(\mathbf{x})$ der Abbildung des *MLP*-Netzes (Gl. 3.41) als Approximation der Jacobimatrix $\mathcal{F}_{\mathbf{x}}(\mathbf{x})$ des dynamischen Systems (Gl. 2.25) eingesetzt. Damit ergibt sich mit dem in Kapitel 2.4 vorgestellten Treppen-Iterations-Algorithmus (Gleichungen (2.29) \cdots (2.33)) folgendes Verfahren zur Berechnung der Lyapunov-Exponenten aus experimentellen Daten unter Verwendung des *MLP*-Netzes als Modell des dynamischen Systems:

0 Initialisierung

- $\mathbf{Q}^0 = \mathbf{I}$
- Evolutionszähler $k=0$
- Anzahl K der Iterationen festlegen

1 Abbildung des Test-Volumenelementes \mathbf{Q} durch das *MLP*-Netz

- Inkrementieren des Evolutionszählers $k := k + 1$
- $\mathbf{A}^k = \tilde{\mathcal{G}}_{\mathbf{x}}(\mathbf{x}^{k-1})\mathbf{Q}^{k-1}$

2 \mathbf{QR} -Zerlegung der Matrix \mathbf{A}

- $\bar{\mathbf{Q}}^k \bar{\mathbf{R}}^k = \mathbf{A}^k$
- $\mathbf{Q}^k := \bar{\mathbf{Q}}^k \text{SIGN}(\bar{\mathbf{R}}^k)$
- $\mathbf{R}^k := \text{SIGN}(\bar{\mathbf{R}}^k) \bar{\mathbf{R}}^k$

3 Berechnung der neuen Näherungslösung λ_i^k

- $\lambda_i^k = \frac{1}{k\Delta T} \sum_{j=0}^{k-1} \log_2 R_{ii}^j$; $i = 1, \dots, d_E$

4 Wenn $k < K$: Weiter bei 1

5 Näherungslösung für Lyapunov-Exponenten

- $\lambda_i \approx \lambda_i^K = \frac{1}{K\Delta T} \sum_{j=0}^{K-1} \log_2 R_{ii}^j$; $i = 1, \dots, d_E$

6 ENDE

Die Anzahl K der zur Ermittlung der Lyapunov-Exponenten betrachteten Punkte auf

dem rekonstruierten Attraktor wird im folgenden als *Anzahl an Iterationen* des Treppen-Iterations-Algorithmus bezeichnet.

In einer Vielzahl von Veröffentlichungen [24, 27, 37, 107] wird der Einsatz dieses Algorithmus zur Berechnung der Lyapunov-Exponenten dynamischer Systeme beschrieben. Im Rahmen dieser Arbeit werden die Abbildungseigenschaften von *MLP*-Netzen zur Modellierung chaotischer dynamischer Systeme und zur Berechnung der Lyapunov-Exponenten dieser Systeme eingesetzt. Im Gegensatz zur Mehrzahl der veröffentlichten Untersuchungen werden hier globale Modelle, die optimierten und nicht mehr adaptierten *MLP*-Netze, und nicht fortlaufend neu berechnete lokale Modelle [27, 37, 107, 129, 146] verwendet.

Das beschriebene Verfahren liefert entsprechend der gewählten Einbettungsdimension d_E auch ebensoviele Lyapunov-Exponenten. Im allgemeinen ist der rekonstruierte Attraktor in einer Mannigfaltigkeit des Einbettungsraumes mit einer Dimension m enthalten, die gemäß Gl.(2.37) kleiner gleich der Einbettungsdimension d_E ist. In diesem Fall entstehen verfahrensbedingt $d_E - m$ zusätzliche Lyapunov-Exponenten, die sogenannten *spurious exponents*, die von den Eigenschaften des dynamischen Systems unabhängig sind. Ein Verfahren zur Trennung der echten von den zusätzlich erzeugten Lyapunov-Exponenten wird von Parlitz [107] (siehe auch [87] und [40]) angegeben.

Dieses basiert auf einem Vergleich der aus dem vorgegebenen Zeitsignal ermittelten Lyapunov-Exponenten $\lambda^{orig} = (\lambda_1^{orig}, \dots, \lambda_{d_E}^{orig})^T$ mit den Lyapunov-Exponenten $\lambda^{rev} = (\lambda_1^{rev}, \dots, \lambda_{d_E}^{rev})^T$, die sich bei zeitlicher Umkehr des Signals ergeben. Parlitz [107] gibt für die bei dieser Vorgehensweise theoretisch erwarteten Werte der Lyapunov-Exponenten folgende Beziehung an

$$\begin{aligned} \lambda_i^{orig} &= -\lambda_{m+1-i}^{rev} \quad ; \quad i = 1, \dots, m \\ \lambda_i^{orig} &\approx \lambda_i^{rev} < 0 \quad ; \quad i = m + 1, \dots, d_E \end{aligned} \quad (5.4)$$

wobei dieser Gleichung die Annahme zugrunde liegt, daß die zusätzlich erzeugten Exponenten kleiner sind als der kleinste echte Lyapunov-Exponent.

Durch die zeitliche Umkehr laufen die Trajektorien in den Raumrichtungen, in denen sie sich zuvor voneinander entfernt hatten, aufeinander zu und umgekehrt. Dadurch ändern die echten Lyapunov-Exponenten des Systems ihr Vorzeichen, bleiben aber dem Betrag nach unverändert. Die durch die Einbettung zusätzlich erzeugten Exponenten zeigen dieses Verhalten nicht, sind aber in beiden Fällen von gleicher Größenordnung.

Die im folgenden beschriebenen Untersuchungen konzentrieren sich auf die Bestimmung der Lyapunov-Exponenten des Lorenz-Systems und der Hénon-Abbildung unter Verwendung des oben angegebenen Verfahrens. Weitergehende Untersuchungen zeigen, daß auch eine Berechnung der Lyapunov-Exponenten aus der Lösung der Mackey-Glass-Differentialgleichung sowohl für $\Delta T = 17$ als auch für $\Delta T = 30$ zu konsistenten Ergebnissen führt.

5.6.1 Lyapunov-Exponenten des Lorenz-Modells

Die im folgenden beschriebenen Untersuchungen zur Berechnung der Lyapunov-Exponenten des Lorenz-Systems basieren sowohl auf verschiedenen Rekonstruktionen des Attraktors aus dem skalaren Zeitsignal $\mathbf{L}_{x,0.1}$ Gl.(A.5), als auch auf Verwendung der „Original“-Zustandsvektoren $\mathbf{L}_{xyz,0.1}$ Gl.(A.7).

Zunächst werden aus dem skalaren Zeitsignal $\mathbf{L}_{x,0.1}$ Folgen von rekonstruierten Zustandsvektoren für die Einbettungsdimensionen $d_E=3, \dots, d_E=12$ betrachtet. Als Approximation der zugrundeliegenden Abbildungen \mathcal{G} der rekonstruierten Attraktoren werden die Abbildungen $\tilde{\mathcal{G}}$ der bereits in Kapitel 5.2 vorgestellten 3-lagigen *MLP*-Netze mit den Netztopologien 3:70:1, \dots , 12:25:1 verwendet. Die in Abbildung 5.17 dargestellten Ergebnisse beschränken sich auf $\tau=1$ für die Wahl der Einbettungskonstante, da entsprechend der Abbildung 5.5 für diesen Wert die besten Approximationsergebnisse erzielt wurden.

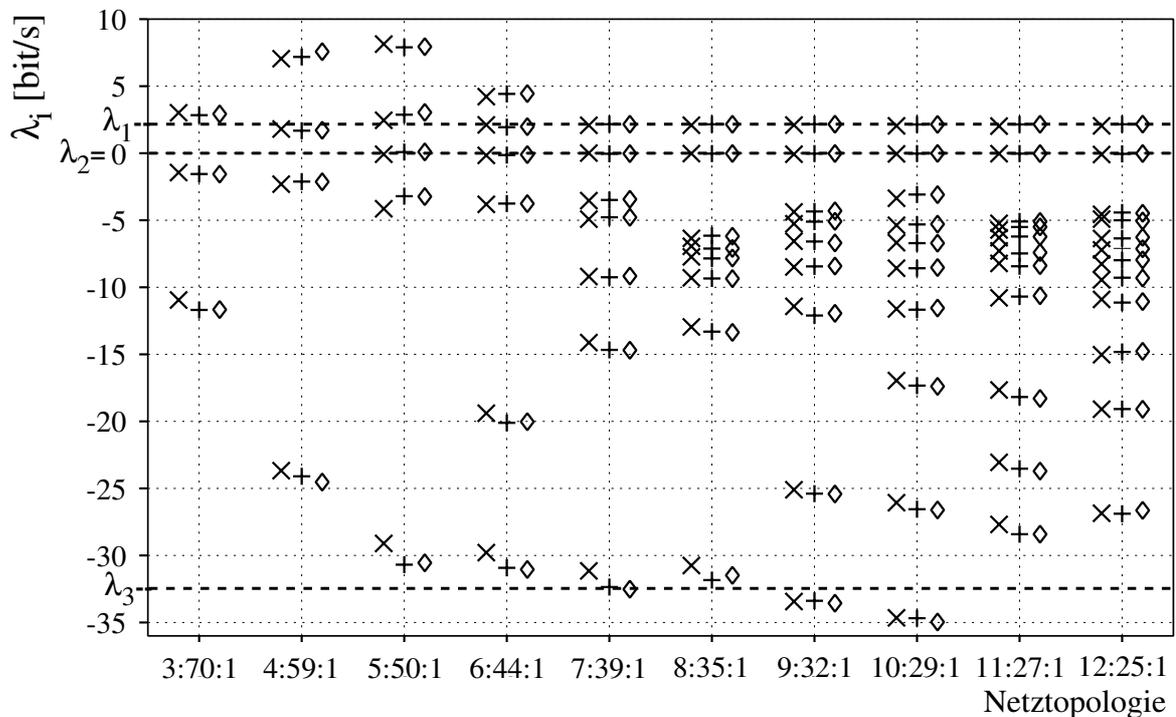


Abbildung 5.17: Lyapunov-Exponenten des Lorenz-Systems berechnet aus rekonstruierten Zustandsvektoren und der Approximation der Abbildung des rekonstruierten Attraktors durch *MLP*-Netze unterschiedlicher Topologie. Die Exponenten wurden nach (x): $K=10^3$, (+): $K=10^4$ und (◇): $K=10^5$ Iterationen des Treppen-Iterations-Algorithmus berechnet. Die theoretischen Werte sind durch gestrichelte Linien gekennzeichnet.

Die dargestellten Lyapunov-Exponenten wurden nach jeweils (x) : $K=10^3$, (+) : $K=10^4$ und (◇) : $K=10^5$ Iterationen berechnet. Hierbei entstammen die bei $K=10^3$ verwendeten rekonstruierten Zustandsvektoren der für die Optimierung des *MLP*-Netzes verwendeten

Trainingsmenge und für $K=10^4$ und $K=10^5$ der 10^5 Muster umfassenden Testmenge. Die theoretischen Werte der Lyapunov-Exponenten sind in Form gestrichelter Linien in diese Darstellung eingetragen. Der Vergleich mit den aus rekonstruierten Zustandsvektoren berechneten Werten liefert folgende Ergebnisse:

- Für die Einbettungsdimensionen $d_E=7, \dots, d_E=12$ werden die Exponenten λ_1 und λ_2 mit hoher Genauigkeit ermittelt (siehe hierzu auch Tabelle 5.10). Die Abweichungen der Werte von λ_1 liegen im Bereich $\pm 0,2\%$ bis $\pm 0,4\%$.
- Die bei diesen Einbettungsdimensionen zusätzlich erzeugten Exponenten (*spurious exponents*) sind negativ, liegen im Bereich zwischen λ_2 und λ_3 und lassen sich bei Kenntnis der echten Exponenten leicht von diesen trennen. Dieses Ergebnis widerspricht der Gl.(5.4) zugrundeliegenden Annahme, wonach die zusätzlich erzeugten Exponenten kleiner als der kleinste echte Lyapunov-Exponent (hier λ_3) sind. Dieses ist auf den globalen Modellansatz und die Verwendung von *MLP*-Netzen zur Approximation zurückzuführen, hat jedoch, wie die folgenden Ausführungen zeigen, keinen Einfluß auf die Qualität der erzielbaren Ergebnisse.
- Der betragsmäßig große negative Exponent λ_3 wird mit guter Genauigkeit von $\pm 5\%$ für die Einbettungsdimensionen $d_E=5, \dots, d_E=9$ berechnet. Generell steigen die numerischen Anforderungen an die Bestimmung und anschließende QR-Zerlegung der Jacobimatrizen $\tilde{\mathcal{G}}_{\mathbf{x}}(\mathbf{x})$ der Abbildung $\tilde{\mathcal{G}}$ des *MLP*-Netzes bei der Berechnung betragsmäßig großer Lyapunov-Exponenten und bei Verwendung großer Einbettungsdimensionen. Diese exponentiellen Abhängigkeiten führen dazu, daß der Exponent λ_3 für $d_E=10, \dots, d_E=12$ zu ungenau bestimmt wird.
- Für Einbettungsdimensionen im Bereich $d_E=3, \dots, d_E=6$ liegt keine Einbettung des Lorenz-Attraktors vor, da die charakteristischen Größen nicht erhalten bleiben und **zusätzliche positive** Lyapunov-Exponenten (größer als λ_1) erzeugt werden.

Ein Vergleich der nach 10^3 , 10^4 und 10^5 Iterationen erzielten Ergebnisse ergibt, daß die Lyapunov-Exponenten bereits nach 10^3 Iterationen annähernd konstante Werte aufweisen. Änderungen für $K=10^4$ und $K=10^5$ Iterationen treten nur bei betragsmäßig großen Exponenten auf, da deren genaue Berechnung aufgrund der exponentiellen Abhängigkeiten numerisch wesentlich aufwendiger ist.

Das Konvergenzverhalten des Treppen-Iterations-Algorithmus über den ersten 2 000 Iterationen für das 7:39:1 *MLP*-Netz ist nochmals in detaillierterer Form in Abbildung 5.20(a) dargestellt. Auch hier verändern sich nur noch die Werte der betragsmäßig großen negativen Exponenten bei mehr als 500 Iterationen merklich. Es wird jedoch deutlich, daß aus dem Konvergenzverhalten keine Aussagen bezüglich der Trennung der echten von den verfahrensbedingt zusätzlich erzeugten Lyapunov-Exponenten abgeleitet werden können.

Die fehlende Möglichkeit zur Identifikation der echten Lyapunov-Exponenten stellt im vorliegenden Fall kein Problem dar, da diese bekannt sind. Sollen jedoch aus experimentell ermittelten Daten unbekannter dynamischer Systeme deren Lyapunov-Exponenten mit dem beschriebenen Algorithmus bestimmt werden, wird ein Verfahren zur Trennung der echten von den zusätzlichen Exponenten notwendig. Dazu wird, wie von Parlitz vorgeschlagen, das zeitlich invertierte Lorenz-Signal $\mathbf{L}_{x,0.1}^{-1}$ gemäß Gl.(A.6) als Basis für die folgenden Untersuchungen verwendet.

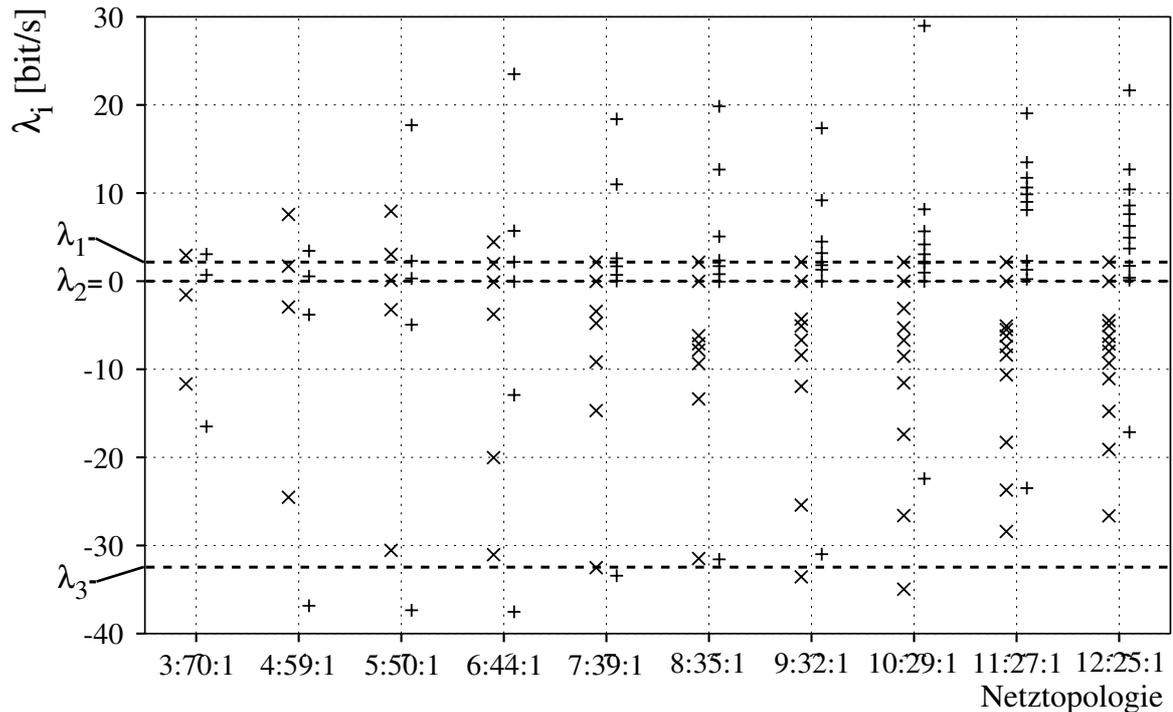


Abbildung 5.18: Lyapunov-Exponenten des Lorenz-Systems berechnet unter Verwendung von *MLP*-Netzen unterschiedlicher Topologie bei (\times): normalem zeitlichen Verlauf und bei ($+$): invertiertem zeitlichen Verlauf der rekonstruierten Zustandsvektoren. Die theoretischen Werte sind durch gestrichelte Linien gekennzeichnet.

Zunächst wurden *MLP*-Netze der bereits zuvor verwendeten Topologien 3:70:1, \dots , 12:25:1 unter Verwendung derselben, nur zeitlich invertierten, 2000 Muster umfassenden Trainingsmenge optimiert. Die nach 10^5 Iterationen des Treppen-Iterations-Algorithmus' berechneten Werte der Lyapunov-Exponenten wurden zum Ausgleich der Wirkung der Zeitumkehr auf die echten Exponenten mit -1 multipliziert (vgl. Gl.(5.4)) und sind in Abbildung 5.18 durch ($+$) gekennzeichnet dargestellt. Zum Vergleich enthält diese Abbildung, gekennzeichnet durch (\times), auch die bereits in Abbildung 5.17 enthaltenen Werte der Lyapunov-Exponenten, die nach 10^5 -Iterationen für den ursprünglichen zeitlichen Verlauf des Lorenz-Signals ermittelt wurden. Zusätzlich ist in Abbildung 5.19 ein Ausschnitt aus Abbildung 5.18 für den Bereich $-8\text{bit/s} < \lambda < 8\text{bit/s}$ wiedergegeben.

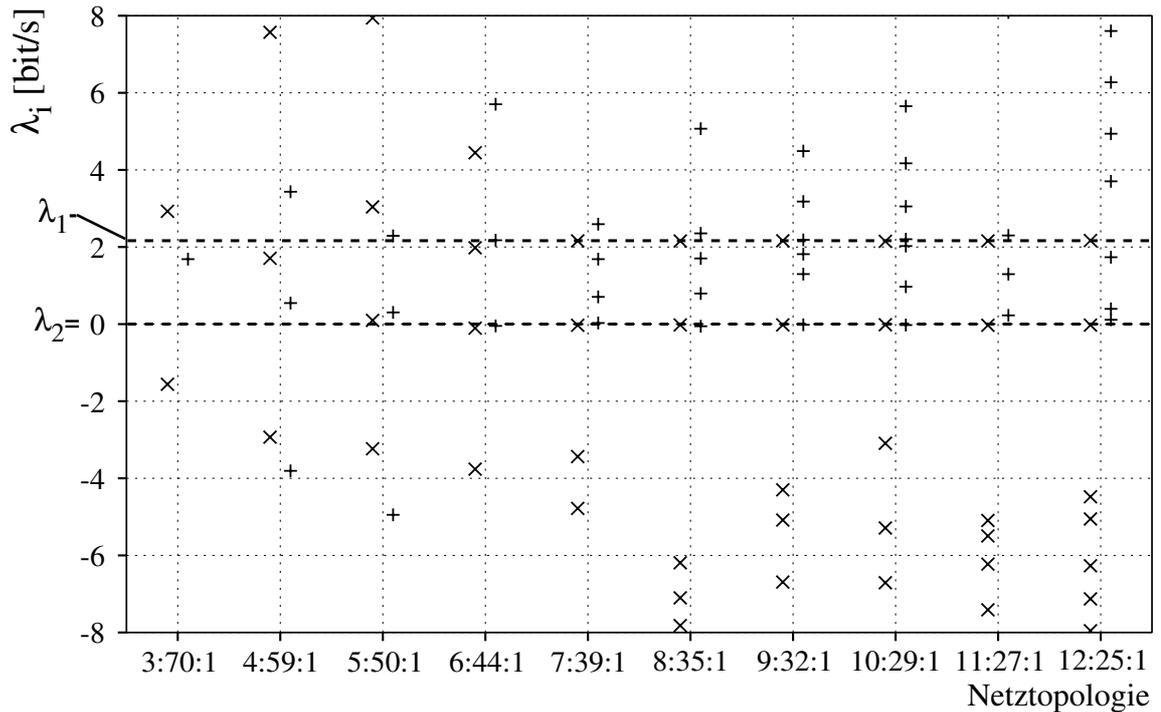


Abbildung 5.19: Ausschnitt aus Abbildung 5.26 für Werte der Lyapunov-Exponenten im Bereich von -8bit/s bis 8bit/s .

Wie bereits zuvor festgestellt wurde, liegt für die Dimensionen im Bereich $d_E=3, \dots, d_E=6$ keine Einbettung des Lorenz-Attraktors vor, da zusätzliche positive Lyapunov-Exponenten auftreten. Das gleiche Verhalten wird entsprechend Abbildung 5.18 auch bei der Untersuchung des zeitlich invertierten Signals beobachtet. Auch hier werden neben dem jetzt positiven Exponenten λ_3 zusätzliche positive Exponenten im Bereich $\lambda_2 < \lambda < -\lambda_3$ erzeugt. Daher werden diese Ergebnisse nicht weiter berücksichtigt.

Für die Einbettungsdimensionen $d_E=7, \dots, d_E=12$ entstehen bei Zeitumkehr keine zusätzlichen positiven Exponenten, so daß die hierfür ermittelten Ergebnisse zur weiteren Auswertung herangezogen werden.

Im Gegensatz zu den von Parlitz durchgeführten Untersuchungen [107] und den Gleichungen (5.4) wird keine konsistente Überlappung der berechneten Lyapunov-Exponente beobachtet. Dieses bedeutet, daß bei Zeitumkehr zusätzliche negative Exponenten entstehen, die jedoch nicht ausschließlich kleiner als $-\lambda_1$ sind, sondern auch Werte im Bereich $-\lambda_1 < \lambda < \lambda_2$ annehmen.

Auf Basis der dargestellten Ergebnisse ist dennoch eine Identifizierung der echten Lyapunov-Exponenten mit hoher Sicherheit möglich, da Überlappungen entsprechend $\lambda_i^{orig} \approx -\lambda_j^{rev}$ nur bei den echten Lyapunov-Exponenten auftreten, deren Werte bei normalem zeitlichen Verlauf sehr genau ermittelt werden. Weiterhin ist festzustellen, daß die Lyapunov-Exponenten bei zeitlich invertiertem Signal wesentlich ungenauer berech-

net werden. Dieses wird wieder auf die exponentiellen Abhängigkeiten bei der Berechnung der Lyapunov-Exponenten und damit auf die Wirkung des bei Zeitumkehr sehr großen positiven Exponenten λ_3 zurückgeführt.

Zur Beurteilung der Konvergenzeigenschaften sind in Abbildung 5.20(b) die bei zeitlich invertiertem Signal berechneten Werte der Lyapunov-Exponenten bei Verwendung des 7:39:1 *MLP*-Netzes über den ersten 2000 Iterationen im Vergleich zu den bei normalem Zeitverlauf ermittelten Werten dargestellt.

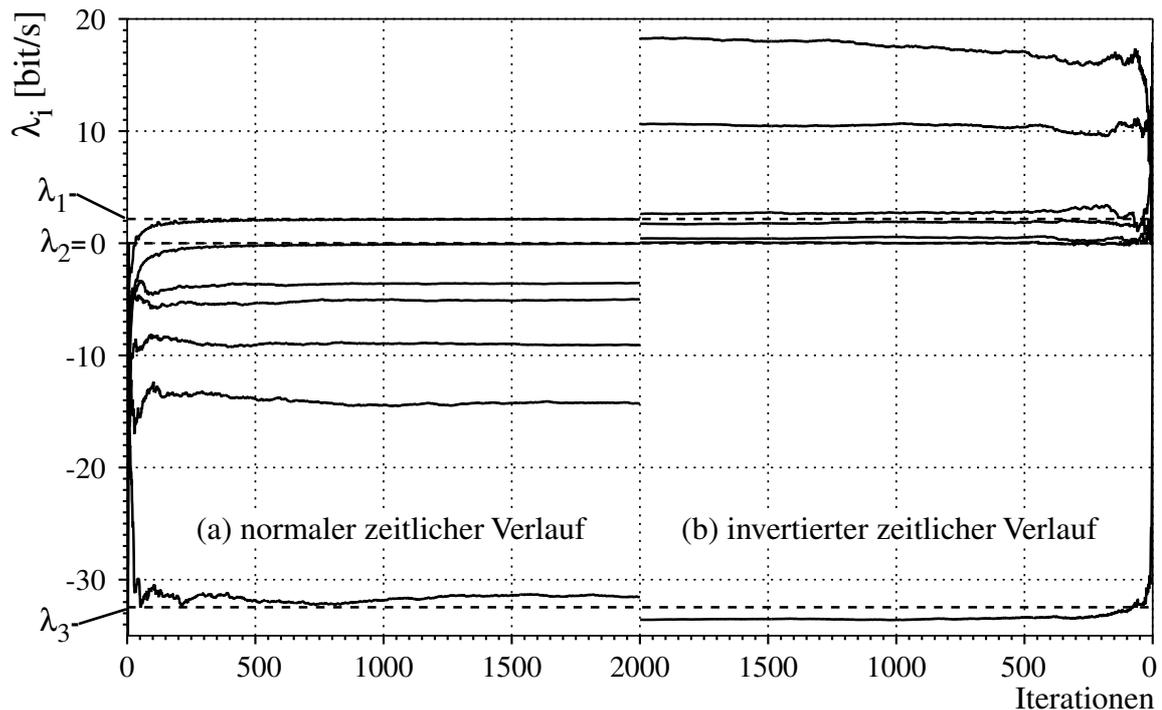


Abbildung 5.20: Konvergenzverhalten des Treppen-Iterations-Algorithmus bei der Berechnung der Lyapunov-Exponenten des Lorenz-Signals unter Verwendung rekonstruierter Zustandsvektoren und eines 3-lagigen 7:39:1 *MLP*-Netzes zur Approximation der Systemabbildung. Es sind jeweils 2000 Iterationen bei (a) : normalem zeitlichen Verlauf und (b) : bei invertiertem zeitlichen Verlauf des Lorenz-Signals dargestellt. Im Fall (b) sind die ermittelten Lyapunov-Exponenten mit dem Faktor -1 multipliziert.

Das Konvergenzverhalten zeigt in beiden betrachteten Fällen keine wesentlichen Unterschiede. Es werden auch bei Verwendung des zeitlich invertierten Signals merkliche Änderungen bei mehr als 500 Iterationen nur bei den betragsmäßig großen negativen Exponenten beobachtet. Dabei ist zu beachten, daß diese aufgrund des Faktors -1 in Abbildung 5.20(b) positiv dargestellt sind. Aus dem Vergleich der Werte der Lyapunov-Exponenten nach 2000 Iterationen für normalen und invertierten Zeitverlauf ist eine Identifizierung der echten Lyapunov-Exponenten λ_1 , λ_2 und λ_3 eindeutig möglich.

Zur Prädiktion der „Original“-Zustandsvektoren $\mathbf{L}_{xyz,0.1}$ und damit zur Approximation des „Original“-Lorenz-Attraktors wurde ein 3-lagiges 3:50:3 *MLP*-Netz mit 353 Gewichten sowie ein 4-lagiges 3:15:15:3 *MLP*-Netz mit 348 Gewichten verwendet. Beide Netze stellen vektorielle Prädiktoren mit jeweils $d_S=3$ Knoten in der Ein- und Ausgangslage gemäß Gl.(2.45) dar. Die hiermit erzielten Prädiktionsergebnisse für eine 2000 Muster umfassende Trainingsmenge sowie eine 100 000 Muster beinhaltende Testmenge sind in Tabelle 5.10 zusammengefaßt. Weiterhin sind dort die Lyapunov-Exponenten wiedergegeben, die nach jeweils $K=10^5$ Iterationen des Treppen-Iterations-Algorithmus ermittelt wurden.

Zum Vergleich mit den Ergebnissen, die bei Verwendung rekonstruierter Zustandsvektoren erzielt wurden, sind aus den Untersuchungen zu Abbildung 5.5 und 5.17 zusätzlich die Prädiktionsgewinne sowie die Lyapunov-Exponenten für die 7:39:1, 8:35:1 und 9:32:1 *MLP*-Netze in Tabelle 5.10 dargestellt.

Topologie			Q	$G_{tr}[dB]$	$G_{te}[dB]$	λ_1	λ_2	λ_3
3:50:3	$d_S=3$	—	353	61,66	60,39	2,116	-0,045	-32,137
3:15:15:3	$d_S=3$	—	348	62,82	61,17	2,136	0,013	-33,961
7:39:1	$d_E=7$	$\tau = 1$	352	66,89	47,43	2,158	-0,031	-32,519
8:35:1	$d_E=8$	$\tau = 1$	351	70,33	43,41	2,156	-0,021	-31,481
9:32:1	$d_E=9$	$\tau = 1$	353	70,16	46,80	2,161	-0,021	-33,561
theor. Werte	—	—	—	—	—	2,164	0,0	-32,46

Tabelle 5.10: Vergleich der berechneten Lyapunov-Exponenten bei Verwendung von
 (a) „Original“-Zustandsvektoren ($d_S=3$) und
 (b) rekonstruierten Zustandsvektoren ($d_E=7, d_E=8$ und $d_E=9$)
 mit den theoretischen Lyapunov-Exponenten des Lorenz-Systems.

5.6.2 Lyapunov-Exponenten der Hénon-Abbildung

Die in Abschnitt 5.4 mit den optimierten *MLP*-Netzen erzielten extrem hohen Prädiktionsgewinne für das \mathbf{H}_x -Signal zeigen, daß die dynamischen Eigenschaften der zugrundeliegenden Hénon-Abbildung sehr gut durch die *MLP*-Netze modelliert werden. Daher ist zu vermuten, daß ebenfalls die Lyapunov-Exponenten der Hénon-Abbildung auf Basis

dieser *MLP*-Netze mit hoher Genauigkeit bestimmt werden können.

In Abbildung 5.21 sind die nach 100 000 Iterationen des Treppen-Iterations-Algorithmus berechneten Lyapunov-Exponenten sowohl für das Hénon-Signal \mathbf{H}_x bei normalem zeitlichen Verlauf als auch für das zeitlich invertierte \mathbf{H}_x^{-1} -Signal dargestellt. Hierzu wurden die bereits in Abschnitt 5.4 beschriebenen *MLP*-Netze der Topologie 3:70:1, 4:59:1 ... 12:25:1 für den Wert $\tau=1$ der Einbettungskonstanten verwendet.

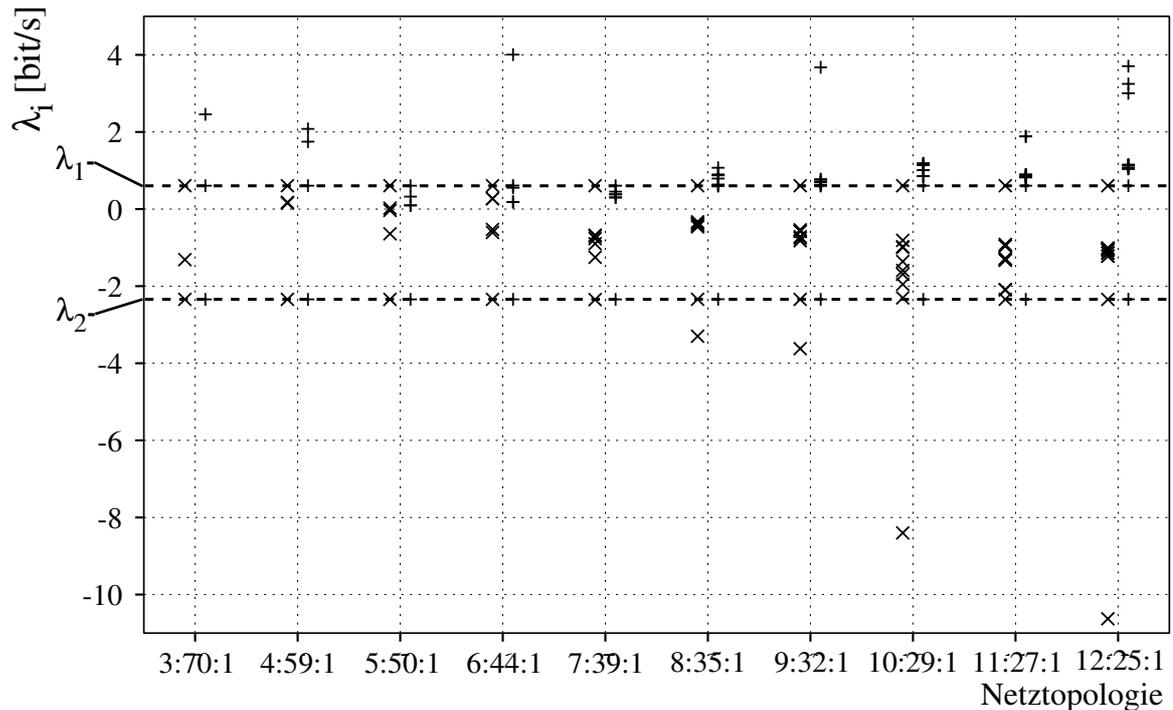


Abbildung 5.21: Lyapunov-Exponenten der Hénon-Abbildung berechnet nach jeweils $K=10^5$ Iterationen des Treppen-Iterations-Algorithmus unter Verwendung von *MLP*-Netzen unterschiedlicher Topologie bei (x): normalem zeitlichen Verlauf und bei (+): invertiertem zeitlichen Verlauf der rekonstruierten Zustandsvektoren. Die theoretischen Werte sind durch gestrichelte Linien gekennzeichnet.

Da die Qualität der Nachbildung der Abbildungseigenschaften der Hénon-Abbildung durch das *MLP*-Netz in sehr hohem Maße von der Einbettungskonstanten beeinflusst wird, werden bei Verwendung von *MLP*-Netzen, die für einen Wert der Einbettungskonstante $\tau \neq 1$ optimiert wurden, wesentlich ungenauere Werte für die Lyapunov-Exponenten bestimmt.

Die Auswertung der in Abbildung 5.21 dargestellten Ergebnisse ergibt wie zuvor für das Lorenz-System auch für die hier betrachtete Hénon-Abbildung eine eindeutige Identifizierung der echten Lyapunov-Exponenten. Diese werden für alle verwendeten *MLP*-Netze in Verbindung mit den entsprechenden Folgen rekonstruierter Zustandsvektoren mit sehr

hoher Genauigkeit ermittelt.

λ_i	$K = 10^2$	$K = 500$	$K = 10^3$	$K = 10^4$	$K = 10^5$	theoretisch
λ_1	0,66479	0,64571	0,63757	0,61057	0,60597	0,603
λ_2	-2,41199	-2,38310	-2,37506	-2,34856	-2,34294	-2,34

Tabelle 5.11: Lyapunov-Exponenten für die Hénon-Abbildung, geschätzt aus der x-Komponente des Attraktors für $K=10^2$ bis $K=10^5$ Iterationen. Zur Modellierung wurde ein *MLP*-Netz mit der Topologie 3:70:1 eingesetzt.

Tabelle 5.11 gibt die für das 3:70:1 *MLP*-Netz erzielten Lyapunov-Exponenten der Hénon-Abbildung nach 10^2 bis 10^5 Iterationen des Verfahrens im Vergleich zu den theoretischen Werten wieder. Nach 10^5 Iterationen weicht der positive Exponent nur um 0,5% und der negative Exponent nur um 0,1% vom theoretischen Wert ab. Diese sehr genauen Werte demonstrieren die Leistungsfähigkeit des beschriebenen Verfahrens und zeigen, daß die optimierten *MLP*-Netze die Abbildungseigenschaften des zugrundeliegenden dynamischen Systems sehr genau nachbilden.

6 Zusammenfassung

Im Mittelpunkt dieser Arbeit steht der Einsatz von *MLP*-Netzen zur Nachbildung der Abbildungseigenschaften nichtlinearer dynamischer Systeme. Unter Verwendung des Verfahrens der zeitverzögerten Komponenten [100, 136] wird aus einem vorliegenden chaotischen Zeitsignal eine Folge von Zustandsvektoren rekonstruiert. Bei geeigneter Rekonstruktion repräsentiert der auf diese Weise rekonstruierte Attraktor gemäß dem Einbettungstheorem von Takens [136] die charakteristischen Eigenschaften des zugrundeliegenden dynamischen Systems.

Die Optimierung der *MLP*-Netze erfolgt auf Basis einer Trainingsmenge, welche eine Teilmenge der rekonstruierten Zustandsvektoren darstellt und aus K Eingangs-Ausgangs-Paaren besteht. Die Trainingsmenge muß alle relevanten Abbildungseigenschaften des rekonstruierten Attraktors und damit des zugrundeliegenden dynamischen Systems charakterisieren. Approximiert die Abbildung $\tilde{\mathcal{G}}(\mathbf{x}^k)$ des optimierten *MLP*-Netzes die Abbildung $\mathcal{G}(\mathbf{x}^k)$ des rekonstruierten Attraktors hinreichend genau, kann die Vorhersage des zukünftigen Verlaufs des betrachteten chaotischen Zeitsignals mit hoher Genauigkeit erfolgen. Das *MLP*-Netz wird hier als nichtlinearer Prädiktor eingesetzt, wobei die Minimierung des mittleren quadratischen Prädiktionsfehlers auf ein nichtlineares Optimierungsproblem zur Bestimmung optimaler Netzgewichte führt. Neben der Trainingsmenge wird aus den rekonstruierten Zustandsvektoren eine Testmenge gebildet. Diese enthält ausschließlich nicht in der Trainingsmenge enthaltene rekonstruierte Zustandsvektoren, wodurch geprüft wird, ob das optimierte *MLP*-Netz die geforderten globalen Abbildungseigenschaften des zugrundeliegenden dynamischen Systems hinreichend genau nachbildet.

Zur nichtlinearen Optimierung der Netzgewichte wird das neuartige Verfahren der *Optimierung Lage für Lage* (*OLL*-Verfahren) vorgestellt, welches speziell auf die Optimierung von *MLP*-Netzen ausgerichtet ist. Dieses neue Verfahren beruht auf dem Ansatz, die iterative Berechnung der Netzgewichte getrennt nach Lagen durchzuführen. Für die Bestimmung der Gewichte der Ausgangslage führt dieser Ansatz auf ein lineares Problem. Zusätzlich wird das nichtlineare Optimierungsproblem für die verdeckten Lagen durch Linearisierung der Aktivierungsfunktionen auf ein iteratives lineares Problem zurückgeführt. Durch Einführung eines Strafterms wird der resultierende Linearisierungsfehler begrenzt. Im Vergleich zu den parallel eingesetzten *BP*-, *CG*- und *QN*-Verfahren besitzt das *OLL*-Verfahren eine um zwei bis fünf Zehnerpotenzen höhere Konvergenzgeschwindigkeit, wodurch eine erhebliche Reduktion des Rechenzeitbedarfs erzielt wird.

Am Beispiel von vier chaotischen Zeitsignalen (vgl. Anhang A), welche aus den Lösungen des Lorenz-Systems, der Mackey-Glass-Differentialgleichung sowie der Hénon-Abbildung abgeleitet sind, wird nachgewiesen, daß *MLP*-Netze die den Signalen zugrundeliegenden komplexen nichtlinearen Abhängigkeiten weitgehend ausnutzen und demzufolge sehr hohe Prädiktionsgewinne bei der Vorhersage des zukünftigen zeitlichen Verlaufs der Signale

erzielen. Hierzu werden 3-, 4- und 5-lagige *MLP*-Netze unter Verwendung des *OLL*-Verfahrens optimiert und als nichtlineare Prädiktoren eingesetzt. Die Topologie der Netze wurde bei diesen Untersuchungen so variiert, daß die Anzahl der Netzgewichte und damit die Ordnung des Optimierungsproblems in allen Fällen annähernd 350 betrug.

Abhängig von der Rekonstruktion und der Topologie des Netzes werden maximale Prädiktionsgewinne von 65,5 *dB* für das $L_{x,0,1}$ -Signal bei einer Prädiktionsweite von $pw=0,1$, von 139,8 *dB* bei $pw=1$ für das H_x -Signal sowie von 90,7 *dB* für das MG_{17} -Signal und 86 *dB* für das MG_{30} -Signal bei einer Prädiktionsweite von $pw=6$ erzielt. Der Einsatz linearer Prädiktoren liefert demgegenüber nur Prädiktionsgewinne von 4 *dB*, 1,4 *dB*, 20,7 *dB* sowie 20,5 *dB*. Diese Ergebnisse zeigen, daß lineare Prädiktoren zur Prädiktion dieser chaotischen Zeitsignale ungeeignet sind, da nahezu keine oder nur geringe lineare Bindungen vorhanden sind.

Für die einzelnen Signale werden trotz annähernd konstanter Anzahl an Netzgewichten bei der nichtlinearen Prädiktion stark variierende Prädiktionsergebnisse erzielt. Daher wurden unter Verwendung des *OLL*-Verfahrens systematische Untersuchungen durchgeführt, um zu analysieren, welchen Einfluß die Rekonstruktionsparameter d_E und τ sowie der Umfang der Trainingsmenge auf die Höhe der erzielbaren Prädiktionsgewinne haben und inwieweit Rückschlüsse aus der Kenntnis charakteristischer Kenngrößen des zugrundeliegenden dynamischen Systems auf eine optimale Wahl dieser Einflußgrößen gezogen werden können.

Als erstes Ergebnis ist festzustellen, daß die Variation der Einbettungsdimension und in sehr starkem Maße auch die Variation der Einbettungskonstanten entscheidenden Einfluß auf die Höhe der erzielbaren Prädiktionsgewinne haben. Dieses Verhalten wurde in der Literatur bisher nicht beschrieben und die damit verbundenen negativen Auswirkungen auf die erzielbaren Prädiktionsergebnisse demzufolge nicht berücksichtigt. Als Dimensionierungsrichtlinien für die Wahl günstiger Einbettungsparameter lassen sich aus der Kenntnis der fraktalen Dimension des zugrundeliegenden dynamischen Systems sowie aus dem Verlauf der Autokorrelationsfunktion des betrachteten chaotischen Zeitsignals folgende Bedingungen angeben (vgl. Abschnitt 5.3.1):

- Die Einbettungsparameter d_E und τ sind so zu wählen, daß die durch die rekonstruierten Zustandsvektoren in Verbindung mit der Prädiktionsweite erfaßte Zeitspanne (Fensterlänge) in der Größenordnung des ersten Minimums der normierten AKF des skalaren Zeitsignals liegt.
- Die Einbettungsdimension ist gemäß Takens und Mañé größer oder gleich dem Zweifachen der fraktalen Dimension plus Eins zu wählen.

Hinsichtlich der Festlegung des Umfangs der Trainingsmenge liefern die Untersuchungen folgende Ergebnisse:

- Die Anzahl der Trainingsmuster sollte in jedem Fall größer sein als die Ordnung des Optimierungsproblems. Anderenfalls werden die Zielwerte der Trainingsmenge nur gespeichert und reproduziert, ohne daß die *MLP*-Netze die Abbildungseigenschaften des dynamischen Systems auch nur annähernd nachbilden.
- Fallen die Prädiktionsgewinne für die Testmenge trotzdem deutlich geringer aus als die für die Trainingsmenge, sind drei Fälle zu unterscheiden. Wird die Optimierung der *MLP*-Netze nicht rechtzeitig beendet, tritt u. U. der Overlearning-Effekt auf, was zur Folge hat, daß die Abbildungseigenschaften des *MLP*-Netzes zu stark an die speziellen Eigenschaften der Trainingsmenge angepaßt werden. Die für die Prädiktion der Testmenge erforderliche Nachbildung der globalen Abbildungseigenschaften des zugrundeliegenden dynamischen Systems erfolgt nur unzureichend. Der Overlearning-Effekt tritt nicht mehr auf, wenn der Umfang der Trainingsmenge zum drei- bis fünffachen der Ordnung des Optimierungsproblems gewählt wird.
- Weiterhin kann die Trainingsmenge ungünstig oder zu klein gewählt worden sein, so daß die Abbildungseigenschaften des zugrundeliegenden Systems nicht in ausreichender Weise repräsentiert sind. In diesem Fall ist die Trainingsmenge entweder zu vergrößern, oder aus einem anderen Teil der rekonstruierten Zustandsvektoren zu bilden.

Einen zweiten Schwerpunkt dieser Arbeit bildet der Vergleich des *OLL*-Verfahrens hinsichtlich erzielbarer Prädiktionsgewinne, Konvergenzgeschwindigkeit und Robustheit mit aus der Literatur bekannten Optimierungsverfahren. Hierzu wird ein 5:22:9:1 *MLP*-Netz für die Prädiktion des Lorenz-Signals $\mathbf{L}_{x,0.1}$ und ein 10:29:1 *MLP*-Netz für die Prädiktion des Mackey-Glass-Signals \mathbf{MG}_{17} ausgewählt. Aus der Vielzahl der mit dem *OLL*-Verfahren durchgeführten Untersuchungen werden mit diesen *MLP*-Netzen für das jeweilige Signal die höchsten Prädiktionsergebnisse erzielt.

Zur Optimierung werden neben dem *BP*- und *BD*-Verfahren vier Varianten der klassischen *CG*-Verfahren von Polak-Ribière und Fletcher-Reeves sowohl mit als auch ohne Restart-Strategie eingesetzt. Als Beispiele für modernere *CG*-Verfahren kommen das gedächtnislose *BFGS*-Verfahren sowie das Verfahren der skalierten konjugierten Gradienten nach Shanno sowohl mit als auch ohne Restart-Strategie zum Einsatz. Aus den Varianten der *QN*-Verfahren wird das von Broyden, Fletcher, Goldfarb und Shanno beschriebene *BFGS*-Verfahren ausgewählt.

Der Rechenzeitbedarf für diese umfangreichen Untersuchungen ist aufgrund der teilweise extrem geringen Konvergenzgeschwindigkeiten der betrachteten Optimierungsverfahren sehr hoch, weshalb sich dieser Vergleich auch auf die genannten zwei Beispiele beschränkt. Insgesamt waren hierfür ca. 350 Tage Rechenzeit auf einer UNIX-Workstation mit einer Rechenleistung von ca. 1 bis $2 \cdot 10^6$ Fließkommaoperationen pro Sekunde notwendig.

Die Auswertung der Ergebnisse führt hinsichtlich der Konvergenzgeschwindigkeit sowie des Rechenzeitbedarfs der Optimierungsverfahren auf eine Einteilung in vier Gruppen. Als das mit Abstand leistungsfähigste Verfahren erweist sich das *OLL*-Verfahren, obwohl die Komplexität und damit der Rechenzeitbedarf je Iteration im Vergleich zu den *BP*-Verfahren um einen Faktor 15 bis 45 höher ist und auch im Vergleich mit dem wesentlich aufwendigeren *BFGS*-Verfahren noch das 2- bis 4-fache beträgt.

Das *OLL*-Verfahren ist dem *BFGS*-Verfahren bei weitem überlegen und erzielt im Vergleich zu diesem eine Reduktion der insgesamt erforderlichen Rechenzeit um den Faktor 26,5 für das $\mathbf{L}_{x,0.1}$ -Signal und 72,5 im Falle des \mathbf{MG}_{17} -Signals. Betrachtet man die Anzahl der Iterationen, so beträgt dieser Faktor sogar 97 bzw. 134 zugunsten des *OLL*-Verfahrens.

Das *BFGS*-Verfahren erweist sich auch gegenüber den 7 untersuchten Varianten der *CG*-Verfahren als deutlich überlegen und liefert bei vergleichbarer Anzahl an Iterationen um ca. 20 dB höhere Prädiktionsgewinne. Nur mit dem *BFGS*-Verfahren gelingt es, die Prädiktionsgewinne, die mit dem *OLL*-Verfahren erzielt werden, wenigstens annähernd zu erreichen. Bezüglich der erforderlichen Rechenzeit erzielt das *OLL*-Verfahren gegenüber den *CG*-Verfahren eine Reduktion um Faktoren zwischen 270 und 334 für die Prädiktion des $\mathbf{L}_{x,0.1}$ -Signals und um Faktoren zwischen 128 und 248 für die Prädiktion des \mathbf{MG}_{17} -Signals. Im Gegensatz zu verschiedenen Veröffentlichungen [131, 90, 5] sind generelle Vor- bzw. Nachteile einzelner *CG*-Verfahren gegenüber anderen *CG*-Verfahren im Rahmen der hier durchgeführten Untersuchungen nicht feststellbar.

Am ungeeignetsten sowohl bezüglich der Konvergenzgeschwindigkeit als auch im Hinblick auf die erzielbaren Prädiktionsgewinne erweisen sich die in der Literatur am häufigsten zur Optimierung von *MLP*-Netzen eingesetzten *BD*- und *BP*-Verfahren, wobei das *BD*-Verfahren gegenüber dem *BP*-Verfahren noch deutlich besser abschneidet. Das *BD*-Verfahren erreicht mit 20,6 dB für das $\mathbf{L}_{x,0.1}$ -Signal und 38 dB für das \mathbf{MG}_{17} -Signal einen um 47,2 dB bzw. 64 dB geringeren Prädiktionsgewinn. Es erzielt damit nach ca. $5 \cdot 10^5$ bzw. 10^6 Iterationen ein Ergebnis, welches mit dem *OLL*-Verfahren bereits nach 2 Iterationen erreicht wird.

Das *BFGS*-Verfahren und die *CG*-Verfahren führen in sehr flachen Regionen der Fehleroberfläche, in denen in Suchrichtung nur extrem kleine Änderungen der Gütefunktion auftreten, verfahrensbedingt nur sehr kleine Änderungen der Netzgewichte durch oder brechen aufgrund numerischer Probleme ab. Die Konvergenzgeschwindigkeit der Gradientenverfahren ist daher in diesen Gebieten und damit auch in der Nähe lokaler oder absoluter Minima extrem gering.

Dieser Fall tritt z. B. bei der Optimierung des 10:29:1 *MLP*-Netzes zur Prädiktion des \mathbf{MG}_{17} -Signals nach 50 635 Iterationen des *BFGS*-Verfahrens auf. Vor dem Abbruch hat der Vektor der Netzgewichte einen Betrag von ca. 83 und das *BFGS*-Verfahren führt so kleine Änderungen der Netzgewichte durch, daß die Beträge der Differenzvektoren

nur 10^{-6} bis 10^{-4} betragen. Bei Start des *OLL*-Verfahrens mit den Netzgewichten, die zum Abbruch des *BFGS*-Verfahrens führen, ändert das *OLL*-Verfahren die Netzgewichte bereits nach einer Iteration so stark, daß der Differenzvektor einen Betrag von 42 hat. Nach 1 000 Iterationen des *OLL*-Verfahrens hat der Vektor der Netzgewichte einen Betrag von 22 048, wobei der Prädiktionsgewinn um weitere $7,2 \text{ dB}$ gesteigert wird. Entsprechende Ergebnisse werden auch erzielt, wenn die mit den *CG*- oder *BP*-Verfahren optimierten *MLP*-Netze als Startwerte des *OLL*-Verfahrens verwendet werden.

Diese Eigenschaft des *OLL*-Verfahrens, bei der Änderung der Netzgewichte von der lokalen Beschaffenheit der Fehleroberfläche unabhängig zu sein, ermöglicht es diesem, im Gegensatz zu den Gradientenverfahren erheblich größere Änderungen der Netzgewichte vorzunehmen. Dies trägt zu der sehr hohen Konvergenzgeschwindigkeit des *OLL*-Verfahrens bei. Weiterhin wird hierdurch die Gefahr des Abbruchs des Verfahrens aufgrund numerischer Probleme sowie das Festlaufen an den stationären Punkten der Fehleroberfläche mit relativ geringem Prädiktionsgewinn erheblich verringert und die Robustheit des Verfahrens erhöht. Die Konvergenzeigenschaften des *OLL*-Verfahrens sind dennoch gesichert, da in Lösungsnähe stets linearisiert wird.

Diese Ergebnisse zeigen deutlich die überlegenen Eigenschaften des *OLL*-Verfahrens auch, oder besonders in Bereichen der Fehleroberfläche, in denen die betrachteten Gradientenverfahren entweder aufgrund numerischer Probleme abbrechen oder wo deren Konvergenzgeschwindigkeit aufgrund der lokalen Beschaffenheit der Fehleroberfläche bereits extrem gering geworden ist.

Zur Bewertung der erzielten Prädiktionsergebnisse werden für die Signale \mathbf{MG}_{17} und \mathbf{MG}_{30} der Mackey-Glass-Differentialgleichung Vergleichswerte aus der Literatur herangezogen, die ebenfalls auf einer vorgegebenen Prädiktionsweite von $pw=6$ beruhen (vgl. Abschnitte 5.3.1 und 5.3.2). Das \mathbf{MG}_{17} -Signal wird u. a. von Day und Davenport [21] untersucht, die erzielbare Prädiktionsgewinne im Bereich von ca. $42,5 \text{ dB}$ bis $46,0 \text{ dB}$ angeben. Im Vergleich hierzu liegt der mit dem *OLL*-Verfahren erreichte Prädiktionsgewinn bei über 90 dB und damit um ca. 45 dB über denen von Day und Davenport. Für das \mathbf{MG}_{30} -Signal liegen Ergebnisse von Lapedes und Farber [71, 72] sowie Sanger [128] vor. Lapedes und Farber verwenden ein 6:10:10:1 *MLP*-Netz und erzielen einen Prädiktionsgewinn von ca. 30 dB bis 32 dB [71]. Sanger [128] verwendet ebenfalls eine Einbettungsdimension von $d_E=6$ und erzielt damit einen Prädiktionsgewinn von ca. 32 dB . Im Vergleich hierzu werden mit dem *OLL*-Verfahren wesentlich höhere Prädiktionsgewinne bis zu 86 dB erzielt. Die von Lapedes und Farber sowie Sanger mit $d_E=6$ zu klein gewählte Einbettungsdimension verursacht diese deutlich schlechteren Prädiktionsergebnisse; sie sollte für das \mathbf{MG}_{30} -Signal mindestens $d_E=9$ betragen.

Die obigen Ausführungen zeigen, daß *MLP*-Netze sehr gut zur Prädiktion chaotischer Zeitsignale geeignet sind. Demzufolge approximiert die Abbildung $\tilde{\mathcal{G}}$ des *MLP*-Netzes

die zugrundeliegende Abbildung \mathcal{G} des rekonstruierten Attraktors sehr genau. Erfüllt die Rekonstruktion zusätzlich die Bedingungen aus Kapitel 2.5 für eine Einbettung, bleiben die charakteristischen Eigenschaften des zugrundeliegenden dynamischen Systems erhalten, so daß aus der Folge von rekonstruierten Zustandsvektoren und der Abbildung $\tilde{\mathcal{G}}$ des *MLP*-Netzes die Lyapunov-Exponenten des Systems berechnet werden können.

Hierzu wird der Treppen-Iterations-Algorithmus (vgl. Abschnitte 2.4 und 5.6) eingesetzt. Dieser liefert jedoch entsprechend der gewählten Einbettungsdimension d_E auch ebenso viele Lyapunov-Exponenten. Zur Trennung der zusätzlichen, verfahrensbedingt auftretenden Exponenten (*spurious exponents*) von den tatsächlichen Lyapunov-Exponenten wird das Verfahren bei unveränderten Rekonstruktionparametern und gleicher Topologie des *MLP*-Netzes auf das chaotische Zeitsignal sowohl bei normalem zeitlichen Verlauf als auch bei zeitlich invertiertem Verlauf angewendet. Durch Vergleich der zwei berechneten Sätze von Exponenten gelingt die Trennung der echten und der verfahrensbedingt erzeugten Lyapunov-Exponenten, da sich durch diese Vorgehensweise der Betrag der echten Lyapunov-Exponenten nicht ändert, wohl aber ihr Vorzeichen. Die auf Basis dieser Vorgehensweise für das Lorenz-System und die Hénon-Abbildung ermittelten Werte der Lyapunov-Exponenten stimmen mit hoher Genauigkeit mit den theoretischen Werten überein, wenn die Rekonstruktion der Zustandsvektoren die zuvor beschriebenen Bedingungen erfüllt.

Die vorgestellten Ergebnisse zur nichtlinearen Prädiktion chaotischer Zeitsignale belegen die überragenden Eigenschaften der *MLP*-Netze nahezu beliebige nichtlineare Abhängigkeiten mit sehr hoher Genauigkeit zu approximieren. Diese Prädiktionsgewinne werden maximiert, wenn zusätzlich die ermittelten Kriterien zur optimalen Wahl der Rekonstruktionsparameter auf Basis der charakteristischen Eigenschaften der chaotischen Zeitsignale berücksichtigt werden. Weiterhin steht mit dem neuen *OLL*-Verfahren ein Optimierungsverfahren für *MLP*-Netze zur Verfügung, welches den bekannten nichtlinearen Optimierungsverfahren hinsichtlich Konvergenzgeschwindigkeit, Rechenzeitbedarf und Robustheit weit überlegen ist. Gerade im Vergleich zu den *BP*-Verfahren erzielt das *OLL*-Verfahren bereits nach wenigen (< 5) Iterationen Prädiktionsgewinne, die mit den *BP*-Verfahren erst nach 10^6 Iterationen erreicht werden (vergl. Abbildung 5.13). Hier ist der Konvergenzverlauf der *BP*-Verfahren bei einer hohen Anzahl an Iterationen bereits so flach, daß fälschlicherweise auf Konvergenz des *BP*-Verfahrens geschlossen werden würde, wenn die mit dem *OLL*-Verfahren erzielten Prädiktionsgewinne dieses nicht bereits ausschließen würden!

Da bei der überwiegenden Anzahl praktischer Anwendungen von *MLP*-Netzen fast ausschließlich die verschiedenen Varianten des *BP*-Verfahrens angewendet werden, ist zu erwarten, daß durch Einsatz des *OLL*-Verfahrens wesentlich bessere Ergebnisse bei gleichzeitiger deutlicher Reduktion des Rechenzeitbedarfs erzielt werden können.

Literatur

- [1] B. Aazhang, B.-P. Paris, G. C. Orsak: “*Neural Networks for Multiuser Detection in Code-Division Multiple-Access Communications*”, IEEE Trans. on Communications, Vol. 40, No. 7, pp. 1212-1222, 1992
- [2] H. D. I. Abarbanel, R. Brown, J. B. Kadtke: “*Prediction in Chaotic Nonlinear Systems: Methods for Time Series with Broadband Fourier Spectra*”, Physical Review A, Vol. 41, No. 4, pp. 1782-1807, 1990
- [3] S. Amari: “*Theory of adaptive pattern classifiers*”, IEEE Trans. Electron. Computers, Vol. 16, pp. 299-307, 1967
- [4] B. E. Ambrose, R. M. Goodman: “*Neural Networks Applied to Traffic Management in Telephone Networks*”, Proceedings of the IEEE, Vol. 84, No. 10, pp. 1421-1429, 1996
- [5] U. Arz: “*Optimierung neuronaler Netze mit dem Verfahren der konjugierten Gradienten*”, Studienarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, Dez. 1992
- [6] R. Battiti: “*Accelerating Backpropagation Learning: Two Optimization Methods*”, Complex Systems 3, pp. 331-342, 1989
- [7] R. Battiti: “*First and Second-Order Methods for Learning: between Steepest Descent and Newtons’s Method*”, submitted to IEEE Trans. on Neural Network
- [8] E. M. L. Beale: “*A Derivation of Conjugate Gradients*”, Numerical Methods of Nonlinear Optimization, Academic Press (London), pp.39-43, 1972
- [9] F. Bärmann, F. Biegler-König: “*On a Class of Efficient Learning Algorithms for Neural Networks*”, Neural Networks, Vol. 5, No.1, 1992
- [10] F. Biegler-König, F. Bärmann: “*On a Class of Efficient Learning Algorithms for Multi-Layered Neural Networks*”, Artificial Neural Networks, Vol. 2, I. Aleksander and J. Taylor (Editors), Elsevier Science Publishers B.B., 1992
- [11] A. Brandstater, J. Swift, H. L. Swinney, A. Wolf, J. D. Farmer, E. Jen, J. P. Crutchfield: “*Low-Dimensional Chaos in a Hydrodynamic System*”, Phys. Rev. Lett., Vol. 51, No. 16, pp. 1442-1445, 1983
- [12] D. S. Broomhead, G. P. King: “*Extracting Qualitative Dynamics from Experimental Data*”, Physica D, Vol. 20, pp. 217-236, 1986

- [13] D. S. Broomhead, G. Lowe: "*Multivariable Functional Interpolation and Adaptive Networks*", Complex Systems, Vol. 2, pp. 321-355, 1988
- [14] J. G. Caputo, B. Malraison, P. Atten: "*Determination of Attractor Dimension and Entropy for Various Flows: An Experimentalist's Viewpoint*", G.Mayer-Kress (editor), Dimensions and Entropies in Chaotic Systems, 1986
- [15] M. Casdagli: "*Nonlinear Prediction of Chaotic Time Series*", Physica D 35, pp. 335-356, 1989
- [16] W. E. Caswell, J. A. Yorke: "*Invisible Errors in Dimension Calculations: Geometric and Systematic Effects*", In G. Mayer-Kress, Editor, Dimensions and Entropies in Chaotic Systems, 1986
- [17] Y. le Cun: "*A Learning Scheme for Asymmetric Threshold Networks*", Proc. Cognitiva, Vol. 85, pp. 599-604, 1985
- [18] C. Cramer, E. Gelenbe: "*Low Bit-Rate Video Compression with Neural Networks and Temporal Subsampling*", Proceedings of the IEEE, Vol. 84, No. 10, pp. 1529-1543, 1996
- [19] J. P. Crutchfield, B. S. McNamara: "*Equation of Motion from Data Series*", Complex Systems, Vol. 1, pp. 417-452, 1987
- [20] G. Cybenko: "*Approximation by Superpositions of a Sigmoidal Function*", Math. Control Signals Systems, Vol. 2, pp. 303-314, 1989
- [21] S. D. Day, M.R. Davenport: "*Continuous-Time Temporal Back-Propagation with Adaptable Time Delays*", IEEE Trans. on Neural Networks, Vol. 4, No. 2, pp. 348-354, 1993
- [22] J. E. Dennis, R. B. Schnabel: "*Numerical Methods for Unconstrained Optimization and Nonlinear Equations*", Prentice Hall, Englewood Cliffs, NJ, 1983
- [23] J. Deppisch, H. U. Bauer, T. Geisel: "*Hierarchical Training of Neural Networks and Prediction of Chaotic Time Series*", Preprint
- [24] J.-P. Eckmann, D. Ruelle: "*Ergodic Theory of Chaos and Strange Attractors*", Reviews of Modern Physics, Vol. 57, No. 3, Part I, Juli 1985.
- [25] T. Endo, L. O. Chua: "*Synchronizing Chaos from Electronic Phased-Locked Loops*", Int. Jour. of Bifurcation and Chaos, Vol. 1, No. 3, pp. 701-710, 1991
- [26] L. Fox, J. H. Wilkinson. *NAG Fortran Library Manual Mark 13*.

- [27] J.-P. Eckmann, S. O. Kamphorst, D. Ruelle, S. Ciliberto: "*Liapunov exponents from time series*", Physical Review A, Vol. 34, No. 6, pp. 4971-4979, December 1986.
- [28] S. Ergezinger, E. Thomsen: "*An Accelerated Learning Algorithm for Multilayer Perceptrons: Optimization Layer by Layer*", IEEE Trans. on Neural Networks, Vol. 6, No. 1, pp. 31-42, Januar 1995
- [29] S. E. Fahlmann: "*Faster Learning Variations on Back-Propagation: An Empirical Study*", Proc. of the 1988 Connectionists Models Summer School, pp. 38-51, 1989
- [30] J. D. Farmer, J. J. Sidorowich: "*Predicting Chaotic Time Series*", Physical Review Letters, Vol. 59, No. 8, pp. 845-848, 1987
- [31] J. D. Farmer, J. J. Sidorowich: "*Predicting Chaotic Dynamics*", Dynamic Patterns in Complex Systems, World Scientific, 1988
- [32] J. D. Farmer, J. J. Sidorowich: "*Optimal Shadowing and Noise Reduction*", Los Alamos National Laboratory, Los Alamos, Technical Report LA-UR-90-653, 1990
- [33] J. D. Farmer, J. J. Sidorowich: "*Optimal Shadowing and Noise Reduction*", Physica D, Vol. 47, pp. 373-392, 1991
- [34] R. Fletcher: "*Practical Methods of Optimization*", Vol. 1: Unconstrained Optimization, John Wiley and Sons, Chichester, 1980
- [35] A. D. Fraser, H. L. Swinney: "*Independent Coordinates for Strange Attractors from Mutual Information*", Phys. Rev. A, Vol. 35, No. 2, pp. 1134-1140, 1986
- [36] A. R. Gallant, H. White: "*There Exists a Neural Network that does not make Avoidable Mistakes*", Proc. of the Int. Conf. on Neural Networks, Vol. I, pp. 657-664, 1988
- [37] K. Geist, U. Parlitz, W. Lauterborn: "*Comparison of Different Methods for Computing Lyapunov-Exponents*", Progress of Theoretical Physics, Vol. 83, No. 5, pp. 875-893, 1990
- [38] V. Gibiat: "*Phase Space Representation of Acoustical Musical Signals*", Journal of Sound and Vibration, Vol. 123, No. 3, pp. 529-536, 1988
- [39] P. E. Gill, W. Murray, M. H. Wright: "*Practical Optimization*", Academic Press Inc. (London), 1981
- [40] K. Goerke: "*Untersuchungen zum Einsatz von rezeptiven Feldern in neuronalen Netzen zur Prädiktion von Musiksignalen*", Diplomarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, Nov. 1991

- [41] P. Grassberger: “*On the Fractal Dimension of the Hénon Attractor*”, Physics Letters, Vol. 97A, No. 6, pp. 224-226, 1983
- [42] P. Grassberger: “*Generalized Dimensions of Strange Attractors*”, Physics Letters, Vol. 97A, No. 6, pp. 227-230, 1983
- [43] P. Grassberger, I. Procaccia: “*Measuring the Strangeness of Strange Attractors*”, Physica D, 9:pp. 189-208, 1983
- [44] P. Grassberger, I. Procaccia: “*Characterization of Strange Attractors*”, Phys. Rev. Lett., Vol. 50, No. 5, pp. 346-349, 1983
- [45] I. W. Habib: “*Applications of Neurocomputing in Traffic Management of ATM Networks*”, Proceedings of the IEEE, Vol. 84, No. 10, pp. 1430-1441, 1996
- [46] H. Haken: “*At Least one Lyapunov-Exponent vanishes if the Trajectory of an Attractor does not contain a Fixed Point*”, Physics Letters, Vol. 94A, No. 2, pp. 71-72, 1983
- [47] S. M. Hammel: “*A Noise Reduction Method for Chaotic Systems*”, Physics Letters A, Vol. 148, No. 8-9, pp. 421-428, 1990
- [48] M. J. Hasler: “*Electrical Circuits with Chaotic Behavior*”, Proceedings of the IEEE, Vol. 75, No. 8, pp. 1009-1021, 1987
- [49] S. Hayes, C. Grebogi, E. Ott: “*Communicating with Chaos*”, Physical Review Letters, Vol. 70, No. 20, pp. 3031-3034, 1993
- [50] S. Haykin: “*Neural Networks Expand SP's Horizons*”, IEEE Signal Processing Magazine, Vol. 13, No. 2, pp. 24-49, 1996
- [51] S. Haykin, X. B. Li: “*Detection of Signals in Chaos*”, Proceedings of the IEEE, Vol. 83, No. 1, pp. 95-122, 1995
- [52] Z. He, H. Li: “*Nonlinear Predictive Image Coding with a Neural Network*”, Proc. 1990 IEEE ICASSP, pp.1009-1012, 1990
- [53] R. Hecht-Nielsen: “*Kolmogorov's Mapping Neural Network Existence Theorem*”, IEEE First Intern. Conf. on Neural Networks, Vol. III, 1987
- [54] R. Hecht-Nielsen: “*Theory of Backpropagation Neural Network*”, Proc. of the Int. Joint Conf. on Neural Networks, Vol. I, pp. 593-611, 1989
- [55] R. Hecht-Nielsen: “*Neurocomputing*”, Addison-Wesley, 1990
- [56] M. Hénon: “*A Two-Dimensional Mapping with a Strange Attractor*”, Commun. Math. Phys., Vol. 50, p. 69, 1976

- [57] H. Heuser: *“Lehrbuch der Analysis, Teil 1 und 2”*, B. G. Teubner, Stuttgart, 9. Auflage, 1991
- [58] J. Hertz, A. Krogh, R. G. Palmer: *“Introduction to the Theory of Neural Computation”*, Addison-Wesley, 1991
- [59] J. Holzfuss, G. Mayer-Kress: *“An Approach to Error-Estimation in the Application of Dimension Algorithms”*, In G. Mayer-Kress, Editor, *Dimensions and Entropies in Chaotic Systems*, 1986
- [60] F. Hunt, F. Sullivan: *“Efficient Algorithms for Computing Fractal Dimensions”*, In G. Mayer-Kress, Editor, *Dimensions and Entropies in Chaotic Systems*, 1986
- [61] B. Irie, S. Miyake: *“Capabilities of Three-layered Perceptrons”*, Proc. of the Int. Conf. on Neural Networks, I, pp. 641-648, IEEE Press, New York, 1988
- [62] K. I. Iso, T. Watanabe: *“Speaker-Independent Word Recognition using a Neural Prediction Model”*, Proc. 1990 IEEE ICASSP, pp.441-444, 1990
- [63] Y. Ito: *“Approximation of Continuous Functions on \mathbb{R}^d by Linear Combinations of Shifted Rotations of a Sigmoid Function With and Without Scaling”*, Neural Networks, Vol. 5, pp. 105-115, 1992
- [64] R. A. Jacobs: *“Increased Rates of Convergence through Learning Rate Adaptation”*, Neural Networks, Vol. 1, pp. 295-307, 1988
- [65] N. S. Jayant, P. Noll: *“Digital Coding of Waveforms”*, Prentice-Hall, New Jersey, 1984
- [66] R. D. Jones, Y. C. Lee, C. W. Barnes, G. W. Flake, K. Lee, P. S. Lewis, S. Qian: *“Functional Approximation and Time Series Prediction with Neural Networks”*, Int. Joint Conf. on Neural Networks, pp. 649-665, 1990
- [67] K. D. Kammeyer: *“Nachrichtenübertragung”*, B. G. Teubner Stuttgart, 1992
- [68] L. Kocarev, K. S. Halle, K. Eckert, L. O. Chua, U. Parlitz: *“Experimental Demonstration of Secure Communications via Chaotic Synchronization”*, Int. Jour. of Bifurcation and Chaos, Vol. 2, No, 3, pp. 709-713, 1992
- [69] A. N. Kolmogorov: *“On the Representation of Continuous Functions of Many Variables by Superposition of Continuous Functions of One Variable and Addition”*, Dokl. Akad. Nauk USSR, 114, pp. 953-956, 1957
- [70] E. J. Kostelich, J. A. Yorke: *“Noise Reduction: Finding the Simplest Dynamical System Consistent with the Data”*, Physica D, Vol. 41, pp. 183-196, 1990

- [71] A. Lapedes, R. Farber: *“Nonlinear Signal Processing Using Neural Networks: Prediction and System Modelling”*, Los Alamos National Laboratory, Los Alamos, Technical Report LA-UR-87-2662, 1987
- [72] A. Lapedes, R. Farber: *“How neural nets work”*, in *Neural Information Processing Systems*, D. Z. Anderson, Ed. New York: Am. Inst. Physics, pp. 442-456, 1988
- [73] S. P. Layne, G. Mayer-Kress, J. Holzfuss: *“Problems Associated with Dimensional Analysis of Electroencephalogram Data”*, G. Mayer-Kress (editor), *Dimensions and Entropies in Chaotic Systems*, pp. 246-256, 1986
- [74] E. Levin: *“Word Recognition using Hidden Control Neural Architecture”*, Proc. 1990 IEEE ICASSP, pp.433-436, 1990
- [75] T. Li, J. A. Yorke: *“Period three implies chaos”*, Am. Math. Monthly,, Vol. 82, pp. 985, 1975
- [76] W. Liebert, H. G. Schuster: *“Proper Choice of the Time Delay for the Analysis of Chaotic Time Series”*, Phys. Letters A, Vol. 142, No. 2,3, pp. 107-111, 1989
- [77] R. P. Lippmann: *“An Introduction to Computing with Neural Nets”*, IEEE ASSP Magazine, Vol. 4, No. 2, ISSN 0740-7467, April 1987
- [78] E. N. Lorenz: *“Deterministic Nonperiodic Flow”*, J. Atmos. Sci., Vol. 20, pp. 130, 1963
- [79] D.G. Luenberger: *“Linear and Nonlinear Programming”*, Addison-Wesley, 1984
- [80] O. Macchi, M. Jaidane-Saidane: *“Adaptive IIR Filtering and Chaotic Dynamics: Application to Audiofrequency Coding”*, IEEE Trans. on Circuits and Systems, Vol. 36, No. 4, pp. 591-599, 1989
- [81] M. C. Mackey, L. Glass: *“Oscillation and Chaos in Physiological Control Systems”*, Science, Vol. 197, pp. 287-289, 1977
- [82] C. Maganza, R. Causse: *“Bifurcations, Period Doublings and Chaos in Clarinetlike Systems”*, Europhysics Letters, Vol. 1, No. 6, pp. 296-302, 1986
- [83] J. Makhoul: *“Linear Prediction: A Tutorial Review”*, Proc. IEEE, Vol. 63, No. 4, pp. 561-580, April 1975
- [84] B. Malraison, P. Atten, P. Berge, M. Dubois: *“Dimension of Strange Attractors: An Experimental Determination for Chaotic Regimes of two Convective Systems”*, J. Phys. Letters, Vol. 44, No. 22, pp. 897-902, 1983

- [85] R. Mañé: “*On the Dimension of the Compact Invariant Sets of Certain Nonlinear Maps,*”, in *Dynamical Systems and Turbulence*, Warwick 1980, Lect. Notes in Mathematics 898, (Springer, New York), pp. 230-242, 1981
- [86] P. Maragos: “*Fractal Aspects of Speech Signals: Dimension and Interpolation,*”, Proc. 1991 IEEE ICASSP, pp.417-420, 1991
- [87] J. Marquardsen: “*Modellierung von Musiksignalen auf der Basis von radialen Basisfunktionen*”, Diplomarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, 1991
- [88] T. Matsumoto: “*Chaos in Electronic Circuits*”, Proceedings of the IEEE, Vol. 75, No. 8, August 1987
- [89] M. Minsky, S. Papert: “*Perceptrons*”, MIT Press, Cambridge MA, 1969
- [90] M. F. Møller: “*A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning*”, PB-339 Preprint, Computer Science Department, University of Aarhus, Denmark, 1990
- [91] J. Moody, Ch. Darken: “*Learning with Localized Receptive Fields*”, Proceedings of the Connectionist Models Summer School, 1988
- [92] J. Moody, Ch. Darken: “*Fast Learning in Networks of Locally-Tuned Processing Units*”, Neural Computation, Vol. 1, No. 2, pp. 281-294, 1989
- [93] J. Moody: “*Fast Learning in Multi-Resolution Hierarchies*”, Advances in Neural Information Processing Systems 1, pp. 29-39, San Mateo, CA: Morgan Kaufmann, 1989
- [94] N. Morgan, H. Bourlard: “*Continuous Speech Recognition using Multilayer Perceptrons with Hidden Markov Models*”, Proc. 1990 IEEE ICASSP, pp.413-416, 1990
- [95] K. S. Narendra: “*Neural Networks for Control: Theory and Practice*”, Proceedings of the IEEE, Vol. 84, No. 10, pp. 1385-1406, 1996
- [96] A. V. Oppenheim, G. W. Wornell, S. H. Isabelle, K. M. Cuomo: “*Signal Processing in the Context of Chaotic Signals*”, Proc. 1992 IEEE ICASSP, pp. IV117-IV120, 1992
- [97] S. S. Oren, E. Spedicato: “*Optimal Conditioning of Self-Scaling Variable Metric Algorithms*”, Math. Programming, Vol. 10, pp. 70-90, 1976
- [98] V. I. Oseledec: “*A multiplicative ergodic theorem. Lyapunov characteristic numbers for dynamical systems*”, Trudy Mosk. Mat. Obsc. 19, 179, 1968

- [99] F. Osterloh: "*Modellierung dynamischer Systeme auf der Basis rückgekoppelter neuronaler Netze*", Diplomarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, 1992
- [100] N. H. Packard, J. P. Crutchfield, J. D. Farmer, R. S. Shaw: "*Geometry from a time series*", Phys. Rev. Lett. 45, 712-716, 1980
- [101] K. K. Paliwal: "*Neural Net Classifiers for Robust Speech Recognition under Noisy Environments*", Proc. 1990 IEEE ICASSP, pp. 429-432, 1990
- [102] N. G. Panagiotidis, D. Kalogeras, S. D. Kollias, A. Stafylopatis: "*Neural Network-Assisted Effective Lossy Compression of Medical Images*", Proceedings of the IEEE, Vol. 84, No. 10, pp. 1474-1487, 1996
- [103] A. Papoulis: "*Probability, Random Variables and Stochastic Processes*", McGraw-Hill, Second Edition, 1984
- [104] D. B. Parker: "*Learning Logic*", Tech. Rep. TGR-47, M.I.T. Sloan School of Management, Cambridge, MA, 1982
- [105] T. S. Parker, L. O. Chua: "*Practical Numerical Algorithms for Chaotic Systems*", Springer, New York, 1989
- [106] U. Parlitz: "*Der Begriff der Resonanz und sein Zusammenhang mit der Torsion des lokalen Flusses bei nichtlinearen Oszillatoren*", Dissertation, Georg-August-Universität zu Göttingen, 1987
- [107] U. Parlitz: "*Identification of True and Spurious Lyapunov-Exponents from Time Series*", Int. Jour. of Bifurcation & Chaos, Vol. 2, No. 1, pp. 155-165, 1992
- [108] U. Parlitz, L. O. Chua, L. Kocarev, K. S. Halle, A. Shang: "*Transmission of Digital Signals by Chaotic Synchronization*", Int. Jour. of Bifurcation and Chaos, Vol. 2, No. 3, pp. 973-981, 1992
- [109] U. Parlitz, S. Ergezinger: "*Robust Communication Based on Chaotic Sequences*", Physics Letters A., Vol. 188, pp. 146-150, 1994
- [110] T. S. Parker, L. O. Chua: "*Chaos: A Tutorial for Engineers*", Proc. IEEE, Vol. 75, No. 8, pp. 982-1008, 1987
- [111] P. Plaschko, K. Brod: "*Höhere mathematische Methoden für Ingenieure und Physiker*", Springer, 1989
- [112] D. C. Plaut, S. Nowlan, G. Hinton: "*Experiments on Learning by Back Propagation*", Technical Report CMU-CS-86-126, Carnegie-Mellon University, 1986

- [113] T. Poggio, F. Girosi: "*Networks for Approximation and Learning*", Proc. of the IEEE, Vol. 78, No. 9, Sept. 1990
- [114] M. J. D. Powell, F. Girosi: "*Some Convergence Properties of the Conjugate Gradient Method*", Math. Programming, Vol. 11, pp. 42-49, 1976
- [115] W. H. Press, B. P. Flannery, S. A. Teukolsky, W. T. Vetterlin: "*Numerical Recipes in C*", Cambridge University Press, 1989
- [116] F. Reinhardt, H. Soeder: "*dtv-Atlas zur Mathematik, Band 1 und 2*", Deutscher Taschenbuch Verlag, 9. Auflage, November 1991
- [117] H. Reiningger, D. Wolf: "*Nonlinear Prediction of Stochastic Processes using Neural Networks*", Signal Processing V: Theories and Applications, Elsevier Science Publishers B. V., pp. 1623-1626, 1990
- [118] A. Rényi: "*On the Dimension and Entropy of Probability Distributions*", In Trans. of the Second Prague Conf. on Information Theory, Statistical Decision Functions, Random Processes, pp. 545-556, 1959
- [119] S. A. Rizvi, L. C. Wang, N. M. Nasrabadi: "*Neural Network Architectures for Vector Prediction*", Proceedings of the IEEE, Vol. 84, No. 10, pp. 1513-1528, 1996
- [120] A. Roebel: "*Beschreibung von Musiksignalen basierend auf der Theorie nichtlinearer Systeme*", Diplomarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, Nov. 1990
- [121] F. Rosenblatt: "*The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain.*", Psychological Review, Vol. 65, pp. 386-408, 1958
- [122] J. C. Roux, R. H. Simoyi, H. L. Swinney: "*Observation of a Strange Attractor*", Physica D, Vol. 8D, No. 1-2, pp. 257-266, 1983
- [123] D. E. Rumelhart, G. E. Hinton, R. J. Williams: "*Learning Internal Representations by Error Propagation*", in D. E. Rumelhart und J. L. McClelland (Eds.) *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, Vol. 1, MIT Press, 1986
- [124] D. Ruelle, F. Takens: "*On the Nature of Turbulence*", Commun. Math. Phys., Vol. 21, p. 21, 1971
- [125] D. Ruelle: "*Characteristic Exponents for a Viscous Fluid subject to Time Dependent Forces*", Commun. Math. Phys., Vol. 93, pp. 285-300, 1984
- [126] D. Ruelle: "*Chaotic Evolution and Strange Attractors*", Cambridge University Press, Cambridge, 1989

- [127] D. Ruelle: "*Deterministic Chaos: the Science and the Fiction*", Proc. R. Soc. Lond. A 427, pp. 241-248, 1989
- [128] T. D. Sanger: "*A Tree-Structured Adaptive Network for Function Approximation in High-Dimensional Spaces*", IEEE Trans. on Neural Networks, Vol. 2, No. 2, Mar. 1991
- [129] M. Sano, Y. Sawada: "*Measurement of Lyapunov Spectrum from a Chaotic Time Series*", Physical Review Letters, Vol. 55, No. 10, Sept. 1985
- [130] H. G. Schuster: "*Deterministic Chaos*", Physik-Verlag, 1984
- [131] D. F. Shanno: "*Conjugate Gradient Methods with Inexact Searches*", Mathematics of Operations Research, Band 3, Nr. 3, S. 244-256, August 1978
- [132] S. Smarsli: "*Untersuchung eines schnellen Lernalgorithmus zur Optimierung mehrlagiger neuronaler Netze*", Diplomarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, 1992
- [133] J. Spitzkowski: "*Prädiktion von Musiksignalen mit neuronalen Netzen*", Diplomarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, 1991
- [134] M. Stinchcombe und H. White: "*Universal Approximation using Feedforward Networks with Non-Sigmoid Hidden Layer Activation Functions*", Int. Joint. Conf. on Neural Networks, June 1989
- [135] P. Strohbach: "*New Forms of Levinson and Schur Algorithms*", IEEE Signal Processing Magazine, pp. 12-36, January 1991
- [136] F. Takens: "*Detecting Strange Attractors in Turbulence*", in Dynamical Systems and Turbulence, Warwick 1980, Lecture Notes in Math. 898 (Springer Verlag 1981), pp. 366-381
- [137] N. Tishby: "*A Dynamical Systems Approach to Speech Processing*", Proc. 1990 IEEE ICASSP, pp. 365-368, 1990
- [138] E. Thomsen: "*Einsatz von hierarchischen neuronalen Netzen zur Prädiktion von Musiksignalen*", Diplomarbeit, Institut für Allgemeine Nachrichtentechnik, Universität Hannover, Nov. 1991
- [139] R. Unbehauen: "*Systemtheorie*", 5. Auflage, Oldenbourg, 1990
- [140] M. de Sousa Vieira, A. L. Lichtenberg, M. A. Lieberman: "*Nonlinear Dynamics of Self-Synchronizing Systems*", Int. Jour. of Bifurcation and Chaos, Vol. 1, No. 3, pp. 691-699, 1991

- [141] T. P. Vogel, J. K. Mangis, A. K. Rigler, W. T. Zink, D. L. Alkon: “*Accelerating the Convergence of the Back-Propagation Method*”, *Biological Cybernetics*, 59, pp. 257-263, 1988
- [142] R. L. Watrous: “*Learning Algorithms for Connectionist Networks: Applied Gradient Methods of Nonlinear Optimization*”, *IEEE Int. Conf. on Neural Networks*, Vol. 2, pp. 619-627, 1987
- [143] A. S. Weigend, B. A. Huberman, D. E. Rumelhart: “*Predicting the Future: A Connectionist Approach*“, *Int. Journal of Neural Systems*, April 1990
- [144] P. Werbos: “*Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Science*“, Ph. D. thesis, Harvard Univ., 1974
- [145] H. Whitney: “*Differentiable Manifolds*”, *Ann. Math.* Vol. 37, p. 645, 1936
- [146] A. Wolf, J. B. Swift, H. L. Swinney, J. A. Vastano: “*Determining Lyapunov-Exponents from a Time Series*”, *Physica 16D*, pp. 285-317, 1985
- [147] S. Wu: “*Chua’s Circuit Family*”, *Proceedings of the IEEE*, Vol. 75, No. 8, pp. 1022-1032, 1987
- [148] L. S. Young: “*Entropy, Lyapunov-Exponents and Hausdorff Dimension in Differential Dynamical Systems*”, *IEEE Trans. on Circuits and Systems*, CAS-30(8), pp. 599-607, 1983
- [149] B. P. Yuhas, M. H. Goldstein, T. J. Sejnowski, R. E. Jenkins: “*Neural Network Models of Sensory Interaction for Improved Vowel Recognition*”, *Proceedings of the IEEE*, Vol. 78, No. 10, pp. 1658-1668, 1990

A Untersuchte chaotische Systeme

In der vorliegenden Arbeit wird die Leistungsfähigkeit der vorgestellten Algorithmen am Beispiel berechneter Referenzsignale beurteilt und verglichen.

Diese Zeitsignale stellen die Lösungen bzw. einzelne Komponenten der Lösungen dreier unterschiedlicher chaotischer Systeme dar. Dieses sind das Lorenz-System, die Hénon-Abbildung sowie die Mackey-Glass-Differentialgleichung.

Die Auswahl der Systeme sowie der die Eigenschaften dieser Systeme beeinflussenden Parameter ist so erfolgt, daß ein Vergleich mit aus der Literatur vorliegenden Untersuchungsergebnissen möglich ist. Die im folgenden kurz beschriebenen chaotischen Systeme bzw. chaotischen Zeitsignale werden in der Literatur häufig verwendet und stellen somit gewissermaßen *Benchmark-Signale* dar.

Die Referenzsignale wurden in doppelt genauer Fließkommadarstellung gemäß IEEE 754 berechnet und im weiteren verwendet. Zusätzlich wurde eine Normierung der Signale durchgeführt, so daß der Gleichanteil über das Gesamtsignal 0,5 betrug und die einzelnen Abtastwerte auf den Bereich $[0,0 ; 1,0]$ beschränkt waren. Die Auswirkungen von Ungenauigkeiten in der Zahlendarstellung wurden nicht näher betrachtet.

A.1 Das Lorenz-System

Eines der bekanntesten dynamischen Systeme, an dem der Meteorologe E. N. Lorenz [78] in den frühen sechziger Jahren als einer der ersten chaotisches Verhalten untersuchte, ist das nach ihm benannte Lorenz-System. Dieses System hat einen $d_S = 3$ -dimensionalen Zustandsraum und wird durch folgende Gleichungen beschrieben:

$$\begin{aligned} \frac{dx(t)}{dt} &= \sigma \cdot (y(t) - x(t)) \\ \frac{dy(t)}{dt} &= r \cdot x(t) - y(t) - x(t)z(t) \\ \frac{dz(t)}{dt} &= x(t)y(t) - b \cdot z(t) \end{aligned} \tag{A.1}$$

Lorenz entwickelte dieses stark vereinfachte Modell ausgehend von Navier-Stokes-Gleichung, Kontinuitäts- und Wärmeleitgleichung für die Bewegung der Erdatmosphäre.

Die Parameter in den Gleichungen hängen mit der Prandl-Zahl (σ), geometrischen Abmessungen (b) und der Temperaturdifferenz (r) zusammen. Die Lorenz-Gleichungen beschreiben für alle Parameterkombinationen, für die $\sigma > 0 \wedge b > 0$ gilt, ein dissipatives System. Für $r < 1$ besitzt das System einen stabilen Gleichgewichtspunkt mit den Koordinaten $(0; 0; 0)$. Wird $r > 1$, so wechselt das System in einen anderen stabilen Gleichgewichtspunkt. Bei weiterer Erhöhung der Temperaturdifferenz entsteht zunächst periodisches Verhalten, welches schließlich in chaotisches Verhalten übergeht.

Für die im Rahmen dieser Arbeit verwendeten Parameter $\sigma=16$, $r=45,92$ und $b=4$ weist das System chaotisches Verhalten auf und besitzt die folgenden charakteristischen Größen:

$$\textit{Fraktale} - \textit{Dimensionen} \quad (\text{A.2})$$

$$d_C = 2,06$$

$$d_K = 2,07$$

$$\textit{Lyapunov} - \textit{Exponenten} \quad (\text{A.3})$$

$$\lambda_1 = 2,16$$

$$\lambda_2 = 0$$

$$\lambda_3 = -32,4$$

Die Lösung der Lorenz-Gleichungen wurde mittels einer Runge-Kutta-Integration 4ter Ordnung berechnet [105]. Bei der Integration wurde die Schrittweite $dt=0,01$ mit dem Anfangswert $(0,0;0,1;0,0)$ gewählt. Um das Einschwingen des Systems auf dem Attraktor zu gewährleisten, wurden die ersten 10 000 Zustandsvektoren verworfen. Für die in den Abschnitten 5.1, 5.2 sowie 5.6.1 durchgeführten Untersuchungen wurden die folgenden Signale mit jeweils $N=110\,000$ Werten bzw. Vektoren verwendet, wobei $x(t)$, $y(t)$ und $z(t)$ die Lösung von Gl.(A.1) darstellen.

$\mathbf{L}_{x,0,01}$: x-Komponente der Lösung des Lorenz-Systems für $T_A=0,01$

$$\mathbf{L}_{x,0,01} = \{x_{0,01}^k\}_1^N \quad \text{mit} \quad x_{0,01}^k = x(100 + k \cdot 0,01) ; k = 1, \dots, N \quad (\text{A.4})$$

$\mathbf{L}_{x,0,1}$: x-Komponente der Lösung des Lorenz-Systems für $T_A=0,1$

$$\mathbf{L}_{x,0,1} = \{x_{0,1}^k\}_1^N \quad \text{mit} \quad x_{0,1}^k = x(100 + k \cdot 0,1) ; k = 1, \dots, N \quad (\text{A.5})$$

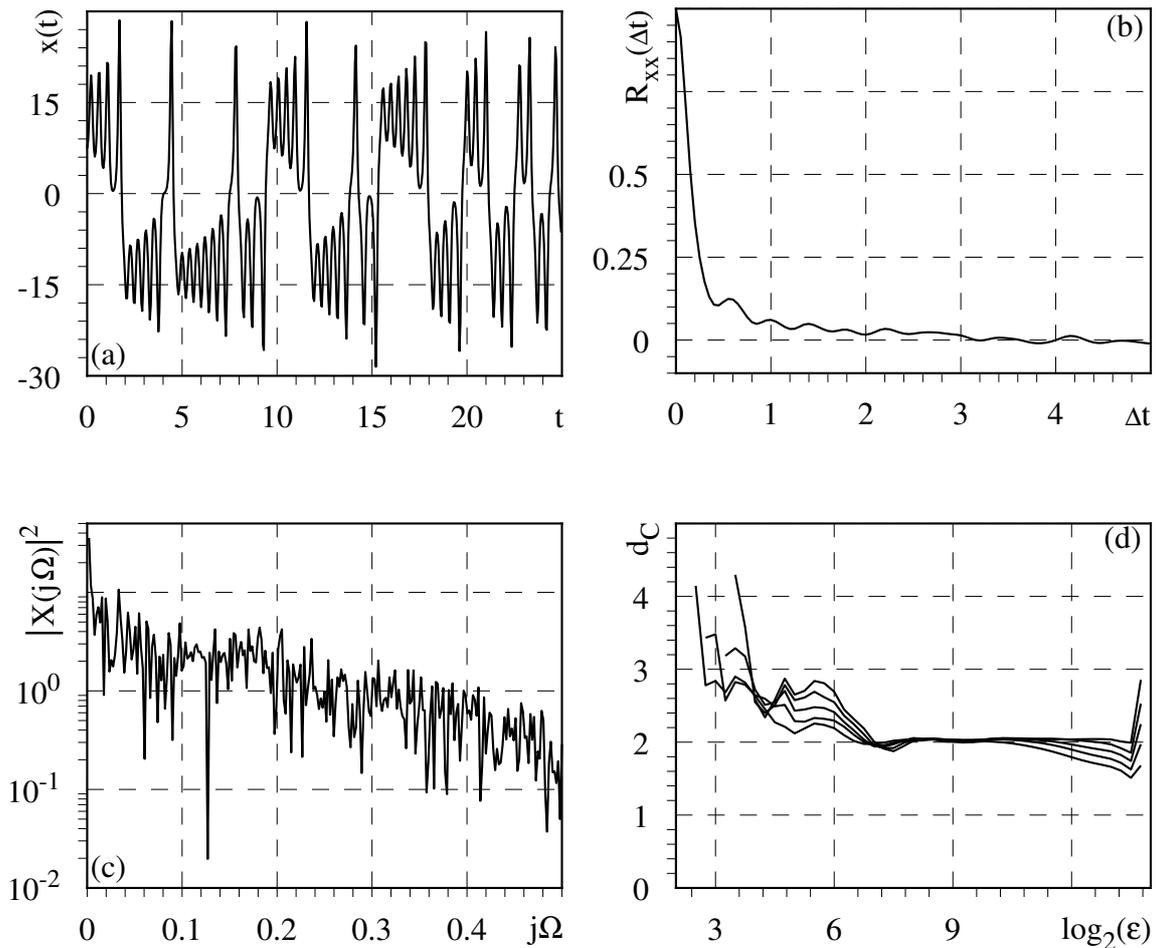
$\mathbf{L}_{x,0,1}^{-1}$: zeitlich invertierte x-Komponente der Lösung des Lorenz-Systems für $T_A=0,1$

$$\mathbf{L}_{x,0,1}^{-1} = \{x_{0,1}^{N+1-k}\}_1^N \quad \text{mit} \quad x_{0,1}^k = x(100 + k \cdot 0,1) ; k = 1, \dots, N \quad (\text{A.6})$$

$\mathbf{L}_{xyz,0,1}$: Lösung des Lorenz-Systems für $T_A=0,1$

$$\begin{aligned} \mathbf{L}_{xyz,0,1} &= \{(x_{0,1}^k, y_{0,1}^k, z_{0,1}^k)^T\}_1^N \quad \text{mit} \quad x_{0,1}^k = x(100 + k \cdot 0,1) ; k = 1, \dots, N \\ & \quad y_{0,1}^k = y(100 + k \cdot 0,1) ; k = 1, \dots, N \\ & \quad z_{0,1}^k = z(100 + k \cdot 0,1) ; k = 1, \dots, N \end{aligned} \quad (\text{A.7})$$

Abbildung A.1 zeigt einige charakteristische Größen der x-Komponente $\mathbf{L}_{x,0,1}$ bzw. $\mathbf{L}_{x,0,01}$ der Lösung der Lorenz-Gleichungen. Der typische Verlauf des Zeitsignals $x(t)$ ist durch die Form des Lorenz-Attraktors begründet. Die graphische Darstellung der Zustandsvektoren $\mathbf{L}_{xyz,0,1}$ und damit des Attraktors ist in Bild 2.5 (a) in Kapitel 2.5 wiedergegeben. Die beiden "Flügel" des Attraktors werden abwechselnd durchlaufen, wobei die Verweildauer immer unterschiedlich ist.

Abbildung A.1: Charakteristische Kenngrößen der x -Komponente des Lorenz-Attraktors

- (a) Zeitsignal (b) normierte AKF
(c) Leistungsdichtespektrum (d) Korrelationsdimension

Entgegen dem ersten Eindruck besitzt das Zeitsignal keine ausgeprägte Periodizität und nur geringe lineare Bindungen. Dieses wird u. a. aus dem Verlauf der normierten Autokorrelationsfunktion und aus dem Leistungsdichtespektrum deutlich. Bild A.1 (d) zeigt die mittlere Steigung des Korrelationsintegrals Gl.(2.14), welches mit dem Verfahren von Grassberger und Procaccia [41, 42, 43, 44] berechnet wurde. Die hieraus gemäß Gl.(2.17) bestimmte Korrelationsdimension stimmt gut mit dem theoretischen Wert $d_C=2,06$ überein.

A.2 Die Mackey-Glass-Differentialgleichung

Ein zweites, in der Literatur ([71], [128], [21] und [91]) häufig untersuchtes Zeitsignal mit chaotischen Eigenschaften entsteht durch numerische Integration der folgenden Differen-

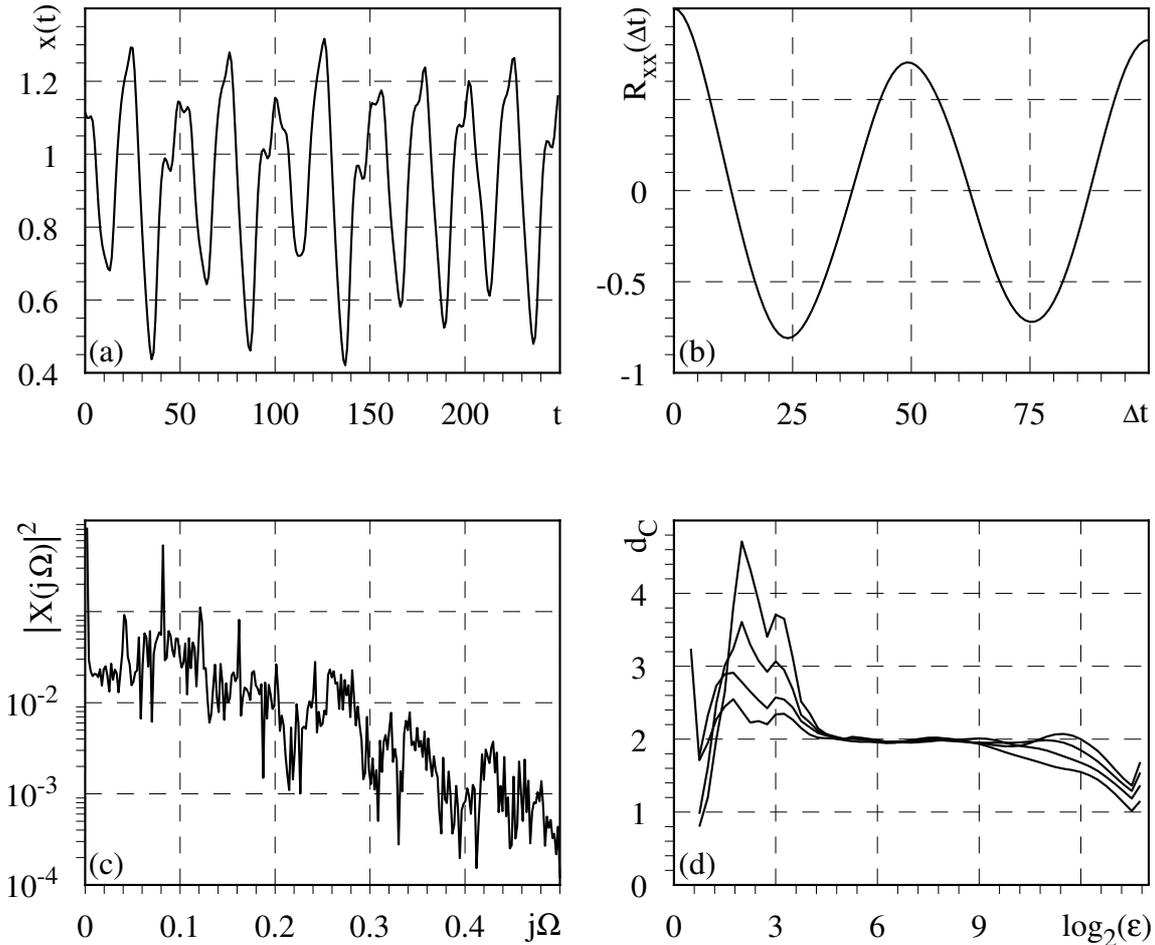


Abbildung A.2: Charakteristische Kenngrößen der Lösung der Mackey-Glass-Differentialgleichung für $\Delta T=17$

- (a) Zeitsignal (b) normierte AKF
(c) Leistungsdichtespektrum (d) Korrelationsdimension

tialgleichung mit retardiertem Argument:

$$\frac{dx(t)}{dt} = \frac{a \cdot x(t - \Delta T)}{1 + x(t - \Delta T)^{10}} - b \cdot x(t) \quad (\text{A.8})$$

Dieses System wurde zuerst von M. Mackey und L. Glass [81] für die Modellierung biologischer Vorgänge beschrieben. Für die Wahl von $a=0,2$ und $b=0,1$ ändert sich mit wachsendem ΔT das asymptotische Verhalten des Systems vom Gleichgewichtspunkt über periodische Lösungen bis hin zu chaotischem Verhalten. In der Literatur wird im wesentlichen der chaotische Bereich mit $\Delta T=17$ und $\Delta T=30$ untersucht.

Auf Grund des retardierten Argumentes $x(t - \Delta T)$ ist der Zustandsraum dieses dynamischen Systems ∞ -dimensional, jedoch ist die stationäre Lösung auf einen niedrigdimensionalen Attraktor beschränkt.

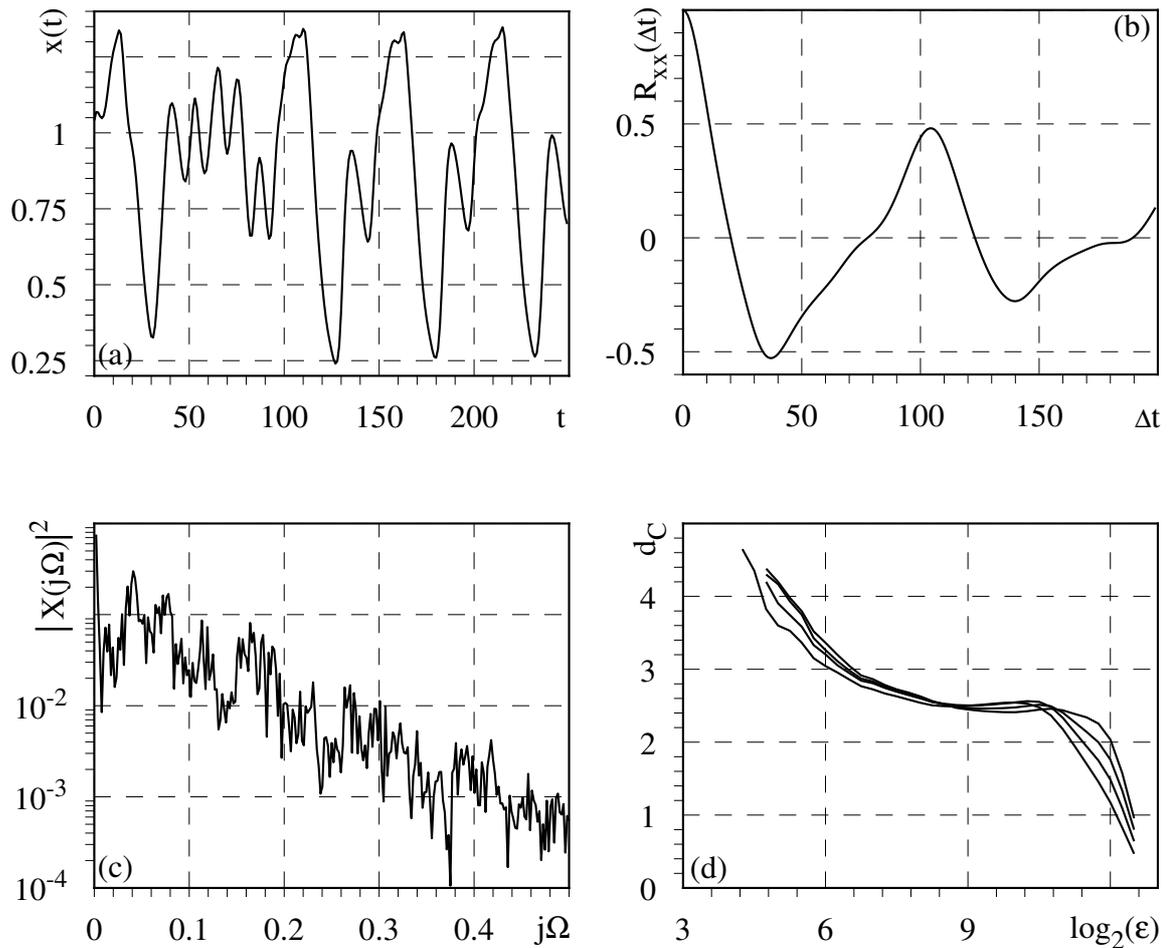


Abbildung A.3: Charakteristische Kenngrößen der Lösung der Mackey-Glass-Differentialgleichung für $\Delta T=30$

- (a) Zeitsignal (b) normierte AKF
(c) Leistungsdichtespektrum (d) Korrelationsdimension

Die Lösung der Mackey-Glass-Differentialgleichung wurde wieder unter Verwendung einer Runge-Kutta-Integration 4ter Ordnung berechnet [105]. Als Anfangsbedingung wurde $x(t)=0,8$ for $t \leq 0$ angesetzt und die Schrittweite zu $dt=1,0$ gewählt. Die ersten 10 000 Werte wurden wiederum verworfen, um das Einschwingen des Systems auf dem Attraktor zu gewährleisten.

Für die in den Abschnitten 5.3.1 und 5.3.2 durchgeführten Untersuchungen wurden die folgenden Signale mit jeweils $N=110\,000$ Werten verwendet, wobei $x(t)$ die Lösung von Gl.(A.8) für $\Delta T=17$ bzw. $\Delta T=30$ darstellt.

MG₁₇ : Lösung der Mackey-Glass-Differentialgleichung für $\Delta T=17$ und $T_A=1$

$$\mathbf{MG}_{17} = \{x^k\}_1^N \quad \text{mit} \quad x^k = x(10000 + k \cdot 1) ; k = 1, \dots, N \quad (\text{A.9})$$

\mathbf{MG}_{30} : Lösung der Mackey-Glass-Differentialgleichung für $\Delta T=30$ und $T_A=1$

$$\mathbf{MG}_{30} = \{x^k\}_1^N \quad \text{mit} \quad x^k = x(10000 + k \cdot 1) ; k = 1, \dots, N \quad (\text{A.10})$$

Die charakteristischen Größen dieser Signale sind in den Abbildungen (A.2) und (A.3) zusammengestellt. Gegenüber der x-Komponente der Lösung des Lorenz-Systems weisen beide Mackey-Glass-Zeitsignale eine deutliche Periodizität auf. Es handelt sich hierbei jedoch nicht um periodische oder quasi-periodische Lösungen, wie den in Bild A.2(c) und A.3(c) dargestellten Leistungsdichtespektren zu entnehmen ist. Beide Systeme besitzen jedoch ausgeprägte lineare Bindungen, was durch den Verlauf der normierten Autokorrelationsfunktionen aus den Abbildungen A.2(b) und A.3(b) unterstrichen wird. Für die in den Abschnitten 5.3.1 und 5.3.2 durchgeführten Untersuchungen zur Prädiktion dieser Signale sind die ersten Minima der normierten Autokorrelationsfunktionen bei $\Delta t=24$ für das \mathbf{MG}_{17} -Signal und bei $\Delta t=37$ für das \mathbf{MG}_{30} -Signal von Bedeutung.

Weiterhin wird aus den Abbildungen A.2(d) und A.3(d) die Korrelationsdimension als Steigung des Korrelationsintegrals gemäß Gl.(2.14) und Gl.(2.17) für das \mathbf{MG}_{17} -Signal zu $d_C(\mathbf{MG}_{17}) \approx 2,0$ und für das \mathbf{MG}_{30} -Signal zu $d_C(\mathbf{MG}_{30}) \approx 3,5$ ermittelt. Diese Werte stimmen gut mit denen bei Lapedes und Farber [71] angegebenen Werten von

$$d_C(\mathbf{MG}_{17}) = 2,1 \quad (\text{A.11})$$

für den *Mackey-Glass-Attraktor* mit $\Delta T=17$, bzw.

$$d_C(\mathbf{MG}_{30}) = 3,54 \quad (\text{A.12})$$

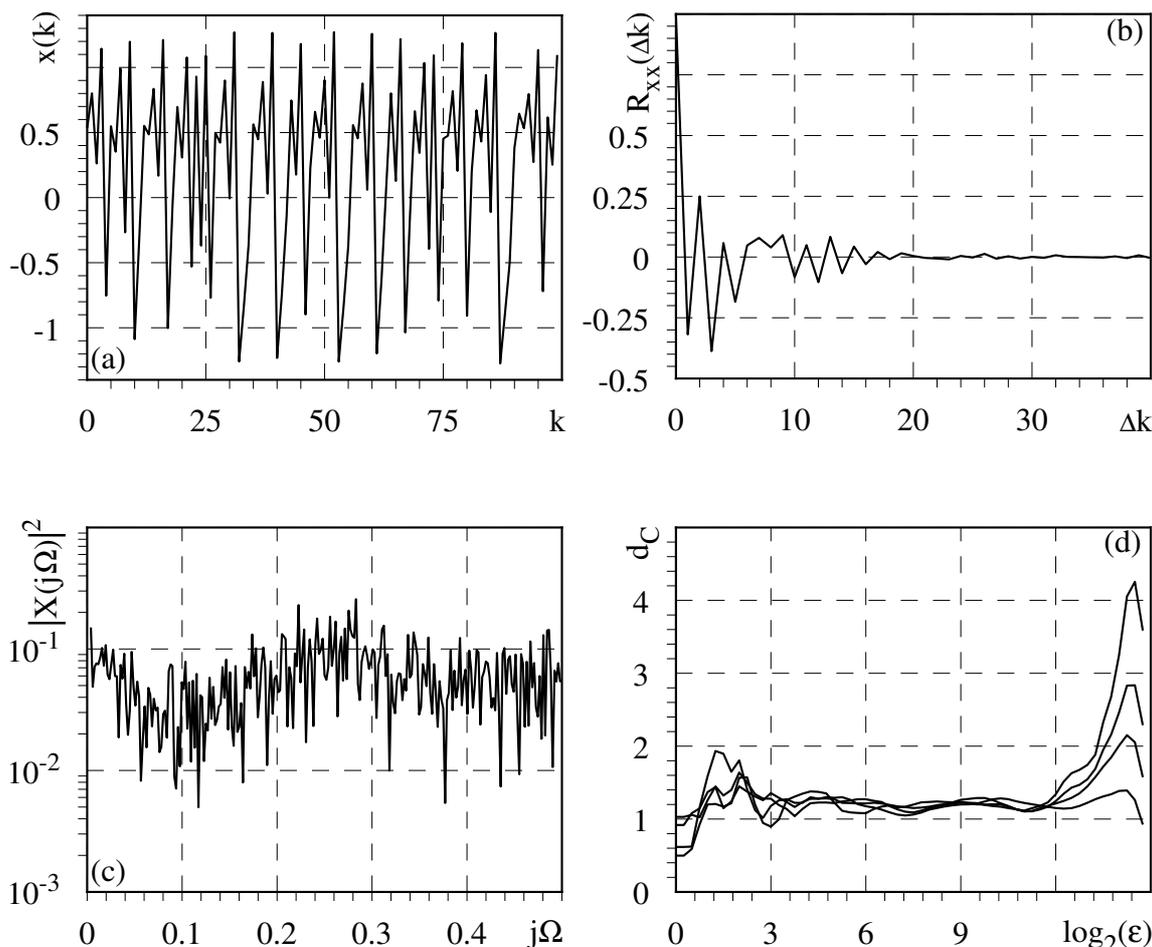
für den Mackey-Glass-Attraktor mit $\Delta T=30$ überein.

A.3 Die Hénon-Abbildung

Als Beispiel für ein zeitdiskretes chaotisches System gemäß Gl.(2.8) wird im Rahmen dieser Arbeit die im Jahre 1976 von Hénon eingeführte und nach ihm benannte 2-dimensionale Hénon-Abbildung untersucht. Diese stellt eine 2-dimensionale Erweiterung der bekannten logistischen Abbildung dar. Die zugehörigen Gleichungen lauten

$$\begin{aligned} x^{k+1} &= 1 - a(x^k)^2 + y^k \\ y^{k+1} &= bx^k. \end{aligned} \quad (\text{A.13})$$

Die Jacobische Determinante der Hénon-Abbildung entspricht dem Betrag des Parameters b . Demzufolge ist die Hénon-Abbildung dissipativ, solange $|b| < 1$ gilt. Für die hier verwendeten Parameter $a=1,4$, $b=0,3$ weist das System chaotisches Verhalten auf und

Abbildung A.4: Charakteristische Kenngrößen der x -Komponente des *Hénon-Attraktors*

- (a) Zeitsignal (b) normierte AKF
(c) Leistungsdichtespektrum (d) Korrelationsdimension

besitzt die folgenden charakteristischen Größen:

$$\textit{Fraktale} - \textit{Dimensionen} \tag{A.14}$$

$$d_C = 1,21$$

$$d_K = 1,26$$

$$\textit{Lyapunov} - \textit{Exponenten} \tag{A.15}$$

$$\lambda_1 = 0,603$$

$$\lambda_2 = -2,34$$

Die Wahl der Startwerte für die Iteration ist unkritisch, solange $x \neq 0 \vee y \neq 0$ erfüllt ist, und die Startwerte in der Umgebung des Attraktors gewählt werden. Hier wurde der Startpunkt $(0, 1; 0, 1)$ verwendet, und die ersten 10 000 berechneten Punkte wiederum nicht berücksichtigt, um das Einschwingen des Systems auf die stationäre Lösung sicherzustellen.

Für die in den Abschnitten 5.4 und 5.6.2 durchgeführten Untersuchungen wurde das folgende Signal mit $N=110\,000$ Werten verwendet.

\mathbf{H}_x : x-Komponente der Lösung der Hénon-Abbildung

$$\mathbf{H}_x = \{x^k\}_1^N \quad (\text{A.16})$$

\mathbf{H}_x^{-1} : zeitlich invertierte x-Komponente der Lösung der Hénon-Abbildung

$$\mathbf{H}_x = \{x^{N+1-k}\}_1^N \quad (\text{A.17})$$

Neben einem Ausschnitt aus dieser Zahlenfolge sind in Abbildung (A.4), wie bereits für die Lorenz- und Mackey-Glass-Signale, als charakteristische Größen die normierte Autokorrelationsfunktion, das Leistungsdichtespektrum sowie die mittlere Steigung des Korrelationsintegrals dargestellt.

Die x-Komponente der Hénon-Abbildung besitzt Bild A.4(b) zufolge nahezu keine linearen Bindungen und damit gemäß Bild A.4(c) ein Leistungsdichtespektrum, welches sehr breitbandig ist und eher an ein Rauschsignal denn an ein deterministisches Signal gemäß Gl.(A.13) erinnert.

Die Korrelationsdimension ergibt sich aus der mittleren Steigung des Korrelationsintegrals entsprechend Bild A.4(d) zu $d_C \approx 1,2$ und stimmt im Rahmen der erzielbaren Genauigkeit mit dem theoretischen Wert von $d_C=1,21$ [41] überein.

B Glossar

Im folgenden Glossar sind Definitionen einiger der verwendeten mathematischen Begriffe aufgeführt. Weiterführende Erläuterungen finden sich in [116, 57, 103, 139].

B.1 abgeschlossene Menge

Eine Menge M heißt *abgeschlossen*, wenn ihr Komplement C_M offen ist. Eine Menge ist genau dann abgeschlossen, wenn sie alle ihre Häufungspunkte enthält.

B.2 absolute Stetigkeit

Eine Funktion F heißt *absolut stetig*, wenn für jede höchstens endliche Menge sich nicht überlappender Intervalle $[\alpha_k, \beta_k]$ aus dem Definitionsbereich von F und für jede Schranke $\epsilon > 0$ ein $\rho > 0$ existiert, für das aus $\sum_{k=1}^n |\beta_k - \alpha_k| < \rho$ die Ungleichung $\sum_{k=1}^n |F(\beta_k) - F(\alpha_k)| < \epsilon$ folgt.

B.3 im allgemeinen gültige Eigenschaft

Unter einer *im allgemeinen gültigen Eigenschaft* der Elemente einer Menge versteht man eine Eigenschaft, die einer dichten (B.6) Teilmenge dieser Menge zukommt.

B.4 beschränkte Menge

Seien (M, \leq) eine halbgeordnete Menge und $X \subseteq M$. Ein Element $s \in M$ heißt *obere Schranke* von X , wenn gilt:

$$x \leq s \quad \forall \quad x \in X \tag{B.1}$$

Ist die Menge der oberen Schranken von X nicht leer, so heißt X *nach oben beschränkt*. Entsprechend definiert man die Begriffe *untere Schranke* von X und *nach unten beschränkte Menge* von X . Ist $X \subseteq M$ nach oben und nach unten beschränkt, so nennt man X (schlechthin) *beschränkt*.

B.5 C^∞ -Funktion

Ist M eine nichtleere offene Teilmenge des \mathbb{R}^n und $m \in \mathbb{N}$, so bezeichnet man mit $C^m(M)$ die Menge aller Funktionen $F : M \rightarrow \mathbb{R}$, die auf M definiert sind und deren partielle Ableitungen der Ordnung $\leq m$ alle auf M vorhanden und stetig sind. Liegt M fest, so nennt man ein $F \in C^m(M)$ auch eine C^m -Funktion. Sind Ableitungen beliebiger Ordnung auf M vorhanden und stetig, so nennt man ein $F \in C^\infty(M)$ auch eine C^∞ -Funktion.

B.6 dichte Teilmenge

Ist eine Menge \mathbf{Y} gegeben und existiert eine Teilmenge $\mathbf{X} \subset \mathbf{Y}$ mit der Eigenschaft, daß jedes Element aus \mathbf{Y} beliebig genau durch ein Element aus \mathbf{X} angenähert werden kann, so nennt man \mathbf{X} eine *dichte Teilmenge* von \mathbf{Y} .

B.7 Diffeomorphismus

Eine stetige Abbildung $\mathbf{F} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ wird als *diffeomorphe Abbildung* oder als *Diffeomorphismus* bezeichnet, wenn sie invertierbar ist, und die inverse Abbildung $\mathbf{F}^{-1} : \mathbb{R}^m \rightarrow \mathbb{R}^n$ ebenfalls stetig ist.

B.8 dissipatives System

Ein physikalisches System heißt *dissipativ*, wenn es zur fortdauernden Entwicklung einer Energiezufuhr bedarf. Bei mathematischen Modellen dynamischer Systeme wird dieses Verhalten abstrahiert. Betrachtet man die Entwicklung eines Volumens im Zustandsraum des Systems, so ist das System dissipativ, wenn dieses Volumen durch die das System beschreibenden Gleichungen kontrahiert wird. Zeitdiskrete Systeme sind genau dann dissipativ, wenn der Betrag der Jacobi Determinante der Abbildung, also der Determinante der Funktionalmatrix, kleiner 0 ist. In mathematischer Form gilt für zeitdiskrete dissipative Systeme

$$|\mathbf{J}_{\mathcal{F}}| = \left| \text{Det} \left(\frac{\partial \mathcal{F}_j}{\partial x_i} \right) \right| < 1.$$

Für kontinuierliche Systeme existiert ein ähnliches Kriterium. Dort wird die Änderung eines Volumenelementes in Abhängigkeit von der Zeit durch das Integral der Divergenz des Vektorfeldes über dieses Volumenelement gegeben. Das System ist dissipativ, wenn die Ableitung des Volumens nach der Zeit negativ ist. Für ein dissipatives System im n -dimensionalen Zustandsraum gilt also

$$\frac{dV}{dt} = \int_V \text{div}(\mathbf{F}(\mathbf{x})) dV < 0.$$

B.9 Einbettung

Eine Eintauchung (B.10) nennt man *Einbettung*, wenn die Abbildung zusätzlich abgeschlossen und umkehrbar eindeutig (bijektiv) ist. Bei einer Einbettung bleiben die qualitativen Eigenschaften der abgebildeten Mannigfaltigkeit erhalten. Es ist allerdings möglich, daß die Mannigfaltigkeit durch die Einbettung beliebig verzerrt, verdreht oder sogar gefaltet wird.

B.10 Eintauchung

Eine Abbildung einer Mannigfaltigkeit \mathbf{X} in eine Mannigfaltigkeit \mathbf{Y} mit $\dim(\mathbf{X}) < \dim(\mathbf{Y})$ heißt *Eintauchung*, wenn für alle $\mathbf{x} \in \mathbf{X}$ die linearisierte Abbildung (Jacobimatrix) regulär ist. Benachbarte Punkte können bei einer Eintauchung also nicht zusammenfallen.

B.11 homöomorph

Ist eine Abbildung $F : \mathbf{X} \rightarrow \mathbf{Y}$ umkehrbar eindeutig (bijektiv), stetig und auch ihre Umkehrung F^{-1} stetig, dann nennt man diese Abbildung *homöomorph*. Homöomorphismen erhalten bestimmte Eigenschaften, die sogenannten topologischen Eigenschaften, wie zum Beispiel Kompaktheit und Zusammenhang.

B.12 Jacobi Determinante

Sei $F : \mathbf{X} \rightarrow \mathbb{R}^m$, $\mathbf{X} \subseteq \mathbb{R}^n$, differenzierbar in $\mathbf{x}_0 \in \mathbf{X}$.

Als *Jacobi-Matrix* oder *Funktionalmatrix* $\mathbf{JF}_{\mathbf{x}_0}$ von F in \mathbf{x}_0 bezeichnet man die $m \times n$ -Matrix:

$$\left(\frac{\partial F_j}{\partial x_i} \right)_{\substack{j=1, \dots, m \\ i=1, \dots, n}} \quad (\text{B.2})$$

Im Falle $m = n$ wird die $\mathbf{JF}_{\mathbf{x}_0}$ zugeordnete Determinante als *Jacobi-Determinante* oder *Funktionaldeterminante* von F in \mathbf{x}_0 bezeichnet. Man schreibt:

$$\text{Det} \mathbf{JF}_{\mathbf{x}_0} = \frac{\partial(F_1, \dots, F_n)}{(\partial x_1, \dots, x_n)}(\mathbf{x}_0) \quad (\text{B.3})$$

$$\mathbf{J}\tilde{\mathcal{G}}_x^k := \frac{\partial \tilde{\mathcal{G}}(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x}=\mathbf{x}^k} = \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_{d_S}}{\partial x_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_1}{\partial x_{d_S}} & \cdots & \frac{\partial y_{d_S}}{\partial x_{d_S}} \end{pmatrix} \quad (\text{B.4})$$

B.13 kompakte Menge

Eine Teilmenge des \mathbb{R}^n ist genau dann *kompakt*, wenn sie beschränkt und abgeschlossen ist. Kompakte Mengen müssen nicht unbedingt zusammenhängend sein.

B.14 Konvergenzklassen

Ein Iterationsprozeß zeigt *Konvergenz der Ordnung r* , wenn der Grenzwert

$$\gamma = \lim_{i \rightarrow \infty} \frac{\|\mathbf{w}^{i+1} - \mathbf{w}^*\|}{\|\mathbf{w}^i - \mathbf{w}^*\|^r} \quad 0 \leq \gamma < \infty \quad (\text{B.5})$$

existiert. Für $r=1$ liegt *lineare*, für $r=2$ *quadratische Konvergenz* vor. Im Falle linearer Konvergenz muß $\gamma < 1$ erfüllt sein, damit Konvergenz eintritt. Ist der Grenzwert γ für $r=1$ exakt 0, liegt *superlineare Konvergenz* vor.

Ist die Bedingung

$$\gamma = \lim_{i \rightarrow \infty} \frac{\|\mathbf{w}^{i+n} - \mathbf{w}^*\|}{\|\mathbf{w}^i - \mathbf{w}^*\|^r} \quad 0 \leq \gamma < \infty \quad (\text{B.6})$$

erfüllt, spricht man von *n -Schritt Konvergenz der Ordnung r* [79]. Für $\gamma=0$ ist *n -Schritt superlineare Konvergenz der Ordnung r* gegeben.

B.15 konjugierte Richtungen

Zwei Vektoren \mathbf{u}^1 und \mathbf{u}^2 heißen *konjugiert* bezüglich einer symmetrischen Matrix \mathbf{H} oder *H-orthogonal*, wenn die Beziehung $(\mathbf{u}^1)^T \mathbf{H} \mathbf{u}^2 = 0$ erfüllt ist. Für $\mathbf{H}=\mathbf{0}$ sind alle denkbaren Vektoren $\mathbf{u}^1 \neq \mathbf{u}^2$ konjugiert, für $\mathbf{H}=\mathbf{I}$ reduziert sich die Eigenschaft der H-Orthogonalität auf den herkömmlichen Orthogonalitätsbegriff $(\mathbf{u}^1)^T \mathbf{u}^2 = 0$. Eine Menge von Vektoren $\mathbf{u}^0, \mathbf{u}^1, \dots, \mathbf{u}^k$ heißt H-orthogonal, wenn $(\mathbf{u}^i)^T \mathbf{H} \mathbf{u}^j = 0$ für alle $i \neq j$ gilt.

B.16 Mannigfaltigkeit

Eine n -dimensionale *Mannigfaltigkeit* ist eine Menge \mathbf{X} , für deren Punkte \mathbf{x} jeweils eine offene Umgebung (B.25) $\mathbf{U} \in \mathbf{X}$ existiert, die eine homöomorphe Abbildung (B.11) einer offenen Teilmenge des \mathbb{R}^n ist. Eine Mannigfaltigkeit heißt differenzierbar, wenn die homöomorphen Abbildungen sich überlappender Umgebungen differenzierbar verträglich sind. Ein 2-Torus ist ein Beispiel für eine 2-dimensionale Mannigfaltigkeit (vgl. Kapitel 2.2).

B.17 Norm eines Vektors

Als *p -Norm* eines n -dimensionalen Vektors \mathbf{w} bezeichnet man

$$\|\mathbf{w}\|_p = \left(\sum_{i=1}^n |w_i|^p \right)^{\frac{1}{p}} \quad (\text{B.7})$$

Üblicherweise werden die *Euklidische Norm* ($p = 2$)

$$\|\mathbf{w}\|_2 = \sqrt{w_1^2 + w_2^2 + \dots + w_n^2} = \sqrt{\mathbf{w}^T \mathbf{w}} \quad (\text{B.8})$$

oder die *Unendlichkeitsnorm* ($p = \infty$)

$$\|\mathbf{w}\|_\infty = \max_{w_i} |w_i| \quad (\text{B.9})$$

verwendet, welche auch als Maximumsnorm bezeichnet wird. Strebt die Dimension n des Vektors \mathbf{w} gegen ∞ , bleibt die Unendlichkeitsnorm beschränkt, die Euklidische Norm dagegen nicht.

B.18 offene Menge

Eine Menge M , die nur innere Punkte enthält heißt *offen*.

B.19 Tangentialraum

Ist \mathbf{X} eine k -dimensionale *Mannigfaltigkeit*, dann ist der *Tangentialraum* $T_x(\mathbf{X})$ ab \mathbf{x} der k -dimensionale Raum, der im Punkt \mathbf{x} tangential zu \mathbf{X} ist. Ist $F : \mathbf{X} \rightarrow \mathbf{Y}$ eine Abbildung, so wird durch die linearisierte Abbildung der Tangentialraum im Punkt \mathbf{x} in den Tangentialraum im Punkt \mathbf{y} überführt $DF(\mathbf{x}) : T_x(\mathbf{X}) \rightarrow T_y(\mathbf{Y})$.

B.20 Topologie

Eine *Topologie* über einer Menge \mathbf{X} ist dadurch definiert, daß jedem Element $\mathbf{x} \in \mathbf{X}$ ein System $\mathcal{U}(\mathbf{x})$ von Teilmengen von \mathbf{X} , sog. Umgebungen U von \mathbf{x} , zugeordnet ist mit folgenden Axiomen:

- $\mathbf{x} \in \mathbf{X}$ für jedes $U \in \mathcal{U}(\mathbf{x})$
- $U \in \mathcal{U}(\mathbf{x}) > V \in \mathcal{U}(\mathbf{x})$ für jedes $V \supset U$
- $U_1, U_2 \in \mathcal{U}(\mathbf{x}) > U_1 \cap U_2 \in \mathcal{U}(\mathbf{x})$
- Zu jedem $U \in \mathcal{U}(\mathbf{x})$ existiert ein $W \in \mathcal{U}(\mathbf{x})$, so daß $U \in \mathcal{U}(\mathbf{y})$ für alle $\mathbf{y} \in W$

B.21 topologischer Raum

Eine Menge \mathbf{X} zusammen mit einer Topologie über \mathbf{X} heißt ein *topologischer Raum*.

B.22 topologische Dimension

Ein metrischer Raum mit abzählbarer Basis heißt:

- 0-dimensional, wenn jede Umgebung \mathbf{U} jedes Punktes eine Umgebung \mathbf{V} umfaßt, deren Rand $\partial\mathbf{V}$ leer ist,
- höchstens n -dimensional ($n \geq 1$), wenn jede Umgebung \mathbf{U} jedes Punktes eine Umgebung \mathbf{V} umfaßt, deren Rand $\partial\mathbf{V}$ höchstens $(n - 1)$ -dimensional ist,
- n -dimensional, wenn er höchstens n -dimensional und nicht $(n - 1)$ -dimensional ist,
- ∞ -dimensional, wenn er für kein $n \in \mathbb{N}$ höchstens n -dimensional ist.

Diese Definition der Dimension wird als *topologische Dimension* bezeichnet.

B.23 Trajektorie

Der Zustandsvektor $\mathbf{x}(t) \in \mathbb{R}^{d_s}$ beschreibt den Zustand eines kontinuierlichen dynamischen Systems. Dieser Zustand stellt in Abhängigkeit des Zeitparameters t eine Kurve im d_s -dimensionalen Zustandsraum des Systems dar, welche als *Trajektorie* bezeichnet wird.

B.24 Überdeckung

Als *Überdeckung* eines topologischen Raumes \mathbf{M} wird jedes System $\mathcal{X} := \{\mathbf{X}_i | i \in I\}$ von Teilmengen $\mathbf{X}_i \subseteq \mathbf{M}$ (bei beliebiger Indexmenge I) mit der Eigenschaft $\bigcup_{i \in I} \mathbf{X}_i = \mathbf{M}$ bezeichnet.

B.25 Umgebung

Eine Teilmenge \mathbf{U} von \mathbf{X} heißt eine *Umgebung* des Punktes $\mathbf{x}_0 \in \mathbf{X}$, wenn es eine Kugelumgebung $U_\varepsilon(\mathbf{x}_0)$ gibt mit $U_\varepsilon(\mathbf{x}_0) \subset \mathbf{U}$.

B.26 Vektorfeld

Sei \mathbf{X} eine nichtleere Teilmenge des \mathbb{R}^n . Als *Vektorfeld* wird jede Funktion $\mathbf{F} : \mathbf{X} \rightarrow \mathbb{R}^n$ bezeichnet.

Lebenslauf

Persönliche Daten:

Name Siegfried Ergezinger
Geburtsdatum, -ort 17.11.59, Dehmke (jetzt Aerzen)
Familienstand verheiratet, eine Tochter

Schulbildung:

04/66 - 07/69 Grundschule in Aerzen
08/69 - 06/78 Gymnasium Bad Pyrmont

Wehrdienst:

07/78 - 06/80 Grundwehrdienst in Goslar, Wehrdienst in Husum

Studium:

10/80 - 09/82 Grundstudium der Elektrotechnik an der Universität Hannover
10/82 - 08/84 sowie Hauptstudium der Nachrichtentechnik mit Schwerpunkt
07/85 - 04/87 Übertragungstechnik an der Universität Hannover
09/84 - 06/85 Studium der Elektrotechnik an der San Diego State University
10/86 - 09/88 Grundstudium der Wirtschaftswissenschaften, Fernuniversität
Hagen, Diplomvorprüfung September 1988

Berufspraxis:

07/82 - 08/84 sowie wissenschaftliche Hilfskraft am Institut für Allgemeine
07/85 - 12/86 Nachrichtentechnik der Universität Hannover
09/84 - 06/85 "Graduate Assistant" an der San Diego State University
05/87 - 12/92 Wissenschaftlicher Mitarbeiter am Institut für
Allgemeine Nachrichtentechnik der Universität Hannover

seit 04/93 Mitarbeiter bei der E-Plus Mobilfunk GmbH, Düsseldorf
seit 12/93 Leiter der Abteilung "Netzdienste"