

**Informationstheoretische
Optimierung
künstlicher neuronaler Netze**
für den Einsatz in Steuergeräten

Dissertation

der Fakultät für Informatik- und Kognitionswissenschaften
der Eberhard-Karls-Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von

Dipl.-Inform. Michael Helmut Froehlich
aus Allensbach am Bodensee

Tübingen
2003

Tag der mündlichen Qualifikation: 23. Juli 2003
Dekan: Prof. Dr. Ulrich Güntzer
1. Berichterstatter: Prof. Dr. Wolfgang Rosenstiel
2. Berichterstatter: Prof. Dr. Andreas Zell

Abstract

Artificial neural networks are usually adapted by repeated exposure to training patterns, which are gained from physical measurements. The size of the network required to map the training space has to be set by the user and depends on the complexity of the data.

In order to analyze the complexity of measurements, the information scale proposed by Claude Shannon in 1948 can be used. Based on the so-called mutual information principle, several methods for data analysis and the creation of network topologies, which are adapted to the complexity of the data, have been developed.

By using mutual information, a method was developed which allows the evaluation of Fourier spectra based on information theory, where elaborate frequency ranges are weighted depending on their information content in relation to the whole spectrum. This weighting offers additional support in the design and development of digital filters.

The evaluation of training patterns, based on information theory, could be applied to develop a method for selectively choosing the necessary network inputs. Finally, based on these experiences, a method could be developed to generate complete network topologies which match the complexity of the training space.

These newly developed methods can be applied effectively in the estimation of injection quantities for a common rail diesel engine. This can be done by analyzing and evaluating measurement data from the rail pressure signal, based on information theory. Thus a small neural network can be created that determines the injected quantity for each combustion and cylinder individually. The achieved accuracy can be compared to the results of much bigger conventional networks. This considerably smaller network topology enables the real time monitoring of injection quantities in engine control units with limited calculation and memory capacities.

Abstrakt

Künstliche neuronale Netze werden in der Regel durch eine wiederholte Präsentation von Trainingspattern adaptiert, welche aus Meßdaten gewonnen wurden. Die zur Abbildung des Trainingsraums erforderliche Größe des Netzes muß dabei in der Regel durch den Anwender vorgegeben werden und hängt stark von der Komplexität der Daten ab.

Zur Bestimmung der Komplexität von Meßdaten kann ein von Claude Shannon schon 1948 vorgeschlagenes Informationsmaß verwendet werden. Aufbauend auf der sogenannten Transinformation werden verschiedene Verfahren zur Datenanalyse und zur Erzeugung einer an die Komplexität angepaßten Netztopologie vorgestellt.

Durch den Einsatz der Transinformation ist zunächst ein Verfahren entwickelt worden, das eine informationstheoretische Bewertung von Fourier-Spektren ermöglicht. Dabei werden einzelne Frequenzbereiche entsprechend ihres Informationsanteils bezogen auf das Gesamtspektrum gewichtet. Diese Gewichtung bietet eine zusätzliche Hilfe bei der Entwicklung digitaler Filter.

Die informationstheoretische Bewertung von Trainingspattern konnte für die Entwicklung einer Methode zur gezielten Auswahl tatsächlich notwendiger Netzeingänge genutzt werden. Aufbauend auf so gewonnenen Erfahrungen ist schließlich ein Verfahren zur Erzeugung vollständiger Netztopologien entwickelt worden. Dabei wird eine Topologie erzeugt, die der Komplexität des Trainingsraums entspricht.

Diese neu entwickelten Verfahren können bei der Bestimmung von Einspritzmengen an einem Common-Rail Dieselmotor erfolgreich eingesetzt werden. Dies geschieht durch eine informationstheoretische Analyse und Auswertung von Meßdaten des Raildrucksignals. Dadurch kann ein kleines neuronales Netz erzeugt werden, das die Einspritzmenge für jede Verbrennung zylinderindividuell bestimmen kann. Die erzielte Genauigkeit ist dabei vergleichbar mit den Ergebnissen sehr viel größerer Netze herkömmlicher Verfahren. Diese wesentlich kleinere Netztopologie ermöglicht eine Echtzeitüberwachung der Einspritzmenge in Motorsteuergeräten mit begrenzter Rechen- und Speicherkapazität.

Danksagung

Die vorliegende Arbeit ist im Rahmen meiner Tätigkeit bei der Daimler Benz AG entstanden, die wie meine Arbeit über die Jahre einen starken Wandel erfahren hat und zur DaimlerChrysler AG gewachsen ist. Folgende Danksagung entspricht in etwa dem chronologischen Mitwirken verschiedener Betreuer, Kollegen und Freunde.

Zunächst möchte ich mich bei Herrn Prof. Dr. W. Rosenstiel nicht nur für seine stets gute Betreuung, sowie wegweisenden Anregungen bedanken, sondern auch für die anfängliche Ermutigung, mich überhaupt an einer Promotion zu versuchen. Eine sehr gute Unterstützung habe ich auch durch unseren Abteilungsleiter Volker Wilhelm, sowie die Teamleiter Hans-Hubert Hemberger, Matthias Scherer und Rainer Müller-Finkeldei erfahren.

Ich konnte stets mit Unterstützung aller Kollegen unserer Abteilung, einiger Nachbarabteilungen, sowie dem Neuro-Team der Uni-Tübingen rechnen, deren Aufzählung den Rahmen dieser Danksagung sprengen würde. Zudem möchte ich nicht Gefahr laufen, jemanden zu vergessen und mich hiermit bei allen bedanken, die beim Gelingen dieser Arbeit mitgewirkt haben.

Mein besonderer Dank gilt meinen beiden Diplomanden Steffen Tacke und Kristina Ahlin, sowie einigen Kollegen, mit deren Unterstützung ich täglich rechnen konnte. Zu ihnen zählen Thomas Lich, Frank Heister, Frank Kirschbaum, Gregor Schock, Bernd Schneider und Dirk Mehlfeldt.

Für die ständigen Ermutigungen meiner Familie und Freunde, sowie das andauernde Verständnis für meine arbeitsbedingten Launen empfinde ich große Dankbarkeit.

Inhaltsverzeichnis

I. Teil - Einführung und Grundlagen	1
1 Einführung und Motivation	1
1.1 Motivation zur Netzwerkoptimierung	1
1.2 Schwerpunkt der Arbeit	2
1.3 Gliederung der Arbeit	3
2 Theoretische Grundlagen	4
2.1 Künstliche neuronale Netze	4
2.1.1 Biologische Motivation für neuronale Netze	4
2.1.2 Mathematisches Neuronenmodell	5
2.1.3 Resilient-Propagation-Lernregel	8
2.1.4 Levenberg-Marquardt-Algorithmus	9
2.1.5 Cascade-Correlation-Architektur	10
2.2 Statistische Maße und Wahrscheinlichkeiten	12
2.2.1 Grundlagen statistischer Maße	12
2.2.2 Grundlagen der Wahrscheinlichkeitsrechnung	15
2.2.3 Fehlermaße und Bewertung	18
2.3 Grundlagen der Signalanalyse	20
2.3.1 Elementarsignale	20
2.3.2 Transformationen auf Signale	23
2.3.3 Faltungsintegrale	23

2.3.4	Fourier-Transformation	24
2.3.5	Oktavbandzerlegung	27
2.4	Verallgemeinerte Transinformation	29
2.4.1	Shannon Entropie und Informationskanal	29
2.4.2	Eigenschaften der Transinformation	31
2.4.3	Gleichverteilung von Zufallsvariablen	35
2.4.4	Schätzverfahren zur Transinformation	39
2.4.5	Bewertung des Schätzverfahrens	43
2.5	Zusammenfassung	46
3	Stand der Technik	47
3.1	Methoden zur Datenanalyse	47
3.1.1	Regression	47
3.1.2	Korrelationsanalyse	49
3.1.3	Hauptkomponentenanalyse	51
3.1.4	Frequenzanalysen	53
3.2	Auswahl und Optimierung von Gewichten	56
3.2.1	Daumenregeln	57
3.2.2	Regularisierung	58
3.2.3	Early Stopping	58
3.3	Topologieoptimierende Algorithmen	60
3.3.1	Pruning-Algorithmen	60
3.3.2	Cascade Correlation	63
3.4	Probleme aktueller Methoden	66
3.4.1	Einschränkungen bei der Datenanalyse	66
3.4.2	Gewichtsoptimierung nur für Spezialfälle	67
3.4.3	Topologieoptimierung ohne Datenbezug	68
3.5	Zusammenfassung	69

II. Teil - Neu entwickelte Methoden und ihre Anwendung	70
4 Anwendungsorientierte Methoden	70
4.1 Problem- und Aufgabenbeschreibung	70
4.2 Informationsoptimierte Frequenzanalyse	72
4.2.1 Invarianz gegenüber Phasenverschiebungen	72
4.2.2 Frequenzbandzerlegung	73
4.2.3 Rekonstruktion der Frequenzbänder	77
4.2.4 Bewertung der informationsoptimierten Frequenzanalyse	80
4.3 Informationsoptimierte Eingangsdimension	84
4.3.1 Mehrdimensionale Transinformation	84
4.3.2 Bewertung der Dimensionsreduzierung	86
4.4 Informationsoptimierte Netztopologie	88
4.4.1 Optimierung von Netzverbindungen	88
4.4.2 Bewertung des informationsoptimierten Trainings	91
4.5 Zusammenfassung	94
5 Anwendungen	95
5.1 Systemgrundlagen	96
5.1.1 Grundlagen zum Common-Rail-System	96
5.1.2 Sensorik und Meßtechnik	99
5.1.3 Abdeckung des Datenraums	101
5.2 Bestimmung des 50%-Umsatzpunktes	102
5.2.1 Aufgabenstellung	102
5.2.2 Neuronale Brennraumdruckauswertung	102
5.2.3 Reduzierung der Eingangsdimensionalität	105
5.2.4 Bewertung der Ergebnisse	107

5.3	Einspritzmengenbestimmung am Common-Rail-System	108
5.3.1	Aufgabenstellung	108
5.3.2	Analyse und Modellierung des Common-Rail-Systems	109
5.3.3	Informationsoptimierte Datenanalyse	117
5.3.4	Reduzierung der Eingangsdimensionalität	122
5.3.5	Informationsoptimierte Netztopologie	126
5.4	Zusammenfassung der Ergebnisse	133
5.4.1	Bewertung der Ergebnisse zum 50%-Umsatzpunkt	134
5.4.2	Bewertung der Ergebnisse zur Datenanalyse	134
5.4.3	Bewertung der Ergebnisse zur Raildruckanalyse	135
5.5	Aufgetretene Probleme	137
6	Zusammenfassung und Ausblick	138
A	Nomenklatur	141
B	Algorithmen	145

Abbildungsverzeichnis

2.1	Vergleich biologisches und mathematisches Netz	5
2.2	Schematischer Aufbau einer Netztopologie	6
2.3	Mathematisches Modell einer Nervenzelle	7
2.4	Cascade-Correlation-Netztopologie	11
2.5	Ausgewählte Elementarsignale	20
2.6	Gleichverteiltes Rauschen	22
2.7	Oktavbandzerlegung eines Chirp-Signals	28
2.8	Informationskanal nach Shannon	29
2.9	Gleichverteilung durch Ranking	37
2.10	Spektralkonsistenz des Rankings	38
2.11	Iterative Bestimmung von Epsilon	42
2.12	Binärmatrix zur Schätzung der Transinformation	43
2.13	Komplexe Zufallsverteilungen zur Bewertung	44
2.14	Vergleich Korrelations- zu Transinformations-Funktion	45
3.1	Auswahl relevanter Druckpositionen durch Korrelation	50
3.2	Ellipsoid zweier Hauptkomponenten	52
3.3	Spektraldarstellungen zur Fourier-Transformation	54
3.4	Spektrogramm der Short-Time-Fourier-Analyse	55
3.5	Generalisierung mit und ohne Abbruch	59
4.1	Phaseninvarianz der verallgemeinerten Transinformation	72
4.2	Frequenzbandzerlegung überlagerter Sinus-Schwingungen	74

4.3	Signalzerlegung durch Frequenzbandzerlegung	76
4.4	Informationsverlust durch Frequenzbandteilungen	79
4.5	Frequenzanalyse von Systemeingängen und -ausgängen	80
4.6	Informationsoptimierte Systemanalyse	81
4.7	Auflösung informationsoptimierte Frequenzanalyse	82
4.8	Mehrdimensionalen Transinformationsfunktion	86
4.9	Durch Transinformation gefundene Kluster	87
4.10	Gewichtete Ein- und Ausgänge eines Neurons	89
4.11	Informationsoptimiertes iteratives Training	92
4.12	Vergleich informationsoptimiertes Training und Cascade Correlation . .	93
5.1	Schematischer Aufbau des CDI-Systems	97
5.2	Abdeckung der Arbeitspunkte des Common-Rail-Motors	101
5.3	Brennraumdruckkurve mit <i>50%-Umsatzpunkt</i>	103
5.4	Äquidistante Abtastung des Brennraumdrucks	103
5.5	Gleichverteilung der Messungen zum <i>50%-Umsatzpunkt</i>	105
5.6	Durch Transinformation ausgewählte Druckpositionen	106
5.7	Drucksignale des Common-Rail-Systems	110
5.8	Normiertes Spektrum des Raildrucksignals	111
5.9	Normiertes Spektrum verschiedener Raildrucksignale	112
5.10	Gesamtmodell des Hochdruckspeichers	113
5.11	Druckrückschläge schließender Injektoren	114
5.12	Modell des Hochdruckspeichers	115
5.13	Vergleich von realen- und Modelldruckschwingungen	116
5.14	Signale zur informationsoptimierten Frequenzanalyse	118
5.15	Gegenüberstellung Fourier- und informationsoptimierte Analyse	119
5.16	Auswertung unterschiedlicher Filtergrenzen	121
5.17	Methoden zur Reduzierung der Eingangsdimensionalität	123
5.18	Auswertung unterschiedlicher Stützstellen	125

5.19 Methoden zur Reduzierung der Netztopologie	128
5.20 Informationsoptimierte Netztopologie	129
5.21 Auswertung unterschiedlicher Stützstellen	131
5.22 Darstellung von Arbeitspunkten mit großer Abweichung	136

Kapitel 1

Einführung und Motivation

Die Analyse, Verarbeitung und Reaktion auf Informationen sind sowohl in der Natur, als auch in technischen Anwendungen von großer Wichtigkeit. Zur Verarbeitung der Informationsflut der Sinne hat sich die Natur das Konzept neuronaler Netze geschaffen. Durch den Einsatz moderner Computertechnologie ist es möglich dieses Konzept durch ein mathematisches Modell neuronaler Netze vereinfacht nachzubilden.

1.1 Motivation zur Netzwerkoptimierung

Trotz der hohen Verarbeitungsgeschwindigkeit heutiger Prozessoren ist, insbesondere für den Einsatz mit Echtzeitanwendung, die Größe simulierbarer künstlicher Netze begrenzt, was zu einer Optimierung solcher Topologien motiviert.

Arbeiten und Erfahrungen der letzten Jahre haben gezeigt, daß eine Überwachung von Verbrennungsprozessen im Fahrzeug durch umfangreiche neuronale Netze prinzipiell möglich ist (siehe [Fro96] und [Köh97]). Dennoch sind heutige in Fahrzeugen verwendete Prozessoren nicht in der Lage, diese Netzstrukturen in Echtzeit zu berechnen. Eine Optimierung solcher Netztopologien ist für einen Echtzeiteinsatz notwendig.

Zahlreiche Arbeiten wie [Cec97] und [Jan92] aus dem Bereich der Neuro-Fuzzy-Systeme sind zur Optimierung ausgewählter Anwendungen zukunftsweisend. Dabei kann wie in [Fro99] für Motordaten ein auf Fuzzyregeln basierender Ansatz erzielt werden, jedoch ist dabei keine Reduzierung des Berechnungsaufwands während der Anwendungsphase gelungen.

Die Anwendung einer informationsoptimierten Klassifikation von Meßdaten zur Optimierung von Netztopologien vor der Anwendungsphase erscheint daher notwendig und zukunftsweisend.

1.2 Schwerpunkt der Arbeit

Eine der wichtigsten und schwierigsten Aufgaben bei der neuronalen Modellierung ist die Wahl der Netzwerktopologie und somit die Bestimmung der Anzahl an freien Parametern des Netzwerks.

Eine klassische Vorgehensweise zur neuronalen Modellierung besteht aus den vier folgenden Schritten, die zur Optimierung der Approximationsfähigkeit auch zyklisch wiederholt werden können:

Schritt 1: Analyse und Vorverarbeitung der Meßdaten

Schritt 2: Spezifikation der Netzwerkarchitektur

Schritt 3: Adaption bzw. Training der Netzwerkparameter

Schritt 4: Diagnose der Approximationsfähigkeit des Netzwerks

Dabei verfügt eine gute Netzwerkarchitektur in der Regel nur über so viele freie Parameter (Verbindungen bzw. Neuronen), daß die zu modellierende Funktion im Rahmen einer gewünschten Genauigkeit approximiert werden kann [And97]. Es wird also eine Netzwerkarchitektur gesucht, die klein genug ist, das Rauschen von Meßreihen nicht zu approximieren, jedoch groß genug ist, den funktionalen Anteil der Daten abzubilden.

Der Schwerpunkt vieler Arbeiten im Zusammenhang mit künstlichen neuronalen Netzen liegt bei der Entwicklung immer leistungsfähigerer Lernalgorithmen, jedoch selten bei der Bestimmung einer optimierten Netztopologie. Im Gegensatz hierzu liegt der Fokus der vorliegenden Arbeit auf den ersten beiden Schritten des oben beschriebenen Vorgehens, nämlich bei der Analyse von Meßdaten und der Spezifikation von den an die Komplexität der Daten angepaßten Netzarchitekturen.

Auf dem Gebiet der Analyse und Vorverarbeitung von Meßdaten kommen Methoden der Zeitreihenanalyse wie Korrelations-, Frequenz- und Hauptkomponentenanalyse zur Diskussion. Aufbauend auf der klassischen Fast-Fourier-Analyse wird eine Erweiterung vorgestellt, die durch Kombination mit einer informationstheoretischen Datenanalyse eine Gewichtung einzelner Frequenzlinien erlaubt.

Mit Hilfe einer informationstheoretischen Datenanalyse lassen sich verschiedene Ansätze zur Spezifikation der Netzwerkarchitektur realisieren. Dies geschieht unter anderem in Kombination mit herkömmlichen Trainingsalgorithmen, wobei neben den Netzwerkparametern auch die Topologie adaptiert wird.

Die mit der vorliegenden Arbeit vorgestellten Methoden ermöglichen eine automatische Klassifikation von Meßdaten noch vor der Anwendungsphase. Die Klassifikation kann sowohl vor als auch während des Trainings neuronaler Netze erfolgen, wobei die Netztopologie eine automatische Optimierung erfährt. Dabei entstehen sowohl leistungsstarke als auch kleine Netzstrukturen, die sich durch einen geringen Berechnungsaufwand während der Anwendungsphase auszeichnen.

1.3 Gliederung der Arbeit

Zur besseren Orientierung des Lesers ist hier eine kurze Beschreibung zur Gliederung der vorliegenden Arbeit zusammengestellt. Der Kern der Arbeit mit seinen neuen Methoden und Anwendungen ist ab Kapitel 4 zu finden. Vorherige Kapitel dienen dem Leser also Orientierung und Grundlagenquelle.

Kapitel 1 Einführung und Motivation:

In diesem Kapitel sind grundlegende Gedanken zur Motivation, zum Schwerpunkt und zur Gliederung der vorliegenden Arbeit zusammengefaßt.

Kapitel 2 Theoretische Grundlagen:

Kurze Einführung in mathematische Grundlagen aus den Bereichen künstliche neuronale Netze, Statistik, Signalanalyse und Informationstheorie. Dabei liegt der Schwerpunkt bei den informationstheoretischen Grundlagen zur Transinformation.

Kapitel 3 Stand der Technik:

Zusammenfassung bestehender Methoden zur Datenanalyse und Datenvorverarbeitung, sowie zur Spezifikation der Topologie von künstlichen neuronalen Netzen, sowohl vor, als auch während und nach dem Netztraining.

Kapitel 4 Anwendungsorientierte Methoden:

Darstellung der im Rahmen dieser Arbeit und der gegebenen Aufgabenstellung neu entwickelten Methoden und Algorithmen aus den Bereichen Signalanalyse und Signalverarbeitung, sowie Optimierung von neuronalen Netzwerktopologien.

Kapitel 5 Anwendungen:

Im Anschluß an eine kurze Einführung in die Motortechnik von *Common-Rail-Systemen* wird die Anwendbarkeit der entwickelten Methoden dargestellt. Dies geschieht anhand verschiedener Aufgaben aus dem Bereich der Signalverarbeitung motorspezifischer Daten.

Nach einem Ausblick auf denkbare Erweiterungen der Methoden ist eine Zusammenstellung der verwendeten Algorithmen im Anhang der Arbeit zu finden.

Kapitel 2

Theoretische Grundlagen

Dieses Kapitel soll einige theoretische Grundlagen für das Verständnis der ausgewählten und erweiterten Methoden schaffen. Zunächst wird ein grober Überblick zu künstlichen, neuronalen Netzen, den verwendeten Topologien, sowie Lernverfahren gegeben. Um die gewonnenen Ergebnisse beurteilen zu können, werden Maße zur Bewertung benötigt. Eine Einführung ausgewählter Bewertungsmaße erfolgt aufbauend auf statistischen Methoden. Zur Verarbeitung gewonnener Meßdaten kommen Methoden der digitalen Signalverarbeitung zur Anwendung. Aus den vielfältigen Methoden sind einige Grundlagen, sowie der Ansatz der Fourier-Analyse zusammengestellt. Die Transinformation ist ein Maß, um den Informationsfluß zwischen Ein- und Ausgang eines beliebigen Systems zu bewerten, deren Grundlagen abschließend vorgestellt werden.

2.1 Künstliche neuronale Netze

Dieser Abschnitt gibt einen kurzen, nicht vollständigen Überblick über künstliche, neuronale Netze, sowie die verwendeten Topologien und Lernverfahren. Aus der Menge an weiterführender Literatur seien hier exemplarisch [Zel93], [Roj93] genannt, an denen sich auch diese kurze Einführung orientiert.

2.1.1 Biologische Motivation für neuronale Netze

Künstliche, neuronale Netze basieren auf der Beobachtung biologischer Nervensysteme und deren Abläufe zwischen den einzelnen Zellen (auch Neuronen genannt). Schon sehr früh wurde versucht, eine algorithmische und mathematische Beschreibung herzuleiten. In den Nervensystemen von Säugetieren findet die Informationsverarbeitung durch sehr viele Nervenzellen statt. Wesentliches Merkmal dabei ist die starke Vernetzung durch Nervenfasern über welche Nervenzellen ihre Information austauschen.

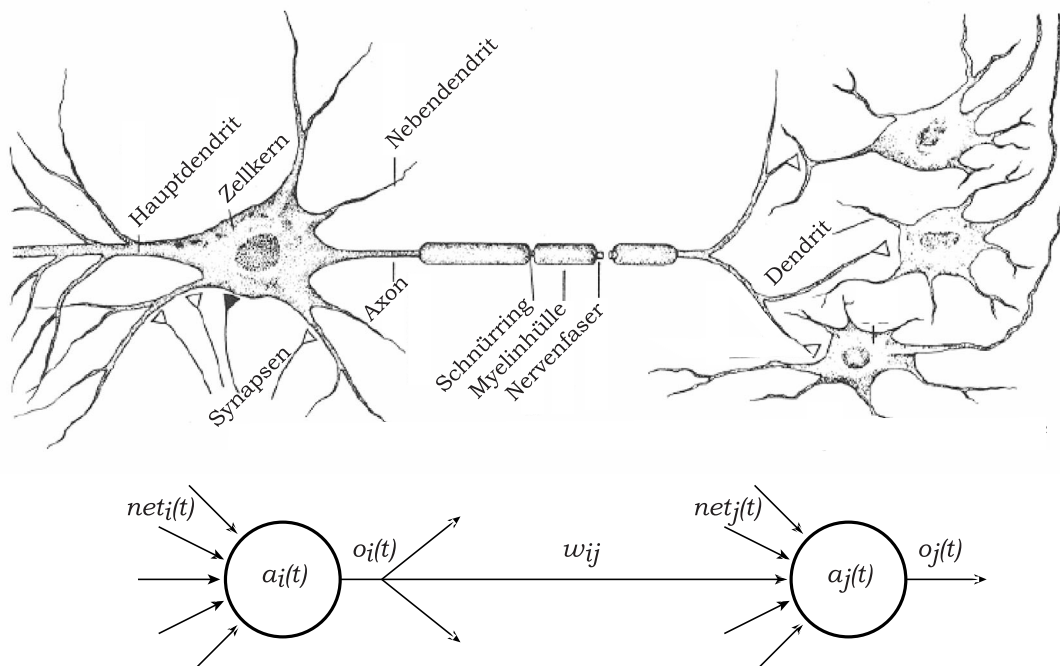


Abbildung 2.1: Vergleich biologisches und mathematisches Netz.

Das mathematische Modell von Nervenzellen ist im Vergleich zum biologischen Vorbild stark vereinfacht. Dies macht sich nicht nur bei der Komplexität der Verbindungen, sondern auch bei der Signalübertragung bemerkbar.

In Abbildung 2.1 wird eine Nervenzelle und ihr mathematisches Modell verglichen. Die Darstellung zeigt eine Nervenzelle mit ihren Verbindungen zu weiteren Neuronen über das sogenannte Axon und die synaptischen Kopplungen.

Gegenüber dem biologischen Vorbild ist die Simulation der Struktur und der Vorgänge in künstlichen neuronalen Netzen sehr stark vereinfacht. So ist die Anzahl der Neuronen und Verbindungen sowie die Parameteranzahl der synaptischen Kopplungen wesentlich geringer. Besonders in den biologisch eher unplausiblen Lernverfahren entspricht das mathematische Modell nicht dem Vorbild. Beibehalten wurden die massive Parallelität vieler einfacher Verarbeitungseinheiten (Neuronen, Zellen), sowie deren synaptische Kopplung in Form von modifizierbaren Verbindungsgewichten.

2.1.2 Mathematisches Neuronenmodell

Bei der mathematischen Beschreibung neuronaler Netze sind die natürlichen Gegebenheiten stark idealisiert und vereinfacht wiedergegeben. Abbildung 2.1 läßt diese starke Vereinfachung schon bei der Flexibilität möglicher Verbindungen erkennen. Weitere

Vereinfachungen werden durch folgende Gliederung der Hauptbestandteile künstlicher Netzwerke deutlich, die dem Schema von [RM86] entspricht.

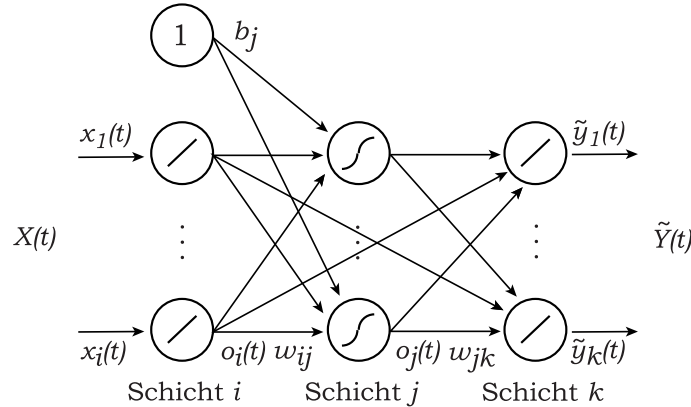


Abbildung 2.2: Schematischer Aufbau einer Netztopologie.

Netzwerk mit Ein- und Ausgabeschicht, sowie einer verdeckten Schicht und den gewichteten Verbindungen.

Netztopologie:

Die Netztopologie kann als gewichteter, gerichteter Graph angesehen werden, wobei die Kanten den Verbindungen zwischen den Nervenzellen entsprechen. Die Verbindungsgewichte zwischen den Zellen i und den Zellen j werden mit w_{ij} bezeichnet. Zur besseren Übersicht lassen sich Zellen oder Neuronen eines neuronalen Netzes in Schichten anordnen:

Eingangsschicht: Über die Zellen der Eingangsschicht i mit linearer Übertragungsfunktion erhält das neuronale Netz seine Eingaben $X(t)$. Eingabezellen mit konstanter Ausgabe dienen über die Gewichte b_j als Schwellwerte der Übertragungsfunktionen der verdeckten Schicht. Alle Zellen der Eingangsschicht stehen am Anfang des gerichteten Graphen.

verdeckte Schicht: Die Zellen der verdeckten Schicht j mit sigmoide Übertragungsfunktionen befinden sich zwischen der Eingangs- und der Ausgangsschicht. Die Wahl der Anzahl und Größe verdeckter Schichten ist im allgemeinen problemabhängig und entscheidend für die Leistungsfähigkeit des gesamten Netzwerks.

Ausgangsschicht: An den Zellen der Ausgangsschicht k mit linearer Übertragungsfunktion kann die Netzantwort oder Ausgabe $\tilde{Y}(t)$ des neuronalen Netzes abgegriffen werden. Sie stehen am Ende des gerichtete Graphen.

Alle verwendeten Netztopologien entsprechen dem Schema der Abbildung 2.2 und verfügen über nur eine verdeckte Schicht.

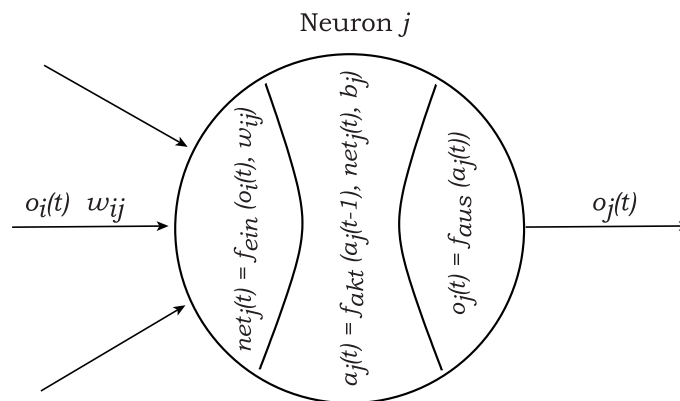


Abbildung 2.3: Mathematisches Modell einer Nervenzelle.

Die Übertragungsfunktion eines künstlichen Neurons läßt sich in drei Schritte einteilen. Die Berechnung der Zelleneingabe aus den gewichteten Ausgaben der Vorgängerzellen, Ermittlung des neuen Aktivierungszustands und die Bestimmung des Ausgabewertes anhand der neuen Aktivierung.

Neuronen:

Sie sind die zentralen Berechnungseinheiten des Netzwerks. Ihre Übertragungsfunktionen können entsprechend Abbildung 2.3 unterteilt sein in:

Eingangsfunktion: Die Eingangsfunktion $f_{ein}(o_i(t), w_{ij})$ berechnet die Netzeingabe $net_j(t)$ des Neurons j aus den Verbindungen und ihren Gewichten. Dabei handelt es sich oft um die gewichtete Summe der Ausgaben o_i vorangehender Neuronen i und den Verbindungsgewichten w_{ij} :

$$net_j(t) = \sum_i o_i(t)w_{ij} \quad (2.1)$$

Aktivierungsfunktion: Die Aktivierungsfunktion $f_{act}(a_j(t-1), net_j(t), b_j)$ bestimmt, wie sich der Aktivierungszustand $a_j(t)$ des Neurons j aus der alten Aktivierung $a_j(t-1)$ und der Netzeingabe $net_j(t)$ berechnet. Dabei ist b_j der Schwellwert bzw. Bias des Neurons, der je nach Umsetzung die Lage bzw. Form der Aktivierungsfunktion beeinflussen kann. Bei der Wahl der Exponentialfunktion zur Umsetzung einer sigmoiden Aktivierungsfunktion ergibt sich:

$$a_j(t) = \frac{2}{1 + \exp(-1 \cdot net_j(t) - b_j)} - 1 \quad (2.2)$$

Ausgangsfunktion: Die Ausgangsfunktion $f_{aus}(a_j(t))$ liefert den Ausgabewert o_j des Neurons j entsprechend seiner Aktivierung. Häufig wird keine Ausgabefunktion benötigt und $o_j(t) = a_j(t)$ gesetzt, was der Identitätsfunktion entspricht.

Lernregel:

Die Lernregel beschreibt den Algorithmus, mit dem das neuronale Netz adaptiert wird, um für jede gegebene Eingabe die gewünschte Ausgabe zu erzeugen. Das Lernen wird meistens nur durch Verändern der Verbindungsgewichte w_{ij} implementiert. Dabei kann unterschieden werden zwischen:

Überwachtes Lernen: Beim überwachten Lernen (engl. *supervised learning*) wird durch Anlegen von Ein- und Ausgabemustern ein absoluter Fehler zwischen tatsächlicher und gewünschter Netzausgabe berechnet. Die Lernregel verändert die Verbindungsgewichte so, daß der Fehler für alle möglichen Eingaben minimiert wird [RM86].

Unüberwachtes Lernen: Das nicht überwachte Lernen (engl. *unsupervised learning*) benötigt zu einer Netzeingabe keine gewünschte Ausgabe. Der Algorithmus hat die Aufgabe, die Eingabedaten zu ordnen. Ziel ist es, Eingabemuster als ähnlich zu identifizieren und Klassen (engl. *Cluster*) zuzuordnen. Aus biologischer Sicht ist dieses Verfahren am plausibelsten ([Koh82]).

Bestärkendes Lernen: Bestärkendes Lernen (engl. *reinforcement learning*) ist eine Mischform der beiden Erstgenannten. Im Gegensatz zum überwachten Lernen wird hier nicht die korrekte Ausgabe zur Verfügung gestellt, sondern nur eine Klassifizierung vorgenommen, ob die Ausgabe "richtig" oder "falsch" ist.

Im folgenden werden einige Lernregeln zum überwachten Lernen beschrieben, die im Zusammenhang mit dieser Arbeit zur Anwendung kamen. Die Beschreibung folgt dabei [Zel93] und [Lud01].

2.1.3 Resilient-Propagation-Lernregel

Resilient Propagation [BR93] ist eine Weiterentwicklung von *Backpropagation* [RM86] und verwendet Ideen der *Backpropagation-Derivate SuperSAB* [Tol90] und *Quickprop* [Fah88]. In der Vergangenheit hat sich *Resilient Propagation* für Approximationsaufgaben durchgesetzt, da es sehr viel schneller und auch stabiler konvergiert als ältere Gradientenabstiegsmethoden [SJW92].

Resilient Propagation verwendet zur Änderung der Gewichte nur das Vorzeichen des Gradienten der Fehlerfunktion, sowie die Richtung der Ableitung zum aktuellen und vorherigen Verarbeitungsschritt. Dazu erhält jedes Gewicht w_{ij} einen zusätzlichen Parameter $\Delta_{ij}(t)$ für die Änderung der Schrittweite. Zur Beeinflussung der Schrittweite stehen zwei Lernraten $0 < \eta^- < 1$ und $\eta^+ > 1$, die entsprechend der letzten Steigung verwendet werden:

$$\Delta_{ij}(t) = \begin{cases} \eta^+ \cdot \Delta_{ij}(t-1); & \text{falls } \frac{\partial E}{\partial w_{ij}}(t-1) \cdot \frac{\partial E}{\partial w_{ij}}(t) > 0 \\ \eta^- \cdot \Delta_{ij}(t-1); & \text{falls } \frac{\partial E}{\partial w_{ij}}(t-1) \cdot \frac{\partial E}{\partial w_{ij}}(t) < 0 \\ \Delta_{ij}(t-1); & \text{sonst} \end{cases} \quad (2.3)$$

Dabei bezeichnet $E = \frac{1}{2} \sum_p \sum_k \delta_{pk}^2$ die zu minimierende Fehlerfunktion mit dem Restfehler $\delta_{pk} = y_{pk} - \widehat{y}_{pk}$ der Zelle k für Trainingsmuster p . Nach dieser Berechnung des aktuellen Betrags erfolgt die Bestimmung der Richtung der Gewichtsänderung $\Delta w_{ij}(t)$ für das Gewicht w_{ij} zwischen den Neuronen i und j . Für positive Gradienten wird das Gewicht reduziert, für negative erhöht:

$$\Delta w_{ij}(t) = \begin{cases} -\Delta_{ij}(t); & \text{falls } \frac{\partial E}{\partial w_{ij}}(t) > 0 \\ +\Delta_{ij}(t); & \text{falls } \frac{\partial E}{\partial w_{ij}}(t) < 0 \\ 0; & \text{sonst} \end{cases} \quad (2.4)$$

Für den Fall, daß ein Vorzeichenwechsel zwischen der letzten ($t-1$) und der aktuellen (t) Steigung der Fehlerfunktion vorliegt, wird die letzte Gewichtsänderung rückgängig gemacht, da offensichtlich ein Minimum übersprungen wurde:

$$\Delta w_{ij}(t) = -\Delta w_{ij}(t-1); \quad \text{falls } \frac{\partial E}{\partial w_{ij}}(t-1) \cdot \frac{\partial E}{\partial w_{ij}}(t) < 0 \quad (2.5)$$

Durch diese Zurücknahme der Gewichtsänderung ist im Folgenden Schritt eine erneute Umkehr der Steigung zu erwarten. Bei *Resilient Propagation* erfolgt die Änderung der Gewichte erst nach der Präsentation aller Trainingsvektoren.

2.1.4 Levenberg-Marquardt-Algorithmus

Der LEVENBERG-MARQUARDT-Algorithmus [Mas95] basiert zum einen auf sehr alten mathematischen Methoden zur Optimierung nichtlinearer kleinster Quadrate [Lev44] und zum anderen auf der GAUSS-NEWTON-Methode [Mar63]. Die moderne Kombination dieser alten Methoden zeigt bei der Verwendung mit künstlichen neuronalen Netzen für viele Approximationsprobleme ein besseres Verhalten als klassische Gradientenabstiegsverfahren. Der LEVENBERG-MARQUARDT-Algorithmus verbindet, selbst bei schlechter Initialisierung, die Stabilität eines Gradientenabstiegsverfahrens mit der quadratischen Konvergenz der GAUSS-NEWTON-Methode, was zu seinem guten Approximationsverhalten führt [Wei99].

Gegeben sei ein Approximationsproblem für die Abbildungsfunktion $y \approx f(x, p)$ mit $\{y_1, \dots, y_m\}$ Zielwerten, $\{x_1, \dots, x_n\}$ Eingabewerten und $\{p_1, \dots, p_q\}$ abhängigen Parametern. Das Problem der nichtlinearen kleinsten Quadrate besteht darin, einen Parametervektor p zu finden, der die Summe der Quadrate zwischen einem vorgegebenen Vektor y und dem Vektor $f(x, p)$ minimiert. Die Güte der Approximation bestimmt die gewichtete Summe der Fehlerquadrate:

$$F(p) = \sum_{i=1}^n \sum_{j=1}^m w_{ij} \cdot r_{ij}^2 \quad (2.6)$$

Dabei sind w_{ij} frei wählbare Gewichte und r_{ij} relative Abweichungen der Approximationsfunktion entsprechend:

$$r_{ij} = \frac{f_i(x_j, p) - y_{ij}}{y_{ij}} \quad (2.7)$$

Zur Minimierung des Approximationsfehlers wird nun bei LEVENBERG-MARQUARDT die ursprüngliche Darstellung der Hessematrix $H = \frac{\partial^2 F(p)}{\partial p_i \partial p_j} \approx 2 \cdot J^T W J$ der GAUSS-NEWTON-Methode ersetzt durch $J^T W J + \lambda D$. Dabei ist D eine Diagonalmatrix, welche die selben Einträge wie $J^T W J$ enthält und λ der skalare LEVENBERG-MARQUARDT-Parameter, der den Einfluß der Diagonalmatrix steuert. Das Iterationsschema hat folgende Form:

$$p^{(t+1)} = p^{(t)} - (J^{(t)T} W J^{(t)} + \lambda D)^{-1} \cdot J^{(t)T} W R^{(t)} \quad (2.8)$$

Dabei ist R die Matrix der relativen Abweichungen r_{ij} der Approximationsfunktion. Der Kern der Methode ist die flexible Mischung aus Gradientenabstieg und GAUSS-NEWTON-Methode, die über den LEVENBERG-MARQUARDT-Parameter λ gewichtet wird. Die Anpassung des λ -Parameters erfolgt mit jeder Iteration abhängig von den Fehlerquadraten der Approximation:

$$\lambda^{(t+1)} = \begin{cases} \lambda^{(t)} \cdot \eta; & \text{falls } F^{(t+1)} < F^{(t)} \\ \frac{\lambda^{(t)}}{\eta}; & \text{sonst} \end{cases} \quad (2.9)$$

Wobei $\eta \in \mathbb{R}$ ein benutzerdefinierter Lernparameter ist.

2.1.5 Cascade-Correlation-Architektur

Cascade-Correlation verbindet zwei Ideen. Zum einen besteht die Netztopologie aus einer Kaskaden-Architektur (engl. *Cascade*), zu der einzelne verdeckte Neuronen hinzugefügt werden. Zum anderen werden die Verbindungsgewichte am Eingang jeder Kandidatenzelle mit Hilfe einer Korrelationssumme (engl. *Correlation*) adaptiert.

Der Lernalgorithmus beginnt zunächst ohne verdeckte Neuronen. Die Gewichte der direkten Verbindungen zwischen Ein- und Ausgangsneuronen werden z.B. durch *Resilient Propagation* über die gesamte Trainingsmenge trainiert. Um den Restfehler zu reduzieren, wird eine Kandidatenzelle j mit trainierbaren Gewichten von allen Vorgängerzellen eingefügt, deren Ausgabe zunächst nur zur Adaption dieser Gewichte dient. Ziel der Adaption ist es, die Korrelationssumme¹ S_j zwischen der Ausgabe o_j der Kandidaten-

¹Dabei handelt es sich nicht wirklich um die Summe der Korrelation, sondern eher der Kovarianz, da einige Normalisierungsterme fehlen. Nach [Fah88] verhält sich das Lernverfahren mit der hier angegebenen Kovarianz in den meisten Fällen besser.

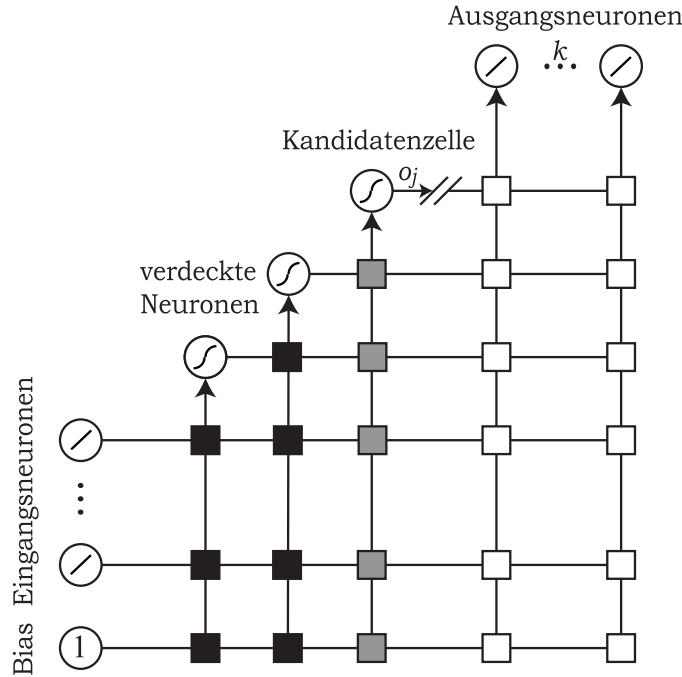


Abbildung 2.4: Cascade-Correlation-Netztopologie.

Vertikale Verbindungen bilden die gewichteten Summen, wobei die Quadrate die Gewichte w_{ij} symbolisieren. Schwarze Gewichte sind eingefroren, graue Gewichte werden beim Einfügen der Kandidatenzelle adaptiert und weiße Gewichte anschließend bei jeder Iteration trainiert.

zelle j und dem Restfehler $\delta_{pk} = y_{pk} - \widetilde{y}_{pk}$ aller Ausgabezellen k und Trainingsmuster p zu maximieren:

$$S_j = \sum_k \left| \sum_p (o_{pj} - \bar{o}_j)(\delta_{pk} - \bar{\delta}_k) \right| \quad (2.10)$$

Um S_j zu maximieren ist die partielle Ableitung nach jedem Gewicht w_{ij} der Eingangsverbindungen zu berechnen:

$$\frac{\partial S_j}{\partial w_{ij}} = \sum_k \sum_p \sigma_k \cdot f'_{akt}(net_{pj}) \cdot o_{pi} \cdot (\delta_{pk} - \bar{\delta}_k) \quad (2.11)$$

Wobei $\sigma_k = \text{sign}(\sum_p (o_{pj} - \bar{o}_j)(\delta_{pk} - \bar{\delta}_k))$ das Vorzeichen der Korrelation zwischen der Ausgabe der Kandidatenzelle j und dem Fehler des Ausgangsneurons k für Trainingsmuster p ist. $f'_{akt}(\cdot)$ bezeichnet die Ableitung der Aktivierungsfunktion der Kandidatenzelle und o_{pi} die Ausgabe der Vorgängerzelle i für Trainingsmuster p . Das Maximieren von S_j kann nun durch ein Gradientenaufstiegsverfahren z.B. *Resilient Propagation* erfolgen.

Um nun den Restfehler zu reduzieren, sind die Vorzeichen der ermittelten Gewichte zu invertieren, da sie die Korrelationssumme zu maximieren versuchen. Sind die Eingangsgewichte der Kandidatenzelle ermittelt und eingefroren, können die Ausgänge mit der Kandidatenzelle verbunden und die Ausgangsgewichte aller verdeckten Neuronen erneut trainiert werden. Ist nach dem Training der verbleibende Restfehler zu groß, kann ein weiteres Neuron eingefügt werden. Eine formale Beschreibung gibt der Algorithmus 3.3.2 in Kapitel 3, Abschnitt 3.3 im Zusammenhang mit weiteren topologieverändernden Verfahren wieder.

2.2 Statistische Maße und Wahrscheinlichkeiten

Um die Ausgaben neuronaler Netze bzw. Ergebnisse eines Vorgehens beurteilen zu können, werden Maße zur Bewertung benötigt. Solche Maße basieren in der Regel auf statistischen Grundlagen.

2.2.1 Grundlagen statistischer Maße

Die hier zusammengetragenen Grundlagen sind nur ein kurzer Überblick über die vielfältigen Methoden der Statistik. Alle folgenden, sowie weitere Betrachtungen lassen sich z.B. in [BHPT95], [Bos94] und [BD84] finden.

Definition 2.2.1 (Häufigkeit)

Gegeben sei eine Stichprobe (Menge) $x = \{x_i\}_{i=1}^n$ von n Beobachtungswerten (Werten). Die Anzahl der in der Stichprobe vorkommenden verschiedenen Merkmalswerte $x^* = \{x_k^*\}_{k=1}^N$ mit $x_1^* \neq x_2^* \neq \dots x_N^*$ sind

$$\begin{aligned} h_k &= h(x_k^*) \quad ; \text{ mit } \sum_{k=1}^N h_k = n \text{ bzw.} \\ r_k &= \frac{h_k}{n} \quad ; \text{ mit } \sum_{k=1}^N r_k = 1 \text{ und } 0 \leq r_k \leq 1 \forall k \end{aligned} \tag{2.12}$$

die absolute bzw. relative Häufigkeit von x_k^* .

Aufbauend auf den Häufigkeiten können folgende Definitionen erfolgen.

Definition 2.2.2 (Summenhäufigkeit und empirische Verteilungsfunktion)

Eine Stichprobe $x = \{x_i\}_{i=1}^n$ mit den Häufigkeiten h_k und r_k der Merkmalswerte $x^* = \{x_k^*\}_{k=1}^N$ sind die Summen

$$\begin{aligned} H_k &= \sum_{i: x_i^* \leq x_k^*} h_i \\ R_k &= \sum_{i: x_i^* \leq x_k^*} r_i \end{aligned} ; \text{ mit } k = 1, 2, \dots, N \quad (2.13)$$

die absolute und relative Summenhäufigkeit von x^* und die für jedes $\check{x} \in \mathbb{R}$ durch

$$\tilde{F}_{x^*}(\check{x}) = \sum_{i: x_i^* \leq \check{x}} h_i \quad (2.14)$$

definierte Funktion $\tilde{F}_{x^*}(\check{x})$ heißt empirische² Verteilungsfunktion der Menge x .

Neben Häufigkeiten und Verteilungsfunktionen lassen sich für Mengen von Stichproben auch folgende Lageparameter angeben.

Definition 2.2.3 (empirischer Mittelwert und Median)

Ist $\{x_i\}_{i=1}^n$ eine Menge von n Werten, dann ist

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.15)$$

der empirische Mittelwert (das arithmetische Mittel) \bar{x} der Werte von x .

Enthält die Menge x sogenannte Ausreißer, ist die Berechnung des Medians sinnvoller. Besteht dazu die Menge aus geordneten Werten $x_1 \leq x_2 \leq \dots \leq x_n$, dann ist

$$\hat{x} = \begin{cases} x_{\frac{n+1}{2}} & , \text{ falls } n \text{ ungerade ist} \\ \frac{x_{\frac{n}{2}} + x_{\frac{n}{2}+1}}{2} & , \text{ sonst} \end{cases} \quad (2.16)$$

der empirische Median oder Zentralwert der Werte von x .

²hier und im Folgenden bezeichnet das Wort empirisch, daß es sich nicht um unendliche, sondern um endliche Mengen handelt.

Zur Bewertung einer Menge $\{x_i\}_{i=1}^n$ liefert die mittlere, absolute Abweichung ein erstes Maß der Werte x_i durch folgende Definition.

Definition 2.2.4 (mittlere, absolute Abweichung)

Ist $\{x_i\}_{i=1}^n$ eine Menge von n Werten mit Mittelwert \bar{x} und Median \tilde{x} , dann sind

$$d\bar{x} = \frac{1}{n} \sum_{i=1}^n |x_i - \bar{x}| \quad \text{und} \quad d\tilde{x} = \frac{1}{n} \sum_{i=1}^n |x_i - \tilde{x}| \quad (2.17)$$

die mittlere, absolute Abweichung bezüglich \bar{x} bzw. \tilde{x} .

Häufig wird nicht die mittlere, absolute Abweichung berechnet, sondern die empirische Varianz und Standardabweichung einer Menge.

Definition 2.2.5 (empirische Varianz und Standardabweichung)

Ist $\{x_i\}_{i=1}^n$ eine Menge von n Werten mit $n \geq 2$ und Mittelwert \bar{x} , dann ist

$$s_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (2.18)$$

die empirische Varianz s_x^2 bzw. Standardabweichung $s_x = \sqrt{s_x^2}$ dieser Menge.

Für eine zweidimensionale Stichprobe (x, y) läßt sich entsprechend der Begriff der empirischen Kovarianz definieren.

Definition 2.2.6 (empirische Kovarianz)

Ist $\{x_i, y_i\}_{i=1}^n$ eine zweidimensionale Menge von n Werten mit $n \geq 2$ und den Mittelwerten \bar{x} und \bar{y} , dann ist

$$s_{xy} = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad (2.19)$$

die empirische Kovarianz von x und y .

Diese statistischen Maße bilden die Basis zur Analyse von Meßwerten, Trainings- und Testergebnissen dieser Arbeit. Neben den statistischen Grundlagen sind für die in Kapitel 2.4 beschriebene verallgemeinerte Transinformation folgende Grundlagen der Wahrscheinlichkeitsrechnung von Nutzen.

2.2.2 Grundlagen der Wahrscheinlichkeitsrechnung

Sei A ein zufälliges Ereignisses des Systems \mathcal{E} von Ereignissen. Dann läßt sich entsprechend [Bos84] aufbauend auf den Definitionen der Statistik eine Funktion $P(A)$ für die Wahrscheinlichkeit des Auftretens von A durch

$$P(A) = \lim_{n \rightarrow \infty} r_n(A)$$

angeben. Entsprechend [Bos84] stößt diese Definition auf einige Probleme, was zu folgender axiomatischen Einführung des Begriffs Wahrscheinlichkeit nach Kolmogoroff aus dem Jahre 1933 führt.

Definition 2.2.7 (Wahrscheinlichkeitsaxiome)

Eine auf einem System von Ereignissen \mathcal{E} definierte Funktion P heißt Wahrscheinlichkeit, wenn folgende Axiome

Axiom I : Für jedes Ereignis $A \in \mathcal{E}$ ist die Wahrscheinlichkeit $P(A)$ eine Zahl im Intervall $[0, 1]$:

$$0 \leq P(A) \leq 1. \quad (2.20)$$

Axiom II : Das sichere Ereignis Ω besitzt die Wahrscheinlichkeit Eins:

$$P(\Omega) = 1. \quad (2.21)$$

Axiom III: Schließen sich die Ereignisse $A \in \mathcal{E}$ und $B \in \mathcal{E}$ einander aus d.h. $A \cap B = \emptyset$, dann gilt:

$$P(A \cup B) = p(A) + p(B). \quad (2.22)$$

erfüllt sind.

Mit Hilfe dieser Axiome und den Bedingungen für das Laplacesche System von Ereignissen \mathcal{E} , welches von endlich vielen gleich wahrscheinlichen, atomaren Ereignissen $A_i \in \mathcal{E}$; mit $i \in \mathbb{N}$ gebildet wird, läßt sich die sogenannte klassische Definition der Wahrscheinlichkeit angeben.

Definition 2.2.8 (Wahrscheinlichkeit)

Die Wahrscheinlichkeit $P(A)$ des Ereignisses $A \in \mathcal{E}$ eines Laplaceschen System von Ereignissen läßt sich durch

$$P(A) = \frac{r}{m} = \frac{\text{Anzahl der für } A \text{ günstigen Ereignisse}}{\text{Anzahl der insgesamt möglichen Ereignisse}} = \frac{|A|}{|\Omega|} \quad (2.23)$$

berechnen.

Zur Verwendung von Wahrscheinlichkeiten mit zufälligen Ereignissen wird der Begriff der Zufallsvariablen eingeführt.

Definition 2.2.9 (Zufallsvariable)

Der Begriff Zufallsgröße ξ bezeichnet die Abbildung $\xi : \Omega \rightarrow \mathbb{R}$, so daß für jedes Intervall $I \subset \mathbb{R}$ die Menge

$$A = \{\omega \in \Omega: \text{mit } \xi(\omega) \in I\} \quad (2.24)$$

zum System der Ereignisse \mathcal{E} gehört, d.h. $A \in \mathcal{E}$ gilt. Die Wahrscheinlichkeit, daß die Zufallsvariable ξ Werte im Intervall I annimmt ist $P(\xi \in I)$.

Unter Verwendung bestimmter Intervalle läßt sich für Zufallsvariablen die Verteilungsfunktion definieren.

Definition 2.2.10 (Verteilungsfunktion)

Für das Intervalle $I = (-\infty, \check{x}]$ läßt sich die Wahrscheinlichkeit durch $P(\xi \leq \check{x})$ beschreiben und die Funktion

$$F_{\xi}(\check{x}) = P(\xi \leq \check{x}); \text{ mit } \check{x} \in \mathbb{R} \quad (2.25)$$

ist die Verteilungsfunktion von ξ .

Für diskrete Zufallsvariablen, deren Wertebereich eine abzählbare Menge ist, sind folgende Begriffe definiert.

Definition 2.2.11 (Wahrscheinlichkeitsverteilung)

Ist ξ eine diskrete Zufallsvariable mit den Werten $\{x_i\}_{i=1}^n$, dann ist

$$\{p_i\}_{i=1}^n = P(\xi = x_i) \quad (2.26)$$

die Einzelwahrscheinlichkeit oder Wahrscheinlichkeitsverteilung der Zufallsvariablen ξ .

Im Fall von zwei- und mehrdimensionale Zufallsvariablen ist folgende Definition von Bedeutung.

Definition 2.2.12 (Verbundwahrscheinlichkeit)

Sind ξ und η diskrete Zufallsvariablen mit den Werten $\{x_i\}_{i=1}^n$ und $\{y_k\}_{k=1}^m$, dann ist

$$\{p_{ik}\}_{i,k=1}^{n,m} = P(\xi = x_i, \eta = y_k) = P(\{\xi = x_i\} \cap \{\eta = y_k\}) \quad (2.27)$$

die Wahrscheinlichkeit dafür, daß sowohl ξ den Wert x_i als auch η den Wert y_k annimmt oder einfach die Verbundwahrscheinlichkeit $\{p_{ik}\}$ der Zufallsvariablen ξ und η .

Für Zufallsvariablen lassen sich analog zu statistischen Stichproben folgende Begriffe definieren.

Definition 2.2.13 (Erwartungswert)

Ist ξ eine Zufallsvariable mit den Werten $\{x_i\}_{i=1}^n$ und den Einzelwahrscheinlichkeiten $p_i = P(\xi = x_i)$ mit $i \in \mathbb{N}$, dann ist

$$\mu = \sum_{i=1}^{\infty} x_i p_i \approx \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.28)$$

der Erwartungswert μ der Stichprobenwerte x_i .

Definition 2.2.14 (Varianz und Standardabweichung)

Ist ξ eine Zufallsvariable mit den Werten $\{x_i\}_{i=1}^n$, dann sind

$$\sigma_x^2 = \mu[(\xi - \mu(\xi))^2] \approx s_x^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (2.29)$$

die Varianz σ_x^2 und $\sigma_x = \sqrt{\sigma_x^2}$ die Standardabweichung dieser Menge.

Für eine zweidimensionale Zufallsvariable läßt sich entsprechend der Begriff der Kovarianz definieren.

Definition 2.2.15 (Kovarianz)

Sind ξ und η Zufallsvariablen der zweidimensionalen Menge $\{x_i, y_i\}_{i=1}^n$ von n Werten mit $n \geq 2$, dann ist

$$\sigma_{xy} = \mu[(\xi - \mu(\xi))(\eta - \mu(\eta))] \approx s_{xy} = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad (2.30)$$

die Kovarianz von ξ und η .

Mit Hilfe der hier gegebenen Definitionen läßt sich die Herleitung der in Kapitel 2.4 dargestellten verallgemeinerten Transinformation aufbauen.

2.2.3 Fehlermaße und Bewertung

Neben den statistischen Maßen sind Angaben über die Abweichung zwischen Sollwerten $\{y_k\}_{k=1}^m$ und Istwerten $\{\tilde{y}_k\}_{k=1}^m$ erforderlich. Die im Folgenden aufgeführten Fehlermaße werden häufig zur Beurteilung herangezogen und lassen sich unter anderem finden bei [And97], [Zel93] und [Fro01].

Ein erstes, einfaches Maß gibt in Anlehnung an die mittlere absolute Abweichung folgende Definition.

Definition 2.2.16 (Fehlermaße)

Seien $\{y_k\}_{k=1}^m$ die Sollwerte und $\{\tilde{y}_k\}_{k=1}^m$ die Istwerte eines Systems, so ist

$$E_{MAE} = \frac{1}{m} \cdot \sum_{k=1}^m |y_k - \tilde{y}_k| \quad (2.31)$$

der mittlere absolute Fehler (engl. Mean Absolute Error) der Istwerte bezogen auf die Sollwerte.

Sind große Abweichungen stärker zu gewichten, ist

$$E_{MSE} = \frac{1}{m} \cdot \sum_{k=1}^m (y_k - \tilde{y}_k)^2 \quad (2.32)$$

der mittlere, quadratische Fehler (engl. Mean Square Error) der Istwerte bezogen auf die Sollwerte.

Soll der mittlere quadratische Fehler um die Varianz der Sollwerte bereinigt sein, so ist

$$E_{MSD} = \frac{E_{MSE}}{s_y^2} \quad (2.33)$$

der um die Varianz bereinigte, mittlere quadratische Abstand (engl. Mean Square Distance) der Istwerte bezogen auf die Sollwerte.

Durch die Quadrierung der Differenz zwischen Soll- und Istwerten geht die ursprüngliche Einheit verloren. Dies kann durch

$$E_{RMS} = \sqrt{\frac{1}{m} \cdot \sum_{k=1}^m (y_k - \tilde{y}_k)^2} \quad (2.34)$$

die Wurzel des mittleren, quadratischen Fehlers (engl. Root Mean Square) der Istwerte bezogen auf die Sollwerte behoben werden.

Neben diesen absoluten Fehlermaßen lassen sich viele andere, auch prozentuale, Angaben definieren. Da der Wertebereich $[0 \dots 100]$ der Zielanwendungen auch kleine Zahlen nahe Null enthält, wird hier auf die Definition prozentualer Maße verzichtet. Prozentuale Fehlerangaben können, insbesondere im Bereich nahe Null, sehr große Werte annehmen, was wenig Information zur Beurteilung der Güte beiträgt.

Definition 2.2.17 (Bewertungsdatensätze)

Zur Beurteilung der Leistungsfähigkeit künstlicher, neuronaler Netze ist es notwendig, die zur Verfügung stehenden Daten $X(t)$ in drei Datensätze zu unterteilen. Dabei ist darauf zu achten, daß alle Datensätze den gesamten Datenraum gut abdecken.:

Trainingsdaten: *Der Trainingsdatensatz $X_{Trn}(t)$ dient dem Training des neuronalen Netzes. Er sollte den größten Umfang haben und möglichst eine Überabdeckung des Datenraums aufweisen. Der Fehler der Netzausgabe während des Trainings wird Lernfehler genannt und kann entsprechend obiger Fehlermaße berechnet werden.*

Validierungsdaten: *Mit Hilfe des Validierungsdatensatzes $X_{Val}(t)$ kann während des Trainings überprüft werden, wie gut das neuronale Netz generalisiert, d.h. auch unbekannte Eingaben richtig klassifiziert. Bei den Anwendungen dient der Validierungsfehler oft als Abbruchkriterium des Trainings.*

Testdaten: *Zur Endkontrolle der Approximations- und Generalisierungsfähigkeit eines vollständig trainierten neuronalen Netzwerks dient der Testdatensatz $X_{Tst}(t)$. Der aus der Netzausgabe dieses Datensatzes bestimmte Testfehler kann als eigentliches Maß für die Güte des neuronalen Netzes angesehen werden.*

Bei der Unterteilung aller verwendeten Datensätze ist jeweils ein Verhältnis von 1/2 Trainingsdaten zu 1/4 Validierungsdaten und 1/4 Testdaten gewählt worden. Eine Verteilung auf den Datenraum der Anwendungen zur Motordiagnose kann der Abbildung 5.2 aus Kapitel 5 Abschnitt 5.1.3 entnommen werden.

2.3 Grundlagen der Signalanalyse

Die in Kapitel 4 Abschnitt 4.2 vorgestellte, informationsoptimierte Frequenzanalyse ist als Erweiterung der herkömmlichen Fourier-Analyse zu verstehen. Um die Funktionsweise des Algorithmus besser verstehen zu können, sind hier einige Grundlagen der Signalverarbeitung zusammengefasst. Die Abhandlung der Thematik ist an [Lük92] angelehnt.

2.3.1 Elementarsignale

Elementarsignale können besonders einfach durch einen mathematischen Ausdruck vollständig beschrieben werden. Für spätere Beispiele und die in Abschnitt 2.3.5 vorgestellte Oktavbandzerlegung [Hof95] und [AARH92] ist die Definition einiger elementarer Signale hilfreich.

Chirp-Signal:

Ein erstes, wichtiges Elementarsignal läßt sich durch folgenden, einfachen algebraischen Ausdruck beschreiben:

$$\text{chirp}(t) = \cos\left(\frac{f_s}{2T} \cdot \pi \cdot t^2\right) \quad (2.35)$$

Dabei handelt es sich um das sogenannte *Chirp*-Signal, das für $t = \{0, \frac{1}{f_s}, \frac{2}{f_s}, \dots, T\}$ im zeitlichen Verlauf alle Frequenzen von $]0 - \frac{f_s}{2}] Hz$ aufsteigend enthält, wobei f_s die Abtastfrequenz bezeichnet. Abbildung 2.5 (a) zeigt ein entsprechendes *Chirp*-Signal.

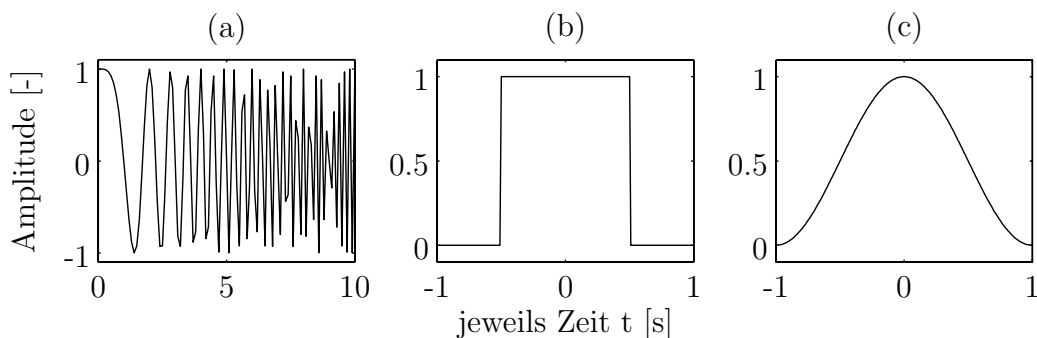


Abbildung 2.5: Ausgewählte Elementarsignale.

- (a) *Chirp*-Signal des Kosinus mit ansteigender Frequenz.
- (b) *Rechteck*-Impuls der Fläche Eins zum Zeitpunkt Null.
- (c) *Hanning*-Fenster der Höhe Eins und Zeitdauer 2 Sekunden.

Folgenden Signale sind stückweise definiert und lassen sich in folgender Weise in normierter Form darstellen.

Rechteck-Impuls:

Der *Rechteck*-Impuls läßt sich stückweise beschreiben durch:

$$\text{rect}(t) = \begin{cases} 1 & ; \text{ falls } |t| \leq \frac{1}{2} \\ 0 & ; \text{ falls } |t| > \frac{1}{2} \end{cases} \quad (2.36)$$

Für $t = \{-1 \dots 1\}$ ist ein *Rechteck*-Impuls in Abbildung 2.5 unter (b) dargestellt.

Hanning-Fenster:

Neben dem *Rechteck*-Impuls ist für die gleiche Zeit t das ebenfalls stückweise definierte *Hanning*-Fenster in Abbildung 2.5 unter (c) dargestellt.

$$\text{hann}(t) = \begin{cases} \frac{1}{2} \cdot (1 - \cos(\pi \cdot (t + 1))) & ; \text{ falls } |t| \leq 1 \\ 0 & ; \text{ falls } |t| > 1 \end{cases} \quad (2.37)$$

Gleichverteiltes Rauschen:

Für einige Beispiele ist die Betrachtung von gleichverteiltem³ Rauschen bzw. von Zufallswerten von Nutzen. Neben einer guten theoretischen Grundlage geben PARK und MILLER in [PM88] unter anderem auch folgende einfache Formel an:

$$s(t + 1) = \text{mod}(a \cdot s(t), m) \quad (2.38)$$

Eine gute Wahl für a und m wird angegeben durch:

$$a = 7^5 = 16807 \text{ und } m = 2^{31} - 102147483647 \quad (2.39)$$

Dieser einfache Generator von Zufallswerten liefert gleichverteilte Zufallswerte und besteht nach PRESS und FARRAR [PTVF97] die meisten statistischen Tests. Obwohl der Generator vielfach erfolgreich eingesetzt werden konnte, erhebt er nicht den Anspruch perfekt zu sein.

Eine Umsetzung ist auf 32-bit Prozessoren nicht möglich, da das Produkt von a und m den Bereich von 32-bit Integern übersteigt. Entsprechend [Sch79] lassen sich diese Probleme durch folgende approximierende Faktorisierung von m lösen:

$$m = a \cdot q + r ; \text{ mit } q = \langle m/a \rangle \text{ und } r = \text{mod}(m, a) \quad (2.40)$$

wobei die spitzen Klammern $\langle \cdot \rangle$ den Integer-Anteil bezeichnen. Für $r < q$ kann gezeigt werden, daß $a \cdot \text{mod}(\check{s}(t), q)$ und $r \cdot \langle \check{s}(t)/q \rangle$ im Intervall $[0, m - 1]$ liegen und

³Zur Definition der Gleichverteilung siehe auch Abschnitt 2.4.3

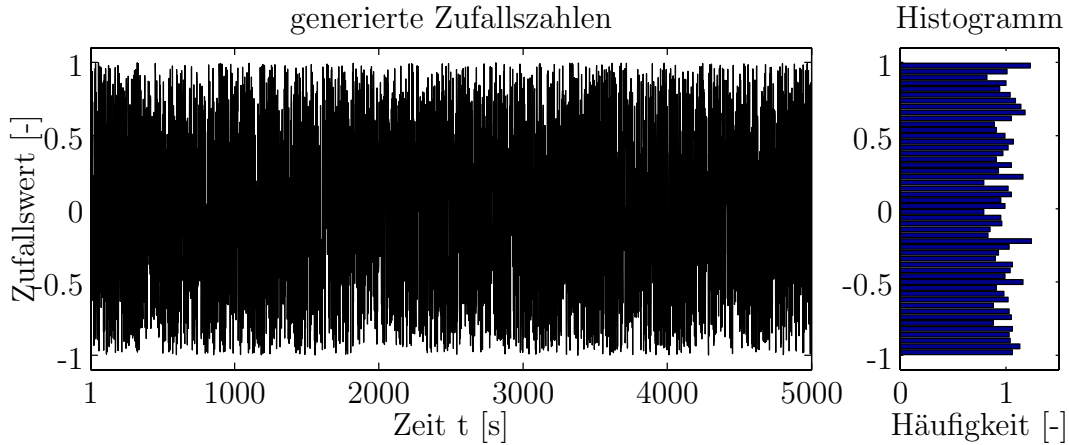


Abbildung 2.6: Gleichverteiltes Rauschen.

Rauschsignal des Zufallszahlengenerators B.0.3. Das Histogramm belegt die nahezu Gleichverteilung, wobei die Ausreißer auf die begrenzte Signallänge zurückzuführen sind.

$$\check{s}(t+1) = \text{mod}(a \cdot \check{s}(t), m) = \begin{cases} a \cdot \text{mod}(\check{s}(t), q) - r \cdot \langle \check{s}(t)/q \rangle & ; \text{für } \check{s}(t+1) \geq 0 \\ a \cdot \text{mod}(\check{s}(t), q) - r \cdot \langle \check{s}(t)/q \rangle + m & ; \text{sonst} \end{cases} \quad (2.41)$$

gilt, wobei jeweils $0 < \check{s}(t) < m - 1$, was auch als Bedingung für $\check{s}(0)$ einzuhalten ist. Für eine gute Approximation von m werden $q = 127773$ und $r = 2836$ vorgeschlagen. Die Umsetzung eines entsprechenden Generators ist im Anhang B.0.3 zu finden.

Abbildung 2.6 zeigt eine von diesem Generator erzeugte Zufallsreihe, sowie die normierte Verteilung der so entstandenen Zufallswerte. Bei exakter Gleichverteilung tritt jeder Zahlenwert gleich häufig auf, wodurch das Histogramm auf der gesamten Höhe konstant Eins betragen würde.

statistisches Maß	realer Wert	theoretischer Wert
Mittelwert	0.003	0.0
Standardabweichung	0.581	0.5
Varianz	0.336	0.3

Tabelle 2.1: Statistische Werte für gleichverteiltes Rauschen.

Durch die begrenzte Länge der Zahlenfolge weichen die statistischen Maße der Tabelle 2.1, sowie die Gleichverteilung von den theoretischen Werten für unbegrenzte Folgen ab.

2.3.2 Transformationen auf Signale

Aus allen Elementarsignalen können zeitlich gedehnte und verschobene Signale durch einfache Koordinatentransformation der Zeitachse gebildet werden. Durch den zusätzlichen Amplitudenfaktor a läßt sich jeder beliebige *Rechteck*-Impuls darstellen.

Beispiel 2.3.1 (Transformation)

Der Rechteck-Impuls aus Gleichung 2.36 kann durch folgende Transformation zu jedem beliebigen Zeitpunkt t_0 mit jeder beliebigen Dauer T und mit jeder Amplitude a erfolgen.

$$a \cdot \text{rect}\left(\frac{t-t_0}{T}\right) = \begin{cases} a & ; \text{ falls } \left|\frac{t-t_0}{T}\right| \leq \frac{1}{2} \\ 0 & ; \text{ falls } \left|\frac{t-t_0}{T}\right| > \frac{1}{2} \end{cases}$$

Dies Transformation läßt sich entsprechend auf alle anderen Signale übertragen.

2.3.3 Faltungsintegrale

Die Herleitung des Faltungsintegrals erfolgt entsprechend [Lük92] anhand linearer zeitinvarianter Systeme. Ein solches System reagiert auf einen Rechteck-Impuls $\text{rect}(t)$ der Dauer T und der Amplitude $\frac{1}{T}$ mit dem Ausgangssignal $g_{\text{rect}}(t)$.

Die Reaktion $g(t)$ des Systems auf ein beliebiges Eingangssignal $s(t)$ läßt sich auf der Basis von $\text{rect}(t)$ näherungsweise bestimmen. Das Eingangssignal $s(t)$ läßt sich so durch eine Treppenfunktion $s_a(t)$ approximieren:

$$s(t) \approx s_a(t) = \sum_{n=-\infty}^{+\infty} s(nT) \cdot \text{rect}(t - nT)T; \text{ mit } n \in \mathbb{N} \quad (2.42)$$

Für lineare zeitinvariante Systeme läßt sich $g(t)$ nach dem Superpositionssatz⁴ approximieren durch:

$$g(t) \approx g_a(t) = \sum_{n=-\infty}^{+\infty} s(nT) \cdot g_{\text{rect}}(t - nT)T; \text{ mit } n \in \mathbb{N} \quad (2.43)$$

Es ist unmittelbar einzusehen, daß die Approximation um so genauer wird, je geringer die Dauer T des Rechteck-Impulses ist. Im Grenzübergang $T \rightarrow 0$ wird der Rechteck-Impuls $\text{rect}(t)$ durch das mathematische Modell des *Dirac*-Stoßes⁵ $\delta(t)$ beschrieben.

⁴Für eine beliebige Transformation $g(t) = \text{Tr}\{s(t)\}$ und beliebige Konstanten a_i gilt [Lük92]:
 $\text{Tr}\{\sum_i a_i s_i(t)\} = \sum_i a_i \text{Tr}\{s_i(t)\} = \sum_i a_i g_i(t)$

⁵Für meßtechnische Zwecke kann der *Dirac*-Stoß als kurzer Rechteck-Impuls gedeutet werden:
 $\delta(t) \approx \lim_{T \rightarrow 0} \frac{1}{T} \cdot \text{rect}\left(\frac{t}{T}\right)$

Das zugehörige Ausgangssignal wird als Stoßantwort $h(t)$ bezeichnet. Aus den Gleichungen 2.42 und 2.43 ergeben sich nach den im Grenzübergang gültigen Beziehungen $nT \rightarrow \tau$ und $T \rightarrow d\tau$ die Faltungsintegrale:

$$\begin{aligned} s(t) &= \int_{-\infty}^{+\infty} s(\tau)\delta(t-\tau)d\tau \\ g(t) &= \int_{-\infty}^{+\infty} s(\tau)h(t-\tau)d\tau \end{aligned} \quad (2.44)$$

Die Faltungsintegrale 2.44 sind eine für lineare zeitinvariante Systeme allgemein geltende Transformation. Sie lassen sich abgekürzt durch sogenannte Faltungsprodukte “*” darstellen:

$$\begin{aligned} s(t) &= s(t) * \delta(t) \\ g(t) &= s(t) * h(t) \end{aligned} \quad (2.45)$$

Die wichtigsten Regeln der Faltungsalgebra entsprechen den Grundregeln der Multiplikation und sind durch folgende Gleichungen zusammengefaßt:

$$\begin{aligned} \text{Eins-Element:} & \quad g(t) = g(t) * \delta(t) \\ \text{Kommutativgesetz:} & \quad g(t) = s(t) * h(t) = h(t) * s(t) \\ \text{Assoziativgesetz:} & \quad g(t) * s(t) * h(t) = [g(t) * s(t)] * h(t) = g(t) * [s(t) * h(t)] \\ \text{Distributivgesetz:} & \quad g(t) * [s(t) + h(t)] = [g(t) * s(t)] + [g(t) * h(t)] \end{aligned}$$

Eine ausführliche Darstellung der Gesetze, sowie des Dirac-Stoßes findet sich in [Lük92].

2.3.4 Fourier-Transformation

Zur Herleitung der Fourier-Transformation wird entsprechend [Lük92] die Eigenfunktion

$$S_E(t) = e^{j2\pi ft} = \cos(2\pi ft) + j \sin(2\pi ft) \quad (2.46)$$

herangezogen. Sie hat die Eigenschaft, daß ihre Form durch Faltung mit einer beliebigen Funktion $s(t)$

$$s(t) * S_E(t) = a \cdot S_E(t), \quad (2.47)$$

abgesehen von der Amplitude a , unverändert bleibt. Die Berechnung des Faltungsintegrals

$$\begin{aligned} s(t) * e^{j2\pi ft} &= \int_{-\infty}^{+\infty} s(\tau)e^{j2\pi f(t-\tau)}d\tau \\ &= \int_{-\infty}^{+\infty} s(\tau)e^{-j2\pi f\tau}d\tau \cdot e^{j2\pi ft} = S(f) \cdot e^{j2\pi ft} \end{aligned} \quad (2.48)$$

liefert nach Ersetzen von τ durch t die sogenannte Fourier-Transformation vom Zeit- in den Frequenzbereich, sowie ihrer Inversen:

$$\begin{aligned} S(f) &= \int_{-\infty}^{+\infty} s(t) e^{-j2\pi ft} dt \\ s(t) &= \int_{-\infty}^{+\infty} S(f) e^{j2\pi ft} df \end{aligned} \quad (2.49)$$

Entsprechend Gleichung 2.47 und der Faltungsalgebra ergibt sich

$$\begin{aligned} [S_E(t) * s_1(t)] * s_2(t) &= [S_1(f) \cdot S_E(t)] * s_2(t) \\ &= S_1(f) \cdot S_2(f) \cdot S_E(t), \end{aligned} \quad (2.50)$$

wodurch sich das Faltungsprodukt zweier Signale im Zeitbereich als algebraisches Produkt im Frequenzbereich darstellen läßt:

$$\begin{aligned} s(t) * h(t) &= g(t) \quad (\text{Zeit-Bereich}) \\ S(f) \cdot H(f) &= G(f) \quad (\text{Frequenz-Bereich}) \end{aligned} \quad (2.51)$$

In symbolischer Darstellung $s(t) \circ \bullet S(f)$ der Gleichung 2.49 des Übergangs vom Zeit- in den Frequenzbereich läßt sich dieser Zusammenhang entsprechend Gleichung 2.52 vereinfacht darstellen. Die umgekehrte Darstellung ist nach dem Multiplikationstheorem ebenfalls gültig:

$$\begin{aligned} s(t) * h(t) &\circ \bullet S(f) \cdot H(f) \\ s(t) \cdot h(t) &\circ \bullet S(f) * H(f) \end{aligned} \quad (2.52)$$

Das Multiplikationstheorem folgt aus der Symmetrie der Fourier-Transformation, die sich wie folgt darstellen läßt:

$$S(t) \circ \bullet s(-f) \quad (2.53)$$

Die Symmetrie ergibt sich aus der Tatsache, daß sich die Inverse der Fourier-Transformation nur durch ein Minus im Exponenten $e^{-j2\pi ft}$ unterscheidet.

Im Folgenden werden einige wichtige Sätze der Fourier-Transformation hergeleitet, die den Umgang vereinfachen.

Satz 2.3.1 (Superpositionsprinzip)

Die Fourier-Transformierte einer Summe von Zeitfunktionen ist gleich der Summe der Fourier-Transformierten der einzelnen Zeitfunktionen.

$$\sum_i a_i s_i(t) \circ \bullet \sum_i a_i S_i(f) \quad (2.54)$$

Beweis 2.3.1

Aus Umformungen der Gleichung

$$\sum_i a_i s_i(t) \circ \bullet \int_{-\infty}^{+\infty} \sum_i a_i s_i(t) \cdot e^{-j2\pi ft} dt \quad (2.55)$$

folgt die Behauptung.

[qed]

Satz 2.3.2 (Ähnlichkeitsprinzip)

Der Zusammenhang zwischen einem gedehnten bzw. gestauchten Signal $s(b \cdot t)$ für ein reelles b und der entsprechenden Fourier-Transformierten ist gegeben durch:

$$s(b \cdot t) \circ \bullet \frac{1}{|b|} S\left(\frac{f}{b}\right) \quad (2.56)$$

Daraus ergibt sich für $|b| > 1$ eine Stauchung der Zeitfunktion $s(t)$ und eine Dehnung des Real- und Imaginärteils, sowie des Betrags und der Phase der Fourier-Transformierten $S(f)$.

Beweis 2.3.2

Der Ansatz

$$s(b \cdot t) \circ \bullet \int_{-\infty}^{+\infty} s(b \cdot t) e^{-j2\pi ft} dt \quad (2.57)$$

führt nach der Substitution $b \cdot t = \theta$ zu:

$$\begin{aligned} s(b \cdot t) &\circ \bullet \frac{1}{b} \int_{-\infty}^{+\infty} s(\theta) e^{-j2\pi\theta f/b} d\theta && ; \text{ für positive } b \\ s(b \cdot t) &\circ \bullet -\frac{1}{b} \int_{-\infty}^{+\infty} s(\theta) e^{-j2\pi\theta f/b} d\theta && ; \text{ für negative } b \end{aligned} \quad (2.58)$$

Die Betragsbildung ergibt die Behauptung.

[qed]

Satz 2.3.3 (Verschiebungsprinzip)

Wird ein Signal $s(t)$ auf der Zeitachse um eine feste Zeit t_0 verzögert, bleibt das Betragsspektrum invariant gegenüber dieser Verschiebung.

$$s(t - t_0) \circ \bullet S(f) \cdot e^{-j2\pi ft_0} \quad (2.59)$$

Der Verschiebungsfaktor $e^{-j2\pi ft_0}$ beeinflusst lediglich das Phasenspektrum.

Beweis 2.3.3

Für $s(t)$ mit fester Verzögerung t_0 gilt:

$$s(t - t_0) \circ \bullet \int_{-\infty}^{+\infty} s(t - t_0) e^{-j2\pi ft} dt \quad (2.60)$$

Mit Hilfe der Substitution $t - t_0 = \theta$ folgt entsprechend obigem Beweis die Behauptung. [qed]

Die hier gegebene Beschreibung der Fourier-Transformation und einiger Eigenschaften findet z.B. in der sogenannten Oktavbandzerlegung Anwendung. Weitere Grundlagen und Anwendungen können in [Lük92] gefunden werden.

2.3.5 Oktavbandzerlegung

Neben der reinen Frequenzanalyse eines Signals kann die Fourier-Transformation zur sogenannten Oktavbandzerlegung [Hof95] verwendet werden. Dazu wird ein Signal nach Transformation in den Frequenzbereich durch Multiplikation mit entsprechenden Filterfunktionen in Oktavbänder zerlegt. Durch Rücktransformation erhält man oktavbandzerlegte Teilsignale. Entsprechend des Superpositionsprinzips (Satz 2.3.1) reproduziert die Komposition der Teilsignale das Originalsignal (siehe auch [Fli93] und [AARH92]).

Abbildung 2.7 veranschaulicht die Zerlegung in Frequenzbänder mit Hilfe von Hanning-Fenstern. Unter (a) ist das Originalsignal zu erkennen. Bei dem Ausgangssignal handelt es sich um das Chirp-Signal, das neben der Fensterfunktion in Abschnitt 2.3.1 beschrieben ist. Durch gleichmäßige Aufteilung des Frequenzraums in hohe wie tiefe Frequenzen entstehen die beiden Teilsignale unter (b). Eine weitere Unterteilung dieser Teilsignale ist mit (c) verdeutlicht. Weitere Unterteilungen führen zu Teilsignalen, die nur aus wenigen Frequenzen aufgebaut sind.

Die beschriebene Methode findet z.B. Anwendung in der Nachrichtenübertragung. Enthält ein Signal nur in manchen Bändern Information, so kann die Codierung der Übertragung auf diese Bänder beschränkt sein.

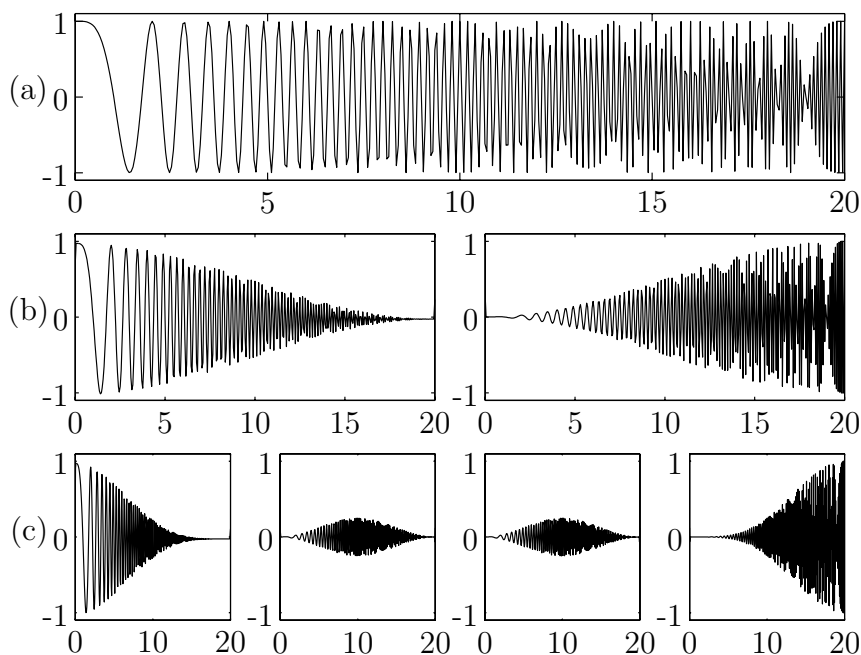


Abbildung 2.7: Oktavbandzerlegung eines Chirp-Signals.

- (a) Chirp-Signal mit allen Frequenzen im Bereich $]0, \dots, F]$ Hz.
- (b) Zerlegung in zwei Oktavanteile. $]0, \dots, \frac{F}{2}]$ und $]\frac{F}{2}, \dots, F]$ Hz.
- (c) Zerlegung in vier Oktavanteile. Bei Rekonstruktion ergeben die dargestellten Signalanteile wieder das Originalsignal.

Für die Hanning-Fenster aus Abschnitt 2.3.1 ergeben sich im Frequenzbereich $f \in \{0, \dots, F\}$ folgende Gleichungen. Dabei ist die Zeitachse durch die Frequenzachse und somit t durch f zu ersetzen:

$$H_n(f) = \text{hann}\left(\frac{\varphi_n}{k}\right) \quad ; \quad \text{mit } \varphi_n = f - f_n$$

$$\sum_{n=1}^u H_n(f) = 1 \quad ; \quad \text{für alle } f \in \{0, \dots, F\} \quad (2.61)$$

mit

$$f_n = n \cdot \frac{F}{u} \quad \text{und} \quad k = \frac{2}{\pi} \cdot \frac{F}{u}; \quad n = \{0, 1, \dots, u\} \quad \text{und} \quad u \in \mathbb{N}$$

Dabei drückt k die Breite der Hanning-Fenster aus, die von der Anzahl u an Unterteilungen des Frequenzbereichs $f \in \{0, \dots, F\}$ abhängt. Die Verschiebungen φ legen die Fenster so, daß ihre Summe über dem definierten Frequenzbereich Eins ergibt⁶.

⁶Bei der Verwendung anderer Fensterfunktionen ist ebenfalls darauf zu achten, daß die Summe der

2.4 Verallgemeinerte Transinformation

Die Qualität neuronaler Modelle hängt stark von der Qualität bzw. dem Informationsgehalt der zum Training verwendeten Daten ab. Da neuronale Netze oft auf der Basis von Meßdaten entwickelt werden, ist eine Methode zur Bestimmung des Informationsgehaltes der Meßdaten von großem Interesse. Dabei interessiert gerade der statistische Zusammenhang zwischen den Ein- und Ausgängen des neuronalen Netzes; mit anderen Worten, die gemeinsame Information der Ein- und Ausgabevektoren, die den Systemzusammenhang am besten beschreibt. Die Berechnung dieser sogenannten Transinformation⁷ kann durch Verfahren zur Bestimmung von Informationsflüssen aus der Informationstheorie erfolgen. Hierbei wird zunächst auf das theoretische Modell eines Informationskanals zurückgegriffen.

2.4.1 Shannon Entropie und Informationskanal

SHANNON [Sha48] führt das Konzept der Transinformation zur allgemeinen, quantitativen Beschreibung der Übertragung von Informationsflüssen über einen abstrakten, gestörten Informationskanal ein, wobei $x(t)$ und $y(t)$ die Ein- und Ausgabesignale darstellen. Dabei werden beide Signale als Zufallsvariablen ξ und η mit den Wahrscheinlichkeitsverteilungen $\{p_m\}$ und $\{q_n\}$ angesehen.

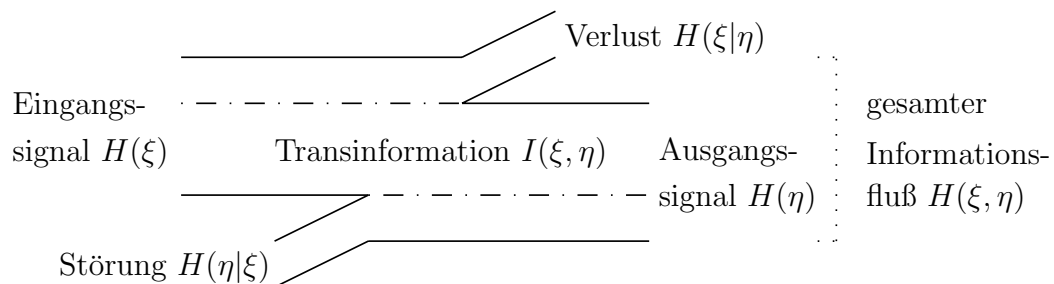


Abbildung 2.8: Informationskanal nach Shannon.

Der Informationskanal zeigt die Zusammenhänge und Bedeutungen der von SHANNON definierten Informationsmaße. Tabelle 2.2 verdeutlicht nochmals die Abdeckung des gesamten Informationsflusses.

Zur Berechnung der Transinformation $I(\xi, \eta)$ des in Abbildung 2.8 dargestellten Kanals hat SHANNON die Informationsmaße $H(\xi)$, $H(\eta)$ und $H(\xi, \eta)$ eingeführt. Für zwei Zu-

Einhüllenden über den gesamten Frequenzbereich konstant Eins beträgt.

⁷Bei der Einführung der Kanaltheorie verwendete SHANNON den Begriff Übertragungsrate (engl. *rate of transmission*). Heute sind auch Bezeichnungen wie *relative Information*, *Syngentropie* und *mutual information* gebräuchlich.

fallsgrößen ξ und η mit den Wahrscheinlichkeitsverteilungen $\{p_m\}_{m=1}^M$, $\{q_n\}_{n=1}^N$ und der Verbundwahrscheinlichkeit $\{s_{mn}\}_{m,n=1}^{M,N}$ beider Zufallsgrößen lassen sich nach SHANNON die Informationsmaße durch folgende Entropien definieren:

$$\begin{aligned} H(\xi) &= - \sum_{m=1}^M p_m \log p_m \\ H(\eta) &= - \sum_{n=1}^N q_n \log q_n \\ H(\xi, \eta) &= - \sum_{m,n=1}^{M,N} s_{m,n} \log s_{m,n} \end{aligned} \quad (2.62)$$

Sieht man nun zwei, möglicherweise mehrdimensionale Signale $x(t)$ und $y(t)$ als Realisierung zufälliger Prozesse an, so kann man mit Hilfe dieser Informationsmaße deren Transinformation bestimmen und somit eine Aussage geben, wie groß der Informationsfluß zwischen den Signalen ist.

$H(\xi, \eta)$		
$H(\xi)$		
	$H(\eta)$	
$H(\xi \eta)$	$I(\xi, \eta)$	$H(\eta \xi)$

Tabelle 2.2: Abdeckung des Informationsflusses durch Informationsmaße.

Tabelle 2.2 stellt die Zusammenhänge der einzelnen Informationsmaße und deren Abdeckungen des gesamten Informationsflusses dar. Dabei ist der Informationsfluß von oben nach unten zu verstehen. Entsprechend Tabelle 2.2 läßt sich die Transinformation $I(\xi, \eta)$ nach folgender Gleichung bestimmen:

$$I(\xi, \eta) \equiv H(\eta) - [H(\xi, \eta) - H(\xi)] \quad (2.63)$$

Dabei ist $H(\eta)$ die A-priori-Unsicherheit über η und $[H(\xi, \eta) - H(\xi)]$ die über η noch verbleibende A-posteriori-Unsicherheit, wenn ξ bekannt ist. Die Differenz $I(\xi, \eta)$ ist die Information über η unter Kenntnis von ξ . Für unabhängige Zufallsvariablen ist $I(\xi, \eta) = 0$. Kann η ganz durch ξ bestimmt werden, so ist $I(\xi, \eta) = 1$.

2.4.2 Eigenschaften der Transinformation

Zur Berechnung der Transinformation aus digitalen Daten betrachten wir einige Eigenschaften für den diskreten Fall. Für eine Betrachtung des kontinuierlichen Falles sei hier auf [Pom97] verwiesen. In Hinblick auf die spätere Implementierung findet insbesondere der Fall $\alpha = 2$ der Rényi-Entropie [Rén77] Beachtung:

$$H_\alpha(\xi) \equiv \begin{cases} \frac{1}{1-\alpha} \log \sum_{m=1}^M p_m^\alpha & ; \alpha \geq 0, \alpha \neq 1 \\ - \sum_{m=1}^M p_m \log p_m & ; \alpha = 1 \end{cases} \quad (2.64)$$

Dabei sei $\{p_m\}_{m=1}^M$ die Wahrscheinlichkeitsverteilung der Zufallsvariablen ξ und der Ordnung $\alpha \in \mathbb{R}$ mit $\alpha \geq 0$.

Bezüglich einer weiteren Wahrscheinlichkeitsverteilung $\{q_n\}_{n=1}^N$ der Zufallsvariablen η und der Verbundwahrscheinlichkeit $\{s_{mn}\}_{m,n=1}^{M,N}$ aus (ξ, η) können nun auf der Basis der Rényi-Entropie verschiedene Informationsmaße berechnet werden:

$$I_\alpha(\xi, \eta) \equiv H_\alpha(\xi) + [H_\alpha(\eta) - H_\alpha(\xi, \eta)] \quad (2.65)$$

Für $\alpha = 1$ sind $H_1(\xi)$ und $H_1(\eta)$ die Informationsmaße nach SHANNON, wobei oft der Logarithmus zur Basis 2 gewählt wird, wodurch die Information die Einheit *bit*⁸ erhält. Die Transinformation nach SHANNON ist somit

$$I_1(\xi, \eta) = \sum_{m,n=1}^{M,N} s_{mn} \log_2 \frac{s_{mn}}{p_m q_n} \quad (2.66)$$

und hat die im Folgenden Satz zusammengefaßten Eigenschaften.

Satz 2.4.1 (Eigenschaften der Transinformation)

Für die Zufallsvariablen ξ und η hat die Transinformation der Gleichung 2.66 folgende Eigenschaften:

1. *Symmetrie:* $I_1(\xi, \eta) = I_1(\eta, \xi)$
2. *Beschränktheit:* $0 \leq I_1(\xi, \eta) \leq \min\{H_1(\xi), H_1(\eta)\}$
3. *Unabhängigkeit:* $I_1(\xi, \eta) = 0 \Leftrightarrow \xi$ und η sind unabhängig
4. *Determiniertheit:* $I_1(\xi, \eta) = H_1(\eta) \Leftrightarrow \eta$ ist eine Funktion von ξ
 $I_1(\xi, \eta) = H_1(\xi) \Leftrightarrow \xi$ ist eine Funktion von η

⁸Das Wort *bit* verwendete SHANNON und bezeichnet auch hier die Information aus zwei gleich wahrscheinlichen Ereignissen: $H_\alpha(\frac{1}{2}, \frac{1}{2}) = \frac{1}{1-\alpha} \log_2(\frac{1}{2^\alpha} + \frac{1}{2^\alpha}) = \frac{1}{1-\alpha} \log_2 2^{1-\alpha} = 1 \text{ bit}$

Beweis 2.4.1

Siehe z.B. [Rén77] Seiten 450 ff. oder [AD75] Seiten 196 ff.

Diese Eigenschaften machen die Transinformation zu einem geeigneten Maß für statistische Abhängigkeiten zwischen Zufallsgrößen. Die Informationsmaße $I_\alpha(\xi, \eta)$ mit $\alpha \geq 0$ und $\alpha \neq 1$ sind jedoch nicht in jedem Fall als Maß der statistischen Abhängigkeit geeignet, was folgender Hilfssatz belegt.

Lemma 2.4.1 (Unbeschränktheit allgemeiner Informationsmaße)

Ist $\{s_{mn}\}$ die endlich diskrete Verbundwahrscheinlichkeit⁹ der Zufallsvektoren ξ und η , so gibt es Verteilungen, für welche die Informationsmaße $I_\alpha(\xi, \eta)$ negative Werte annehmen. Nur für $\alpha = 0$ und $\alpha = 1$ gilt $I_\alpha(\xi, \eta) \geq 0$ für beliebige Verbundwahrscheinlichkeiten.

Beweis 2.4.2

Siehe z.B. [Rén77] Seiten 469 ff. für Fälle $\alpha > 0$ oder [AD75] Seiten 193 ff.

Die für $\alpha = 2$ in $I_2(\xi, \eta)$ eingehende Summe über Quadrate von Wahrscheinlichkeiten kann besonders effektiv aus endlichen Zeitreihen geschätzt werden. Um $I_2(\xi, \eta)$ als Maß für statistische Abhängigkeiten zu verwenden dient der folgende Satz.

Satz 2.4.2 (Eigenschaften des Informationsmaßes $I_2(\xi, \eta)$)

Seien $\{p_m\}_{m=1}^M$, $\{q_n\}_{n=1}^N$ und $\{s_{mn}\}_{m,n=1}^{M,N}$ Wahrscheinlichkeitsverteilungen¹⁰ der Zufallsgrößen ξ und η und sei η gleichverteilt¹¹, also $q_n = \frac{1}{N}$ für alle $n = 1, 2, \dots, N$, dann hat Gleichung 2.64 für $\alpha = 2$, also $I_2(\xi, \eta)$ die Eigenschaften:

1. Symmetrie: $I_2(\xi, \eta) = I_2(\eta, \xi)$
2. Beschränktheit: $0 \leq I_2(\xi, \eta) \leq \min\{H_2(\xi), H_2(\eta)\}$
3. Unabhängigkeit: $I_2(\xi, \eta) = 0 \Leftrightarrow \xi$ und η sind unabhängig
4. Determiniertheit: $I_2(\xi, \eta) = H_2(\eta) \Leftrightarrow \eta$ ist eine Funktion von ξ
 $I_2(\xi, \eta) = H_2(\xi) \Leftrightarrow \xi$ ist eine Funktion von η

Unter der Bedingung der Gleichverteilung von η eignet sich $I_2(\xi, \eta)$ als Maß für statistische Abhängigkeiten der Zufallsvariablen ξ und η .

⁹Es gelte: $\sum_{m=1}^M s_{mn} = q_n$ und $\sum_{n=1}^N s_{mn} = p_m$

¹⁰Die Wahrscheinlichkeitsverteilungen seien immer normiert: $\sum_{m=1}^M p_m = 1$ mit $p_m \geq 0$

¹¹Gleichverteilung für normierte Wahrscheinlichkeitsverteilungen bedeutet (siehe Abschnitt 2.4.3):

$$q_n = \frac{1}{N}, \text{ mit } n = 1, 2, \dots, N \text{ und somit } \sum_{n=1}^N q_n^2 = N \cdot \frac{1}{N^2} = \frac{1}{N}$$

Beweis 2.4.3

Die Symmetrie folgt aus $H_2(\xi, \eta) = H_2(\eta, \xi)$.

Zum Beweis der restlichen Eigenschaften verwenden wir die vorausgesetzte Gleichverteilung von η mit $\sum_{n=1}^N q_n^2 = \frac{1}{N}$ und erhalten:

$$\begin{aligned}
I_2(\xi, \eta) &= \log \sum_{m,n} s_{mn}^2 - \log \sum_m p_m^2 - \log \frac{1}{N} \\
&= \log \frac{\sum_{m,n} s_{mn}^2}{\frac{1}{N} \sum_m p_m^2} \\
&= \log \left(1 + \frac{\sum_{m,n} s_{mn}^2 - \frac{1}{N} \sum_m p_m^2}{\frac{1}{N} \sum_m p_m^2} \right) \\
&= \log \left(1 + \frac{\sum_{m,n} s_{mn}^2 - \frac{2}{N} \sum_m p_m^2 + \frac{1}{N} \sum_m p_m^2}{\frac{1}{N} \sum_m p_m^2} \right) \\
&= \log \left(1 + \frac{\sum_{m,n} s_{mn}^2 - \frac{2}{N} \sum_{m,n} \frac{1}{N} \cdot p_m \cdot p_m + \frac{1}{N} \cdot \frac{1}{N} \sum_{m,n} p_m^2}{\frac{1}{N} \sum_m p_m^2} \right) \\
&= \log \left(1 + \frac{\sum_{m,n} s_{mn}^2 - \frac{2}{N} \sum_{m,n} s_{mn} \cdot p_m + \frac{1}{N^2} \sum_{m,n} p_m^2}{\frac{1}{N} \sum_m p_m^2} \right) \\
&= \log \left(1 + \frac{\sum_{m,n} (s_{mn} - \frac{1}{N} p_m)^2}{\frac{1}{N} \sum_m p_m^2} \right)
\end{aligned}$$

Da das Argument des Logarithmus immer größer als Eins ist, kann $I_2(\xi, \eta)$ nicht negativ werden und die Beschränktheit nach unten ist gegeben.

Nur wenn alle $s_{mn} = \frac{1}{N} \cdot p_m$ für $m \in \{1, 2, \dots, M\}$ sind, gilt $I_2(\xi, \eta) = 0$. Unter dieser Bedingung sind aber auch ξ und η unabhängig und die Unabhängigkeit ist gezeigt.

Sei für alle $m \in \{1, 2, \dots, M\}$ die Relation $S_{mn^*} = p_m$ für genau ein $n^*(m) \in \{1, 2, \dots, N\}$ und sonst $s_{mn} = 0$ dann gilt:

$$\sum_m p_m^2 = \sum_m (\sum_n s_{mn})^2 \geq \sum_{m,n} s_{mn}^2$$

Folglich ist $H_2(\xi, \eta) \geq H_2(\xi)$ und nach Gleichung 2.64 ist $I_2(\xi, \eta) \leq H_2(\eta)$. Unter den gleichen Bedingungen wie oben, jedoch für q_n gilt aber auch:

$$\sum_n q_n^2 = \sum_n (\sum_m s_{mn})^2 \geq \sum_{m,n} s_{mn}^2$$

In diesem Fall folgt $H_2(\xi, \eta) \geq H_2(\eta)$ und $I_2(\xi, \eta) \leq H_2(\xi)$

Damit ist die Beschränktheit nach oben, sowie die Determiniertheit gezeigt.

[qed]

Definition 2.4.1 (Verallgemeinerte Transinformation)

Unter der Bedingung, daß η gleichverteilt ist, heißt das Informationsmaß aus den Wahrscheinlichkeitsverteilungen $\{p_m\}$ und $\{q_n\}$ der Zufallsvariablen ξ und η und ihrer Verbundwahrscheinlichkeit $\{s_{mn}\}$

$$I_2(\xi, \eta) = \log \frac{\sum_{m,n=1}^{M,N} s_{mn}}{\sum_{m=1}^M p_m \cdot \sum_{n=1}^N q_n}$$

verallgemeinerte Transinformation der Zufallsvariablen ξ und η .

Eigenschaft zwischen ξ und η	Korrelationsmaße			
	I_0	I_1	I_2	C^2
η beliebig verteilt:				
$Korrelationsma\beta \geq 0$	+	+	-	+
$Korrelationsma\beta = 0 \Rightarrow$ Unabhängigkeit	-	+	-	-
$Korrelationsma\beta = 0 \Rightarrow$ Unkorreliertheit	-	+	-	+
Unabhängigkeit \Rightarrow $Korrelationsma\beta = 0$	-	+	+	+
Unkorreliertheit \Rightarrow $Korrelationsma\beta = 0$	-	-	-	+
η gleichverteilt:				
$Korrelationsma\beta \geq 0$	+	+	+	+
$Korrelationsma\beta = 0 \Rightarrow$ Unabhängigkeit	-	+	+	-
$Korrelationsma\beta = 0 \Rightarrow$ Unkorreliertheit	-	+	+	+
Unabhängigkeit \Rightarrow $Korrelationsma\beta = 0$	+	+	+	+
Unkorreliertheit \Rightarrow $Korrelationsma\beta = 0$	-	-	-	+

Tabelle 2.3: Eigenschaften unterschiedlicher Korrelationsmaße.

Durch (+) und (-) gekennzeichnetes Vorhandensein von Eigenschaften unterschiedlicher Korrelationsmaße, wobei Rényi-Entropien die I_α ergeben und C^2 quadrierte Korrelationskoeffizienten verkörpert (Quelle: [Pom97]).

Der Vergleich in Tabelle 2.3 vermittelt die Auswirkung einer Gleichverteilung der Zufallsvariablen η . Somit ist das Informationsmaß $I_2(\xi, \eta)$ in gleicher Weise als Maß statistischer Abhängigkeiten geeignet, wie die von SHANNON eingeführte Transinformation $I_1(\xi, \eta)$.

2.4.3 Gleichverteilung von Zufallsvariablen

Die Restriktion auf Gleichverteilung einer Zufallsvariablen erscheint recht hart, insbesondere für Meßreihen, die als Zufallsvariable interpretiert werden sollen.

Definition 2.4.2 (Gleichverteilung)

Sei $\{q_n\}_{n=1}^N$ die normierte¹² Wahrscheinlichkeitsverteilung der Zufallsgröße ξ und gilt

$$q_n = \frac{1}{N}, \text{ mit } n = 1, 2, \dots, N \text{ und somit } \sum_{n=1}^N q_n^2 = N \cdot \frac{1}{N^2} = \frac{1}{N}, \quad (2.67)$$

dann ist die Wahrscheinlichkeitsverteilung q_n gleichverteilt.

Aufbauend auf folgender Definition kann jede beliebig verteilte, endliche Zeitreihe derart transformiert werden, daß ihre Verteilung der Definition 2.4.2 zur Gleichverteilung entspricht.

Definition 2.4.3 (Ranking)

Sei $x(t)$ die Zeitreihe einer Messung zu den Zeitpunkten $t = 0, 1, \dots, T$, dann beschreibt Ranking¹³ die Transformation:

$$x(t) \rightarrow R(t) = \#\{t^* : \text{wobei } x(t^*) \leq x(t) ; \text{ mit } t^* = 0, 1, \dots, T\} \quad (2.68)$$

Die Transformation ist nur genau dann eindeutig umkehrbar, wenn alle Werte der Zeitreihe $x(t)$ paarweise verschieden¹⁴ sind. Dann gilt für die Rangfolge (engl. rank):

$$R(t) \text{ ist eine Permutation von } \{0, 1, \dots, T\}$$

Somit ist die relative Rangfolge $r(t) \equiv \frac{R(t)}{T}$ auf dem Intervall $[0, 1]$ gleichverteilt und die Funktion

$$f : x(t) \rightarrow r(t)$$

ist die empirische, eindimensionale Verteilungsfunktion der Zeitreihe $x(t)$. Im Allgemeinen ist diese Funktion nicht linear.

¹²Für normierte Wahrscheinlichkeitsverteilungen gilt: $\sum_{n=1}^N q_n = 1$ mit $q_n \geq 0$

¹³# bezeichnet die Kardinalität, also die Anzahl der t^* , für die Bedingung $x(t^*) \leq x(t)$ zutrifft.

¹⁴Was vorerst auch gefordert werden kann (siehe Bemerkung 2.4.4).

Eine algorithmische Beschreibung des in Definition 2.4.3 eingeführten *Rankings* kann dem Anhang B entnommen werden.

Der Algorithmus B.0.1 liefert auch für Zeitreihen, deren Werte nicht alle paarweise verschieden sind, gleichverteilte Rangfolgen, was durch folgenden Beweis dargelegt wird.

Beweis 2.4.4 (Rangfolgen paarweise identischer Zeitreihen)

Sei $x(t)$ eine beliebige Zeitreihe mit $x(t) \in \{0, 1, \dots, K\}$ und $t = 0, 1, \dots, T$, dann bezeichne $l(k) = \#\{x(t) = k\}$ mit $k = 1, 2, \dots, K$ die Anzahl der Punkte der Zeitreihe, die identisch sind. Das Erzeugen von Rangzahlen nach Gleichung 2.68 führt dann zu einer Rangfolge:

$$R = \{l(0), l(0) + l(1), \dots, \sum_{k=0}^K l(k)\}; \text{ mit der Verteilung } \frac{l(k)}{T}$$

In der Regel ist dies jedoch nicht die gewünschte Gleichverteilung auf $\{0, 1, \dots, T\}$, welche nur erreicht wird, falls $l(k) = 0$ bzw. $l(k) = 1$ für jedes $k = 0, 1, \dots, K$ ist.

Dieses Problem kann durch folgende Transformation behoben werden. Es seien $x(t_1), x(t_2), \dots, x(t_{l(k)})$ die Punkte der Zeitreihe $x(t)$, die identisch zu k sind. Durch eine Abbildung

$$\begin{aligned} g : x(t_u) &\rightarrow k + a_u \equiv y(t_u) ; \text{ mit} \\ 0 \leq a_1 &< a_2 < \dots < a_{l(k)} < 1 \text{ und } u = \{1, 2, \dots, l(k)\} \end{aligned} \quad (2.69)$$

sind alle Werte von $y(t)$ mit $t = 0, 1, \dots, T$ paarweise verschieden. Durch die Abbildungsfolge

$$h : x(t) \xrightarrow{g} y(t) \xrightarrow{f} R(t) \quad (2.70)$$

ist die relative Rangfolge $\frac{R(t)}{T}$ auf dem Intervall $[0, 1]$ gleichverteilt. Für die Transformation 2.69 kann jede Permutation von $\{a_u\}_{u=1}^{l(k)}$ verwendet werden. Durch die eindeutige Abbildung auf $\{0, 1, \dots, T\}$ des Algorithmus B.0.1 (siehe Anhang B) erfolgt die Auswahl einer solchen Permutation zusätzlich. Unterschiedliche Permutationen erzeugen lediglich verschiedene Rangfolgen, was jedoch keinen Einfluß auf die gewünschte Gleichverteilung hat.

[qed]

Mit dem Algorithmus B.0.1 können beliebig verteilte, endliche Zeitreihen in gleichverteilte Rangfolgen transformiert werden. Dabei bleibt die Charakteristik der Zeitreihe weitgehend unverändert, was Beispiel 2.4.1 belegen soll.

Beispiel 2.4.1 (Ranking)

Abbildung 2.9 (a) zeigt Druckschwingungen realer Meßdaten des Common-Rail-Systems (siehe Kapitel 5, Abschnitt 5.1.1). Durch die Transformation des Algorithmus B.0.1 kann die Zeitreihe als relative Rangfolge (b) dargestellt werden.

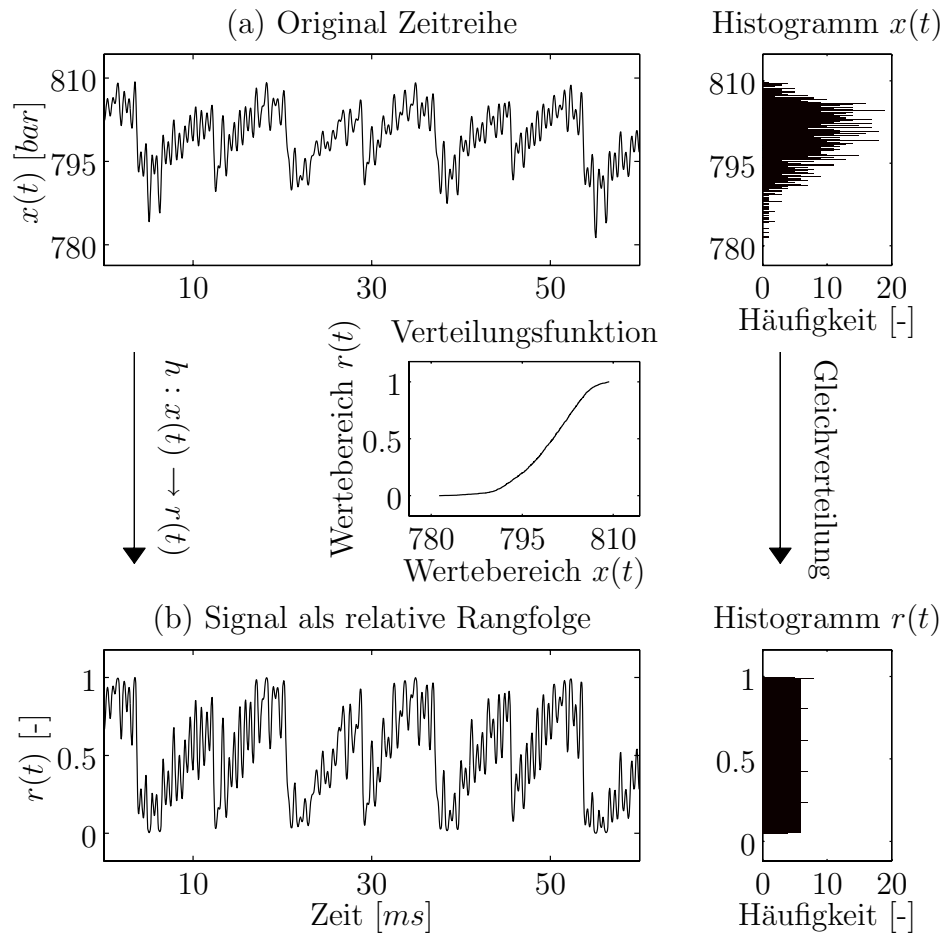


Abbildung 2.9: Gleichverteilung durch Ranking.

Die Zeitreihe $x(t)$ kann durch die gleichverteilte Rangfolge $r(t)$ dargestellt werden. Die Abbildung des Rankings entspricht dabei der Verteilungsfunktion $h: x(t) \rightarrow r(t)$. Im Vergleich der Histogramme ist $r(t)$ im Gegensatz zu $x(t)$ gleichverteilt.

Dabei bleibt die Charakteristik des Signals im wesentlichen bestehen. Die Histogramme belegen die Transformation der Zeitreihe $x(t)$ in die gleichverteilte Rangfolge $r(t)$. Dabei ist eine wichtige Eigenschaft der Gleichverteilung, daß sie keinen Einfluß auf das Informationsmaß $I_2(\xi, \eta)$ hat. Das bedeutet $I_2(\xi, \eta)$ ist Eins, sowohl für den Fall der Identität von ξ und η , als auch für eine Berechnung zwischen ξ und der Rangfolge von ξ als η .

Abbildung 2.10 verdeutlicht, wie sehr sich die Frequenzspektren von Zeitreihe und Rangfolge ähneln. Diese Eigenschaft ist nicht immer gegeben, da Rangfolgen nicht zwingend bandbegrenzt sind.

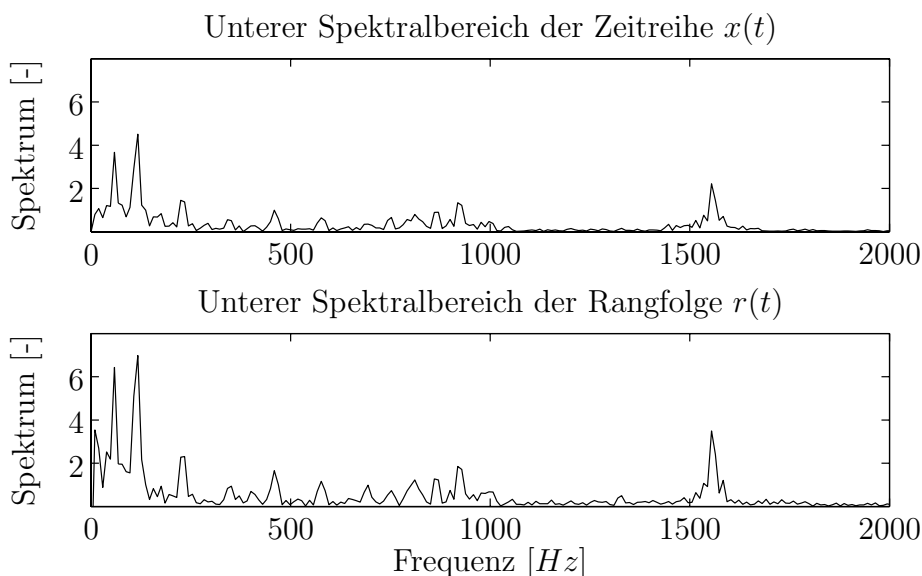


Abbildung 2.10: Spektralkonsistenz des Rankings.

Wie zu erkennen ist, entstehen durch die Transformation des Rankings keine Verschiebungen im Frequenzbereich. Jedoch hat das Ranking einen Einfluß auf die Amplituden, wobei charakteristische Spitzen von $x(t)$ in der Rangfolge $r(t)$ verstärkt zu erkennen sind.

So ist z.B. die Rangfolge der Sinus-Funktion eine Sägezahn-Funktion, die keine Bandbegrenzung hat. Für „verrauschte“ Meßdaten treten diese Probleme selten auf, da die Ableitung der Verteilungsfunktion $\frac{\delta F(x)}{\delta x}$ bei ausreichender Abtastung immer beschränkt bleibt (siehe auch [Pom97] A.3 145 ff.).

Mit der Möglichkeit, aus beliebigen, endlichen Zeitreihen gleichverteilte Rangfolgen zu bilden, ist nun die Basis zur Schätzung der verallgemeinerten Transinformation mit Hilfe des Informationsmaßes H_2 geschaffen.

2.4.4 Schätzverfahren zur Transformation

Die Entwicklung des Schätzverfahrens erfolgt im Hinblick auf eine spätere Anwendung mit mehrdimensionalen Ein/Ausgabe-Vektorpaaren, die häufig als Trainingsvektoren von neuronalen Netzen zum Einsatz kommen.

Grundlagen:

Zur Beschreibung des Schätzverfahrens betrachten wir somit mehrdimensionale Zufallsvektoren

$$\Xi \equiv \{\xi_1, \xi_2, \dots, \xi_{d_1}\},$$

wobei $d_1 \in \mathbb{N}$ die Dimension (die Anzahl der Netzein- bzw. ausgänge) des Zufallsvektors angibt. Dann lassen sich auf einem Vergrößerungsniveau $\epsilon \in \{\epsilon_1, \epsilon_2, \dots, \epsilon_{d_1}\}$ abhängig von der Dimension d_1 Wahrscheinlichkeitsverteilungen

$$\mathbf{P}_m \equiv \{p_{m_1}, p_{m_2}, \dots, p_{m_{d_1}}\}, \text{ mit } \epsilon_1 < \epsilon_2 < \dots < \epsilon_{d_1}$$

angeben. Somit wird mit zunehmender Dimension ein größeres Vergrößerungsniveau gewählt. Auf eine mögliche Wahl von ϵ wird durch Bemerkung 2.4.1 eingegangen.

Interpretieren wir Zeitreihen $\mathbf{X}(t) = \{x_1(t), x_2(t), \dots, x_{d_1}(t)\}_{t=1}^T$ als Zufallsvariablen, so lassen sich die Wahrscheinlichkeitsverteilungen durch das sogenannte Korrelationsintegral

$$\mathbf{P}_m(t) \approx c_{\xi, T}(t) \equiv \frac{\#\{t^* : \text{wobei } \|\mathbf{X}(t) - \mathbf{X}(t^*)\| < \frac{\epsilon}{2} ; \text{ mit } t^* = 1, 2, \dots, T\}}{T}$$

abschätzen, vorausgesetzt das verwendete ϵ ist klein, d.h. in Anbetracht weiterer Dimensionen gilt $\max\{\epsilon_1, \epsilon_2, \dots, \epsilon_{d_1}\} \rightarrow 0$ und die Zeitreihen sind lang genug, also $T \rightarrow \infty$.

Setzen wir diesen naiven Dichteschätzer in Gleichung 2.64 der Rényi-Entropie ein, erhalten wir in Abhängigkeit von $\alpha \geq 0$ die Abschätzung:

$$H_\alpha(\xi) \approx \begin{cases} \frac{1}{1-\alpha} \log \frac{1}{T} \sum_{t=1}^T c_{\xi, T}^{\alpha-1}(t) & ; \text{ falls } \alpha \geq 0, \alpha \neq 1 \\ -\frac{1}{T} \sum_{t=1}^T \log c_{\xi, T}(t) & ; \text{ falls } \alpha = 1 \end{cases} \quad (2.71)$$

Für den Fall $\alpha = 2$ vereinfacht sich Gleichung 2.71 so, daß nur noch der negative *Logarithmus* der Summe

$$C_{\xi, T} = \frac{1}{T} \sum_{t=1}^T c_{\xi, T}(t) \quad (2.72)$$

zu berechnen ist. Sie gibt die relative Anzahl von Punktepaaren der Zeitreihenschar $\mathbf{X}(t)$ an, deren Abstand, abhängig von ihrer Dimension, kleiner als $\epsilon \in \{\epsilon_1, \epsilon_2, \dots, \epsilon_{d_1}\}$ ist.

Betrachten wir nun weitere Zeitreihen $\mathbf{Y}(t) = \{y_1(t), y_2(t), \dots, y_{d_2}(t)\}_{t=0}^T$ als Realisierung der Zufallsvariablen η , so kann die verallgemeinerte Transinformation gemäß Definition 2.4.1 durch

$$I_2(\xi, \eta) \approx \log_2 \frac{C_{\xi \wedge \eta, T}}{C_{\xi, T} \cdot C_{\eta, T}} \quad (2.73)$$

abgeschätzt werden. Zur Bestimmung des Verbundkorrelationsintegrals $C_{\xi \wedge \eta, T}$ wird zunächst der $D = d_1 + d_2$ dimensionale Verbundvektor

$$\mathbf{Z}(t) = \{x_1(t), x_2(t), \dots, x_{d_1}(t), y_1(t), y_2(t), \dots, y_{d_2}(t)\}_{t=1}^T$$

gebildet. Somit läßt sich eine Approximation der Verbundwahrscheinlichkeit $\{\mathbf{S}_{mn}\}$ der Zeitreihen $\mathbf{X}(t)$ und $\mathbf{Y}(t)$ durch

$$\mathbf{S}_{mn}(t) \approx c_{\xi \wedge \eta, T}(t) \equiv \frac{\#\{t^* : \text{wobei } \|\mathbf{Z}(t) - \mathbf{Z}(t^*)\| < \frac{\epsilon}{2} ; \text{ mit } t^* = 1, 2, \dots, T\}}{T}$$

bestimmen. Dabei ist $\epsilon \in \{\epsilon_1, \epsilon_2, \dots, \epsilon_D\}$ mit $D = d_1 + d_2$ die Summe der Dimension beider Zeitreihen.

Definition 2.4.1 verlangt, daß wenigstens eine Zufallsvariable gleichverteilt ist. Deshalb wird bei der folgenden Betrachtung stets von Rangfolgen ausgegangen. Sind beide Zufallsvariablen ξ und η durch absolute Rangfolgen $R_\xi(t)$ und $R_\eta(t)$ repräsentiert, so ergeben sich zusätzliche, numerische Vorteile für das folgende Matrixkalkül. Dabei ist zu beachten, daß folgende Abstandstests äquivalent sind:

$$|r_d(t) - r_d(t^*)| < \frac{\epsilon_d}{2} \text{ und } |R_d(t) - R_d(t^*)| < \mathcal{E}_d, \\ \text{mit } \mathcal{E}_d = \frac{\epsilon_d T}{2}, \text{ für } d = 1, 2, \dots, D$$

Weiterhin sind die ϵ_d für alle $d = 1, 2, \dots, D$ so klein zu wählen, das immer $0 < \mathcal{E}_d \ll T$ gilt.

Matrizenverfahren:

Nach Gleichung 2.73 besteht das algorithmische Problem zur Schätzung der *verallgemeinerten Transinformation* in einer möglichst effektiven Berechnung der Summe von Abständen.

Gegeben sei die Realisierung einer Zufallsvariablen ξ als mehrdimensionale Rangfolge $\mathbf{R}_\xi(t) = \{R_{\xi_1}(t), R_{\xi_2}(t), \dots, R_{\xi_{d_1}}(t)\}_{t=1}^T$. Zur direkten Bestimmung des Korrelationsintegrals 2.72 sind für alle T^2 Paare $\{\mathbf{R}_\xi(t), \mathbf{R}_\xi(t^*)\}_{t, t^*=1}^T$ die Abstände Koordinatenweise

zu bewerten. Dazu werden zunächst für jede Dimension $d = 1, 2, \dots, d_1$ folgende *Rang-Abstands-Matrizen* gebildet:

$$\Delta_{\xi_d} = \begin{pmatrix} \delta_{d,11} & \cdots & \delta_{d,1t^*} & \cdots & \delta_{d,1T} \\ \vdots & \ddots & \vdots & & \vdots \\ \delta_{d,t1} & \cdots & \delta_{d,tt^*} & \cdots & \delta_{d,tT} \\ \vdots & & \vdots & \ddots & \vdots \\ \delta_{d,T1} & \cdots & \delta_{d,Tt^*} & \cdots & \delta_{d,TT} \end{pmatrix}, \text{ mit } \delta_{d,tt^*} \equiv |R_{\xi_d}(t) - R_{\xi_d}(t^*)|$$

In Abhängigkeit vom Abstandsparameter $0 < \mathcal{E}_d \ll T$ ergeben sich folgende *binäre Rang-Abstands-Matrizen*:

$$\mathcal{B}_{\xi_d} = \begin{pmatrix} \beta_{d,11} & \cdots & \beta_{d,1t^*} & \cdots & \beta_{d,1T} \\ \vdots & \ddots & \vdots & & \vdots \\ \beta_{d,t1} & \cdots & \beta_{d,tt^*} & \cdots & \beta_{d,tT} \\ \vdots & & \vdots & \ddots & \vdots \\ \beta_{d,T1} & \cdots & \beta_{d,Tt^*} & \cdots & \beta_{d,TT} \end{pmatrix}, \text{ mit } \beta_{d,tt^*} = \begin{cases} 1 & : \delta_{d,tt^*} < \mathcal{E}_d \\ 0 & : \text{sonst} \end{cases}$$

Alle Matrizen sind symmetrisch. Das bedeutet für die Binärmatrizen $\beta_{d,t^*} = \beta_{d,t^*t}$, wobei die Hauptdiagonalen mit $\beta_{d,tt} = 1$ für alle $t = 1, 2, \dots, T$ besetzt sind. Somit genügt die Berechnung des oberen Dreiecks der Matrizen. Für die jeweiligen *Gewichte* gilt:

$$\sum_{t < t^*} \beta_{d,tt^*} = \frac{\sum_{t,t^*} \beta_{d,tt^*} - T}{2} \quad (2.74)$$

Um das Gewicht über alle Matrizen einer mehrdimensionalen Rangfolge zu bilden, eignen sich die folgenden UND-Konjunktionen:

$$\begin{aligned} \mathcal{B}_{\xi_1} \wedge \mathcal{B}_{\xi_2} &\equiv \{\beta_{1,tt^*} \wedge \beta_{2,tt^*}\}_{t < t^*} \\ \bigwedge_{d=1}^{d_1} \mathcal{B}_{\xi_d} &\equiv \mathcal{B}_{\xi_1}, \mathcal{B}_{\xi_2}, \dots, \mathcal{B}_{\xi_{d_1}} \end{aligned}$$

Die gewünschten Korrelationsintegrale der Gleichung 2.72 ergeben sich aus den relativen Gewichten der so verknüpften *binären Rang-Abstands-Matrizen* unter Berücksichtigung der Rangfolgenlänge T und der Beziehung 2.74:

$$\begin{aligned} C_{\xi,T} &= \frac{1}{T} + \frac{2}{T^2} \cdot \sum_{t < t^*} \bigwedge_{d=1}^{d_1} \mathcal{B}_{\xi_d} \\ C_{\eta,T} &= \frac{1}{T} + \frac{2}{T^2} \cdot \sum_{t < t^*} \bigwedge_{d=1}^{d_2} \mathcal{B}_{\eta_d} \\ C_{\xi \wedge \eta, T} &= \frac{1}{T} + \frac{2}{T^2} \cdot \sum_{t < t^*} \bigwedge_{d=1}^D \mathcal{B}_{\xi_d \wedge \eta_d} \end{aligned}$$

Diese Schätzung der Korrelationsintegrale und der zugehörigen Entropien $H(\xi), H(\eta)$ und $H(\xi, \eta)$ sind durch den Algorithmus B.0.2 im Anhang B umgesetzt.

Durch eine Normierung der verallgemeinerten Transinformation auf $\min\{H_2(\xi), H_2(\eta)\}$ liegen die Werte immer im Bereich $[0, 1]$. Dies ist möglich, da laut Satz 2.4.2 (Beschränktheit) $0 \leq I_2(\xi, \eta) \leq \min\{H_2(\xi), H_2(\eta)\}$ gilt. Durch die Verwendung der Entropien erhalten wir die folgende, einfache Gleichung zur Berechnung der normierten, verallgemeinerten Transinformation:

$$\hat{I}_2(\xi, \eta) \approx \frac{H_2(\xi) + H_2(\eta) - H_2(\xi, \eta)}{\min\{H_2(\xi), H_2(\eta)\}} \quad (2.75)$$

Bemerkung 2.4.1 (Bestimmung von Epsilon)

Die Bestimmung eines geeigneten Wertes für \mathcal{E}_d kann iterativ erfolgen. Um ein von der Dimension unabhängiges Maß zu erhalten, sollte der Wert des Korrelationsintegrals $C_{\xi \wedge \eta, T}$ unabhängig von den gewählten Dimensionen d_1 und d_2 einen kleinen, von Null verschiedenen, konstanten Wert annehmen. Dies ist besonders wichtig, da in unseren Anwendungen oft Transinformationswerte von Zeitreihen unterschiedlicher Dimension verglichen werden sollen.

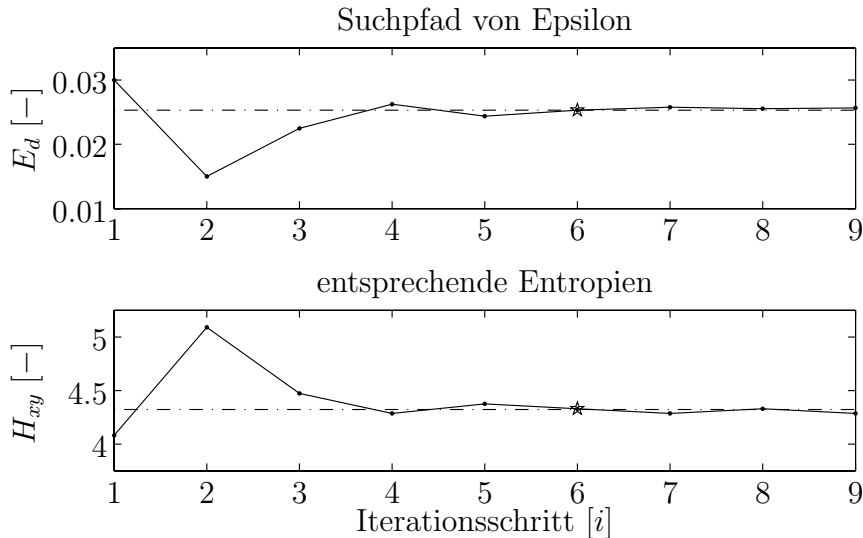


Abbildung 2.11: Iterative Bestimmung von Epsilon.

Um die gemeinsame Entropie $H_{\xi, \eta}$ mit Hilfe von \mathcal{E}_d auf eine Werte von $-\log_2(0.05) \approx 4,3$ einzustellen kann eine iterative Suche erfolgen.

In der Praxis haben sich für $C_{\xi \wedge \eta, T}$ prozentuale Angaben bewährt. Abbildung 2.11 veranschaulicht die iterative Suche nach einem \mathcal{E}_d , so daß $H_2(\xi, \eta) = -\log_2(0.05) \approx 4,3$ ist. Die Suche wurde nach 9 Iterationsschritten abgebrochen. Der beste Wert für \mathcal{E}_d ergab sich nach 6 Iterationen. Somit beträgt $C_{\xi \wedge \eta, T} = 0.05$ und die Binärmatrix $\mathcal{B}_{\xi_d \wedge \eta_d}$ ist zu 5 % besetzt. Eine entsprechende Binärmatrix der Daten aus Beispiel 2.4.1 ist in Abbildung 2.12 dargestellt.

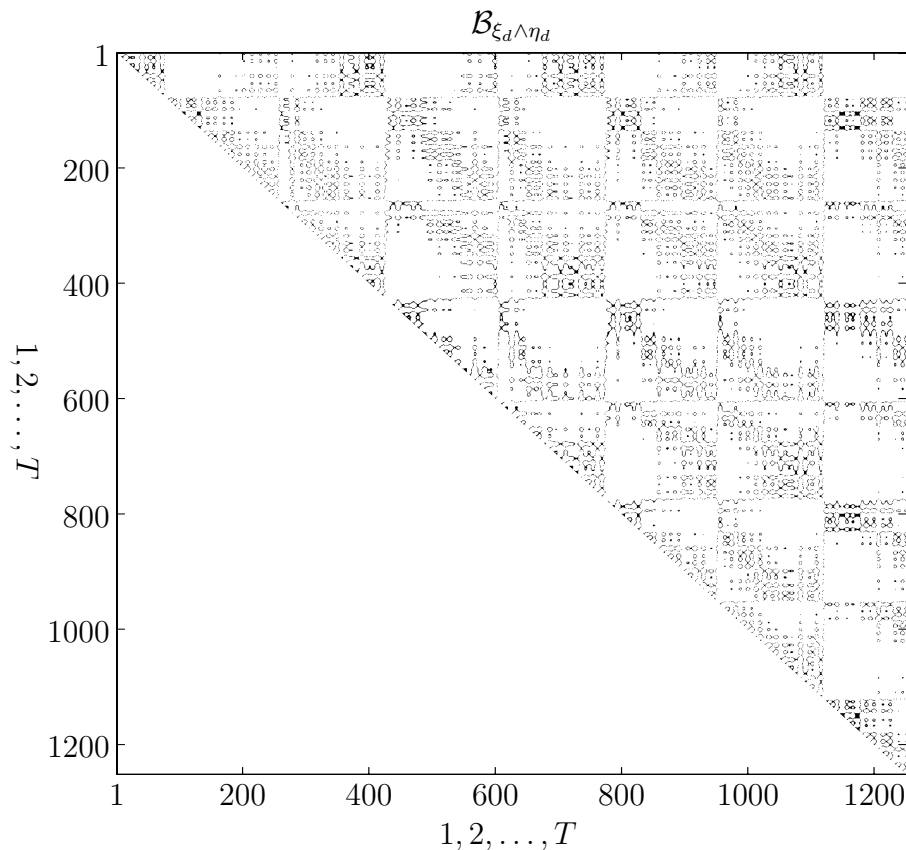


Abbildung 2.12: Binärmatrix zur Schätzung der Transformation.

Für das in Abbildung 2.11 ermittelte \mathcal{E}_d ist die Binärmatrix zu 5% gefüllt. Somit beträgt $C_{\xi \wedge \eta, T} = 0.05$ und $H_2(\xi, \eta) = -\log_2(0.05)$.

2.4.5 Bewertung des Schätzverfahrens

Abschließend soll die verallgemeinerte Transformation als Maß für statistische Abhängigkeit mit einer bekannten Methode, der linearen Korrelation verglichen werden.

Bewertung statistischer Zusammenhänge:

Für einen ersten Vergleich dienen zunächst zwei konstruierte Datenmengen, die entsprechend Abbildung 2.13 folgende statistische Abhängigkeiten aufweisen:

- (a) 2000 zufällig auf 5 Kreisflächen verteilte Datenpunkte. Somit erfüllen sie die Bedingung $r^2 = x^2 + y^2$. Die Zentren dieser Kreisflächen sind wiederum gleichmäßig auf einem Kreis verteilt.
- (b) 2000 zufällig um das Polynom $y = -0.7 \cdot x^3 + 0.7 \cdot x^2 - 0.1 \cdot x$ angeordnete Datenpunkte.

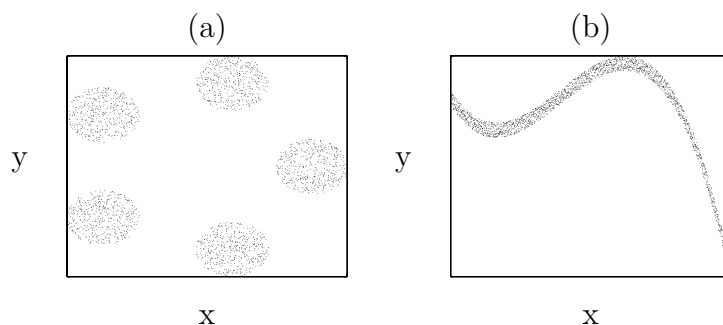


Abbildung 2.13: Komplexe Zufallsverteilungen zur Bewertung.
Zufällige Verteilung von 2000 Datenpunkten zur statistischen Bewertung mittels Transinformation im Vergleich zur Korrelation:

- (a) Um ein Zentrum angeordnete Kreisflächen $r^2 = x^2 + y^2$.
- (b) Um das Polynom $y = -0.7 \cdot x^3 + 0.7 \cdot x^2 - 0.1 \cdot x$ gestreute Punkte.

Tabelle 2.4 verdeutlicht, wie sich die Varianzen, die Kovarianz und damit die Korrelation im Vergleich zur verallgemeinerten Transinformation verhalten.

Statistische Maße		(a)	(b)
Varianz	s_x^2	0.0880	0.0845
Varianz	s_y^2	0.0821	0.0383
Kovarianz	s_{xy}	0.0002	0.0138
Korrelation	$\frac{s_{xy}}{s_x \cdot s_y}$	0.0018	0.2435
Transinformation	$I(x, y)$	0.3519	0.6413

Tabelle 2.4: Vergleich statistischer Maße.

Statistische Maße zur Beurteilung der Datenmengen (a) und (b) aus Abbildung 2.13.

Im Datensatz (a) kann nur schwer ein linearer Zusammenhang zwischen x und y gefunden werden, was sich auch im statistischen Maß der Korrelation widerspiegelt. Der Wert der Transinformation liegt ungleich höher, was auf einen statistischen Zusammenhang zwischen x und y schließen läßt.

Der Datensatz (b) läßt erkennen, daß zwar im Vergleich zum Datensatz (a) stärkere, lineare Abhängigkeiten bestehen, jedoch ist wiederum das Ergebnis der Transinformation deutlicher.

Aus einem Vergleich zwischen Anordnung der Datenmengen und entsprechenden Zahlenwerten ist ebenfalls ersichtlich, daß die Transformation höhere Werte erzielt, je kompakter die Datenpunkte beieinander liegen. Dabei können sie sich auch auf mehrere Kluster aufteilen.

Beispiel 2.4.2 (Anwendungsbeispiel:)

Im abschließenden Beispiel kommen beide Methoden an realen Meßdaten zur Anwendung. Bei den Meßdaten handelt es sich um Druckschwingungen des Einspritzsystems aus Kapitel 5, Abschnitt 5.1.1. Die Schwingungen werden durch Einspritzungen ange-regt. Nun soll untersucht werden, welche Positionen des Drucksignals eine besonders hohe statistische Abhängigkeit von den jeweiligen Einspritzmengen haben (siehe auch Abschnitt 5.3.4).

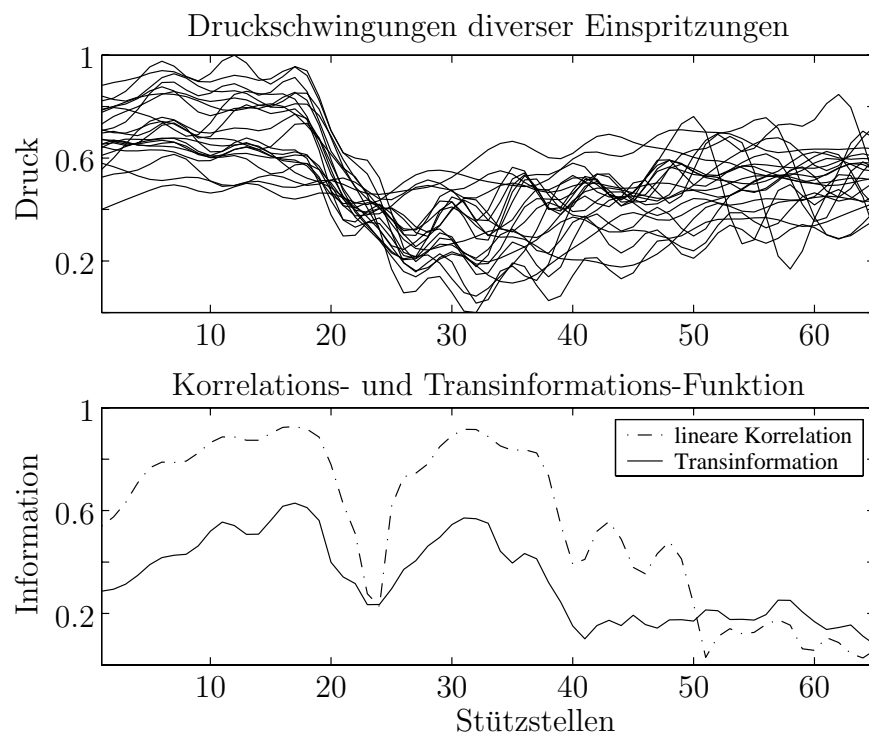


Abbildung 2.14: Vergleich Korrelations- zu Transformations-Funktion.

Darstellung der statistischen Abhängigkeit von Stützstellen realen Rail-druckschwingungen zu den zugehörigen Einspritzmengen.

In Abbildung 2.14 ist gut zu erkennen, daß sich die Verläufe der Korrelations- bzw. Transformations-Funktion ähneln. Offensichtlich stehen einige Positionen in einem starken linearen Zusammenhang mit den Einspritzmengen, da ihre Korrelationskoeffizienten recht hohe Werte erzielen. Hingegen fallen die Zahlenwerte der Transformation geringer aus, was auf eine schwache Klusterung schließen läßt. Dafür sind die relevanten Positionen schärfer umgrenzt.

2.5 Zusammenfassung

Das Kapitel beginnt mit einer kurzen Einführung zu künstlichen neuronalen Netzen mit ihrem mathematischen Modell und diversen Lernalgorithmen. Aufbauend auf dem beschriebenen Modell neuronaler Netze sind neue, automatische Verfahren zur Datenverarbeitung und zum Konstruieren solcher Strukturen entstanden, die im Kapitel 4 vorgestellt werden.

Zum besseren Verständnis der Theorie zur verallgemeinerten Transinformation aus Abschnitt 2.4 wurden einige Grundlagen der Statistik und Wahrscheinlichkeitsrechnung dargestellt. Sie dienen auch als Grundlage zur Bewertung künstlicher, neuronaler Netze.

Für die in Kapitel 4 neu vorgestellte Möglichkeit zur informationstheoretischen Bewertung von Spektren der Fourier-Transformation sind einige Grundlagen der Signalanalyse von Nutzen. Die neue Bewertung ist das Ergebnis einer Kombination aus Fourier-Transformation, Oktavbandzerlegung und verallgemeinerten Transinformation.

Abschließend ist ein Verfahren zur Schätzung der verallgemeinerten Transinformation vorgestellt worden. Dieses Verfahren eignet sich zur Ermittlung statistischer Abhängigkeiten zwischen mehrdimensionalen Zufallsvariablen. Dabei werden Rényi-Entropien der 2. Ordnung verwendet, was die Gleichverteilung wenigstens einer Zufallsvariablen erfordert. Diese Voraussetzung kann durch eine Darstellung als Rangfolge geschaffen werden. Die verallgemeinerte Transinformation kann im Gegensatz zur Berechnung von Korrelationskoeffizienten auch nichtlineare Zusammenhänge erkennen. Werden Meßreihen realer Systeme als Zufallsvariablen interpretiert, so kann das Verfahren zum Schätzen statistischer Zusammenhänge verschiedener Systemeinzugänge herangezogen werden.

Aufbauend auf der verallgemeinerten Transinformation konnten neue Methoden zur Optimierung von Netztopologien entwickelt werden. Eine Beschreibung dieser Methoden folgt in Kapitel 4, sowie deren Anwendung in Kapitel 5.

Kapitel 3

Stand der Technik

Die mit der vorliegenden Arbeit vorgestellten Methoden befassen sich mit der Analyse von Meßdaten, der Auswahl relevanter Daten bzw. der Optimierung von Topologien neuronaler Netze. Sie bewegen sich also auf dem Grenzgebiet der Datenanalyse, Vorverarbeitung und Präsentation von Daten zum Training neuronaler Netzstrukturen, sowie entsprechender Lernverfahren. Somit ist der Stand der Technik verschiedener Fachgebiete zu betrachten.

3.1 Methoden zur Datenanalyse

Um beim Einsatz künstlicher, neuronaler Netze gute Ergebnisse zu erzielen, ist oft eine Vorverarbeitung der Daten erforderlich. Sinn der Vorverarbeitung ist die Reduzierung von Störgrößen. Um solche Störgrößen identifizieren zu können, ist eine ausgiebige Datenanalyse durchzuführen. Einige Methoden der Signalanalyse werden im Folgenden kurz beschrieben.

3.1.1 Regression

Bei der linearen Regression wird davon ausgegangen, daß sich der Zusammenhang zwischen unabhängigen Eingangswerten $\{x_i\}_{i=1}^n$ und den zugehörigen Ausgangswerten $\{y_i\}_{i=1}^n$ durch eine Gerade approximieren läßt. Dieser Zusammenhang läßt sich unter Berücksichtigung von Fehlertermen oder Residuen E_i darstellen durch:

$$y_i = a \cdot x_i + b + E_i \quad (3.1)$$

Die Parameter a und b lassen sich durch die Methode der kleinsten Quadrate bestimmen. Somit ist die Fehlerfunktion

$$f(a, b) = \sum_{i=1}^n (y_i - a \cdot x_i - b)^2 = \sum_{i=1}^n E_i^2 \quad (3.2)$$

zu minimieren. Unter der Annahme die Residuen E_i seien gleichverteilt¹ (siehe Abschnitt 2.3.1 Rauschen) kann Gleichung 3.1 zu

$$\bar{y} = a \cdot \bar{x} + b \quad (3.3)$$

umgeformt² werden und in Gleichung 3.2 läßt sich b ersetzen:

$$\begin{aligned} f(a) &= \sum_{i=1}^n (y_i - \bar{y} - a(x_i - \bar{x}))^2 \\ &= \sum_{i=1}^n (y_i - \bar{y})^2 - 2 \sum_{i=1}^n (y_i - \bar{y})(x_i - \bar{x}) \cdot a + a^2 \sum_{i=1}^n (x_i - \bar{x})^2 \end{aligned} \quad (3.4)$$

Durch Lösen der Differentialgleichung $\frac{\partial f(a)}{\partial a} = 0$ läßt sich ein Schätzwert für a berechnen.

$$\begin{aligned} \frac{\partial f(a)}{\partial a} &= -2 \sum_{i=1}^n (y_i - \bar{y})(x_i - \bar{x}) + 2a \sum_{i=1}^n (x_i - \bar{x})^2 = 0 \\ \Rightarrow a &= \frac{\sum_{i=1}^n (y_i - \bar{y})(x_i - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2} \end{aligned} \quad (3.5)$$

Somit läßt sich a aus der Kovarianz von $\{x_i, y_i\}$ und der Varianz von $\{x_i\}$ berechnen. Durch Einsetzen von a in Gleichung 3.3 ergeben sich die Koeffizienten:

$$a = \frac{s_{xy}}{s_x^2} \quad \text{und} \quad b = \bar{y} - \frac{s_{xy}}{s_x^2} \cdot \bar{x} \quad (3.6)$$

Häufig lassen sich Zusammenhänge nicht durch eine Regressionsgerade approximieren. In solchen Fällen kann Gleichung 3.1 mit einem Polynom oder einer Exponentialfunktion angesetzt werden:

$$y_i = a_p x_i^p + \dots + a_2 x_i^2 + a_1 x_i + b + E_i \quad (3.7)$$

$$y_i = a_1 \cdot e^{a_2 x_i} + b + E_i \quad (3.8)$$

Für einen erfolgreichen Einsatz solcher Regressionen ist es erforderlich, den Zusammenhang zwischen den Eingangswerten $\{x_i\}_{i=1}^n$ und den Ausgangswerten $\{y_i\}_{i=1}^n$ zumindest annähernd zu kennen. So muß z.B. zumindest der Grad des Polynoms bekannt sein, um eine gute Approximation des Zusammenhangs zu erzielen.

¹dann gilt, $\frac{1}{n} \sum_{i=1}^n E_i = 0$.

²der Mittelwert der y_i ist $\bar{y} = \frac{1}{n} \sum_{i=1}^n (a x_i + b + E_i) = a \frac{1}{n} \sum_{i=1}^n x_i + b + \frac{1}{n} \sum_{i=1}^n E_i$
und somit $\bar{y} = a \bar{x} + b + 0$.

3.1.2 Korrelationsanalyse

Der lineare Korrelationskoeffizient gibt Auskunft darüber, wie stark zwei stochastische Größen in einem linearen Zusammenhang stehen. Ist der Betrag des Koeffizienten gleich Eins, so läßt sich eine Größe direkt über eine lineare Gleichung aus der anderen bestimmen. Ist er gleich Null, so besteht kein linearer Zusammenhang zwischen den beiden Größen.

Aufbauend auf den Definitionen in Kapitel 2 Abschnitt 2.2.1 der Varianz und Kovarianz, läßt sich der lineare Korrelationskoeffizient entsprechend Gleichung 3.9 einer zweidimensionalen Menge $\{x_i, y_i\}_{i=1}^n$ bestimmen.

$$r_{xy} = \frac{s_{xy}}{s_x \cdot s_y} \quad (3.9)$$

Der lineare Korrelationskoeffizient hat folgende Eigenschaften:

1. Er ist durch $-1 \leq r_{xy} \leq 1$ begrenzt.
2. Aus $|r_{xy}| = 1$ folgt die Beziehung: $y_i = ax_i + b$. Dabei sind $a = r_{xy} \cdot \frac{s_y}{s_x} = \frac{s_{xy}}{s_x^2}$ und $b = \bar{y} - r_{xy} \bar{x} \cdot \frac{s_y}{s_x} = \bar{y} - \frac{s_{xy}}{s_x^2} \cdot \bar{x}$, die Koeffizienten der Regressionsgerade.
3. Aus $y_i = ax_i + b$ folgt, daß die Punkte $\{x_i, y_i\}$ auf einer Geraden mit positiver bzw. negativer Steigung liegen:

$$r_{xy} = \begin{cases} 1 & ; a > 0 \text{ positive Steigung} \\ -1 & ; a < 0 \text{ negative Steigung} \end{cases}$$

4. Der Korrelationskoeffizient ist wegen $s_y = 0$ für $a = 0$ nicht definiert.

Mit Hilfe solcher Korrelationskoeffizienten lassen sich z.B. n Eingangszeitreihen $X(t) = \{x_1(t), x_2(t), \dots, x_n(t)\}_{t=t_0}^T$ bezüglich ihrer n Ausgangsgrößen $Y = \{y_1, y_2, \dots, y_n\}$ auf Positionen mit großem linearen Zusammenhang untersuchen. Dies geschieht durch die fortschreitende Berechnung der linearen Korrelation zwischen den Vektoren $\{x_i(t_0)\}_{i=1}^n$ und $\{y_i\}_{i=1}^n$ für alle $t = \{t_0, \dots, T\}$.

Die Anwendung der linearen Korrelation nach der Methode des Beispiels 3.1.1 geht davon aus, daß in der Nähe linearer Abhängigkeiten auch weitere Abhängigkeiten vorhanden sind. Eine Analyse von Brennraumdruckdaten mit der in Kapitel 2 Abschnitt 2.4 dargestellten Methode der verallgemeinerten Transinformation zeigt, daß diese Annahme nicht optimale Ergebnisse liefert. Ein entsprechender Vergleich ist mit dem BMBF-Projekt [DMF01] entstanden (siehe auch Kapitel 5, Abschnitt 5.2).

Beispiel 3.1.1 (Brennraumdruckanalyse)

Gesucht sind die Positionen des Brennraumdrucks mit maximalem Zusammenhang zur Verschiebung des Druckmaximums in Bezug auf den Zündzeitpunkt. Abbildung 3.1 veranschaulicht das Ergebnis einer solchen Berechnung.

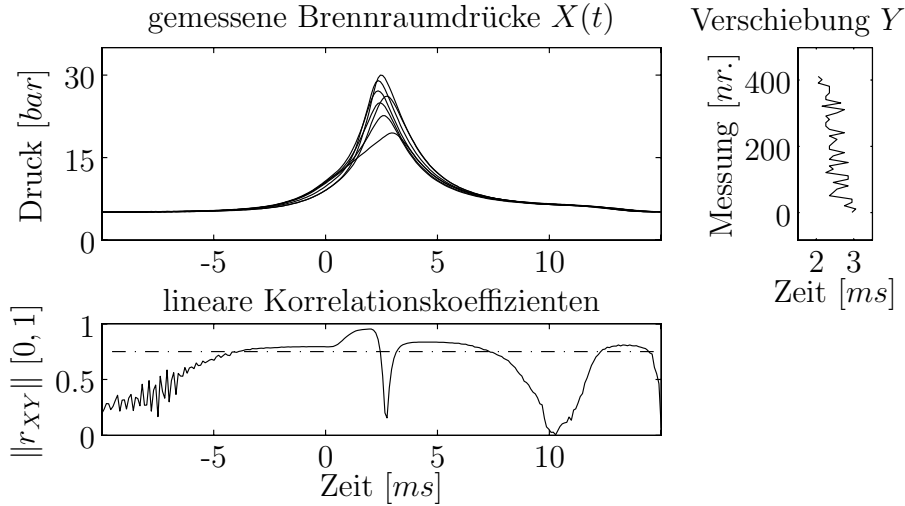


Abbildung 3.1: Auswahl relevanter Druckpositionen durch Korrelation.

Durch Berechnung der Korrelation zu jedem Zeitpunkt t können die Druckpositionen $X(t)$ mit großem linearem Zusammenhang zu Y bestimmt werden.

Wird ein Absolutbetrag der Korrelation von 0.75 als hoch bewertet (siehe gestrichelte Bewertungslinie), dann werden alle Positionen der Hochdruckphase der Verbrennung durch die Korrelation als relevant klassifiziert.

Ist der gemeinsame lineare Zusammenhang von mehreren Positionen $X(t_1, \dots, t_m)$ mit Y zu ermitteln, so besteht die Möglichkeit zur Berechnung einer mehrdimensionalen Korrelation. Dabei wird $\{x_i(t_1)\}_{i=1}^n$ abgekürzt als x_1 bezeichnet.

$$r_{(x_1, \dots, x_m)y} = \sqrt{R_{xy} \cdot \mathcal{R}_{xx}^{-1} \cdot R_{xy}^T} \quad (3.10)$$

mit

$$R_{xy} = (r_{x_1y}, \dots, r_{x_my}) \quad \text{und} \quad \mathcal{R}_{yy}^{-1} = \begin{pmatrix} 1 & r_{x_1x_2} & \cdots & r_{x_1x_m} \\ r_{x_2x_1} & 1 & \cdots & r_{x_2x_m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{x_nx_1} & r_{x_nx_2} & \cdots & r_{x_nx_m} \end{pmatrix}^{-1}$$

Die lineare Korrelation ist nicht geeignet, um nichtlineare Zusammenhänge zu identifizieren. In Kapitel 2 Abschnitt 2.4 wird mit der verallgemeinerten Transinformation ein derartiges Maß vorgestellt.

3.1.3 Hauptkomponentenanalyse

Die Hauptkomponentenanalyse besteht darin, eine orthogonale Transformation eines mehrdimensionalen Zufallsvektors $X = \{x_1, \dots, x_n\}$ in eine neue Menge unkorrelierter Vektoren, sogenannte Hauptkomponenten $Y = \{y_1, \dots, y_m\}$, zu bestimmen. Dabei sind die Hauptkomponenten Linearkombinationen der ursprünglichen Vektoren, die aufgrund der Orthogonalität untereinander linear unabhängig sind. Durch diese Unabhängigkeit besteht die Chance, daß weniger Vektoren mit $m < n$ ausreichen, um den größten Teil der Variation der Originalvektoren abzudecken.

Sei X die Beobachtungsmatrix einer Stichprobe mit der Kovarianzmatrix S deren Eigenwert-Eigenvektor-Paare $(\lambda_1, \mathbf{e}_1), (\lambda_2, \mathbf{e}_2), \dots, (\lambda_n, \mathbf{e}_n)$, entsprechend $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \geq 0$ sortiert sind, dann ist die i -te Hauptkomponente³ gegeben durch:

$$y_i = \mathbf{e}_i^T \mathbf{x} = e_{1i}x_1 + e_{2i}x_2 + \dots + e_{ni}x_n; \quad \text{mit } i = 1, 2, \dots, n \quad (3.11)$$

Es kann gezeigt werden [JW92], daß für die Varianz $\text{var}(y_i)$ und die Kovarianz $\text{cov}(y_i, y_k)$ folgendes gilt:

$$\text{var}(y_i) = \mathbf{e}_i^T S \mathbf{e}_i = \lambda_i; \quad \text{mit } i = 1, 2, \dots, n \quad (3.12)$$

$$\text{cov}(y_i, y_k) = \mathbf{e}_i^T S \mathbf{e}_k = 0; \quad \text{mit } i \neq k \quad (3.13)$$

Für die Gesamtvarianzen der Stichprobe gilt $\sum \text{var}(x) = \sum \text{var}(y) = \lambda_1 + \lambda_2 + \dots + \lambda_n$. Somit läßt sich der prozentuale Anteil der k -ten Hauptkomponente an der Gesamtvarianz bestimmen durch:

$$\text{Anteil}(k) = 100\% \cdot \frac{\lambda_k}{\sum_{i=1}^n \lambda_i} \quad (3.14)$$

Wenn der größte Teil (z.B. 80-90%) der Gesamtvarianz durch die ersten $m < n$ Hauptkomponenten abgedeckt wird, so können diese m Komponenten die n Ausgangsvektoren ohne großen Informationsverlust ersetzen.

Entsprechend [Wed99] läßt sich die quadratische Distanz zwischen \mathbf{x} und $\bar{\mathbf{x}}$ für standardisierte Vektoren mit $\bar{\mathbf{x}} = 0$ schreiben als:

$$c^2 = \mathbf{x}^T S^{-1} \mathbf{x} = \frac{1}{\lambda_1} (\mathbf{e}_1^T \mathbf{x})^2 + \frac{1}{\lambda_2} (\mathbf{e}_2^T \mathbf{x})^2 + \dots + \frac{1}{\lambda_n} (\mathbf{e}_n^T \mathbf{x})^2 \quad (3.15)$$

Unter Verwendung der Gleichung 3.11 beschreibt c^2 für $\lambda_i > 0$ ein Hyperellipsoid im Koordinatensystem mit y_i als Achsen, die in den Richtungen \mathbf{e}_i liegen:

$$c^2 = \frac{1}{\lambda_1} y_1^2 + \frac{1}{\lambda_2} y_2^2 + \dots + \frac{1}{\lambda_n} y_n^2 \quad (3.16)$$

Da λ_1 der größte Eigenwert ist, kommt die Hauptachse des Hyperellipsoids in Richtung \mathbf{e}_1 zu liegen. Für den Fall $n = 2$ ist dieser Zusammenhang in Abbildung 3.2 veranschaulicht.

³dabei bezeichnet \mathbf{e}_i^T denn transponierten Eigenvektor zu \mathbf{e}_i .

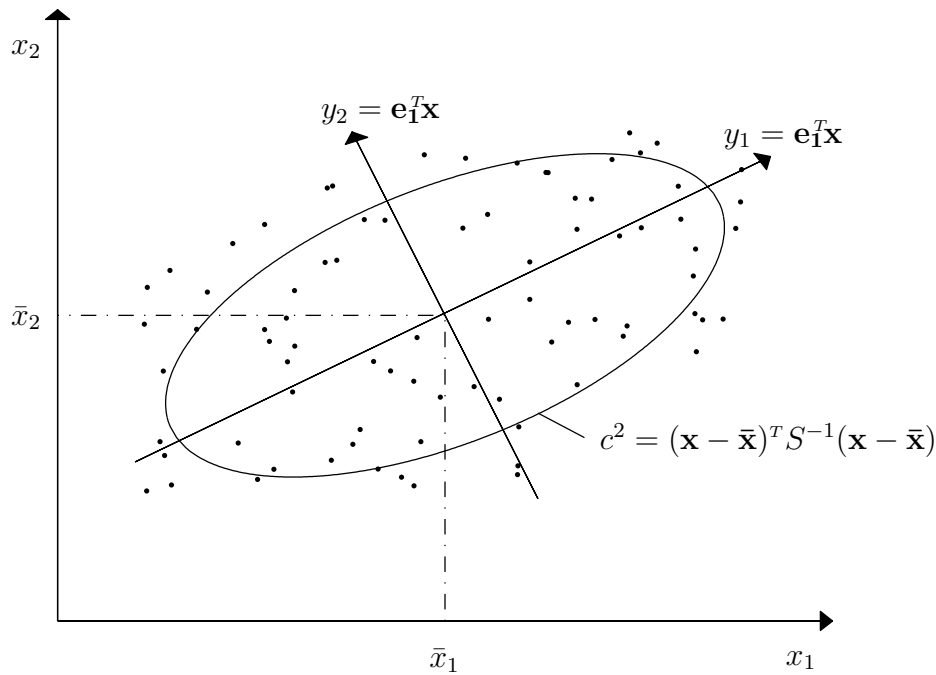


Abbildung 3.2: Ellipsoid zweier Hauptkomponenten.

Die beiden Hauptkomponenten y_1 und y_2 bilden die Achsen des Ellipsoids c^2 , das einen Großteil der Variation des dargestellten Datensatzes umschließt. Für $\lambda_1 > \lambda_2$ bildet y_1 die Haupt- und y_2 die Nebenachse.

Neben der Möglichkeit, die Anzahl von Vektoren eines mehrdimensionalen Zufallsvektors mit Hilfe der Hauptkomponentenanalyse zu reduzieren, kann nach [Wed99] das Hyperellipsoid der Gleichung 3.16 zur Überwachung der Gültigkeitsdomäne⁴ eines künstlichen neuronalen Netzes verwendet werden. Durch die Transformation der Trainingsvektoren in Hauptkomponenten wird der ursprüngliche Datenraum auf ein Hyperellipsoid abgebildet, dessen Begrenzung leicht zu überwachen ist. In der Anwendungsphase kann so überprüft werden, ob der aktuelle Eingangsvektor innerhalb der Gültigkeitsdomäne, also des vom Trainings aufgespannten Hyperellipsoids liegt, bzw. wie weit er davon entfernt ist. So können Eingangsvektoren abgelehnt werden, die außerhalb der Extrapolationsmöglichkeiten des neuronalen Netzes liegen.

Sowohl bei der Verwendung der Hauptkomponentenanalyse zur Reduzierung der Eingangsdimensionalität künstlicher, neuronaler Netze, als auch bei der Überwachung der Gültigkeitsdomäne ist eine Transformation der Eingangsvektoren auch während der Anwendungsphase erforderlich. Dies stellt einen gravierenden Nachteil für den Einsatz mit schnellen Echtzeitanwendungen dar.

⁴Unter Gültigkeitsdomäne wird in [Wed99] der Datenraum verstanden, der durch die Trainingsvektoren aufgespannt wird.

3.1.4 Frequenzanalysen

Frequenzanalysen ermitteln, welche Frequenzanteile in einem Signal enthalten sind und welchen Anteil die einzelnen Frequenzen am Gesamtsignal haben. Aufbauend auf der Theorie zur Fourier-Transformation in Kapitel 2 Abschnitt 2.3.4 lässt sich die Transformation der Gleichung 2.49 auch für den diskreten Fall formulieren. Sei $s(t)$ ein Signal der Zeitdauer T , dann lassen sich die Fourier-Transformation und ihre Inverse wie folgt berechnen:

$$S(f) = \sum_{t=0}^T s(t) \cdot e^{\frac{-j2\pi ft}{T}} \quad ; \quad \text{mit } f = 0 \dots F \quad (3.17)$$

$$s(t) = \frac{1}{T} \sum_{f=0}^F S(f) \cdot e^{\frac{j2\pi ft}{T}} \quad ; \quad \text{mit } t = 0 \dots T$$

Dabei sind t und f linear aufsteigend, haben die gleiche Länge und F bezeichnet die Abtastfrequenz des Signals $s(t)$. Um die so gewonnenen Frequenzanteile, also das Frequenzspektrum von $s(t)$ zu analysieren, sind diverse Darstellungsformen üblich, von denen im Folgenden zwei diskutiert werden:

$$\text{Amplitudenspektrum}(f) = \frac{2}{T} \cdot \sqrt{\text{real}(S(f))^2 + \text{imag}(S(f))^2} \quad (3.18)$$

$$\text{Energiedichtespektrum}(f) = \frac{1}{T} \cdot S(f) \cdot \text{conj}(S(f)) \quad (3.19)$$

Beide Spektren sind eine Darstellung der Koeffizienten der durch die Transformation ermittelten Frequenzanteile. Dabei ist 3.18 eine normierte Darstellung der absoluten Amplituden. Zur Darstellung der Energieverteilung auf die Frequenzen dient das Spektrum der Gleichung 3.19.

Beispiel 3.1.2 (Frequenzspektren)

Für $t = \{0, \frac{1T}{F}, \frac{2T}{F}, \dots, T\}$ mit einer Abtastfrequenz $F = 1000 \text{ Hz}$ sei folgendes Signal $s(t)$ gegeben:

$$s(t) = a_1 \cdot \sin(2\pi f_1 \cdot t) + a_2 \cdot \sin(2\pi f_2 \cdot t) + a_3 \cdot \sin(2\pi f_3 \cdot t) \quad (3.20)$$

Dabei seien die Amplituden gegeben durch $a_1 = 0.5$, $a_2 = 1.0$ und $a_3 = 0.7$, sowie die Frequenzen mit $f_1 = 50 \text{ Hz}$, $f_2 = 150 \text{ Hz}$ und $f_3 = 300 \text{ Hz}$. Dann können die Koeffizienten der Fourier-Transformation entsprechend den Spektren 3.18 und 3.19 dargestellt werden.

Abbildung 3.3 dient zum Vergleich der unterschiedlichen Information, die aus den Spektren gewonnen werden kann. Durch die normierte Darstellung der Amplituden können die Amplituden $a_1 = 0.5$, $a_2 = 1.0$ und $a_3 = 0.7$ korrekt abgelesen werden.

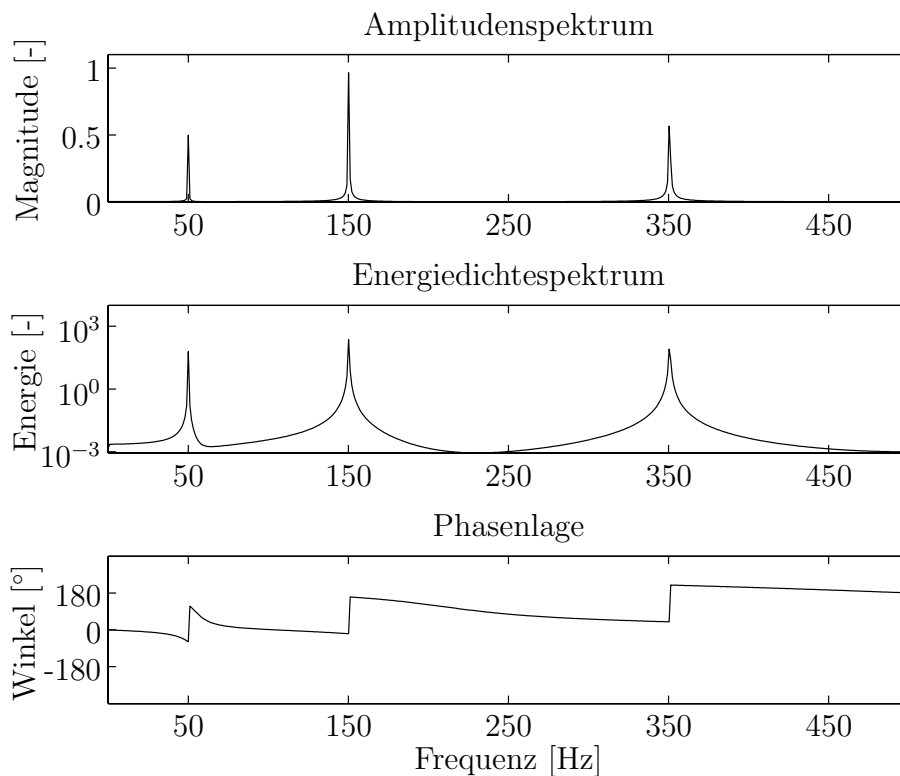


Abbildung 3.3: Spektraldarstellungen zur Fourier-Transformation.

Im Amplitudenspektrum sind die Anteile der Signalamplituden auftretender Frequenzen sichtbar. Das Energiedichtespektrum gibt Auskunft über die Energieverteilung der einzelnen Frequenzen. Dabei haben die einzelnen Frequenzen zur Rekonstruktion des Originalsignals eine unterschiedliche Phasenlage.

Aus der Darstellung der Energiedichte kann abgelesen werden, daß die drei Frequenzen $f_1 = 50 \text{ Hz}$, $f_2 = 150 \text{ Hz}$ und $f_3 = 300 \text{ Hz}$ in etwa den gleichen Energiebeitrag zum Gesamtsignal leisten. Neben diesen Informationen kann aus den komplexen Koeffizienten der Fourier-Transformation auch die Phasenlage der zur Rekonstruktion des Originalsignals verwendeten Frequenzen dargestellt werden.

Ein großer Nachteil dieser Darstellung ist der Verlust der zeitlichen Information, wann welche Frequenz im Originalsignal auftritt. Im Beispiel 3.1.2 spielt diese Information keine Rolle, da alle Frequenzen über den gesamten Zeitraum des Signal auftreten. Bei Meßdaten ist dies in der Regel nicht gegeben.

Um den zeitlichen Bezug nicht zu verlieren, kann die Fourier-Transformation gefenstert erfolgen. Dazu wird ein kurzes Zeitfenster (*engl. Short-Time*) des Originalsignals betrachtet, das langsam über den gesamten Zeitraum des Signals geschoben wird. Die Darstellung der sogenannten *Short-Time-Fourieranalyse* wird auch Spektrogramm genannt und wird mit folgendem Beispiel beschrieben:

Beispiel 3.1.3 (Short-Time-Fourieranalyse)

Zur Analyse von Schwingungen innerhalb eines Hochdruckspeichers zur Kraftstoffeinspritzung ist der Zeitpunkt des Auftretens signifikanter Frequenzen von großer Bedeutung. Anhand solcher Druckdaten ist der Informationsgewinn der *Short-Time Fourier-Analyse* gut erkennbar.

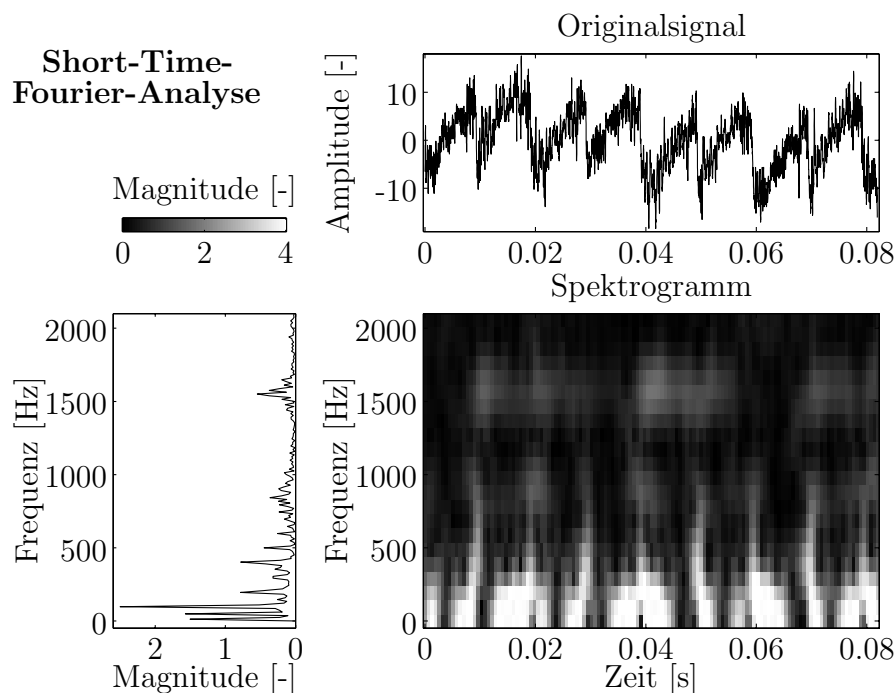


Abbildung 3.4: Spektrogramm der Short-Time-Fourier-Analyse.

Mit Hilfe des Spektrogramms läßt sich das zeitliche Auftreten einzelner Frequenzen darstellen. Im Gegensatz dazu liefert das Amplitudenspektrum nur eine gemittelte Information über den gesamten Zeitraum des Originalsignals.

Über den zeitlichen Verlauf des Originalsignals der Abbildung 3.4 sind 8 Druckeinbrüche zu erkennen. Diese zeitlich begrenzten Druckschwankungen sind auch als helle Fahnen im Spektrogramm der *Short-Time Fourier-Analyse* zwischen 500 und 1100 Hz gut getrennt sichtbar. Auch tiefere Frequenzen spiegeln diese Druckveränderungen wieder, jedoch sind sie hier nicht so gut getrennt. Im Bereich um 1500 Hz werden ebenfalls Schwingungen angeregt, jedoch lassen sie sich nur sehr schwer trennen und zuordnen.

Im Gegensatz dazu liefern gewöhnliche Amplitudenspektren nur eine gemittelte Information über den gesamten Zeitraum des Originalsignals, jedoch keine Aussage darüber, welche Frequenz wann zur Auslösung bestimmter Schwingungen führt.

Die beschriebenen Darstellungsformen zur Frequenzanalyse sind Standardmethoden zur Daten- und Signalanalyse und werden vielseitig eingesetzt. Durch die Erweiterung von Amplitudenspektren zu Spektrogrammen können auch zeitliche Zusammenhänge untersucht werden, jedoch ist eine informationstheoretisch basierte Frequenzanalyse auf diesem Weg nicht realisierbar. Darunter ist die Aussage zu verstehen, welche Frequenzen des Eingangssignals eines Systems relevant sind und somit eine gewünschte Systemantwort des System erzeugen können.

Eine Beschreibung solcher Systemzusammenhänge und einer eigens dafür neu entwickelten Methode zur *informationsoptimierten Frequenzanalyse* ist im Kapitel 4, Abschnitt 4.2 detailliert dargestellt.

3.2 Auswahl und Optimierung von Gewichten

Auf dem Gebiet neuronaler Netze besteht, neben anderem, ein sehr großes Forschungsinteresse in der Steigerung der Trainingsgeschwindigkeit. Der Reduzierung bzw. Optimierung von Netztopologien wird weitaus weniger Beachtung geschenkt. Dabei ist gerade die Spezifikation einer bestmöglichen Netzwerktopologie einer der schwierigsten Schritte bei der Modellierung. Neben der Entwicklung effizienter Lernalgorithmen sind auch einige Ideen zur Topologieoptimierung entwickelt und umgesetzt worden.

Bei der Optimierung neuronaler Netze lassen sich bezüglich der Topologie drei Gesichtspunkte betrachten, unabhängig davon, ob die Anpassung vor, während oder nach dem Training erfolgt:

- Wie viele Neuronen, insbesondere verdeckter Schichten, benötigt die Approximationsaufgabe?
- Wie klein darf das neuronale Netz sein, ohne sein Generalisierungsverhalten zu verlieren?
- Wie sieht eine optimale Vernetzung der Neuronen aus?

Oftmals laufen Algorithmen zur Optimierung der Netztopologie Hand in Hand mit einem Lernverfahren. Dabei muß nicht immer ein neues Verfahren implementiert, sondern nur geschickt genutzt werden. Eine gute Zusammenstellung bekannter Methoden liefert [And97], wovon hier einige vorgestellt werden.

3.2.1 Daumenregeln

Bei der Auswahl von Netztopologien sind bis heute, auf Grund ihrer Einfachheit, Daumenregeln immer noch eines der beliebtesten Mittel. Dabei verläßt sich der Anwender auf seine Erfahrung und sein Geschick, die richtige Topologie durch Probieren zu schätzen. Dabei stützt er sich auf eine häufig zitierte Daumenregel, wobei für die Anzahl der verdecken Neuronen das geometrische Mittel aus der Anzahl n der Ein- und m der Ausgabeneuronen zu wählen ist. Ein neuronales Netz mit einer verdeckten Schicht hat somit

$$\#\{w_{ij}\} = \frac{n^2 + 2n \cdot m + m^2}{2} + n \cdot m \quad (3.21)$$

Gewichte bzw. Parameter, um die gesuchte Funktion zu approximieren. Dabei sind direkte Verbindungen zwischen Eingang und Ausgang bereits berücksichtigt.

Es gibt viele ähnliche Regeln, von denen jedoch keine der Komplexität der Approximationsaufgabe Rechnung tragen kann. Solche Daumenregeln liefern nur eine vage Idee für einen ersten Ansatz einer Netzstruktur. Ein Maß für die Komplexität kann die Dimension des zu approximierenden Datensatzes sein.

Definition 3.2.1 (Dimension eines Datensatzes)

Sei $\mathbf{E}_d = \{\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_d\}$ die orthogonale Basis eines d -dimensionalen Datenraums. Läßt sich die Varianz der Ausgangsvariablen eines Datensatzes durch keine kleinere orthogonale Basis als \mathbf{E}_d erklären, dann ist d die Dimension des Datensatzes.

Eine weitere Möglichkeit zur Bestimmung der Dimension kann über den Begriff Informationsdimension erfolgen, der von SHANNON geprägt wurde [Spe95]. Diese Möglichkeit machen sich die neuen Methoden zur informationsoptimierten Netztopologie des Kapitels 4 zu Nutzen.

Neben der Netzstruktur hängt auch die Anzahl T der zum Training benötigten Datensätze von der Komplexität der Aufgabe ab. Man spricht auch vom Fluch der Dimension [Göp97]. Die Schätzung der Anzahl T kann nach folgender Daumenregel erfolgen:

$$T = q^{-2 - \frac{d}{p}} \quad (3.22)$$

Dabei drückt q die Qualität der Approximation und p die Glattheit der zu approximierenden Funktion aus. Auch wenn der Zahlenwert höchstens empirisch korrekt ist, beschreibt er die Auswirkung der Dimension und der Glattheit der Daten auf die Anzahl der notwendigen Trainingsdaten, was generell zu beachten ist.

Zur Reduzierung der Dimension kann die in Abschnitt 3.1.3 beschriebene Hauptkomponentenanalyse verwendet werden. Im Kapitel 4 werden zur Reduzierung der Dimension neue Methoden, basierend auf Informationsmaßen nach SHANNON vorgestellt.

3.2.2 Regularisierung

Eine sehr verbreitete Methode besteht darin, die Topologie eines neuronalen Netzes indirekt durch Regularisierung der Gewichte zu optimieren. Die bekanntesten Methoden schwächen Gewichte (engl. *Weight Decay*) und entfernen Gewichte (engl. *Weight Elimination*). Beide Methoden versuchen die Gewichte eines Netzwerkes möglichst nahe Null zu halten. Grundlage der Verfahren ist die Annahme, daß große Gewichte bzw. Gewichtsunterschiede zu komplexen, instabilen Netzen führen. Um die Gewichte zu regulieren wird die Fehlerfunktion E um einen zusätzlichen Fehlerterm erweitert, der die Komplexität eines Netzwerkes bestrafen soll und zusätzlich zu minimieren ist:

$$\hat{E} = E + \frac{\beta}{2} \sum_{i,j} w_{ij}^2 \quad (3.23)$$

Somit ist die Fehlerfunktion mit dem Gewichtungsfaktor β und jede Gewichtsänderung abhängig von der Größe der aktuellen Verbindungsgewichte. Eine Übersicht über weitere Bestrafungsterme bietet [Bis95]. Auch wenn die Idee zu dieser Methode schon lange besteht, so beinhalten dennoch zahlreiche aktuelle Lernverfahren ähnliche Terme, um Gewichte zu kontrollieren (z.B. [Bar97]).

Die Komplexität der Netztopologie kann durch Entfernen von Verbindungen mit kleinen Gewichten erfolgen. Die Wahl eines entsprechenden Schwellwertes entscheidet dann über die Art der Optimierung.

Das Problem von Regularisierungsmethoden besteht darin, daß sie aus heuristischen Überlegungen resultieren. Warum sollen alle Gewichte beim *Weight Decay* ausgerechnet nahe Null liegen? Manche Gewichte liegen unter Umständen optimaler Weise weit entfernt von Null.

3.2.3 Early Stopping

Ein weiteres Verfahren, die Leistungsfähigkeit einer Netztopologie zu steigern, besteht in einer Überwachung des Trainingsvorgangs mit einem frühzeitigen Abbruch (engl. *early stopping*). Dazu wird die Datenmenge in eine Trainings- und eine Validierungsmenge aufgeteilt (siehe Definition 2.2.17 in Kapitel refsec:Theorie, Abschnitt 2.2.3). Schon hier ergeben sich zwei Probleme. Welche Datensätze kommen in welche Menge, und auf welcher Größe soll man die Trainingsmenge belassen? Die Methode besteht nun darin, während des iterativen Trainingsprozesses den Fehler der Trainingsmenge mit dem Fehler der Validierungsmenge zu vergleichen. Abbildung 3.5 veranschaulicht diesen Vergleich.

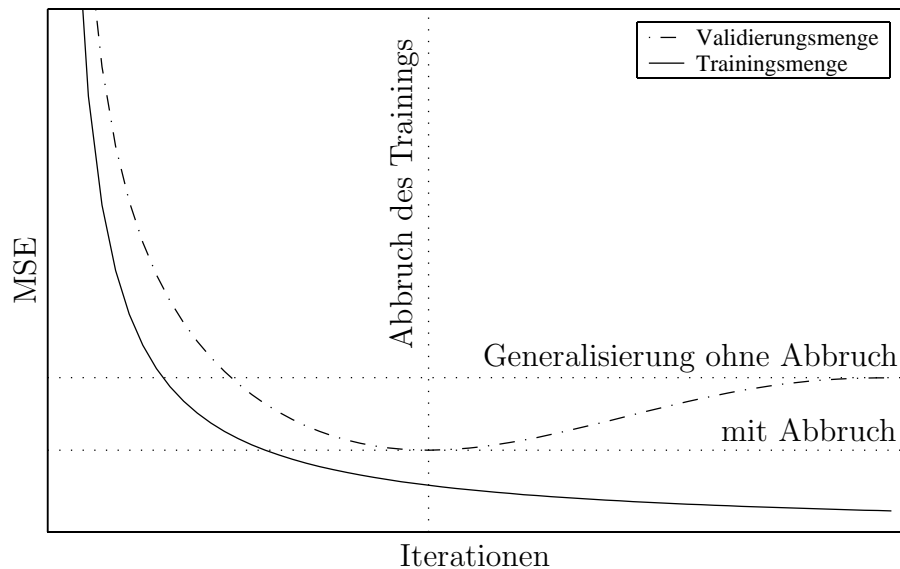


Abbildung 3.5: Generalisierung mit und ohne Abbruch.

Zur Steigerung der Generalisierung eines neuronalen Netzes kann das Training frühzeitig abgebrochen werden. Häufig ist zu beobachten, daß mit fortschreitendem Lernen der Fehler der Validierungsdaten ansteigt. Wird das Training vor einem Anstieg abgebrochen, kann das Netz Eingaben außerhalb der Trainingsmenge besser klassifizieren. Über völlig unbekannte Daten kann jedoch keine Aussage gemacht werden, da die Überwachung mit der Validierungsmenge erfolgt.

Die Gewichte eines bewußt zu groß gewählten Netzwerks werden durch den Trainingsalgorithmus nur auf Grund des Fehlers der Trainingsmenge angepaßt. Kann der Fehler der Validierungsmenge nicht weiter minimiert werden, wird das Training frühzeitig abgebrochen. Dies ist notwendig, da große Netze dazu neigen, sich nur an die Trainingsmenge anzupassen, dafür aber mit einem sehr kleinen Restfehler. Durch den Abbruch kann das übermäßige Anpassen an die Trainingsmenge verhindert werden ([Sar95] und [AMM⁺97]).

Das Hauptproblem dieser Vorgehensweise besteht darin, daß für eine gegebene Topologie lediglich ein optimaler Abbruchzeitpunkt festgestellt werden kann, jedoch keine Optimierung der Topologie erfolgt. Ob die gewählte Topologie optimal, ist bleibt offen.

3.3 Topologieoptimierende Algorithmen

Neben den vorgestellten Möglichkeiten eine problemspezifische Topologie auszuwählen bzw. indirekt zu beeinflussen, bestehen auch Verfahren, die direkten Einfluß auf die Topologie haben.

Topologieoptimierende Algorithmen können grob in zwei Klassen eingeteilt werden:

- Beschneidende Algorithmen (engl. *Pruning*) reduzieren ein bereits vollständig trainiertes neuronales Netz.
- Konstruierende Algorithmen (engl. *Constructive*) bauen eine angepaßte Netzstruktur während des Trainings auf.

Die folgende Darstellung ausgewählter Algorithmen beider Klassen soll lediglich einen groben Überblick geben. Genauere Beschreibungen können den entsprechenden Literaturangaben entnommen werden.

3.3.1 Pruning-Algorithmen

Ziel von *Pruning*-Algorithmen ist es, in einem vollständig trainierten neuronalen Netz nicht signifikante Verbindungen bzw. Neuronen zu identifizieren und zu entfernen. Damit lassen sich Pruningverfahren in zwei Klassen einteilen:

- *Node-Pruning*-Verfahren versuchen, nicht relevante Neuronen bzw. Knoten (engl. *Node*) zu finden und zu eliminieren.
- *Link-Pruning*-Verfahren bewerten Verbindungsgewichte (engl. *Link*) der Netzstruktur und entfernen schwache Verknüpfungen.

Zu den beiden Klassen werden im Folgenden zwei Methoden beschrieben. Die Skelettierung (engl. *Skeletonization*) zum Löschen von Neuronen und die sogenannte optimale Hirn Schädigung (engl. *Optimal Brain Damage*), eine Methode zum Entfernen von Verbindungen.

Skeletonization:

Dieses Verfahren zum *Node*- oder *Neuronen-Pruning* wurde von MOZER und SMOLENSKY [MS89] entwickelt. Es zeichnet sich durch eine hohe Rate an gelöschten Neuronen pro Zyklus aus. Reduziert wird das Netzwerk durch eine Abschätzung der Fehlerfunktion beim Entfernen einzelner Zellen (engl. *units*) aus der Netzstruktur.

Für jeden Knoten wird zur Berücksichtigung seiner Stärke das Maß α_i eingeführt:

$$net_j = \sum_j w_{ij} \cdot \alpha_i \cdot o_i \quad (3.24)$$

Die Auswirkung auf die Fehlerfunktion beim Entfernen von Neuronen aus der Netzstruktur führt zu folgender Betrachtung der Relevanz:

$$\rho_i = E_{\alpha_i=0} - E_{\alpha_i=1} \quad (3.25)$$

Durch Ableitung der Relevanz ρ_i nach α_i kann an der Stelle $\alpha_i = 1$ für ρ_i eine Abschätzung hergeleitet werden. In der praktischen Anwendung schwankt dieser Wert mit der Zeit sehr stark. Eine stabilere Schätzung wird durch folgende, exponentielle Gewichtung erreicht:

$$\rho_i(t+1) \approx 0.8 \cdot \rho_i(t) + 0.2 \cdot \frac{\partial E}{\partial \alpha_i} \quad (3.26)$$

Die Neuronen mit der geringsten Relevanz ρ_i können nun aus dem Netzwerk entfernt werden. Das neuronale Netzwerk wird erneut trainiert und der Vorgang kann, abhängig von den gegebenen Abbruchkriterien wiederholt werden. Auch in aktuelleren Arbeiten werden ähnliche Maße definiert, die einen Anstieg des Fehlers durch Entfernen von Neuronen bewerten [MMJ94].

Optimal Brain Damage:

Das Verbindungs- oder Netzwerk-*Pruning* wurde von LECUN, DENKER und SOLLA [LDS90] entwickelt und optimale Hirn-Schädigung (engl. *Optimal Brain Damage*) genannt. Eine Verallgemeinerung dieser Methode durch HASSIBI und STORK [HD93] wird als optimaler Hirn-Chirurg (engl. *Optimal Brain Surgeon*) bezeichnet. Das Vorgehen von LECUN geht davon aus, daß bei Auswahl geeigneter Gewichte keine Wechselwirkungen zu anderen Verbindungen bestehen und somit durch gezieltes Entfernen kein Informationsverlust entsteht. Folgender Algorithmus versucht, durch erneutes Trainieren des neuronalen Netzes etwaige Verluste zu kompensieren.

Algorithmus 3.3.1 (Optimal Brain Damage)

Der Algorithmus entfernt aus einem mit den Meßreihenpaaren X und Y vollständig trainierten Netzwerk fortschreitend einzelne Neuronen. Die Auswahl erfolgt durch Berechnung einer sogenannten Saliency aus den Diagonalelementen der Hessematrix der Fehlerfunktion. In jeder Iteration wird das reduzierte Netzwerk erneut mit den Meßreihenpaaren X und Y trainiert und der Ausgabefehler überwacht.

Eingangsparameter:

- Die Eingangsmatrix $X(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}_{t=1}^{T_1}$ aus Spaltenvektoren $x(t)$ der einzelnen Messungen am Systemeingang. Dabei ist N die Anzahl der Messungen und T_1 die Länge der Zeitreihen des Systemeingangs bzw. maximale Dimension des Netzeingangs.
- Die Ausgangsmatrix $Y(t) = \{y_1(t), y_2(t), \dots, y_N(t)\}_{t=1}^{T_2}$ aus Spaltenvektoren $y(t)$ der einzelnen Messungen am Systemausgang. Dabei ist T_2 die Länge der Zeitreihen des Systemausgangs bzw. die Dimension des Netzausgangs.
- Die Anzahl n der Trainingsepochen bis zum Validierungstest.
- Trainiertes Netzwerk NET_{trn} mit minimalem Ausgabefehler MSE_{trn} .

Algorithmus:

1. Berechne die Diagonalelemente der Hessematrix H der Fehlerfunktion von NET_{trn} .
2. Berechne die Saliency S für jedes Gewicht und entferne das Gewicht mit der geringsten Saliency.
3. Trainiere reduziertes Netzwerk für n Epochen mit Lerndaten $X_{trn}(t) \subset X(t)$ und entsprechender Trainingsausgabe $Y_{trn}(t)$.
4. Speichere Ausgabefehler MSE_{opt} der Netzausgabe bezüglich Validierungsausgabe $Y_{val}(t)$ mit ausgewählten Validierungsdaten $X_{val}(t) \subset X(t)$, wobei $X_{trn}(t) \cap X_{val}(t) = \emptyset$ gilt.
5. Wiederhole ab 2. falls Ausgabefehler⁵ MSE_{opt} kontinuierlich fällt.
6. Wiederhole ab 1. falls Ausgabefehler $MSE_{opt} \leq MSE_{trn}$.

Ausgabeparameter:

- Netzwerk NET_{opt} mit minimalem Ausgabefehler MSE_{opt} .

Beschreibung:

Zur Bestimmung der Relevanz (engl. *Saliency*) eines Gewichtes werden zunächst die Diagonalelemente der Hessematrix H der Fehlerfunktion eines zuvor vollständig trainierten neuronalen Netzes berechnet. Aus den Diagonalelementen lassen sich über die *Saliency* Gewichte des Netzwerkes identifizieren, die nicht für den Zusammenhang der Meßreihenpaare X und Y signifikant sind. Nach Entfernen eines Gewichtes wird das Netz nachtrainiert, um die Strukturveränderung auszugleichen. Das *Pruning* wird in Abhängigkeit des aktuellen Validierungsfehlers abgebrochen.

⁵Neben dem MSE sind auch andere Fehlermaße als Abbruchkriterium denkbar.

Berechnung der Saliency:

Die Herleitung der *Saliency* erfolgt durch eine Approximation der Fehleränderung ΔE durch ein Taylorpolynom zweiter Ordnung:

$$S = \sum_k g_k \Delta w_k + \frac{1}{2} \sum_k h_{kk} \Delta w_k^2 + \frac{1}{2} \sum_{k \neq l} h_{kl} \Delta w_k \Delta w_l \approx \Delta E \quad (3.27)$$

Dabei bezeichnet Δw_k die Gewichtsänderung eines Verbindungsgewichtes w_{ij} . Die Komponenten des Gradienten der Fehlerfunktion bezüglich dieser Gewichte werden mit

$$g_k = \frac{\partial E}{\partial w_k} \quad \text{und} \quad h_{kl} = \frac{\partial^2 E}{\partial w_k \partial w_l}$$

der Hessematrix bezeichnet.

Vorausgesetzt das *Pruning* wird an einem vollständig trainierten neuronalen Netzwerk vorgenommen, geht der Gradient g_k und somit der erste Term der Gleichung 3.27 gegen Null. Durch die Annäherung der Hessematrix an eine Diagonalmatrix ergibt sich die Vereinfachung:

$$S \approx \frac{1}{2} \sum_k h_{kk} \Delta w_k^2 \quad (3.28)$$

Die *Saliency* S_k , also die Relevanz eines Gewichtes, ergibt sich entsprechend $S_k = \frac{1}{2} h_{kk} w_k^2$, nach Berechnung der zweiten Ableitung.

Eine ausführlichere Zusammenstellung der *Pruning*-Algorithmen *Skeletonization*, *Optimal Brain Damage* und *Optimal Brain Surgeon* kann in [And97] und [Sch96] gefunden werden.

3.3.2 Cascade Correlation

Ein Algorithmus mit "Korrelierten" Kaskaden (engl. *Cascade Correlation*) ist ein von SCOTT FAHLMAN und CHRISTIAN LEBIÈRE [FL90] entwickelter Lernalgorithmus, der nicht nur die Gewichte eines neuronalen Netzes adaptiert, sondern auch die Topologie verändert. Neben *Cascade Correlation* sind in der Literatur noch weitere konstruierende Algorithmen wie z.B. *Perceptron Cascade* [Bur94], DNAL [Bar94] und DCN [RL93] zu finden. Da *Cascade Correlation* das bekannteste Verfahren ist, soll im Folgenden näher auf den Algorithmus eingegangen werden.

Algorithmus 3.3.2 (Cascade Correlation)

Der Algorithmus erzeugt und trainiert für Meßreihenpaare X und Y eine online wachsende Netztopologie. Dabei werden nur Gewichte neu hinzugefügter Neuronen trainiert. Das Training bzw. Wachstum ist abgeschlossen, wenn sich der Netzfehler nicht weiter verbessert bzw. der Validierungsfehler zu wachsen beginnt.

Eingangsparameter:

- Die Eingangsmatrix $X(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}_{t=1}^{T_1}$ aus Spaltenvektoren $x(t)$ der einzelnen Messungen am Systemeingang. Dabei ist N die Anzahl der Messungen und T_1 die Länge der Zeitreihen des Systemeingangs bzw. maximale Dimension des Netzeingangs.
- Die Ausgangsmatrix $Y(t) = \{y_1(t), y_2(t), \dots, y_N(t)\}_{t=1}^{T_2}$ aus Spaltenvektoren $y(t)$ der einzelnen Messungen am Systemausgang. Dabei ist T_2 die Länge der Zeitreihen des Systemausgangs bzw. Dimension des Netzausgangs.
- Die Anzahl n der Trainingsepochen bis zum Validierungstest.

Algorithmus:

1. Erzeuge initialisiertes Netzwerk NET mit T_1 Eingangs- und T_2 Ausgangszellen, sowie Bias Neuron.
2. Trainiere alle Gewichte zu den Ausgabeneuronen für n Epochen mit Lerndaten $X_{trn}(t) \subset X(t)$ und entsprechender Trainingsausgabe $Y_{trn}(t)$.
3. Speichere Ausgabefehler MSE_{net} der Netzausgabe bezüglich Validierungsausgabe $Y_{val}(t)$ mit ausgewählten Validierungsdaten $X_{val}(t) \subset X(t)$, wobei $X_{trn}(t) \cap X_{val}(t) = \emptyset$ gilt.
4. Wiederhole ab 2. falls Ausgabefehler⁶ MSE_{net} kontinuierlich fällt.
5. Speichere NET_{opt} und Ausgabefehler MSE_{opt} für Netzausgabe $X_{val}(t) \subset X(t)$ mit minimalem Ausgabefehler MSE_{net} bezüglich Validierungsausgabe $Y_{val}(t)$.
6. Erweitere Netzwerk um Kandidatenzelle j durch Verbinden mit allen Ein- und Ausgangszellen, sowie mit den Ausgängen zuvor erzeugter Neuronen.
7. Adaptiere Eingangsgewichte der Kandidatenzelle so, daß die Korrelation der Kandidatenausgabe o_j und des Restfehlers δ (ohne Einfluß der Kandidatenzelle) der Netzausgabe bezüglich $X_{trn}(t) \subset X(t)$ maximiert wird.
8. Wiederhole ab 2. falls Ausgabefehler MSE_{opt} kontinuierlich fällt.

Ausgabeparameter:

- Netzwerk NET_{opt} mit minimalem Ausgabefehler MSE_{opt} .

⁶Neben dem MSE sind auch andere Fehlermaße als Abbruchkriterium denkbar.

Beschreibung:

Zunächst generiert der Algorithmus eine minimale Netztopologie ohne verdeckte Neuronen, jedoch mit durch die Daten vorgegebener Dimension für Eingangs- und Ausgangsschicht. Die Ein- und Ausgangsneuronen, sowie das Bias Neuron werden vollständig verbunden und initialisiert. Daraufhin kann das Netzwerk mit einem beliebigen Lernalgorithmus trainiert werden. Dabei dient der Trainingsdatensatz X_{trn} dem Lernen und der Validierungsdatensatz X_{val} der Überwachung des Trainingserfolgs. Kann der Validierungsfehler der Netzausgabe nicht weiter minimiert werden, wird ein neues Neuron j erzeugt und mit allen Eingangsneuronen, sowie den zuvor generierten Neuronen verbunden. Dabei werden die Eingangsgewichte entsprechend Gleichung 2.11 aus Abschnitt 2.1.5 so bestimmt, daß die Korrelation der Ausgabe o_j des neuen Neurons und des Restfehlers δ der Netzausgabe bezüglich $X_{trn}(t) \subset X(t)$ maximiert wird. Für das weitere Training werden diese Gewichte zu den Vorgängerzellen eingefroren. Der Ausgang des neuen Neurons wird mit allen Ausgabezellen verbunden und zusammen mit allen bisherigen Ausgangsgewichten adaptiert. Es werden so lange neue Neuronen generiert bis der Validierungsfehler durch die Trainingsphase nicht weiter minimiert werden kann.

Gewichtsberechnung der Kandidatenzelle:

Ziel des Trainings der Eingangsgewichte der Kandidatenzelle ist es, die Summe S_j der Beträge der Korrelation⁷ zwischen der Ausgabe o_j und dem Restfehler δ_t aller T_2 Ausgabezellen zu maximieren.

$$S_j = \sum_{t=1}^{T_2} \left| \sum_{n=1}^N (o_{nj} - \bar{o}_j)(\delta_{nt} - \bar{\delta}_t) \right| \quad (3.29)$$

Um S_j zu maximieren, ist die partielle Ableitung von S_j nach jedem Gewicht w_{ij} zwischen den Vorgängerzellen i und der Kandidatenzelle j zu berechnen.

$$\frac{\partial S_j}{\partial w_{ij}} = \sum_{t=1}^{T_2} \left(\operatorname{sgn} \left(\sum_{n=1}^N (o_{nj} - \bar{o}_j)(\delta_{nt} - \bar{\delta}_t) \right) \sum_{n=1}^N \frac{\partial o_{nj}}{\partial w_{ij}} (\delta_{nt} - \bar{\delta}_t) \right) \quad (3.30)$$

Mit der Definition $\sigma_t = \operatorname{sgn} \left(\sum_{n=1}^N (o_{nj} - \bar{o}_j)(\delta_{nt} - \bar{\delta}_t) \right)$ für das Vorzeichen der Korrelation ergibt sich als partielle Ableitung von S_j nach dem Gewicht w_{ij} die Gleichung:

$$\frac{\partial S_j}{\partial w_{ij}} = \sum_{t=1}^{T_2} \sum_{n=1}^N \sigma_t \cdot f'_{act}(net_{nj}) \cdot o_{ni} \cdot (\delta_{nt} - \bar{\delta}_t) \quad (3.31)$$

Hierbei ist $f'_{act}(net_{nj})$ die Ableitung der Aktivierungsfunktion der Kandidatenzelle j nach der Netzeingabe net_{nj} und o_{ni} die Ausgabe der Vorgängerzelle i für Messung n .

⁷Dabei handelt es sich in Gleichung 3.29 um eine Kovarianz, da nach FAHLMAN [FL90] dies oft zu besseren Ergebnissen führt als die Verwendung der Korrelation.

Nachdem $\partial S_j / \partial w_{ij}$ für jedes Eingangsgewicht des Kandidaten j berechnet ist, kann ein Gradientenaufstieg erfolgen, um S_j durch Änderung der Gewichte w_{ij} zu maximieren.

Eine ausführliche Beschreibung des Algorithmus mit diversen Variationen kann in [Zel93] gefunden werden. So sind auch Kombinationen von *Cascade Correlation* und diversen *Pruning* Algorithmen realisierbar. Dabei wird die Anzahl an Verbindungen jeder neuen Kandidatenzelle nach erfolgreichem Training durch *Pruning* reduziert.

3.4 Probleme aktueller Methoden

Durch die hohen Anforderungen an Verarbeitungsgeschwindigkeit und Auswertungsgenauigkeit bei der Verbrennungsüberwachung moderner Motoren können die dargestellten aktuellen Methoden keine ausreichenden Ergebnisse liefern. Die dabei auftretenden Probleme werden in den folgenden Abschnitten diskutiert.

3.4.1 Einschränkungen bei der Datenanalyse

Die bekanntesten Methoden zur Datenanalyse, die lineare Regressions- und Korrelations-, sowie die Hauptkomponentenanalyse liefern nur für Daten mit linearem Zusammenhang gute Ergebnisse. Dies ist ein entscheidender Nachteil für den Einsatz mit Meßdaten unbekannter Zusammensetzung. Auch wenn sich die Methoden auf polynomiale oder exponentielle Zusammenhänge erweitern lassen, muß der Anwender den funktionalen Zusammenhang grob kennen und angeben. Die klassischen Frequenzanalysen mit Hilfe der Fourier-Transformation liefern wertvolle Ergebnisse zum Signalverständnis, jedoch wird kein Bezug zwischen Signalein- und ausgang hergestellt. Bei der Verwendung der in Kapitel 4 vorgestellten Methoden zur informationsoptimierten Datenanalyse wird ein beliebiger Zusammenhang zwischen Ein- und Ausgang der Daten auch ohne jegliche Angaben des Anwenders erkannt. Die dadurch abgedeckten Nachteile klassischer Methoden sollen hier ihren Vorteilen gegenübergestellt werden.

Vorteile:

- Die Berechnung der linearen Regression und Korrelation ist einfach und effizient.
- Sowohl Korrelations- als auch Hauptkomponentenanalyse ermöglichen neben der Datenanalyse auch eine Reduktion der Ein- und Ausgangsparameter.
- Durch die Hauptkomponententransformation des Datenraums auf Hyperellipsoide ist eine Überwachung der Gültigkeitsdomäne möglich.
- Schnelle Berechnung der Fourier-Transformation (engl. *Fast-Fourier-Transformation*) durch Ausnutzung von Symmetrien möglich.

Nachteile:

- Lineare Regression und Korrelation sind nur für Daten mit linearem Zusammenhang im vollem Umfang einsatzfähig.
- Die Analyse polynomialer oder exponentieller Zusammenhänge ist nur mit Vorkenntnis des Anwenders möglich.
- Die Verwendung der Hauptkomponentenanalyse erfordert eine Berechnung auch während der Anwendungsphase.
- Die Frequenzanalyse der Fourier-Transformation ist nur getrennt für Singnalein- und ausgang möglich, ohne Darstellung von möglichen Zusammenhängen im Frequenzbereich.

3.4.2 Gewichtsoptimierung nur für Spezialfälle

Bei der Auswahl von Anzahl, Lage und Wertigkeit der Gewichte künstlicher, neuronaler Netze muß sich der Anwender in der Regel auf seine Erfahrung und das Trainingsverfahren verlassen. Die Entwicklung neuer Methoden beschränkt sich meist auf eine Effizienzsteigerung von Trainingsverfahren. Zur Auswahl der Gewichtsanzahl können in der Literatur grobe Schätzungen gefunden werden, die jedoch keinen oder nur wenig Bezug zur Komplexität der Datensätze besitzen. In einzelnen, speziellen Fällen können auch sie zu guten Ergebnissen führen. Durch eine informationstheoretisch gestützte Auswahl der Netztopologie können folgende Nachteile behoben werden, wodurch jedoch der Vorteil des geringen Berechnungsaufwands verloren geht.

Vorteile:

- Keine Berechnungszeit beim Einsatz von Daumenregeln erforderlich.
- Regularisierung läßt sich durch einfache Überwachung der Gewichte während des Trainings realisieren.
- Trainingsüberwachung und Abbruch beim Erreichen "optimaler" Gewichte ohne großen Aufwand durchführbar.

Nachteile:

- Daumenregeln nehmen keinen oder nur unvollständigen Bezug auf die Komplexität der durch die Datensätze gegebenen Approximationsaufgabe.
- Eine Schätzung der Komplexität über die Dimension der Daten ist höchstens empirisch korrekt.

- Regularisierungsterme stützen sich auf Hypothesen aus heuristischen Überlegungen, die nicht für jede Anwendung Gültigkeit besitzen müssen.
- Beste Ergebnisse bei der Trainingsüberwachung sind nur durch Beobachtung und manuellen Eingriff des Anwenders erzielbar.
- Aufteilung der Datensätze ist intuitiv, wobei zusätzlich eine Verkleinerung der Trainingsmenge erfolgt.

3.4.3 Topologieoptimierung ohne Datenbezug

Die bekannten Verfahren zur Topologieoptimierung, (*Pruning*) und (*Cascade Correlation*) nehmen nur wenig Bezug auf die Beschaffenheit der Daten bzw. die Approximationsaufgabe. Sie stützen sich lediglich auf die Beobachtung des Restfehlerverhaltens beim Entfernen bzw. Hinzufügen von Neuronen. Neben den Vorteilen seien hier auch die Nachteile im Vergleich zu einer informationstheoretisch optimierten Netztopologie genannt.

Vorteile:

- *Pruning* kann durch Angabe einfacher Schwellen „schlankere“ Netzwerke erzeugen, die einen vorgegebenen Restfehler nicht übersteigen.
- *Cascade Correlation* beginnt mit einer kleinen Netztopologie und stoppt selbstständig beim Erreichen einer ausreichenden Größe.
- Beide Verfahren lassen sich kombinieren und können für bestimmte Datensätze kleine, robuste Netztopologien erzeugen.

Nachteile

- *Pruning*-Algorithmen erwecken den Anschein ein gutes Ergebnis nur durch Probieren vieler Kombinationen zu erzielen.
- Die Optimierung durch *Cascade Correlation* besteht im Wesentlichen durch die Bildung von Korrelationen bzw. Kovarianzen, deren Berechnung in der Regel einen linearen Charakter aufweist, was nicht für jeden Datensatz optimal ist.
- Beide Verfahren nehmen keinen Bezug auf den tatsächlichen Zusammenhang der Daten und somit auf die Komplexität der Approximationsaufgabe.

3.5 Zusammenfassung

In diesem Kapitel sind gängige Methoden zur Analyse, Spezifikation, Adaption und Diagnose von Daten bzw. neuronalen Netzen dargestellt worden. Neben einigen Vorteilen haben diese Methoden, insbesondere im Hinblick auf die Zielanwendung der Verbrennungsüberwachung, gravierende Nachteile. So sind sie für eine schnelle Berechnung optimiert und können schon nach kurzer Zeit ein befriedigendes Ergebnis liefern. Die hohe Geschwindigkeit wird dabei oft durch die Annahme erzielt, der Zusammenhang der Daten lasse sich linear approximieren. Diese Annahme trifft für Daten der angestrebten Überwachung nicht zu. Zwar lassen sich viele der dargestellten Methoden auf nichtlineare Fälle erweitern, jedoch muß dem Anwender dann der grundlegende Charakter der Zusammenhänge bekannt sein.

Aufbauend auf den theoretischen Grundlagen der in Kapitel 2.4 beschriebenen verallgemeinerten Transinformation lassen sich vielseitige Methoden zur Bewertung aller informationstheoretischen Zusammenhänge von Meßdaten entwickeln. Durch eine Kombination bestehender Methoden mit der verallgemeinerten Transinformation kann so eine Erweiterung geschaffen werden, die keine Begrenzung auf lineare, polynomiale oder exponentielle Zusammenhänge hat. Dies geschieht vollständig automatisch, ohne jegliche Vorkenntnis des Anwenders über die Beschaffenheit der Daten. Damit haben die in den folgenden Kapiteln beschriebenen Methoden einen großen Vorteil gegenüber allen Methoden, die sich auf angenäherte Zusammenhänge stützen, die vom Anwender vermutet oder durch zeitaufwendige Voruntersuchungen ermittelt wurden.

Kapitel 4

Anwendungsorientierte Methoden

Dieses Kapitel ist der Beschreibung neu entwickelter Methoden gewidmet, die auf der Basis der in Kapitel 2.4 beschriebenen, verallgemeinerten Transinformation entstanden sind. Bei ihrer Entwicklung lag der Schwerpunkt stets bei Anwendungen zur Datenauswertung von Motorparametern zur Verbrennungsüberwachung. Dies erfordert Verfahren, die es ermöglichen, schnelle und robuste Algorithmen zu erhalten. Für die Anwendung neuronaler Netze bedeutet dies, möglichst kleine, jedoch robuste Netztopologien zu ermitteln.

4.1 Problem- und Aufgabenbeschreibung

In den letzten Jahren haben sich insbesondere im Hinblick auf Verbrauch und geringen Schadstoffausstoß Dieselmotoren mit Direkteinspritzung bewährt. Bei dieser Technologie setzt die DAIMLERCHRYSLER AG auf ein System mit Druckspeicherung (engl. *Common-Rail*) und elektronischer Injektoreinspritzung. Bei Motoren mit solchen *Common-Rail*-Einspritzsystemen (CR-System) ist die zylinderindividuell und arbeitspielaufgelöste, tatsächlich eingespritzte Kraftstoffmenge eine Größe, deren Überwachung bei der Erfüllung der EURO III¹ Abgasnorm wünschenswert ist.

Um zukünftigen Anforderung gewachsen zu sein, konnte in den Jahren 1997 bis 1999 im Rahmen einer Dissertation bei der DAIMLERCHRYSLER AG ein Verfahren zur arbeitspielaufgelösten Überwachung der eingespritzten Kraftstoffmenge an *Common-Rail*-Systemen entwickelt werden (siehe [Köh97] und [Fro96]). Dieses Verfahren zeichnete sich durch einige positive Eigenschaften aus, bietet aber genügend Potential für notwendige Verbesserungen.

¹Durch die EURO III Norm werden seit Anfang des Jahres 2000 Ausstoßmengen von $0.05 \frac{g}{km}$ Rußpartikel und $0.6 \frac{g}{km}$ Stickoxide und Kohlenwasserstoffe gefordert.

Positive Eigenschaften:

- Durch die rein softwarebasierte Umsetzung ist keine weitere Sensorik am Fahrzeug notwendig.
- Die Modellierung des System mit Hilfe von künstlichen neuronalen Netzen ist in Grenzen² robust gegenüber Fertigungstoleranzen.
- Die tatsächlich eingespritzte Kraftstoffmenge kann in jedem Arbeitspunkt³ mit einer Abweichung von weniger als 2 mm^3 geschätzt werden.

Neben diesen durchaus positiven Eigenschaften hat das Verfahren jedoch auch einige Schwachstellen.

Negative Eigenschaften:

- Die variable, drehzahlabhängige Abtastung erfordert im oberen Drehzahlbereich sehr hohe Abtastfrequenzen, und somit hohe Anforderungen an die AD-Wandler.
- Durch die hohen Abtastfrequenzen ist die Eingangsdimension und somit die gesamte Netzstruktur für eine Umsetzung im Fahrzeug zu groß.
- Zum Training einer robusten Netzstruktur mit herkömmlichen Lernverfahren ist eine sehr große Anzahl von Trainingsdaten erforderlich.

Erst durch den Einsatz im Fahrzeug (engl. *onboard*) ist die wesentliche Forderung nach einer *Onboard*-Umsetzung der Euro III Abgasnorm erfüllt. Somit sind weitere Bestrebungen auf die Reduzierung der Größe der neuronalen Struktur zu konzentrieren.

Angestrebte Maßnahmen:

- Da die Abtastfrequenz maßgeblich an der Größe der Netztopologie beteiligt ist, sind Schritte bei der Signalanalyse und Vorverarbeitung notwendig, um schon auf diesem Weg die Eingangsdimensionalität des neuronalen Netzes zu reduzieren.
- Auch die Verkleinerung der gesamten Netztopologie ist von großem Interesse, da der Aufwand zur Berechnung im Wesentlichen proportional zum Produkt der Neuronenanzahl aller Schichten ist.
- Ein Ansatz zur Reduzierung der Anzahl notwendiger Datenpunkte ist für den Einsatz bei der vielseitigen Motorpalette wünschenswert. Im Ausblick der vorliegenden Arbeit wird ein entsprechender Vorschlag geliefert.

Um die angestrebten Maßnahmen zu erfüllen, sind neue Methoden entwickelt worden, die auf Basis der verallgemeinerten Transinformation künstliche, neuronale Netze für einen Einsatz im Fahrzeug optimieren.

²Fertigungsbedingte Toleranzen der Injektoren mit Abweichungen der Einspritzmenge von bis zu 3% können abgedeckt werden.

³Die Bandbreite der Einspritzmengen erstreckt sich von ca. $0.5 - 70\text{ mm}^3$.

4.2 Informationsoptimierte Frequenzanalyse

Unter *informationsoptimierter* Frequenzanalyse ist eine Darstellung von Frequenzspektren mit informationstheoretischer Gewichtung einzelner Frequenzbänder zu verstehen. Sie ist als zusätzliches Tool zu bestehenden Methoden wie Fast-Fourier und Short-Time-Fourier-Analyse anzusehen. Durch Analyse statistischer Abhängigkeiten von frequenzbandbegrenzenden Teilsignalen ermöglicht sie eine Klassifikation von Frequenzlinien bezüglich ihrer Relevanz für das Gesamtsystem.

Die Methode beruht auf der in Kapitel 2, Abschnitt 2.4.4 vorgestellten Schätzung der verallgemeinerten Transinformation und ist somit statistischen Schwankungen unterworfen. Dies macht sich besonders bei kurzen Zeitreihen in der Qualität der Auflösung und somit bei der Trennung von informationstheoretisch relevanten Frequenzen bemerkbar.

4.2.1 Invarianz gegenüber Phasenverschiebungen

Eine wichtige Eigenschaft der Transinformation liegt in ihrer Invarianz gegenüber Phasenverschiebungen, da viele reale Systeme solche Verschiebungen verursachen. Schon einfache Filter verursachen Phasenverschiebungen, die einen nichtlinearen Systemzusammenhang verursachen. Damit ist diese Eigenschaft besonders wichtig für die *informationsoptimierte* Frequenzanalyse, da sie einen intensiven Einsatz von Bandpaßfiltern erfordert.

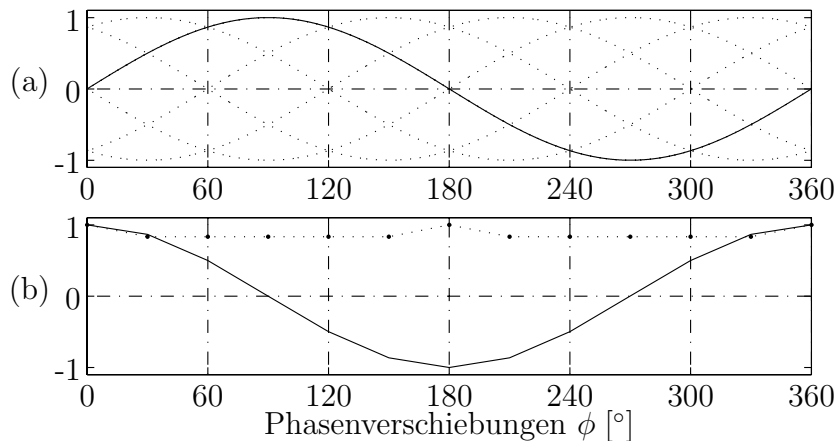


Abbildung 4.1: Phaseninvarianz der verallgemeinerten Transinformation.

- (a) Um je 60° phasenverschobene Sinus-Signale mit Amplituden $[-1, 1]$.
- (b) Gegenüberstellung der Phasenabhängigkeit des linearen Korrelationskoeffizienten (Linie) und der Transinformation (Punkte).

Abbildung 4.1 veranschaulicht die Invarianz der Transformierung gegenüber diversen Phasenverschiebungen. Unter (a) sind Sinus-Schwingungen mit unterschiedlichen Phasenverschiebungen dargestellt. Die Kurve unter (b) zeigt die Werte der Korrelationskoeffizienten des phasenfreien Sinus und der Phasenverschobenen Sinus-Schwingungen. Für die Phasen 0° und 360° beträgt der Koeffizient Eins, für 180° jedoch minus Eins. Alle anderen Phasenverschiebungen liefern Werte im Intervall $] -1, 1[$. Die Punkte im Werte-Bereich auf Höhe der Eins stellen die Ergebnisse der verallgemeinerten Transformation dar. Sie bleiben für alle Phasenverschiebungen nahezu konstant Eins.

4.2.2 Frequenzbandzerlegung

Aufbauend auf der in Kapitel 2, Abschnitt 2.3.5 beschriebenen Oktavbandzerlegung kann durch Anwendung einer Rechteck-Funktion eine Frequenzbandzerlegung erzielt werden. Die Zerlegung in solche Frequenzbänder und nicht in Oktavbänder hat für die *informationsoptimierte* Frequenzanalyse folgende Vorteile:

- Im Gegensatz zur Oktavbandzerlegung wird die eindeutige Zuordnung zu Frequenzbereichen möglich.
- Der Energie- und somit auch der Informationsverlust verhält sich bei exakter Teilung des Spektrums quadratisch.
- Die gewählte Fensterfunktion kann unverändert bei der Zusammensetzung der Frequenzanteile verwendet werden.

Im folgenden wird die Frequenzbandzerlegung beschrieben, wodurch die Unterschiede zu der in Kapitel 2, Abschnitt 2.3.5 dargestellten Oktavbandzerlegung deutlich werden. Abbildung 4.2 veranschaulicht die Zerlegung in Frequenzbereiche mit Hilfe von Rechteck-Fenstern. Bei der Verwendung anderer Fensterfunktionen⁴ ist darauf zu achten, daß die Summe der Funktionswerte der Fensterfunktion über den gesamten Frequenzbereich Eins ergibt. Für Rechteck-Fenster ergeben sich im Frequenzbereich $f \in \{0, \dots, F\}$ folgende Gleichungen⁵:

$$R_n(f) = \text{rect}\left(\frac{\varphi_n}{k}\right) \quad ; \quad \text{mit } \varphi_n = f - f_n$$

$$\sum_{n=1}^u R_n(f) = 1 \quad ; \quad \text{für alle } f \in \{0, \dots, F\} \quad (4.1)$$

mit

$$f_n = \pm(n-1) \cdot \frac{F}{2^{u+1}}; \quad k = \frac{F}{2^u}; \quad n = \{2^1, 2^2, \dots, 2^{u+1}\} \text{ und } u \in \mathbb{N}$$

⁴Bei der Verwendung von z.B. Hanning- bzw. Dreieck-Fenstern ist bei der Rekonstruktion die Kompensation des Energieverlustes zu adaptieren (siehe Abschnitt 4.2.3).

⁵Dabei ist die Zeitachse durch die Frequenzachse und somit t durch f zu ersetzen

Dabei drückt k die Breite der Fenster aus, die von der Anzahl u an Unterteilungen des Frequenzbereichs $f \in \{0, \dots, F\}$ abhängt. Die Verschiebungen φ legen die Fenster so, daß ihre Summe über dem definierten Frequenzbereich Eins ergibt.

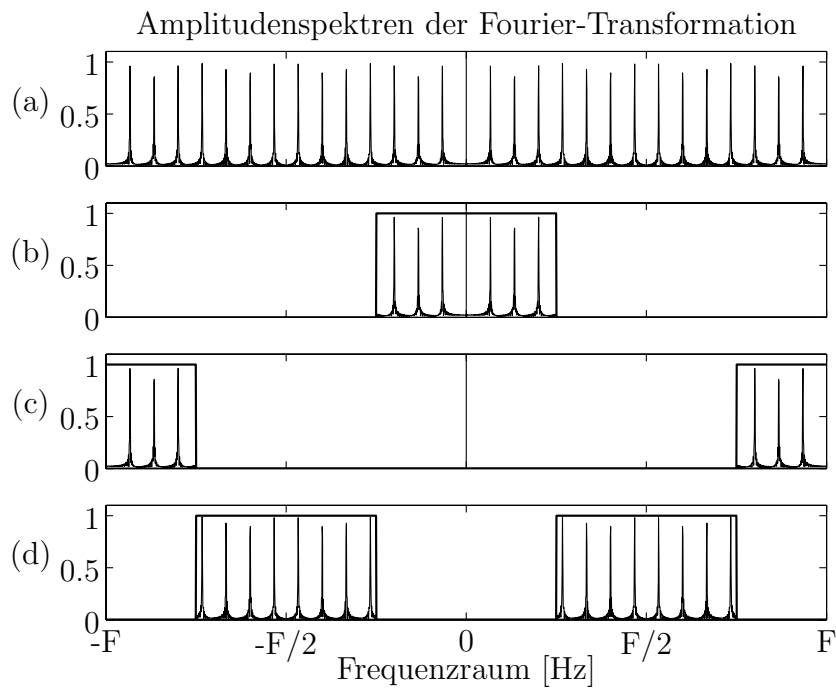


Abbildung 4.2: Frequenzbandzerlegung überlagerter Sinus-Schwingungen. Überlagerte Sinus-Schwingungen mit regelmäßig auftretenden Frequenzanteilen, zerlegt in verschiedene Frequenzbereiche. Zur Rekonstruktion kann das Originalsignal (a) aus dem tiefpaßgefilterten Signal (b), dem hochpaßgefilterten Signal (c) und dem bandpaßgefilterten Signal (d) zusammengesetzt werden.

Abbildung 4.2 zeigt unter (a) das normierte Frequenzspektrum überlagerter Sinus-Schwingungen, deren Frequenzen gleichmäßig auf das Spektrum verteilt sind. Die Darstellung unter (b) verdeutlicht die Anwendung der Rechteck-Funktion, um eine Fensterung der tiefen Frequenzen zu realisieren. Unter (c) ist der korrespondierende Ausschnitt hoher Frequenzen zu erkennen. Durch weitere Unterteilungen entstehen Bänder tiefer, hoher und mittlerer Frequenzen. Unter (d) sind nur die Bereiche mittlerer Frequenzen dargestellt. Bei der *informationsoptimierten* Frequenzanalyse werden diese Frequenzbänder mit jeder Iteration schmaler gewählt. Die Summe aller Teilsignale ergibt dabei jeweils das Originalsignal.

Durch sukzessive Unterteilung des Frequenzspektrums, unter Verwendung der Rechteck-Funktion, entstehen nach Rücktransformation aus dem Frequenzbereich ideale tief-, band- und hochpaßgefilterte Zeitsignale. Die Frequenzbandzerlegung kann durch folgenden Algorithmus erzielt werden:

Algorithmus 4.2.1 (Frequenzbandzerlegung)

Der Algorithmus liefert für eine Zeitreihe $s(t)$ eine Matrix $X(t)$ von tief-, band- und hochpaßgefilterten Zeitreihen. Dabei wird der Frequenzraum durch Halbierung in $u + 1$ gleiche Teile zerlegt. Die Filterung erfolgt durch Multiplikationen mit Rechteck-Funktionen im Frequenzbereich.

Eingangsparameter:

- Zeitreihe eines Signals $s(t) = \{s_t\}_{t=1}^T$, wobei $\log_2(T) \in \mathbb{N}$.
- Anzahl gewünschter Unterteilungen $u < \log_2(T)$.

Algorithmus:

1. Falls $\log_2(T) \notin \mathbb{N}$ kürze $s(t)$, so daß $\log_2(T) \in \mathbb{N}$.
2. Prüfe ob $u < \log_2(T)$, sonst $u = \log_2(T) - 1$.
3. Bilde Fourier-Transformierte der Zeitreihe $s(t) \circ \bullet S(f)$.
4. Bestimme Matrix an Rechteck-Funktionen im Frequenzbereich $R_n(f) = \text{rect}(\frac{\varphi_n}{k})$, mit $\varphi_n = f - f_n$; $f_n = \pm(n-1) \cdot \frac{F}{2^{u+1}}$; $k = \frac{F}{2^u}$; $n = \{2^1, 2^2, \dots, 2^{u+1}\}$.
5. Bilde Rücktransformationen $S(f) \cdot R_n(f) \bullet \circ s(t) * r(t)$.

Ausgangsparameter:

- Matrix der gefilterten Zeitreihen $X_n(t) = \{s(t) * r_n(t)\}$ mit $n = \{2^1, 2^2, \dots, 2^{u+1}\}$.
- Matrix der verwendeten Rechteck-Funktionen $R_n(f) = \text{rect}(\frac{\varphi_n}{k})$.

Beschreibung:

Nach der Prüfung der Eingangsparameter erfolgt eine Fourier-Transformation des Signals $s(t)$ aus dem Zeit- in den Frequenzbereich. Die Filterung kann gemäß Multiplikationstheorem (siehe Kapitel 2, Abschnitt 2.3.4) durch einfache Multiplikation mit Rechteck-Funktionen entsprechend Gleichung 4.1 erfolgen. Durch die Rücktransformation in den Zeitbereich können die gefilterten Zeitreihen als Matrix $X(t)$ angegeben werden.

Beispiel 4.2.1 (Frequenzbandzerlegung)

Durch Abbildung 4.3 ist das Ergebnis für ein Chirp-Signal im Zeitbereich dargestellt. Unter (a) ist das Chirp-Signal zu sehen, das die Frequenzen von Null bis zur Hälfte der Abtastfrequenz durchläuft. Damit ist das Abtasttheorem erfüllt und die Frequenzbandzerlegung kann angewendet werden. In (b) und (c) sind die ersten zwei Zerlegungen zu erkennen.

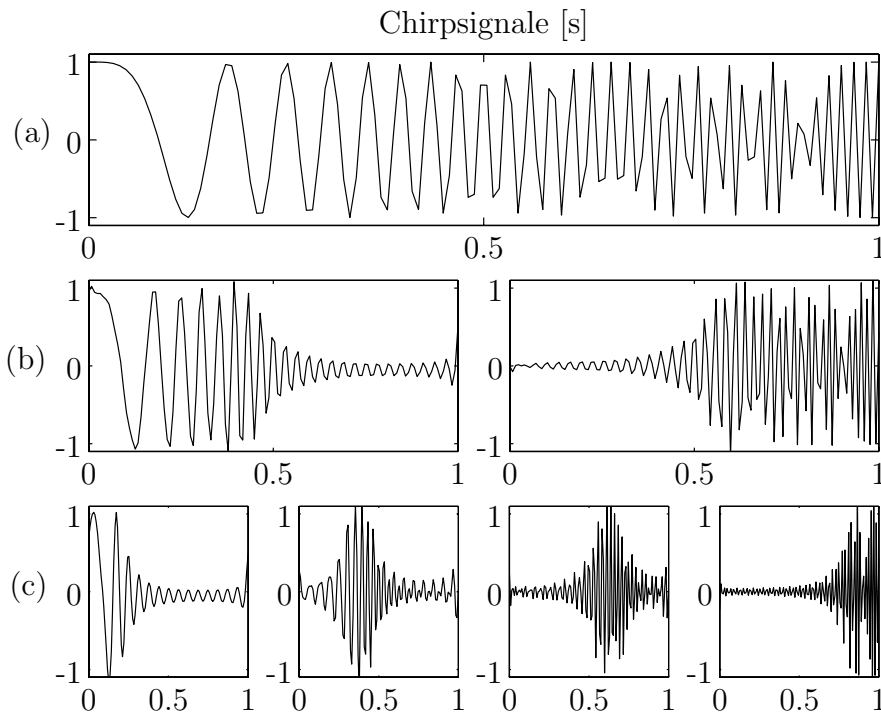


Abbildung 4.3: Signalzerlegung durch Frequenzbandzerlegung.

Unter (a) ist ein *Chirp*-Signal über den gesamten Frequenzbereich dargestellt. Die ersten beiden Frequenzbandteilungen sind unter (b) und (c) angedeutet.

Auf Grund des Superpositionsprinzips ergibt die Überlagerung der vier Zeitsignale aus (c) wieder das originale Chirp-Signal. Dies ist möglich, da die Zerlegung in die Frequenzbänder mit Hilfe von Rechteck-Fenstern erfolgt, deren Summe gemäß Gleichung 4.1 über dem gesamten Frequenzbereich Eins ergibt.

Die Zerlegung in Bänder durch Rechteck-Funktionen hat auch bei der Rekonstruktion der Frequenzbereiche unter Verwendung von Informationsanteilen der verallgemeinerten Transformation Vorteile, da sie eine schärfere Abgrenzung der Frequenzbereiche ermöglichen. Steile Filterflanken verursachen zwar große Phasenverschiebungen, die jedoch nur einen geringen Einfluß auf das verwendete Informationsmaß haben (siehe Abschnitt 4.2.1).

4.2.3 Rekonstruktion der Frequenzbänder

Unter Verwendung der Frequenzbandzerlegung kann mit Hilfe der verallgemeinerten Transinformation eine informationstheoretische Bewertung der Frequenzbänder angegeben werden. Der folgende Algorithmus ist das Kernstück der *informationsoptimierten* Frequenzanalyse.

Algorithmus 4.2.2 (Informationsoptimierte Frequenzanalyse)

Der Algorithmus liefert für eine Zeitreihe $x(t)$, bezogen auf eine weitere Zeitreihe $y(t)$, Graphen $G_u(f)$ informationstheoretisch gewichteter Frequenzbänder. Zur Teilung in Frequenzbänder wird das Signal $x(t)$ mit Hilfe des Algorithmus 4.2.1 hoch-, tief- und bandpaßgefiltert. Der Informationsgehalt der gefilterten Signale bezogen auf $y(t)$ wird mit Hilfe der verallgemeinerten Transinformation ermittelt.

Eingabeparameter:

- Meßreihe $x(t) = \{x_t\}_{t=1}^T$ des Systemeingangs, wobei $\log_2(T) \in \mathbb{N}$.
- Parallel aufgezeichnete Meßreihe $y(t) = \{y_t\}_{t=1}^T$ des Systemausgangs.
- Die Belegungsdichte $\text{proz} \in [0, 1]$ als indirekte Angabe der Suchboxgröße.

Algorithmus:

1. Prüfe, ob $\log_2(T) \in \mathbb{N}$ für die Zeitreihen $x(t)$ und $y(t)$ gilt.
2. Setze maximal mögliche Anzahl an Unterteilungen $u = \log_2(T) - 1$.
3. Bilde Rangfolge der Zeitreihe $y(t)$.
4. Bestimme iterativ geeignetes Epsilon ϵ , so daß $H_{2,\epsilon}(y) = \log_2(\text{proz})$.
5. Zerlege $x(t)$ durch u -fache Frequenzzerlegung mit Algorithmus 4.2.1
 $X_n(t) = \{x(t) * r_n(t)\}$ mit $n = \{2^1, 2^2, \dots, 2^{u+1}\}$.
6. Bilde spaltenweise Rangfolgen der Matrix $X_n(t)$.
7. Berechne die Transinformationsfunktion
 $I_{2,\epsilon}(n) = I_{2,\epsilon}(X_n(t), y(t))$ mit $n = \{2^1, 2^2, \dots, 2^{u+1}\}$.
8. Bilde informationstheoretisch gewichteten Graphen $G(f)$ der Frequenzen
 $f = \{0 \dots F\}$
 $G_u(f) = I_{2,\epsilon}(n) \cdot R_n(f)$ mit $n = \{2^1, 2^2, \dots, 2^{u+1}\}$.
9. Wiederhole ab 5. falls $u := u - 1 \geq 1$.

Ausgabeparameter:

- Matrix der Graphen $G_u(f)$ mit informationstheoretisch gewichteten Frequenzbändern.

Beschreibung:

Haben die Zeitreihen $x(t)$ und $y(t)$ die gleiche Länge, kann die Anzahl u maximal möglicher Unterteilungen bestimmt werden. Da die Zeitreihe $y(t)$ über die gesamte Berechnung konstant bleibt, kann ihre Rangfolge und ein auf die Daten angepaßtes Epsilon getrennt bestimmt werden. Nach Zerlegung der Zeitreihe $x(t)$ in $2^u + 1$ frequenzbandgefilterte Signale $X(t)$ werden deren Rangfolgen gebildet. Ist die Transinformationsfunktion zwischen $X(t)$ und $y(t)$ bestimmt, kann mit Hilfe der bei der Frequenzbandzerlegung verwendeten Fenster $R(f)$ der Graph $G(f)$ der *informationsoptimierten* Frequenzanalyse angegeben werden. Dieses Vorgehen wird für alle Teilungsverhältnisse des Frequenzraums F wiederholt.

Die Anwendung der verallgemeinerten Transinformation zum Auffinden relevanter Frequenzen aus bandpaßgefilterten Zeitsignalen ist im Wesentlichen aus folgenden Gründen möglich:

- Die verallgemeinerte Transinformation findet alle statistischen Abhängigkeiten.
- Phasenverschiebungen gefilterter Signale stellen keine Einschränkung dar.

Ein Problem des hier beschriebenen Algorithmus zur Frequenzanalyse auf Basis einer iterativen Frequenzbandhalbierung ist ein mit jeder Halbierung fortschreitender Informationsverlust. Wie der Abbildung 4.4 (b) des Beispiels 4.2.2 zu entnehmen ist, folgt der Informationsverlust dem Verlauf

$$G_{\text{Verlust}}(u) = \min_{\substack{0 \leq f \leq F \\ 1 \leq u \leq U}} (G_u(f)) + \frac{1 - \min_{\substack{0 \leq f \leq F \\ 1 \leq u \leq U}} (G_u(f))}{2^u} \quad (4.2)$$

wobei $u = \{0, 1, \dots, U\}$ die Anzahl der Spektralhalbierungen angibt. Da die Schätzung der Transinformation auf der Auswertung der Entropie zwischen Signalen beruht, was auch als Schätzung der wechselseitigen Energie angesehen werden kann, deutet dieser Kurvenverlauf auf eine sukzessive Energiehalbierung hin. Das Ergebnis der Fourier-Transformation kann als Energiedichtespektrum interpretiert werden. Somit ist die Energiehalbierung verständlich, da durch die Frequenzbandzerlegung das Spektrum mit jeder Iteration halbiert wird, wodurch ein Kurvenverlauf von $\frac{1}{2^u}$ zu erwarten ist. Da jedoch durch das Schätzverfahren immer eine Restinformation gefunden wird, muß die Kurve um den Minimalanteil $\min(G_u(f))$ korrigiert werden.

Beispiel 4.2.2 (Informationsverlust)

Die in Beispiel 4.2.1 dargestellte Chirp-Funktion wird verwendet, um eine einschränkende Eigenschaft der Informationsanalyse aus Frequenzbändern zu veranschaulichen. Dazu wird dem Algorithmus 4.2.2 das Chirp-Signal sowohl als Systemeingang, als auch -ausgang übergeben. Zu erwarten ist, daß jede Frequenz informationstheoretisch den gleichen Anteil trägt.

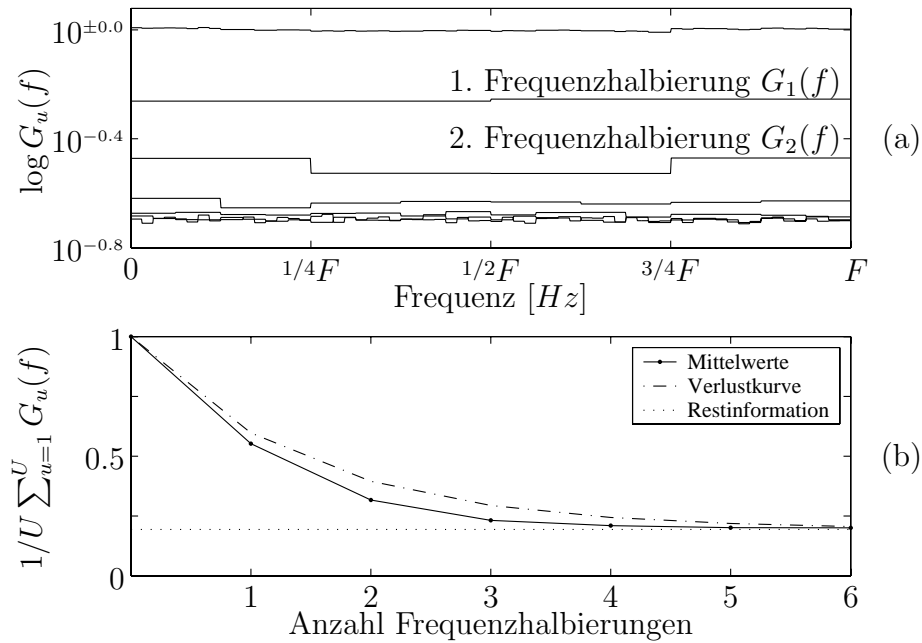


Abbildung 4.4: Informationsverlust durch Frequenzbandteilungen.

- (a) Mit jeder Frequenzbandhalbierung sinkt der mittlere Informationsgehalt zwischen Ein- und Ausgabesignal.
- (b) Über alle Frequenzen gemittelter Informationsverlust durch sukzessives Halbieren des Frequenzbereichs.

In Abbildung 4.4 (a) ist zu erkennen, daß die erwartete Gleichverteilung der Frequenzanteile eintritt und im statistischen Mittel jede Frequenz die gleiche Information trägt. Ebenfalls zu erkennen ist, daß mit jeder Halbierung des Frequenzbereichs der Informationsgehalt zwischen Ein- und Ausgabesignal sinkt. Unter (b) sind die gemittelten Informationsgehalte für verschiedene Unterteilungen aufgetragen. Der Kurvenverlauf deutet auf einen ständigen Informationsverlust durch Halbieren des Frequenzspektrums hin.

Durch Korrektur der Informationsanteile mit dem Kehrwert der Funktion 4.2 kann der Energieverlust korrigiert werden. Die Gleichung

$$G(f) = \frac{1}{U} \sum_{u=1}^U \tilde{G}_u(f); \quad \text{mit} \quad \tilde{G}_u(f) = G_u(f) \cdot \frac{2^u}{1 + \min_{\substack{0 \leq f \leq F \\ 1 \leq u \leq U}} (G_u(f))(2^u - 1)} \quad (4.3)$$

liefert das erwartete Ergebnis eines maximalen, konstanten Informationsgehaltes über den gesamten Frequenzbereich. Dieses Ergebnis ist in Abbildung 4.4 (a) als Linie bei $10^{\pm 0.0} = 1$ zu erkennen. Durch den statistischen Charakter der Transinformationsauswertung kann natürlich keine glatte Kurve erwartet werden.

4.2.4 Bewertung informationsoptimierte Frequenzanalyse

Um den Stellenwert der *informationsoptimierten* Frequenzanalyse gegenüber der Fourier-Frequenzanalyse darzustellen, werden im Folgenden zwei Beispiele diskutiert.

Beispiel 4.2.3 (Systemanalyse)

Für ein nicht bekanntes System existiere nur eine Messung am Systemeingang $x(t)$ und am Systemausgang $y(t)$. Aus meßtechnischen Gründen sei $x(t)$ tiefpaßgefiltert. Für das hier beschriebene Beispiel besteht die Messung des Systemeingangs aus einem Chirp-Signal mit Frequenzen von 1 – 5000 Hz.

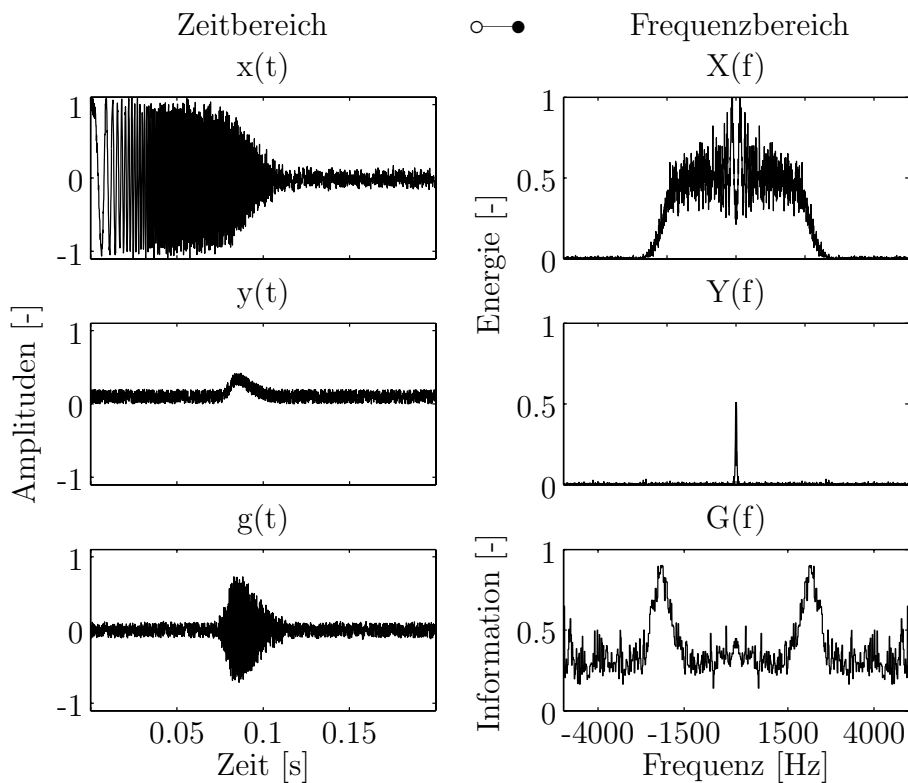


Abbildung 4.5: Frequenzanalyse von Systemeingängen und -ausgängen.

Darstellung des Systemeingangs $x(t)$, des Systemausgangs $y(t)$, sowie des Ausgangs der Transinformativanalyse $g(t)$ des Systemzusammenhangs der Abbildung 4.6. Auf der linken Seite sind jeweils die Zeitsignale dargestellt, auf der Rechten die Frequenzspektren.

Auf die Frage, ob alle Frequenzen des Systemeingangs für die gemessene Systemantwort notwendig sind, kann eine gewöhnliche Fourier-Frequenzanalyse keinen Aufschluß geben. Diese Aussage belegt Abbildung 4.5, in der neben den Zeitreihen $x(t)$ und $y(t)$ auch die entsprechenden normierten Fourier-Spektren $X(f)$ und $Y(f)$ dargestellt sind.

Die informationsoptimierte Frequenzanalyse des Algorithmus 4.2.2 kann die gewünschte Information liefern. Dazu dienen lediglich das Eingangssignal $x(t)$ und das Ausgangssignal $y(t)$ als Eingabe für den Algorithmus zur Frequenzanalyse. Zusammen mit der informationsoptimierten Gewichtung der Frequenzlinien ergibt sich ein abgerundetes Bild des Systemverhaltens bezogen auf das Frequenzspektrum.

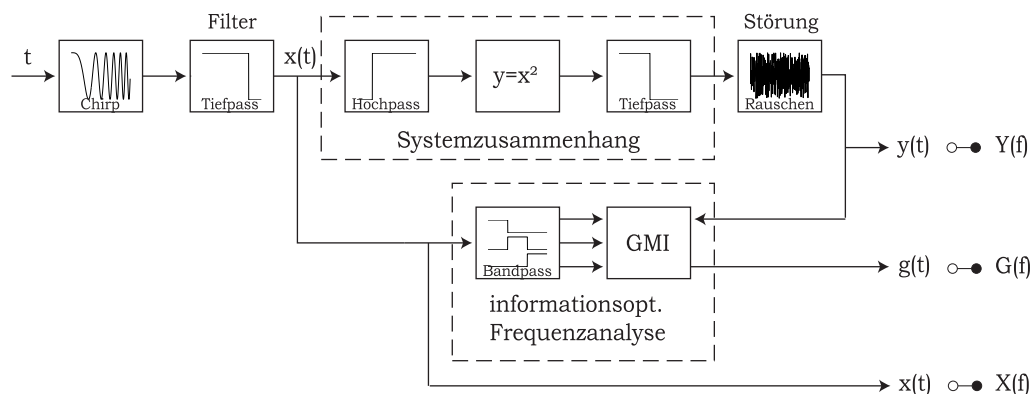


Abbildung 4.6: Informationsoptimierte Systemanalyse.

Durch die Kombination aus Tief- und Hochpaßfilter am Eingang und die Signalquadrierung erzeugt das System die Einhüllende eines bandpaßgefilterten *Chirp*-Signals. Das System benötigt als Signaleingang $x(t)$ also nur Frequenzen aus dem mittleren Frequenzbereich, um den gleichen Signal- ausgang $y(t)$ zu erzeugen. Diese Eigenschaft kann den Spektren $X(f)$ und $Y(f)$ nicht entnommen werden, jedoch dem *informationsoptimierten* Frequenzspektrum $G(f)$ (siehe Abbildung 4.5).

Eine Antwort auf die zu Beginn gestellte Frage kann mit Hilfe des Graphen $G(f)$ in Abbildung 4.5 gegeben werden. Ein entsprechend bandbegrenztetes Signal $g(t)$ am Systemeingang genügt, um die gewünschte Systemantwort $y(t)$ zu erhalten.

Beispiel 4.2.3 veranschaulicht, wie durch die *informationsoptimierte* Gewichtung von Frequenzspektren zusätzliche Informationen über ein System gewonnen werden. Der tatsächliche Systemzusammenhang ist in Abbildung 4.6 dargestellt. Das System ermittelt nach einer Hochpaßfilterung die Einhüllenden des bandbegrenzten Signals von $x(t)$. Dazu wird das Signal quadriert und der Frequenzbereich durch Tiefpaßfilterung halbiert. Die Bandbegrenzung durch den Tiefpaßfilter am Systemeingang kann durch die *informationsoptimierte* Frequenzanalyse ermittelt werden.

Das folgende Beispiel demonstriert mit welcher Auflösung das Verfahrens einzelne Frequenzlinien isolieren bzw. trennen kann.

Beispiel 4.2.4 (Frequenzauflösung)

Die Zeitreihe des Systemeingangs $x(t)$ der Abbildung 4.7 ist aus folgenden Sinusschwingungen mit 31 Hz, 53 Hz und 71 Hz konstruiert.

$$\left. \begin{aligned} s_1(t) &= \sin(2\pi \cdot t \cdot 31) \\ s_2(t) &= \sin(2\pi \cdot t \cdot 53) \\ s_3(t) &= \sin(2\pi \cdot t \cdot 71) \end{aligned} \right\} x(t) = 3 \cdot s_1(t) + s_2(t) + 2 \cdot s_3(t)$$

Die Ausgangssignale $y_1(t)$ und $y_2(t)$ entstehen durch Systeme mit folgenden Abbildungsvorschriften.

$$\begin{aligned} y_1(t) &= s_2(t)^2 \\ y_2(t) &= \frac{s_1(t)}{1 - s_3(t)} \end{aligned}$$

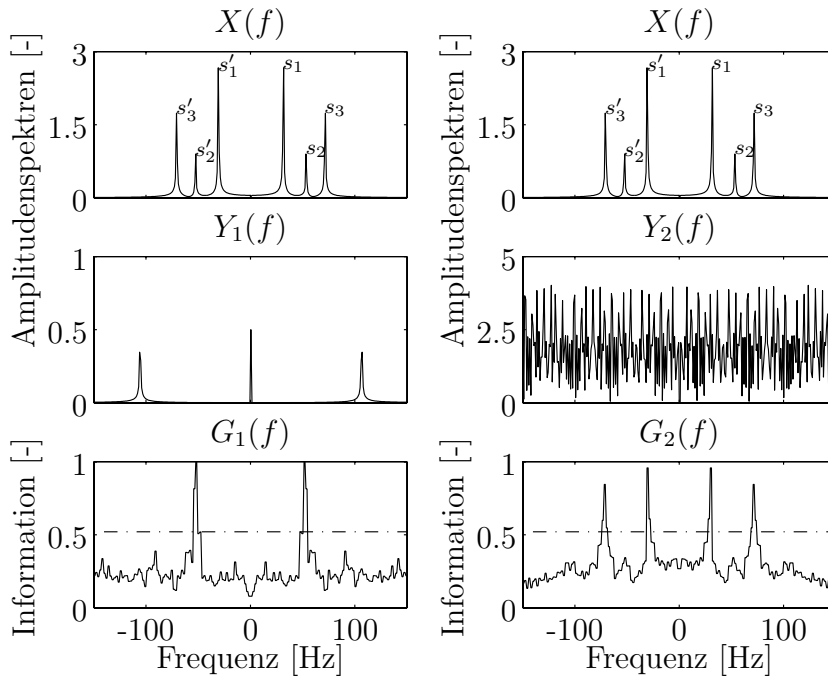


Abbildung 4.7: Auflösung informationsoptimierte Frequenzanalyse.

Gegenüberstellung der herkömmlichen Fourier-Frequenzanalyse und der *informationsoptimierten* Frequenzanalyse des Eingangssignals $x(t)$ und der Ausgangssignale $y_1(t)$ und $y_2(t)$. Es ist gut zu erkennen, daß nur die *informationsoptimierten* Spektren $G_1(f)$ und $G_2(f)$ Rückschluß auf die relevanten Frequenzanteile geben.

Beide Systeme verwirklichen nichtlineare Funktionen, wobei die Systemantwort $y_2(t)$ zusätzlich aus den Anteilen $s_1(t)$ und $s_3(t)$ gebildet wird.

Abbildung 4.7 stellt Amplitudenspektren der Fourier-Transformation des Eingangs $x(t)$ und der Systemantworten $y_1(t)$ und $y_2(t)$ gegenüber. Aus diesen Spektren kann nicht erkannt werden, welches der Signale $s_1(t), s_2(t)$ bzw. $s_3(t)$ Anteile an den jeweiligen Systemantworten trägt. Erst mit Hilfe der informationsoptimierten Frequenzanalyse werden die relevanten Signalanteile des Eingangs $x(t)$ herausgestellt.

Bewertungshilfe:

Die gestrichelte Linie in Abbildung 4.7 dient der Bewertung der Graphen $G_1(f)$ bzw. $G_2(f)$. Sie gibt an, ab welchem Informationsgehalt die gefundenen Frequenzanteile über dem mittleren Informationsgehalt aller genormten Teilgraphen $\tilde{G}_u(f)$ der Spektralhalbierungen liegen.

$$\begin{aligned}
 G_{min}(u) &= \min_{0 \leq f \leq F} (\tilde{G}_u(f)) \\
 G_{max}(u) &= \max_{0 \leq f \leq F} (\tilde{G}_u(f)) \\
 \text{Bewertung} &= \sum_{u=1}^U G_{min}(u) + G_{max}(u)
 \end{aligned} \tag{4.4}$$

Durch die Beispiele 4.2.3 und 4.2.4 konnte gezeigt werden, wie die *informationsoptimierte* Frequenzanalyse unter Verwendung der verallgemeinerten Transinformation zusätzliche Information zum gewöhnlichen Spektrum der Fourier-Analyse liefern kann. Dabei ist jedoch immer zu bedenken, daß die lange Berechnungszeit der *informationsoptimierten* Frequenzanalyse durch den Nutzen dieser Analyse gerechtfertigt sein sollte.

4.3 Informationsoptimierte Eingangsdimension

In Kapitel 3 sind verschiedene Methoden zur Optimierung der Netzwerktopologie dargestellt worden. Die Methode zur informationsoptimierten Eingangsdimension basiert auf der in Kapitel 2, Abschnitt 2.4 beschriebenen verallgemeinerten Transinformation und dient der informationstheoretisch gestützten Auswahl relevanter Eingangsvektoren.

4.3.1 Mehrdimensionale Transinformation

Zur Reduzierung der Eingangsdimensionalität ist eine mehrdimensionale Anwendung der Schätzung zur verallgemeinerten Transinformation erforderlich. Ein- und Ausgangsmatrizen von Meßreihen können durch folgenden Algorithmus auf informationstheoretische Zusammenhänge analysiert werden. Als zusätzliche Eingangsparameter sind lediglich zwei weitere Angaben erforderlich. Die prozentuale Belegung der Binärmatrizen als indirektes Maß für die Größe der Suchboxen und der Grad der zu erreichenden Gesamtinformation in Prozent als Abbruchkriterium. Eine ähnliche Vorgehensweise ist bei [Sob99] zu finden.

Algorithmus 4.3.1 (Informationsoptimierte Eingangsdimension)

Der Algorithmus liefert für Meßreihenpaare X und Y relevante Positionen für die Zeitreihen aus X , an denen die statistische Abhängigkeit von Y maximal ist. Die Schätzung der statistischen Abhängigkeit erfolgt auf Grund der verallgemeinerten Transinformation $I_2(X, Y)$.

Eingangsparameter:

- Die Eingangsmatrix $X(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}_{t=1}^{T_1}$ aus Spaltenvektoren $x(t)$ der einzelnen Messungen am Systemeingang. Dabei ist N die Anzahl der Messungen und T_1 die Länge der Zeitreihen des Systemeingangs bzw. bisherige Dimension des Netzeingangs.
- Die Ausgangsmatrix $Y(t) = \{y_1(t), y_2(t), \dots, y_N(t)\}_{t=1}^{T_2}$ aus Spaltenvektoren $y(t)$ der einzelnen Messungen am Systemausgang. Dabei ist T_2 die Länge der Zeitreihen des Systemausgangs bzw. Dimension des Netzausgangs.
- Die Belegungsdichte $proz \in [0, 1]$ als indirektes Maß der Suchboxgröße.
- Die Abbruchbedingung $grad \in [0, 1]$ als Angabe der zu erreichenden Gesamtinformation.

Algorithmus:

1. Initialisiere $Xmax = \{\emptyset\}$ und $pos = \{\emptyset\}$.
2. Prüfe Dimensionen von $X(t)$ und $Y(t)$.
3. Setze $proz = 0.05$ und $grad = 0.8$, falls nicht vorhanden.
4. Bilde zeilenweise Rangfolgen der Matrizen $X(t)$ und $Y(t)$.
5. Bestimme iterativ geeignetes ϵ , so daß
 $H_2(Xmax, Y) = \log_2(proz)$.
6. Berechne die Transinformationsfunktion
 $I(t, \epsilon) = I_2(\{Xmax, X(t)\}, Y)$ mit $t = \{1, 2, \dots, T_1\}$.
7. Erweitere $Xmax = \{Xmax, X(m)\}$, wobei
 $I(m, \epsilon) = \max\{I(t, \epsilon)\}_{t=1}^{T_1}$ für $m \in \{1, 2, \dots, T_1\}$.
8. Speichere gefundene Position $pos = \{pos, m\}$.
9. Wiederhole ab 5. falls $\max\{I(t, \epsilon)\}_{t=1}^{T_1} \leq grad$.

Ausgabeparameter:

- Vektor pos der Positionen maximaler Information.

Beschreibung:

Nach der Initialisierung der Matrix $Xmax$ und des Ausgabevektors pos erfolgt eine kurze Prüfung auf Erfüllung der Voraussetzungen. Sind die Belegungsdichte und das Abbruchkriterium nicht angegeben, werden Standardwerte gesetzt. Der Algorithmus betrachtet jeweils den Schnitt über alle Meßreihen, weshalb die Rangfolgenbildung zeilenweise geschieht. Die iterative Suche nach einem geeigneten ϵ erfolgt entsprechend Abschnitt 2.4.4 unter Betrachtung aller bisher gefundenen Maximalvektoren $Xmax$ und der Ausgangsmatrix $Y(t)$. Die Transinformationsfunktion bei nur einem Ausgang besteht aus den geschätzten Transinformationen für jeden Zeilenvektor der Eingangsmatrix $X(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}_{t=1}^T$ und dem Ausgangsvektor Y . Bei nur einem Ausgang ergibt sich für die 1. Iteration:

$$I(t) = I_2(\{x_1(t), x_2(t), \dots, x_N(t)\}_{t=1}^T, \{y_1, y_2, \dots, y_N\})$$

Sei m die Position des Maximums $I(m) = \max\{I(t)\}_{t=1}^T$ der ersten Iteration, dann ergibt sich die Transinformationsfunktion der 2. Iteration aus:

$$I(t) = I_2(\{x_1(m), \dots, x_N(m), x_1(t), \dots, x_N(t)\}_{t=1}^T, \{y_1, y_2, \dots, y_N\})$$

Für jede weitere Iteration kommt ein weiterer Vektor $X(m)$ für die Berechnung hinzu. Die gefundenen Positionen m werden als Vektor pos nach Erreichen einer maximalen Gesamtinformation ausgegeben.

4.3.2 Bewertung der Dimensionsreduzierung

Folgendes Beispiels soll das Vorgehen anhand einer Brennraumdruckanalyse veranschaulichen und als Grundlage einer Bewertung der neuen Methode dienen.

Beispiel 4.3.1 (Brennraumdruckanalyse)

Gesucht sind die Positionen des Brennraumdrucks mit maximalem, statistischem Zusammenhang zur Verschiebung des Druckmaximums in Bezug auf den Zündzeitpunkt.

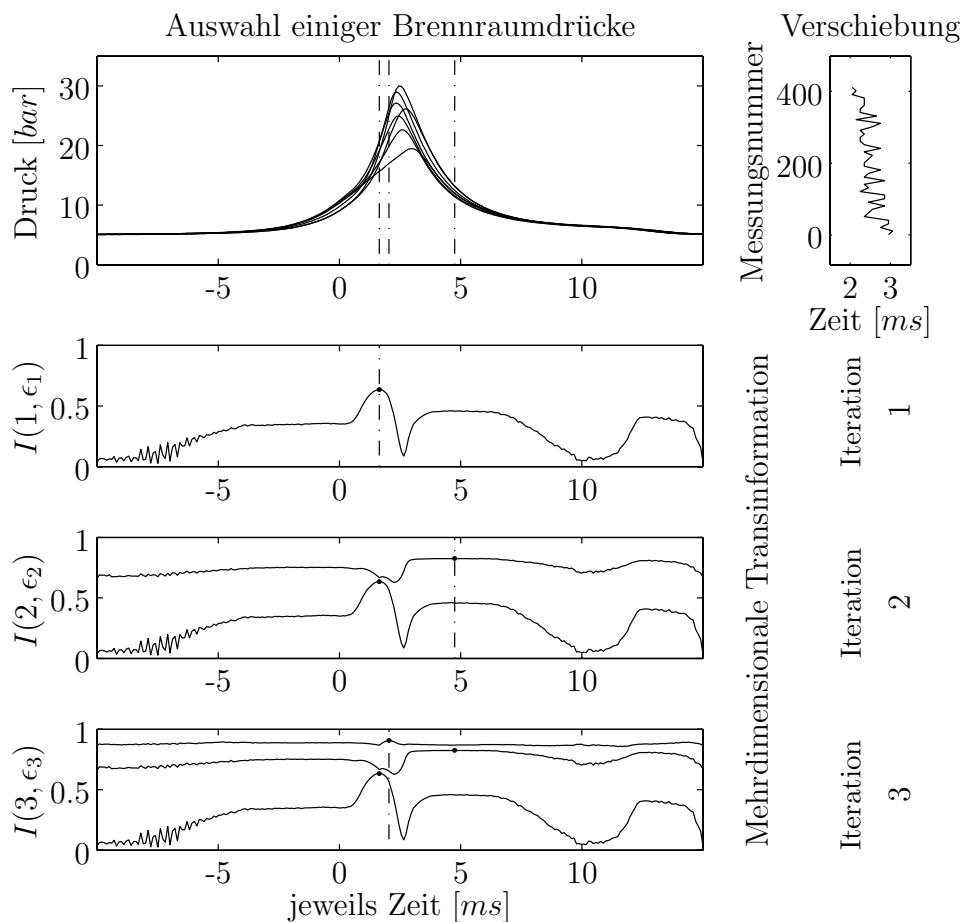


Abbildung 4.8: Mehrdimensionalen Transinformatiionsfunktion.

Die Abbildung zeigt nur die ersten 3 Iterationsschritte zur Auswahl von Druckpositionen. Dabei sind drei Druckpositionen mit hohem Informationsgehalt bezogen auf die Verschiebung der Druckmaxima des Brennraumdrucksignal ausgewählt worden.

Der fortschreitende Anstieg der Transinformatiionsfunktion $I(t, \epsilon)$ durch Auswahl zusätzliche Stützstellen ist in Abbildung 4.8 gut zu erkennen.

Für eine übersichtlichere Darstellung sind darüber nur einzelne, ausgewählte Meßreihen des Brennraumdrucks waagrecht dargestellt. Daneben ist der Vektor der zugehörigen Verschiebungen des Maximaldrucks erkennbar. Zur Berechnung der Transintransformationsfunktionen wird für diesen Vektor und jede Abtastposition aller Meßreihen die verallgemeinerte Transintransformationsfunktion berechnet. Die Vektoren der ausgewählten und in die Berechnung höherer Iterationen einfließenden Positionen mit bisher maximalem Informationsgewinn sind durch die gestrichelten Linien markiert.

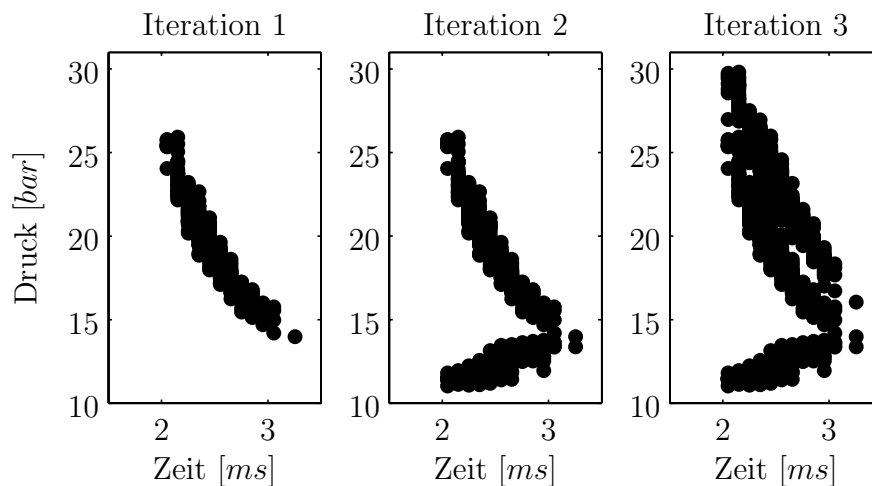


Abbildung 4.9: Durch Transintransformations gefundene Cluster.

Darstellung der ersten 3 Iterationen bei der Brennraumdruckanalyse mit größter statistischer Abhängigkeit zu der zeitlichen Verschiebung der Druckmaxima. Die Nichtlinearität und statistische Unabhängigkeit der ausgewählten Positionen ist gut zu erkennen.

Die im Beispiel zur Brennraumdruckanalyse gefundenen Abhängigkeiten zwischen Druck und zeitlicher Verschiebung der Druckmaxima sind in Abbildung 4.9 dargestellt. Die Punktwolken oder Cluster veranschaulichen die maximalen, statistischen Abhängigkeiten der ersten drei Iterationen des Verfahrens. Es ist gut zu erkennen, daß insbesondere mit der zweiten Iteration kein starker, linearer Zusammenhang besteht. Anhand der Clusterung ist jedoch eine relativ starke, nichtlineare Abhängigkeit erkennbar.

Das Beispiel 4.3.1 veranschaulicht die Funktionsweise der Dimensionsreduzierung mittels mehrdimensionaler Transintransformations. Im Vergleich zu einer äquidistanten Abtastung kann durch die vorgestellte Methode die Anzahl der Stützstellen stark reduziert werden. Unter Verwendung einer Korrelationsanalyse können zwar ebenfalls Stützstellen mit geringem statistischem Zusammenhang eliminiert werden, jedoch kann keine

Aussage über statistische Zusammenhänge unter den Stützstellen gemacht werden. Solche Informationen kann eine Berechnung von Hauptkomponenten liefern.

Die Auswahl relevanter Eingangskomponenten mit Hilfe der mehrdimensionalen Transinformationsanalyse bietet ähnliche Vorteile wie die Berechnung von Hauptkomponenten. Die Methode ist zwar sehr zeitaufwendig, hat jedoch den wesentlichen Vorteil, daß die Berechnungskosten nur bei der Analyse der Daten anfallen. In der Anwendungsphase sind keine weiteren Berechnungen nötig.

Dem in Kapitel 2, Abschnitt 2.4.4 vorgestellten Verfahren zur Schätzung der Transinformation sind, insbesondere im mehrdimensionalen Fall, statistische Grenzen gesetzt. Das hat zur Folge, daß schon ab ca. 10 Komponenten jede weitere Auswahl zunehmend statistischen Schwankungen unterliegt und die Informationsmaxima weniger deutlich heraustreten (siehe Kapitel 5, Abschnitt 5.3.4). Somit ist die Anzahl an Stützstellen begrenzt, die durch das Verfahren ermittelt werden können.

4.4 Informationsoptimierte Netztopologie

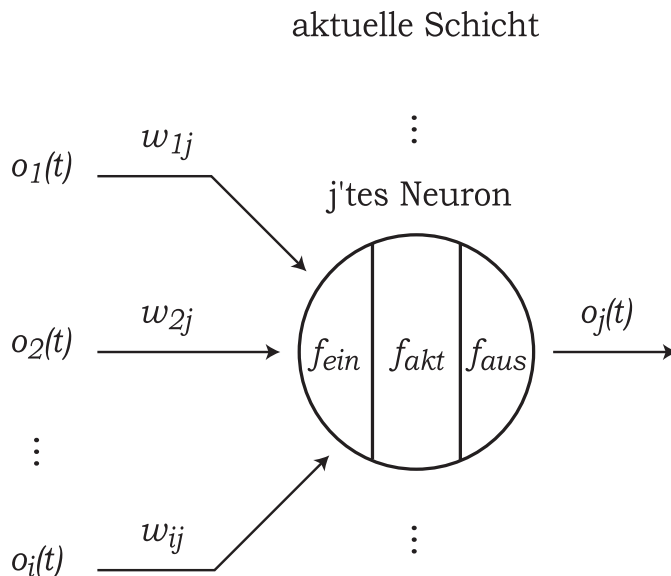
Aufbauend auf der Optimierung von Eingangsdimensionalitäten stellt dieser Abschnitt eine Methode vor, bei der die Transinformation zur informationstheoretischen Optimierung der gesamten Netztopologie Verwendung findet. Ähnlich wie bei dem bekannten Lernverfahren *Cascade Correlation* [FL90] werden hierbei zur bestehenden Netzstruktur Neuronen hinzugefügt und einzeln trainiert. Der hier beschriebene Algorithmus generiert dabei neuronale Netze mit nur einer verdeckten Schicht. Das Verfahren kann jedoch leicht zur Generierung tieferer Netzstrukturen erweitert werden.

4.4.1 Optimierung von Netzverbindungen

Grundlage des Algorithmus zum *informationsoptimierten* Netztraining ist die informationstheoretische Bewertung von Netzverbindungen. Dazu wird der statistische Zusammenhang zwischen allen Verbindungen vorangehender Neuronen und der Ausgabe des aktuell betrachteten Neurons abgeschätzt. Diese Abschätzung erfolgt mit Hilfe des Algorithmus 4.4.1 zur Dimensionsreduktion auf Basis der verallgemeinerten Transinformation.

Abbildung 4.10 soll die Anwendung der Dimensionsreduktion am Beispiel eines Neurons verdeutlichen. Am Eingang des dargestellten Neurons j liegt die Menge gewichteter Eingangsfolgen $\{w_{ij} \cdot o_i(t)\}_{i=1}^N$ aller N vorangehenden Neuronen an. Nach Durchlaufen der Eingangs-, Aktivierungs- und Ausgangsfunktion liefert das Neuron die Ausgangsfolge $o_j(t)$. Dabei können Eingangs-, Ausgangs-, sowie Aktivierungsfunktion beliebig gewählt sein. Werden Eingangsfolge $w_{ij} \cdot o_i(t)$ als Parameter $X(t)$ und Ausgangsfolge

$o_j(t)$ als Parameter $Y(t)$ des Algorithmus 4.3.1 gewählt, so liefert die Dimensionsreduktion die Verbindungen zum Neuron j , die zur Erzeugung der Ausgabefolge $o_j(t)$ die größte Relevanz haben. Ohne weitere Parameter werden die Verbindungen ausgewählt, die z.B. 80% der relevanten Information tragen. Ist diese Information ausreichend, können alle anderen Verbindungen gelöscht werden. Eine solche Optimierung wird oft



Bei der *informationsoptimierten* Dimensionsreduktion wird die verallgemeinerte Transinformation verwendet, um die Relevanz der Eingänge $o_1(t) \dots o_i(t)$ für die Neuronenausgabe $o_j(t)$ zu bewerten. Nicht relevante Verbindungen und deren vorangestellte Neuronen können dann entsprechend einer Schwelle entfernt werden.

Abbildung 4.10: Gewichtete Ein- und Ausgänge eines Neurons.

auch als Pruning bezeichnet. Bei Pruning-Algorithmen werden jedoch zunächst große Netzstrukturen gewählt, um sie erst nach vollständigem Training auszudünnen. Im hier beschriebenen Algorithmus erfolgt die Ausdünnung jedoch schon während des Trainings, was einen deutlichen Unterschied zu Pruning-Algorithmen darstellt.

Algorithmus 4.4.1 (Informationsoptimiertes Netztraining)

Der Algorithmus liefert für Meßreihenpaare X und Y eine informationsbezogen optimale und trainierte Netztopologie. Dabei erfolgt die Auswahl der optimalen Netzverbindungen auf Grund der verallgemeinerten Transinformation. Das Training des Netzes kann durch jeden beliebigen Algorithmus erfolgen.

Eingangsparameter:

- Die Eingangsmatrix $X(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}_{t=1}^{T_1}$ aus Spaltenvektoren $x(t)$ der einzelnen Messungen am Systemeingang. Dabei ist N die Anzahl der Messungen und T_1 die Länge der Zeitreihen des Systemeingangs bzw. maximale Dimension des Netzeingangs.

- Die Ausgangsmatrix $Y(t) = \{y_1(t), y_2(t), \dots, y_N(t)\}_{t=1}^{T_2}$ aus Spaltenvektoren $y(t)$ der einzelnen Messungen am Systemausgang. Dabei ist T_2 die Länge der Zeitreihen des Systemausgangs bzw. Dimension des Netzausgangs.
- Die Anzahl n der Trainingsepochen bis zum Einfügen eines neuen Neurons.
- Die Belegungsdichte $proz \in [0, 1]$ als indirektes Maß der Suchboxgröße.
- Die Abbruchbedingung $grad \in [0, 1]$ als Angabe der zu erreichenden Gesamtinformation.

Algorithmus:

1. Erzeuge Netzwerk NET mit T_1 Eingangs- und T_2 Ausgangszellen.
2. Initialisiere nicht trainierte Verbindungen.
3. Trainiere Netzwerk für n Epochen mit Lerndaten $X_{trn}(t) \subset X(t)$ und entsprechender Trainingsausgabe $Y_{trn}(t)$.
4. Speichere Ausgabefehler MSE_{net} der Netzausgabe ausgewählter Validierungsdaten $X_{val}(t) \subset X(t)$ wobei $X_{trn}(t) \cap X_{val}(t) = \emptyset$ bezüglich Trainingsausgabe $Y_{val}(t)$.
5. Wiederhole ab 3. falls Ausgabefehler MSE_{net} ⁶ kontinuierlich fällt.
6. Optimierte Netzverbindungen aller Neuronen durch Algorithmus 4.3.1 zur Dimensionsreduktion.
7. Trainiere optimiertes Netzwerk für n Epochen mit Lerndaten $X_{trn}(t) \subset X(t)$ und entsprechender Trainingsausgabe $Y_{trn}(t)$.
8. Speichere NET_{opt} und Ausgabefehler MSE_{opt} der Netzausgabe $X_{val}(t) \subset X(t)$ mit minimalem Ausgabefehler MSE_{net} bezüglich Trainingsausgabe $Y_{val}(t)$.
9. Wiederhole ab 7. falls Ausgabefehler MSE_{opt} kontinuierlich fällt.
10. Erweitere Netzwerk um ein Neuron durch Verbinden mit allen Ein- und Ausgangszellen.
11. Springe zu 2. falls Ausgabefehler $MSE_{opt} < MSE_{net}$.

Ausgabeparameter:

- Netzwerk NET_{opt} mit minimalem Ausgabefehler MSE_{opt} .

⁶Neben dem MSE sind auch andere Fehlermaße als Abbruchkriterium denkbar.

Beschreibung:

Zunächst generiert der Algorithmus eine minimale Netztopologie ohne verdeckte Neuronen jedoch mit von den Daten vorgegebener Dimension für Eingangs- und Ausgangsschicht. Die Neuronen werden vollständig verbunden. Die Gewichte aller neuen Verbindungen werden initialisiert. Die Methode ist beliebig, so sind z.B. zufällige oder topologiebezogene Initialisierungsvorschriften denkbar. Daraufhin kann das Netzwerk mit einem beliebigen Lernalgorithmus trainiert werden. Dabei dient der Trainingsdatensatz X_{trn} dem Lernen und der Validierungsdatensatz X_{val} der Überwachung des Trainingserfolgs. Kann der Validierungsfehler der Netzausgabe nicht weiter minimiert werden, beginnt die Optimierung der Netztopologie. Für alle Neuronen der verdeckten aber auch der Ausgabeschicht werden mit Hilfe des Algorithmus 4.3.1 die Eingänge mit größter Relevanz ermittelt. Alle anderen Eingangsverbindungen werden gelöscht. Das so optimierte Netzwerk wird erneut trainiert. Dieser Ablauf wird so lange wiederholt, bis das Hinzufügen vollständig verbundener, einzelner Neuronen in der verdeckten Schicht keine Verbesserung des Netzfehlers liefert.

Nach dem so abgeschlossenen Training kann das Ergebnis mit Hilfe eines Testdatensatzes X_{tst} überprüft werden. Den gesamten Ablauf verdeutlicht das Beispiel 4.4.1 der Bewertung.

4.4.2 Bewertung des informationsoptimierten Trainings

Zur Beurteilung des *informationsoptimierten* Trainings werden Brennraumdruckdaten der Brennraumdruckanalyse des Beispiels 4.3.1 mit der neuen Methode trainiert und mit einer gängigen Methode, dem *Cascade Correlation* Algorithmus verglichen.

Beispiel 4.4.1 (Informationsoptimiertes Training)

Zum besseren Verständnis des Ablaufs sollen die ersten drei Iterationen eines einfachen Trainingsbeispiels diskutiert werden. Der Trainingsdatensatz verfügt über fünf Eingangs- und einen Ausgangsvektor. Abbildung 4.11 stellt die Netztopologien der ersten drei Iterationen jeweils vor und nach der Optimierung dar.

In der 1. Iteration besteht die Netztopologie nur aus einer Eingangs- und einer Ausgangsschicht. Nach dem Training des Netzwerks (a) konnten zwei Netzverbindungen als nicht relevant klassifiziert und aus dem Netzwerk (b) entfernt werden.

Mit der 2. Iteration kommt das erste verdeckte Neuron hinzu. Zunächst wird es mit allen Vorgänger- und Nachfolgerneuronen des Netzwerks (c) verknüpft. Auch hier können nach erfolgreichem Training einige Verbindungen eliminiert werden, wobei andere Vorgängerneuronen der Topologie (d) für das verdeckte Neuron als relevant erscheinen.

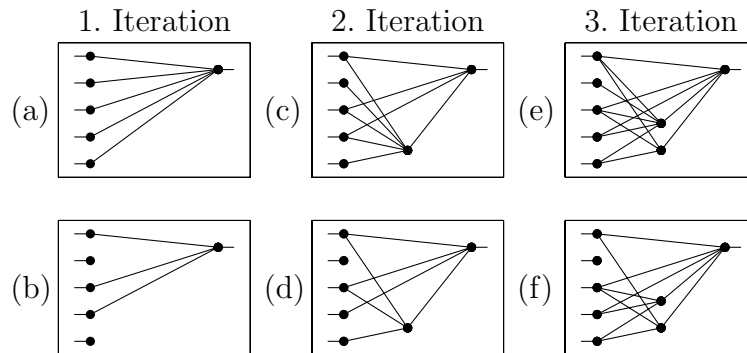


Abbildung 4.11: Informationsoptimiertes iteratives Training.

Darstellung der ersten 3 Iterationen vor und nach informationstheoretischer Optimierung von Netzwerkverbindungen. Mit den Iterationen 2 und 3 kommt jeweils ein neues Neuron hinzu.

Dieses Vorgehen wird auch bei der 3. Iteration wiederholt. Wie die Netztopologie (f) zeigt, trägt der Eingabevektor eines Neurons weniger relevante Information als seine Nachbarn. Auch das zweite verdeckte Neuron bezieht andere Informationen als das Ausgabeneuron bzw. sein Nachbarneuron.

Die im Beispiel 4.4.1 gewonnene Netztopologie hat im Vergleich zu einer vollständig verknüpften Topologie mit zwei verdeckten Neuronen $1/3$ weniger Verbindungen. In Abbildung 4.11 ist ebenfalls gut zu erkennen, wie nicht nur die Netztopologie, sondern auch die Eingangsdimension wächst, obwohl mit jeder Iteration nur 3 Stützstellen mit hohem Informationsgehalt ausgewählt werden. Damit kommt das Verfahren zur Schätzung der Transinformation nicht an seine in den Abschnitten 4.3.2 und 5.3.4 erwähnten statistischen Grenzen. Zusätzlich ist durch eine ständige Überwachung des Validierungsfehlers gewährleistet, daß sich die Abbildungsfunktion des neuronalen Netzes nicht auf die Trainingsdaten spezialisiert.

Im Vergleich zu den in Kapitel 3 beschriebenen *Pruning*-Algorithmen wird die Netztopologie bei der vorgestellten Methode schon während des Trainings optimiert. Der im gleichen Kapitel beschriebene *Cascade Correlation* Algorithmus [FL90] paßt ebenfalls die Netztopologie während des Trainings an, jedoch bestehen Unterschiede zu der hier vorgestellten Methode. *Cascade Correlation* erzeugt Netzstrukturen mit vielen Schichten, die jeweils nur ein Neuron enthalten. Dabei ist jedes Neuron mit allen nachfolgenden Neuronen vollständig verbunden.

Abbildung 4.12 verdeutlicht den Unterschied der entstandenen Netzstrukturen. So erzeugt *Cascade Correlation* schon nach 3 Iterationen wesentlich mehr Verbindungen, als das *informationsoptimierte* Training. Zudem wird mit jedem neuen Neuron eine

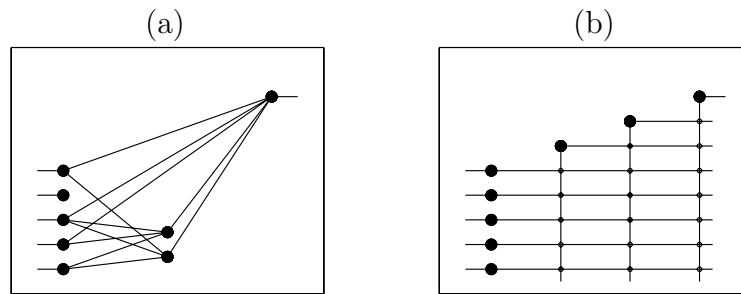


Abbildung 4.12: Vergleich informationsopt. Training - *Cascade Correlation*.

Die Gegenüberstellung der Ergebnisse des *informationsoptimierten* Trainings (a) und des *Cascade Correlation* Algorithmus (b) zeigt, daß *Cascade Correlation* schon nach 3 Iterationen wesentlich mehr Verbindungen erzeugt. Im gegebenen Beispiel sind 18 gegenüber 11 Verbindungen entstanden. Dies kann erst durch späteres Pruning behoben werden.

weitere Schicht in das Netzwerk eingefügt, die die Ausgabe aller Vorgängerzellen verarbeitet. Um die Netzgröße zu beschränken, kann *Cascade Correlation* mit *Pruning* Algorithmen kombiniert werden, die nach jeder Iteration versuchen, die Netzstruktur zu optimieren.

Der *Cascade Correlation* Algorithmus hat somit im Vergleich zum *informationsoptimierten* Training folgende Nachteile:

- Die Bestimmung von Verbindungen und deren Gewichtung neuer Neuronen erfolgt auf Basis von linearen Korrelations- bzw. Kovarianzberechnungen.
- Das Verfahren erzeugt zunächst Netze mit großer Schichtanzahl und vielen Verbindungen.
- Die Verbindungsanzahl kann nur durch nachträgliches oder iteratives Pruning reduziert werden.
- Pruning Verfahren sind entsprechend Kapitel 3 jedoch selbst problembehaftet.

Der wesentliche Vorteil der einfacheren und somit schnelleren Berechnung durch *Cascade Correlation* soll jedoch nicht verschwiegen werden. Auch wenn das *informationsoptimierte* Netztraining einen höheren Berechnungsaufwand hat, so erzeugt das neue Verfahren im Vergleich jedoch kleinere, leistungsstarke Netzstrukturen. Dies wird durch die informationstheoretische Optimierung aller Netzverbindungen erzielt.

4.5 Zusammenfassung

Im Gegensatz zu vielen aktuellen Arbeiten, die sich mit der Optimierung bzw. Neuentwicklung von Trainingsalgorithmen beschäftigen, liegt der Schwerpunkt der vorliegenden Arbeit bei der Bestimmung, bzw. Erzeugung optimaler Netztopologien. Damit die vorgestellten Algorithmen zur vollen Leistungsfähigkeit gelangen, ist eine für die Verfahren optimale Datenvorverarbeitung erforderlich. Zu diesem Zweck ist die Spektralanalyse mit Hilfe der Fourier-Transformation um informationstheoretische Anteile erweitert worden, was den Entwurf optimierter Filter ermöglicht.

Alle beschriebenen, neu entwickelten Methoden, basieren auf dem in [Pom97] vorgestellten und im Kapitel 2, Abschnitt 2.4 dargestellten Schätzverfahren zur verallgemeinerten Transinformation. Sie sind als Erweiterung dieses Verfahrens bzw. als einer Kombination mit weiteren Methoden zu verstehen. Dabei sind folgende Algorithmen entstanden:

Informationsoptimierte Frequenzanalyse:

Diese Methode ist eine Erweiterung der gewöhnlichen Fourier-Analyse. Dabei fließen mit Hilfe der verallgemeinerten Transinformation statistische Abhängigkeiten als zusätzliche Information in die Spektren ein. Durch diese informationstheoretische Ergänzung läßt sich ablesen, welche Frequenzen der Eingangssignale tatsächlich erforderlich sind, damit ein System ein gewünschtes Ausgangsverhalten hat.

Informationsoptimierte Eingangsdimension:

Zur *informationsoptimierten* Reduzierung der Eingangsdimensionalität neuronaler Netze ist eine mehrdimensionale Anwendung der verallgemeinerten Transinformation notwendig. Durch diesen Ansatz entsteht eine Art mehrdimensionale, nicht lineare Korrelationsanalyse, die zur Optimierung der Eingangsschicht einer Netzstruktur herangezogen werden kann.

Informationsoptimierte Netztopologie:

Auch zur Optimierung der gesamten Netzstruktur künstlicher neuronaler Netze kann die verallgemeinerte Transinformation eingesetzt werden. Der neu vorgestellte Algorithmus erzeugt eine Netztopologie, bei der die informationstheoretische Abhängigkeit der Ein- und Ausgänge aller Neuronen optimiert ist.

Dieses Kapitel konnte vielseitige, neue Methoden vorstellen, die sich die Eigenschaften der verallgemeinerten Transinformation zu Nutzen machen. Die Berechnung der informationstheoretischen Zusammenhänge erfolgt dabei auf Basis eines in [Pom97] vorgestellten Schätzverfahrens. So konnten Erweiterungen zur Signalanalyse, aber auch Optimierungen von Netztopologien entwickelt und vorgestellt werden.

Kapitel 5

Anwendungen

Schwerpunkt der folgenden Anwendungen sind Projekte aus dem Bereich der Motordiagnose. Die Arbeit hat dabei das Ziel, ohne große Mehrkosten kommende Forderungen der EURO-III-Abgasnormen zu erfüllen. Im Rahmen des CO_2 -Reduktionsprogramms der EUCAR¹ für Automobile ist eine Überwachung aller schadstoffrelevanten Motorgrößen gefordert. Das Reduktionsprogramm ist eine Reaktion auf die politische Forderung der Europäischen Union (EU), den Flottenausstoß von $120\text{ g }CO_2/km$ bis zum Jahr 2012 zu erreichen.

Zur Erfüllung dieser Forderungen ist somit eine Überwachung aller verbrennungsrelevanten Parameter erforderlich. Zu solchen Parametern gehören auch der *50%-Umsatzpunkt*² (USP), der Einspritzzeitpunkt und die tatsächlich eingespritzten Kraftstoffmengen, wobei jeweils eine zylinderindividuelle und arbeitsspielgenaue Auflösung wünschenswert sind. Da die Menge des eingespritzten Kraftstoffs, sowie Fehlerzustände im Einspritzsystem wesentlich zur Qualität der Verbrennung und somit zum Schadstoffausstoß beitragen, ist eine genaue Bestimmung der eingespritzten Kraftstoffmenge in Zukunft ein fester Bestandteil der Europäischen Fahrzeugdiagnosesysteme (engl. *On-Board-Diagnose*).

Die neu entwickelten Methoden sind auf die Anforderungen solcher *On-Board-Diagnose* Systeme zugeschnitten. Sie sind jedoch allgemein genug, um auch für andere Aufgaben mit ähnlicher Signalverarbeitung zum Einsatz zu kommen. Die Anwendbarkeit der Methodik wird entsprechend der Aufgabenstellung am sogenannten *Common-Rail*-System demonstriert, dessen Beschreibung zunächst erfolgt.

¹Die Abkürzung EUCAR steht für *European Council for Automotive Research and Development*. Die EUCAR Initiative wurde am 27. Mai 1994 als Nachfolger des Joint Research Committee (JRC) der Europäischen Automobilhersteller gegründet. Der wissenschaftlichen Linie der JRC folgend, hat EUCAR damit begonnen, strategische Kooperationen in der Forschungs- und Technologieentwicklung zu pflegen.

²Der *50%-Umsatzpunkt* ist der Zeitpunkt zu dem 50% der Energie der zur Verfügung stehenden Kraftstoffmenge durch die Verbrennung umgesetzt wurde (Schlüsselgrößen zur Überwachung des Verbrennungsprozesses).

5.1 Systemgrundlagen

In den letzten Jahren haben sich insbesondere in Hinblick auf Verbrauch und geringen Schadstoffausstoß Dieselmotoren mit Direkteinspritzung bewährt.

DAIMLERCHRYSLER setzt bei dieser Technologie auf ein System mit Druckspeicherung (engl. *Common-Rail*) und elektromagnetischer Injektoreinspritzung (engl. *Injection*). Die Komponenten und Funktionsweise dieser Technologie, sowie die zur Auswertung benötigte Meßtechnik werden im Folgenden vorgestellt.

5.1.1 Grundlagen zum *Common-Rail*-System

Die in Kapitel 4 vorgestellten neuen Methoden sind bei der Entwicklung neuer *Onboard*-Überwachungsverfahren für *Common-Rail*-Systeme zum Einsatz gekommen. Zum besseren Verständnis dieser Einspritztechnologie werden in diesem Abschnitt einige Grundlagen grob dargestellt.

Im Vergleich zu Otto-Motoren sind Dieselmotoren selbstzündende Verbrennungsmaschinen. Die Entzündung erfolgt im normalen Betrieb alleine durch die Kompression des Luft- Kraftstoffgemisches. Der Ablauf einer solchen Verbrennung wird Arbeitsspiel genannt und kann in Ansaugphase, Verdichtungsphase, Verbrennungsphase und Ausstoßphase unterteilt werden.

Bei direkteinspritzenden Motoren erfolgt die Einspritzung des Kraftstoffs gegen Ende der Verdichtungsphase. Dabei sind sowohl Einspritzzeitpunkt, als auch Einspritzmenge für eine vollständige und saubere Verbrennung von höchster Wichtigkeit. Die Überwachung des Zündzeitpunktes³ kann z.B. durch die Bestimmung des *50%-Umsatzpunktes* erfolgen [HF99]. Die Einspritzmenge kann mit der vorliegenden Arbeit aus Druckschwingungen des Kraftstoffsystems zylinderindividuell und arbeitsspielaufgelöst in Echtzeit ermittelt werden.

Um den Kraftstoff in den verdichteten Brennraum einzubringen, muß dieser unter Druck eingespritzt werden. Herkömmliche, nockenwellengetriebene Verfahren sind z.B. die Reiheneinspritzpumpe (*PE*) und die Radialkolben-Verteilerpumpe (*VP44*). Bei der Reiheneinspritzpumpe handelt es sich um ein sogenanntes kantengesteuertes System, das überwiegend in Nutzfahrzeugen zum Einsatz kommt. Einspritzdauer und Einspritzmenge sind hierbei durch die Lage einer Steuerkante bezogen auf eine Steuerbohrung veränderlich. Beim zweiten System, der Radialkolben-Verteilerpumpe hängen Einspritzdauer und Einspritzmenge bereits von der Öffnungszeit eines Magnetventils ab. Beide Systeme erzeugen den nötigen Druck jedoch nur zum Zeitpunkt der Einspritzung.

³Durch die Selbstentzündung liegt kein durch Zündkerzen exakt gesteuerter Zündzeitpunkt vor.

Im Unterschied zu den genannten Systemen ist das CDI-System (engl. *Common Rail Diesel Injection*) ein elektronisch gesteuertes Speichereinspritzsystem. Hierbei wird der Druck in einem Hochdruckspeicher ständig aufrechterhalten. Zusammen mit den elektromagnetisch angesteuerten Injektoren ermöglicht dies eine sehr flexible Steuerung der Dauer und Menge jeder einzelnen Einspritzung.

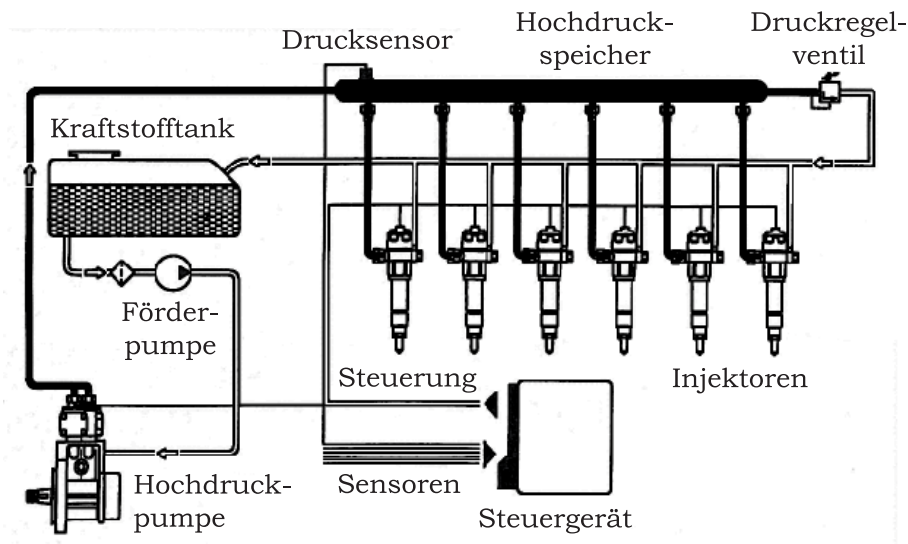


Abbildung 5.1: Schematischer Aufbau des CDI-Systems.

Darstellung aller wichtigen Komponenten des *Common-Rail*-Systems mit Hochdruckkreis (schwarz), Niederdruckrücklauf (weiß), sowie Sensoren und Steuergerät.

Allgemeiner Aufbau und Arbeitsweise:

In Abbildung 5.1 sind die wesentlichen Komponenten eines sechs Zylinder CDI-Systems dargestellt. Die Hauptkomponenten sind Hochdruckspeicher, Hochdruckpumpe, sowie die sechs Injektoren und das Motorsteuergerät. Die Hochdruckpumpe erzeugt im Hochdruckspeicher, je nach Drehzahl und Last⁴ bzw. Kraftstoffbedarf, einen Druck zwischen 250 und 1350 *bar*, der jederzeit zur Einspritzung über die Injektoren zur Verfügung steht. Nicht benötigter Kraftstoff gelangt über einen Niederdruckrücklauf zurück in den Tank. Um das System korrekt zu steuern, benötigt das Steuergerät neben dem Raildrucksignal eine Reihe weiterer Motorsensordaten, wie Drehzahl der Kurbelwelle, Phase der Nockenwelle, Fahrpedalstellung, Ladedruck, Luftmasse, Luft- und Kühlmittemperatur.

⁴Drehzahl und Last bzw. erforderliches Drehmoment werden gemeinsam auch als Arbeitspunkt bezeichnet.

Im folgenden werden die wesentlichen Komponenten des Systems näher erläutert:

Druckgeregelter Hochdruckspeicher:

Der Hochdruckspeicher versorgt zum einen die Injektoren mit Kraftstoff, dient durch sein Volumen aber auch dem Druckausgleich. Im Vergleich zu Einspritzmengen von max. 100 mm^3 pro Zylinder und Arbeitszyklus hat die Common Rail ein beachtliches Volumen von ca. 40.000 mm^3 um Druckschwankungen auszugleichen. Dennoch können am Raildrucksensor einspritzungsbedingte Druckschwankungen von bis zu $\pm 50\text{ bar}$ gemessen werden. Im Vergleich zum Maximaldruck von 1350 bar sind diese Schwankungen jedoch gering, was eine Auswertung zur Einspritzmengenbestimmung erschwert. Über ein Druckregelventil wird der am Drucksensor gemessene Druck durch das Steuergerät möglichst konstant gehalten.

Saugdruckgeregelte Hochdruckpumpe:

Bei der verwendeten Hochdruckpumpe handelt es sich um eine Dreitempelpumpe. Auf die Nockenwelle aufgeflanscht, pressen abwechselnd ihre drei Kolben Kraftstoff in den Hochdruckspeicher. Die Besonderheit der verwendeten Pumpe ist die saugseitige Regelung. Bisherige Common Rail Hochdruckpumpen lieferten immer Maximaldruck. Die zu viel geförderte Menge wurde über das Druckregelventil der Common Rail und den Niederdruckkreislauf in den Tank zurückgeführt. Saugdruckgeregelte Pumpen liefern einen variablen Druck, der durch das Motorsteuergerät vorgegeben wird. Somit wird das System flexibler, weniger Drucküberschuß wird produziert, jedoch ist die Analyse bzw. Modellierung wesentlich komplexer.

Elektromagnetische Injektoren:

Die Injektoren haben die Aufgabe, die Zylinder des Motors mit Kraftstoff zu versorgen. Für eine optimale Verbrennung muß der Injektor zeitlich korrekt durch einen Steuerstrom geöffnet werden. Um den Brennraumgedruck und den hohen Raildruck zu beherrschen, kann die Injektornadel nur indirekt angesteuert werden. Durch den Steuerstrom wird über ein Ventil ein Teil des Raildrucks angesteuert und dazu verwendet, die Nadel des Injektors zu bewegen, um die eigentliche Einspritzung auszulösen. Durch diese zweistufige Ansteuerung stellt sich der Injektor ebenfalls als ein schwer zu modellierendes Bauteil mit einem stark nichtlinearen Verhalten dar.

Ein Modell vom Drucksensor bis zur tatsächlich eingespritzten Kraftstoffmenge besteht somit aus komplizierten Komponenten und bildet unter Berücksichtigung des Zusammenspiels ein hoch komplexes System, für das es bis zum heutigen Tag noch kein vollständiges Modell gibt. Für eine ausführliche Beschreibung des CDI-Systems sei hier auf [Tac99]⁵ verwiesen.

⁵Eine gute Darstellung des *Common-Rail*-Systems ist bei der ROBERT BOSCH GmbH erhältlich.

5.1.2 Sensorik und Meßtechnik

Modelle neuronaler Netze können nur so gut sein, wie die Meßdaten, die zu ihrem Training zur Verfügung stehen. Bei den Messungen muß aus diesem Grund sehr großer Wert auf eine hohe Qualität der Aufzeichnungen gelegt werden. Dabei spielen besonders die Sensoren, aber auch die Meßtechnik und das Vorgehen eine große Rolle. Die im Zusammenhang mit den Anwendungen verwendeten Meßverfahren und Sensoren sind im Folgenden zusammengestellt.

Messungen zur Bestimmung des 50%-Umsatzpunktes:

Die Aufzeichnung der Brennraumdrücke zur Bestimmung der 50%-Umsatzpunkte erfolgte direkt im Fahrzeug am befeuerten Motor. Dazu sind zusätzliche Bohrungen am Motor für das Einsetzen der Brennraumdrucksensoren notwendig. Ein entsprechendes Fahrzeug mit Sensoren, Verkabelung und Meßtechnik⁶ stand zur Erstellung dieser Arbeit zur Verfügung.

Brennraumdrucksensor:

Brennraumdrucksensoren können zur Zeit noch nicht serienmäßig angeboten werden, da ihre Lebensdauer den gewünschten Anforderungen nicht entspricht. Wegen der hohen Temperaturdifferenzen im Brennraum sind aufwendige Algorithmen zur Driftkorrektur notwendig. Die Messungen sind kurbelwinkelsynchron mit einer Abtastung von $1^\circ KW$ erfolgt.

Brennverlaufsrechnung:

Zur Bestimmung des tatsächlichen 50%-Umsatzpunktes ist eine sogenannte Brennverlaufsrechnung erforderlich. Sie gibt neben dem Umsatzpunkt auch Aufschluß über viele weitere verbrennungsrelevante Größen, wie zum Beispiel auch die tatsächlich eingespritzte Kraftstoffmenge. Leider ist ihre Berechnung zu zeitaufwendig für einen Echtzeiteinsatz.

Sowohl Lebensdauer als auch Berechnungsaufwand stellen die vielversprechende Auswertung von Brennraumdrücken derzeit als noch nicht serientauglich dar. Die Reduzierung des Berechnungsaufwands kann entsprechend [MBGH96] durch den Einsatz von neuronalen Netzen erreicht werden.

⁶Die Abteilung FT2/EA verfügt über Fahrzeuge, die als Testträger für derartige Messungen ausgestattet sind.

Messungen zur Einspritzmengenbestimmung:

Die Messungen des Rail- und Injektordrucks erfolgen nicht direkt am Motor, sondern an einem Pumpenprüfstand. Auf einem Pumpenprüfstand wird der Kraftstoff nicht direkt in den Brennraum der Zylinder eingespritzt, sondern in sogenannte Mengenindikatoren. Die Aufzeichnung erfolgt mit einem Transientenrekorder mit Mehrkanal-AD-Wandlern. Eine genaue Beschreibung des Prüfstands mit seiner Meßtechnik und den verwendeten Indikatoren kann in [Fro96] und [Tac99] gefunden werden.

Raildrucksensor:

Zur Aufzeichnung des Raildrucks kam der serienmäßig eingebaute BOSCH-Drucksensor zum Einsatz. Dieser Sensor ist recht kostengünstig und wird zur Regelung des verwendeten Hochdruckspeichers genutzt. Der meßbare Bereich liegt zwischen 200 und 3000 *bar*.

Injektordrucksensor:

Die Druckmessung an den Injektoren erfolgt nicht serienmäßig. Zur Aufzeichnung dieser Drücke sind am Pumpenprüfstand an Zuleitungen zwischen Hochdruckspeicher und Injektor Drucksensoren der Firma KISTLER angebracht. Diese Sensoren sind wesentlich genauer als der verwendete Raildrucksensor, jedoch auch bedeutend teurer. Die Aufzeichnung erfolgt somit nicht direkt im Injektor. Eine solche Messung ist in Zukunft durch den Einsatz von Piezo-Injektoren denkbar.

Einspritzmengenindikator:

Mit Hilfe der heutigen Indikatoren kann die Einspritzmenge mit einer Genauigkeit von 1% gemessen werden. Eine höhere Genauigkeit ist derzeit nicht notwendig, da die fertigungsbedingte Streuung der Injektoren im Bereich von 3% liegt.

Die Aufzeichnung aller Daten ist mit einer Abtastung von 20 *kHz* erfolgt. Für qualitativ hochwertige Messungen ist eine ständige Überwachung der Drift, der Drucksensoren, sowie der Mengenindikatoren erforderlich.

5.1.3 Abdeckung des Datenraums

Damit ein neuronales Netz für jede mögliche Dateneingabe gute Ergebnisse liefern kann, ist es notwendig, daß der Datenraum durch das Netztraining abgedeckt ist. Für den Einsatz neuronaler Netze im Motormanagement bedeutet dies, daß der Trainingsdatensatz den Datenraum aller möglichen Arbeitspunkte des Motors gut abdeckt. Die Betriebspunkte eines *Common-Rail*-Systems lassen sich durch einen dreidimensionalen Datenraum aus Drehzahl, Raildruck und Einspritzmenge abdecken und darstellen.

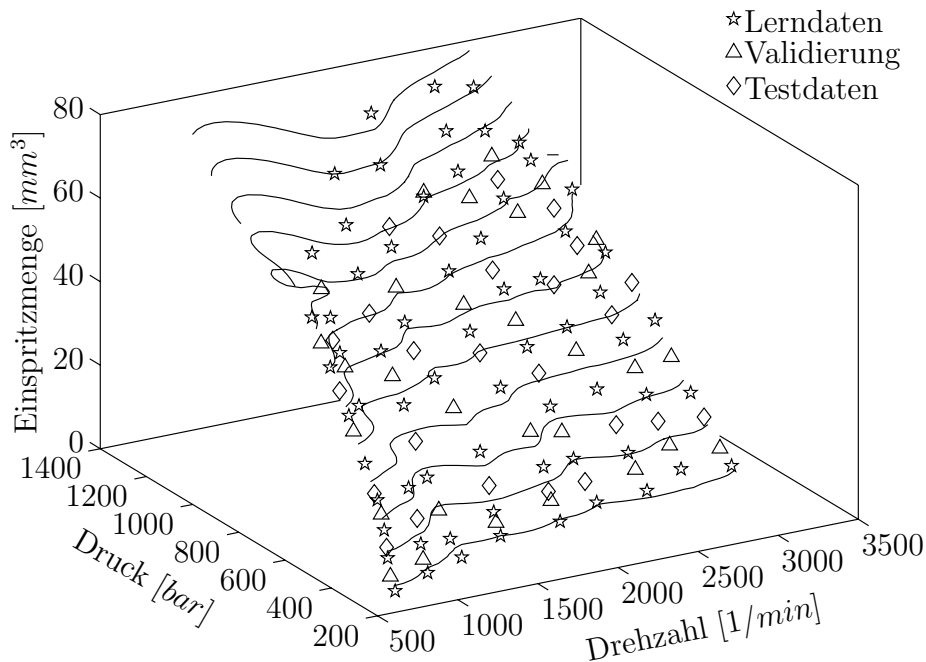


Abbildung 5.2: Abdeckung der Arbeitspunkte des Common-Rail-Motors.

Für ein zuverlässiges Netztraining ist es erforderlich, daß der Datenraum aller möglichen Netzeingaben gut abgedeckt ist. Dies gilt sowohl für Trainingsdaten, als auch für Validierungs- und Testdaten.

Abbildung 5.2 veranschaulicht die Verteilung der Meßpunkte auf der Ebene, die durch diese drei Parameter aufgespannt wird. Dabei handelt es sich lediglich um den Einstellungsraum der Motorparameter und nicht um den von den neuronalen Netzen zu approximierenden Datenraum. Dieser ist wesentlich komplexer und wird durch die Daten aufgespannt, die in den hier dargestellten Arbeitspunkten aufgezeichnet werden.

Zum Training der Netze dient nur ein Teil der Datensätze. Die Verteilung der Arbeitspunkte auf Trainings-, Validierungs- und Testdaten (siehe Definition 2.2.17 in Kapitel 2, Abschnitt 2.2.3) ist durch unterschiedliche Symbole markiert. Die Verwendung der Aufteilung entspricht dabei der Beschreibung aus Kapitel 2, Abschnitt 2.2.3 zur Bewertung neuronaler Netze.

5.2 Bestimmung des *50%-Umsatzpunktes*

Ein großes Problem bei Verbrennungsmotoren sind die Schwankungen im Ablauf der Verbrennung, die auch im stationären Fall auftreten. Dieses als Zyklenschwankung bezeichnete Verhalten beeinflusst den Wirkungsgrad eines Verbrennungsmotors deutlich. Die Gründe für Zyklenschwankungen sind sehr vielfältig. Fast jede Komponente des Motors hat Einfluß auf dieses Verhalten. Aus diesem Grund werden Verfahren gesucht, die diese Schwankungen beschreiben, um ihnen durch Regeleinriffe wie z.B. bei der Einspritzzeit entgegenzuwirken.

5.2.1 Aufgabenstellung

Zur Überwachung des Verbrennungsprozesses und somit auch der Zyklenschwankungen ist ein Verfahren zur Bestimmung des aktuellen *50%-Umsatzpunktes* (USP) gesucht. Der *50%-Umsatzpunkt* gilt als eine Schlüsselgröße zur Überwachung des Verbrennungsprozesses. Er ist der Zeitpunkt zu dem 50% der zur Verfügung stehenden, im Kraftstoff gespeicherten, Energie durch die Verbrennung umgesetzt wurde. Durch eine Regelung der Einspritzung⁷ kann die Differenz zwischen aktuellem und optimalem *50%-Umsatzpunkt* ausgeglichen werden, was zu einer energetisch verbesserten Ausnutzung der Verbrennung führt. Dabei haben frühere Untersuchungen gezeigt, daß eine optimale, kurbelwinkelbezogene Position bei $8^\circ KW$ nach dem oberen Totpunkt der Kompressionsphase liegt.

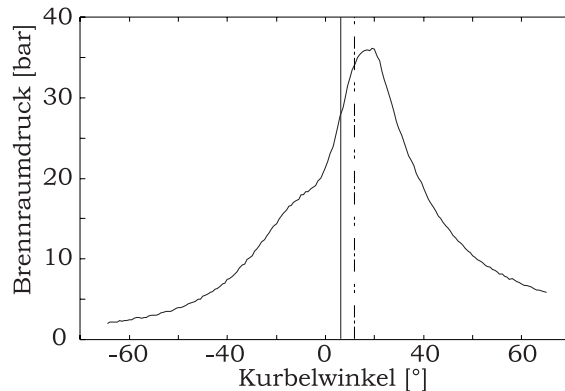
5.2.2 Neuronale Brennraumdruckauswertung

In den Jahren 1998 bis 2000 konnte im Rahmen des BMBF⁸-Projektes „Technische Anwendungen der Regelung chaotischer Systeme und der Zeitreihenanalyse“ [DMF01] bei der DAIMLERCHRYSLER AG⁹ ein echtzeitfähiges Verfahren zur Bestimmung des tatsächlichen *50%-Umsatzpunktes* entwickelt werden. Das Verfahren beruht auf der Messung und Auswertung von Brennraumdruckkurven. Die Bestimmung der Umsatzpunkte erfolgt durch ein künstliches, neuronales Netz. *50%-Umsatzpunkte* lassen sich auch durch Brennverlaufsrechnung aus Brennraumdruckkurven bestimmen, was momentan noch nicht in Echtzeit möglich ist und somit keine Regelung zuläßt. Zum Adaptieren der Netze leisten berechnete Umsatzpunkte als Trainingsvektor jedoch gute Dienste.

⁷Bei dem genannten BMBF-Projekt ist der Zündzeitpunkt anhand von Brennraumdruckkurven eines Otto-Motors geregelt worden. Eine entsprechende Regelung kann für Diesel-Motoren durch den Einspritzzeitpunkt erfolgen.

⁸BMBF steht für das Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie, das jedes Jahr zukunftsweisende Neuentwicklungen mit Forschungsgeldern unterstützt.

⁹In Zusammenarbeit der Abteilungen FT2/EA, FT2/LP und FT1/AF.



Der tatsächliche *50%-Umsatzpunkt* ist als gestrichelte Linie dargestellt. Er ist bezogen auf den optimalen *Umsatzpunkt* bei 8° KW um ca. 5° KW verzögert.

Abbildung 5.3: Brennraumdruckkurve mit *50%-Umsatzpunkt*.

Abbildung 5.3 veranschaulicht anhand einer Brennraumdruckkurve die kurbelwinkelbezogene Differenz zwischen optimalem und ermitteltem Umsatzpunkt. Durch eine kurbelwinkelsynchrone Abtastung sind die Meßdaten drehzahlunabhängig, was die Datenvorverarbeitung erleichtert, da somit die Brennraumdruckkurven ähnlich sind. Bei einer Abtastung in 1° KW-Schritten ergeben sich 180 Meßwerte pro Arbeitsspiel, um die gesamte Kompressions- und Verbrennungsphase des Motors abzudecken. Es stehen insgesamt 18 Datensätze repräsentativer Arbeitspunkte zur Abdeckung des realen Motorbetriebs zur Verfügung. In jedem Arbeitspunkt können 2000 Arbeitsspiele als Trainingsdaten und je 100 Arbeitsspiele für Test- und Validierungsdatensätze verwendet werden.

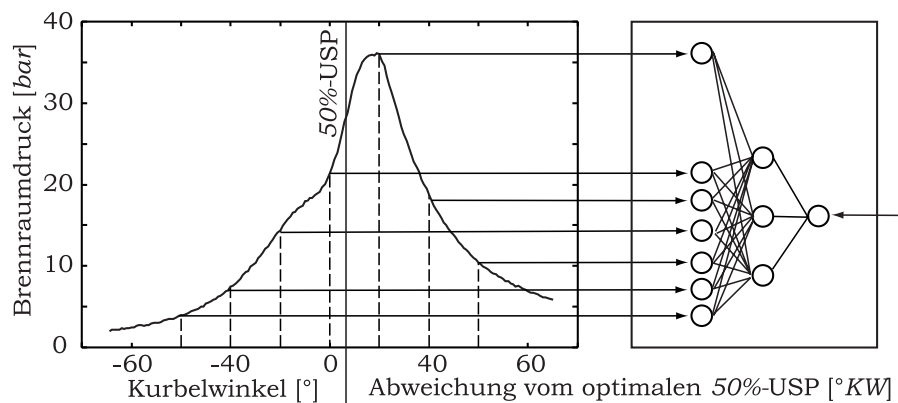


Abbildung 5.4: Äquidistante Abtastung des Brennraumdrucks.

7 von 33 äquidistant abgetasteten Druckwerte als Eingangsvektor für das neuronale Netz. Der aus den Brennraumdruckkurven berechnete aktuelle *50%-Umsatzpunkt* dient als Trainingseingabe am Ausgang des Netzes.

Abbildung 5.4 veranschaulicht eine äquidistante Abtastung der Brennraumdruckkurve als Eingangsvektor für ein neuronales Netz. Dabei sind nur 7 der 33 tatsächlich verwen-

deten Stützstellen dargestellt. Trainingseingang ist der durch Brennverlaufsrechnung gewonnene, aktuelle *50%-Umsatzpunkt*. Dabei ist zu erkennen, daß eine zu geringe Abtastung die steil ansteigenden Flanken der Hochdruckphase schlecht abdeckt. Korrelationsuntersuchungen haben gezeigt, daß gerade dieser Bereich lineare Abhängigkeiten zum *50%-Umsatzpunkt* aufweist (siehe auch Beispiel 3.1.1 in Kapitel 3). Eine äquidistante Abtastung der Hochdruckphase mit $2^\circ KW$ erwies sich als ausreichend, woraus sich Vektoren mit 33 Druckpositionen als Eingabe für ein neuronales Netz ergeben. Die Konfiguration des Netzes mit dem geringsten Restfehler kann der Tabelle 5.1 entnommen werden.

Argument	Parameter
X-Translation	OT getriggerte Abtastung
Y-Translation	Brennraumdruckminimum
Simulator	MATLAB
Netztopologie	$33 \times 50 \times 1$ (feed-forward)
Transferfunktion	'tansig' (Tangens sigmoid)
Initialisierung	'initnw' (Nguyen-Widrow)
Lernregel	'trainlm' (Levenberg Marquardt)
Trainingsparameter	Defaultwerte
Abbruchbedingung	ansteigender Validierungsfehler

Tabelle 5.1: Beschreibung der Trainingsparameter zum *50%-Umsatzpunkt*.

Das so trainierte Netz kann den aktuellen *50%-Umsatzpunkt* mit einer Genauigkeit von $0.2^\circ KW$ in allen Arbeitspunkten bestimmen, was den gewünschten Anforderungen genügt. Liegt der aktuelle Umsatzpunkt nicht in der Nähe von $8^\circ KW$ nach dem oberen Totpunkt¹⁰ (OT), kann seine Lage durch Regelung der Einspritzung korrigiert werden. Durch die Verwendung eines neuronalen Netzes ist die Bestimmung des aktuellen Umsatzpunktes mit einer Workstation bereits in Echtzeit möglich. Für einen Einsatz im Fahrzeug ist jedoch eine weitere Reduzierung der Netztopologie und somit der Berechnungszeit erforderlich.

¹⁰Der obere Totpunkt ist der obere Umkehrpunkt des Kolbens im Zylinder. In der Regel ist der obere Totpunkt der Verdichtungsphase ein wichtiger Referenzpunkt.

5.2.3 Reduzierung der Eingangsdimensionalität

Die in Kapitel 4, Abschnitt 4.3 vorgestellte Methode zur informationsoptimierten Reduzierung der Eingangsdimensionalität kann bei der Auswahl geeigneter Positionen des Brennraumdrucks gute Dienste leisten. Dabei werden solche Druckpositionen ausgewählt, die den größten Informationsanteil bezogen auf den *50%-Umsatzpunkt* beitragen.

Da die Reduzierung der Eingangsdimensionalität auf einer Schätzung der verallgemeinerten Transinformation beruht, ist entsprechend Kapitel 2, Abschnitt 2.4.3 eine Gleichverteilung der Daten erforderlich. Dabei genügt es, den Ausgabevektor bzw. den Trainingsvektor des neuronalen Netzes durch das dort beschriebene *Ranking* zu transformieren. Eine entsprechende Umsetzung ist auch durch den Algorithmus B.0.1 im Anhang gegeben.

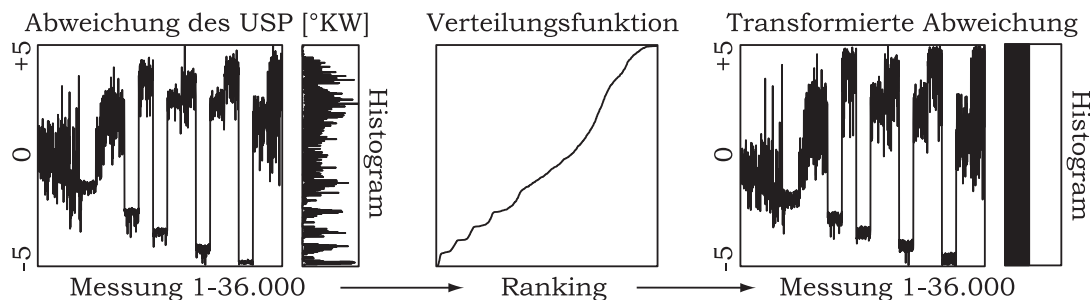


Abbildung 5.5: Gleichverteilung der Messungen zum *50%-Umsatzpunkt*.

Für einen erfolgreichen Einsatz der verallgemeinerten Transinformation zur informationsoptimierten Reduzierung der Eingangsdimensionalität ist es erforderlich, die zufällige Verteilung des Trainingsvektors durch *Ranking* in eine Gleichverteilung zu transformieren.

Abbildung 5.5 veranschaulicht die Transformation einer zufällig verteilten Abweichung, zwischen gemessenen und optimalen *50%-Umsatzpunkten*, in eine gleichverteilte Abweichung. Die gleichverteilten Abweichungen dienen als Trainingsvektoren zusammen mit den korrespondierenden 180 Druckpositionen aller 36.000 Messungen als Eingabe für den Algorithmus 4.3.1 aus Kapitel 4 zur informationsoptimierten Eingangsdimension. Der Algorithmus liefert die Brennraumdruckpositionen, die den größten statistischen Zusammenhang, d.h. den größten Informationsgehalt bezogen auf die Abweichung vom optimalen *50%-Umsatzpunkt* haben. Unter Verwendung der ersten 5 Druckpositionen kann ein neuronales Netz mit Parametern entsprechend Tabelle 5.2 erfolgreich trainiert werden.

Das so trainierte Netz ist ebenfalls in der Lage, für alle Arbeitspunkte den *50%-Umsatzpunkt* mit durchschnittlich 0.2° KW Abweichung zu identifizieren. Damit ist es gelungen, nicht nur die Eingangsdimensionalität, sondern die gesamte Netztopologie zu reduzieren, ohne Einschränkungen bei der Qualität der Identifikation in Kauf nehmen zu müssen.

Argument	Parameter
X-Translation	OT getriggerte Abtastung
Y-Translation	Brennraumdruckminimum
Simulator	MATLAB
Netztopologie	$5 \times 50 \times 1$ (feed-forward)
Transferfunktion	'tansig' (Tangens sigmoid)
Initialisierung	'initnw' (Nguyen-Widrow)
Lernregel	'trainlm' (Levenberg Marquardt)
Trainingsparameter	Defaultwerte
Abbruchbedingung	ansteigender Validierungsfehler

Tabelle 5.2: Beschreibung der Trainingsparameter mit optimierter Topologie.

Die für das Training ausgewählten Druckpositionen sind in Abbildung 5.6 skizziert. Als Trainingseingabe dient wie zuvor die Abweichung vom optimalen Umsatzpunkt. Ist das neuronale Netz adaptiert, kann die Netzausgabe direkt als Regelungsgröße zur Korrektur der Einspritzzeiten verwendet werden.

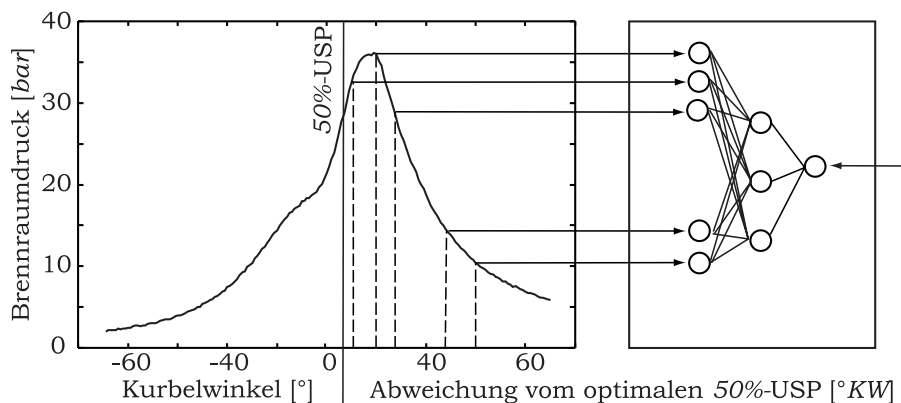


Abbildung 5.6: Durch Transinformation ausgewählte Druckpositionen.

Durch eine informationsoptimierte Auswahl von Stützstellen ist es möglich ein neuronales Netz mit nur 5 von 33 ausgewählten Druckpositionen zu trainieren, ohne Einbußen bei der Bestimmung des *50%-Umsatzpunktes*.

Durch die geringe Größe der gewonnenen Netztopologie ist eine Implementierung der *50%-Umsatzpunkt* Identifikation im Fahrzeug und in Echtzeit möglich. Mit Hilfe dieser

Identifikation kann dann eine Echtzeitregelung der Einspritzung und somit des *50%-Umsatzpunktes* und der Verbrennung erfolgen.

Eine derartige, neuronale Regelung im Fahrzeug und in Echtzeit ist zuvor, ohne eine informationstheoretische Optimierung der Eingangsdimensionalität, noch nicht gelungen. Eine ausführliche Zusammenfassung der Ergebnisse, sowie der Regelungsstrategie kann dem Technischen Bericht [DMF01] zum BMBF-Projekt „Technische Anwendungen der Regelung chaotischer Systeme und der Zeitreihenanalyse“ entnommen werden.

5.2.4 Bewertung der Ergebnisse

Anhand der Bestimmung des *50%-Umsatzpunktes* konnte die Leistungsfähigkeit der neu entwickelten Methode 4.3 aus Kapitel 4 zur Reduzierung der Eingangsdimensionalität demonstriert werden. Durch die mit dem Algorithmus 4.3.1 informationsoptimierte Auswahl von 5 Druckpositionen ist es gelungen, die Topologie von $33 \times 50 \times 1$ auf $5 \times 50 \times 1$ zu beschränken, ohne Qualitätseinbußen bei der Bestimmung des *50%-Umsatzpunktes*. Da die Berechnungszeit während der Anwendungsphase im Wesentlichen von der Anzahl der Netzwerkverbindungen abhängt [Fro96], konnte eine Verkürzung um den Faktor 5.6 erzielt werden. Die ursprüngliche Topologie benötigt bei gleichem Ergebnis 1700 Verbindungen, wohingegen die optimierte Topologie mit 300 Verbindungen auskommt.

Die Methode der Brennraumdruckauswertung ist sehr vielversprechend, da aus Brennraumdruckdaten zahlreiche verbrennungsrelevante Informationen abgeleitet werden können. So können neben der Überwachung der Einspritzzeiten z.B. auch Antiklopff- und Lambdaeregelung realisiert werden. Derzeit existieren jedoch noch keine preisgünstigen Brennraumdrucksensoren, die den Druck- und Temperaturschwankungen des Brennraums gewachsen sind.

Mit einem zuverlässigen Signal des Brennraumdrucks würden sich nicht nur die Einspritzzeitpunkte, sondern auch die Einspritzdauer und somit die Einspritzmenge mit Hilfe von neuronalen Netzen zuverlässig überwachen lassen (siehe [MBGH96] und [Köh96]). In Hinblick auf die Problematik beständiger Brennraumdrucksensoren verfolgt die folgende Methode zur Bestimmung der Einspritzmenge einen anderen Ansatz.

5.3 Einspritzmengenbestimmung

Im Rahmen von *Onboard*-Diagnose-Normen, wie der amerikanischen OBD-II oder der europäischen E-OBD und des CO₂-Reduktionsprogramms der EUCAR¹¹, sind alle abgasrelevanten Komponenten im Fahrzeug (engl. *Onboard*) zu überwachen. Bei Motoren mit *Common-Rail*-Einspritzsystemen (CR-System) ist die zylinderindividuell und arbeitsspielaufgelöste, tatsächlich eingespritzte Kraftstoffmenge eine Größe, deren Überwachung bei der Erfüllung der EURO-III-Abgasnorm hilfreich sein kann.

Um zukünftigen Anforderungen gewachsen zu sein, wurde in den Jahren 1996 bis 1998 bei der DAIMLERCHRYSLER AG im Rahmen der Dissertation [Köh98] ein Verfahren entwickelt, mit dessen Hilfe an CR-Systemen die eingespritzte Kraftstoffmenge arbeitsspielaufgelöst überwacht werden kann. Die mit der vorliegenden Dissertation vorgestellten Methoden konnten das Verfahren soweit optimieren, daß eine Echtzeitumsetzung im Fahrzeug möglich wird. Erst durch eine *Onboard*-Umsetzung ist die Erfüllung der EURO-III gewährleistet.

5.3.1 Aufgabenstellung

Um mit der schnellen Entwicklung moderner Dieselmotoren schritthalten zu können, erfolgt die Entwicklung der neuen Verfahren an einem Prototyp der neuesten *Common-Rail*-Technologie. Das untersuchte *Common-Rail*-System unterscheidet sich von früheren Systemen im Wesentlichen durch einen höheren Maximaldruck von bis zu 2000 bar. Dabei erfolgt die Druckerzeugung entsprechend Kapitel 5.1.1 durch eine saugdruckge-regelte Hochdruckpumpe.

Die Verwendung höherer Drücke ermöglicht eine bessere Zerstäubung des Kraftstoffs, was zu einer homogeneren Beladung des Brennraums führt und somit zu einer schadstoffärmeren Verbrennung. Die Saugdruckregelung macht jedoch eine Auswertung der Druckschwingungen gegenüber herkömmlicher Systeme aufwendiger. Eine weitere Schwierigkeit der neuen Konfiguration besteht in der Erhöhung der Zylinderanzahl von vier auf sechs, deren Signale überlagert werden und schwerer zu trennen sind.

Die Einspritzmengenbestimmung soll aus Druckmessungen am Hochdruckkreislauf erfolgen und folgendes Prinzip nutzen. Durch das Öffnen der Einspritzventile entsteht ein Druckabfall, der sowohl am Injektor, als auch am Hochdruckspeicher (kurz engl. *Rail*) als Schwingung gemessen werden kann.

¹¹Die Abkürzung EUCAR steht für *European Council for Automotive Research and Development*. Die EUCAR Initiative wurde am 27. Mai 1994 als Nachfolger des Joint Research Committee (JRC) der Europäischen Automobilhersteller gegründet. Der wissenschaftlichen Linie der JRC folgend, hat EUCAR damit begonnen, strategische Kooperationen in der Forschungs- und Technologieentwicklung zu pflegen.

Beide Signale sind zu betrachten, wobei der Schwerpunkt der Untersuchungen auf die *Raildruck*-Analyse zu legen ist. Diese Trennung führt zur Betrachtung folgender Unterschiede:

Raildruck:

Bei der Auswertung des Raildrucks ist keine weitere Software am *Common-Rail*-System notwendig. Die Auswertung erfolgt mit Hilfe der am serienmäßig verwendeten Raildrucksensor gemessenen Daten. Aus der Messung an nur einem Punkt der Rail ergibt sich das Problem einer Signalüberlagerung aller Einspritzungen. Durch die unterschiedlichen Laufzeiten und Dämpfungen der Schwingungen von ihren Quellen zum Drucksensor erhalten die charakteristischen Schwingungen der Einspritzungen jedes Injektors eine unterschiedliche Ausprägung. Diese Unterschiede machen eventuell den Einsatz unterschiedlicher Netze erforderlich.

Injektordruck:

Durch den höheren technischen Einsatz von Drucksensoren an jedem Injektor ist die charakteristische Schwingungsausprägung der Einspritzungen aller Zylinder vergleichbar. Durch die Nähe zur Schwingungsquelle und somit besseren Signalausprägung besteht ein größeres Potential, Voreinspritzungen klassifizieren zu können. Diese Vorteile müssen sich durch einen höheren Aufwand für Sensoren und deren Verkabelung erkauft werden. Auch ist der Hardware-Aufwand des Steuergerätes höher.

Im Raildrucksignal sind die Einspritzungen aller Injektoren sichtbar. Zudem liegt der Sensor sehr weit von den Schwingungsquellen entfernt. Dadurch machen sich neben Überlagerungen der Injektorsignale auch geometriebedingte Störungen bemerkbar. Die Komplexität des Signals erschwert eine analytische Auswertung, was den Einsatz von neuronalen Netzen rechtfertigt. Für die erfolgreiche Auswertung von Meßsignalen eines derart komplexen Systems ist eine ausführliche Analyse erforderlich.

5.3.2 Analyse und Modellierung des *Common-Rail*-Systems

Zum weiteren Verständnis des *Common-Rail*-Systems ist eine Analyse der am System aufgenommenen Meßdaten hilfreich. Dazu erfolgt zunächst eine Betrachtung im Zeitbereich. Zusätzlich sollen Analysen im Frequenzbereich, sowie eine hydraulische Betrachtung und Modellierung das Systemverständnis erweitern.

Analyse im Zeitbereich:

Abbildung 5.7 zeigt jeweils ein Arbeitsspiel der am *Common-Rail*-System aufgezeichneten Drucksignale. Dabei setzt sich das am Hochdruckspeicher gemessene Raildrucksignal p_{rail} aus dem Pumpendrucksignal p_{pmp} und den Injektordrucksignalen $p_{inj1} \dots p_{inj6}$ zusammen.

Im Pumpendrucksignal sind pro Arbeitsspiel drei volle Schwingungen zu erkennen. Sie spiegeln die bautechnisch bedingten Druckschwingungen einer Dreistempelpumpe wieder (siehe [Fro96]). Diese Schwingung ist auch im Raildrucksignal zu erkennen, wobei sie den Druckeinbruch der 1., 3. und 5. Einspritzung verstärkt.

Von den 6 Injektordrucksignalen ist nur das Signal des 1. Injektors abgebildet. Die von der Einspritzung ausgelöste gedämpfte, harmonische Schwingung ist sehr gut zu erkennen. Der Einfluß aller 6 Einspritzungen auf das Raildrucksignal tritt über das Arbeitsspiel verteilt ab ca. 33, 42, 51, 60, 69 und 78 ms auf.

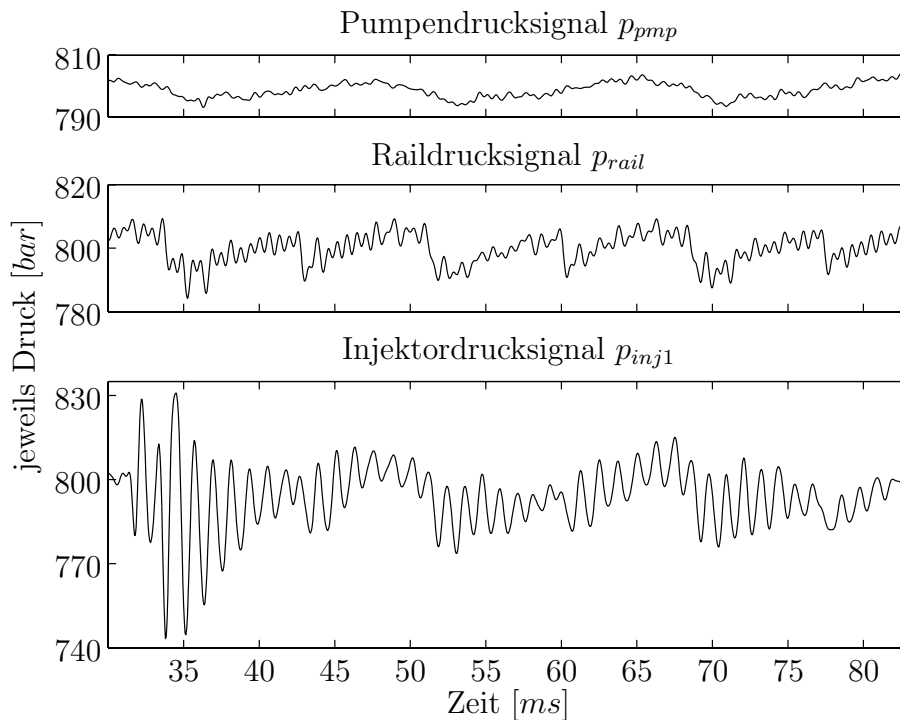


Abbildung 5.7: Drucksignale des *Common-Rail*-Systems.

Gegenüberstellung der gemessenen Drucksignale des *Common-Rail*-Systems. Das am Hochdruckspeicher gemessene Raildrucksignal p_{rail} setzt sich aus dem an der Pumpe gemessenen Pumpendrucksignal p_{pmp} und den Injektordrucksignalen $p_{inj1} \dots p_{inj6}$ zusammen, von denen nur das Signal des 1. Injektors dargestellt ist.

Allgemein treten die Schwingungen im Raildrucksignal stark gedämpft und überlagert auf, was sich durch geringere Amplituden und veränderte Signalverläufe abzeichnet. Das Auftreten all dieser Schwingungen ist auch durch folgende Analysen im Frequenzbereich belegt.

Analyse im Frequenzbereich:

Bei der Analyse der im Raildrucksignal auftretenden Frequenzen lassen sich die bereits identifizierten Schwingungen wiederfinden. So zeigt die Frequenzspitze (a) in Abbildung 5.8 die Schwingung des Pumpendrucksignals, die der dreifachen Nocken-drehzahl entspricht. Der Peak bei (b) entspricht der Frequenz von 6 Einspritzungen pro Arbeitsspiel. Die Frequenzen um (c) entsprechen den durch die Einspritzungen angeregten, gedämpften, harmonischen Injektordruckschwingung.

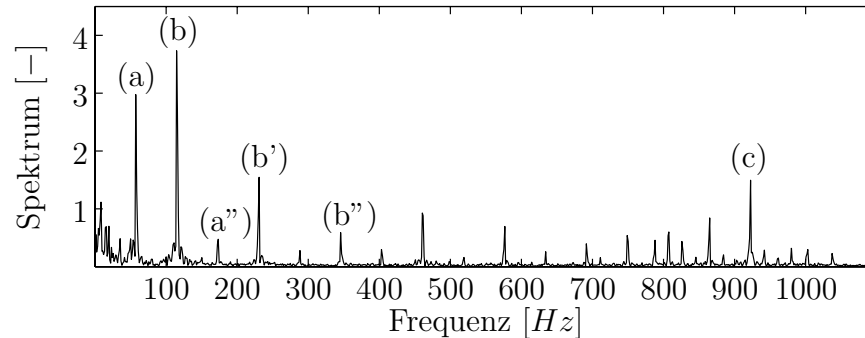


Abbildung 5.8: Normiertes Spektrum des Raildrucksignals.

Im Spektrum des Raildrucksignals ist die Frequenz der Pumpe (a), der 6 Einspritzungen (b), sowie der harmonischen Injektordruckschwingung (c) zu erkennen.

Neben den Grundfrequenzen lassen sich auch Oberschwingungen identifizieren. Die 1. Vielfache der Frequenz des Pumpendrucksignals fällt mit der Frequenz der Einspritzungen (a) zusammen. Aber auch weitere Vielfache der Pumpen- und Einspritzfrequenz wie (a'') und (b'') sind zu erkennen.

Analyse hydraulischer Vorgänge:

Größen wie Kompressibilitätsmodul, Dichte, Schallgeschwindigkeit und Frequenzen von Fluiden sind druck- und temperaturabhängig [Boh91]. Dies gilt begrenzt auch für Öl und somit für Diesel-Kraftstoffe, wobei der Einfluß erst bei hohen Temperaturen oder Drücken relevant wird. Dieser Druckbereich wird nach [CF96] im *Common-Rail*-System bereits erreicht und beeinflusst Schallgeschwindigkeit v bzw. Eigenfrequenzen ν entsprechend den Gleichungen:

$$v(p) = \sqrt{\frac{1}{\rho(p) \cdot \beta_p}} \quad \text{und} \quad \nu_k(p) = v(p) \cdot \frac{k}{2\lambda} ; \quad \text{mit } k \in \mathbb{N} \quad (5.1)$$

Dabei bezeichnet λ die Wellenlänge bzw. die Rohrlängen des Systems. Die druckabhängige Dichte $\rho(p)$ und das Kompressibilitätsmodul¹² β_p lassen sich für Diesel-

¹²Für das Kompressibilitätsmodul β_p existiert nach [Kun77] keine konkrete Formel. Das in Gleichung 5.2 angegebene Polynom ist in [CF96] aus Meßdaten hergeleitet.

kraftstoffe durch folgende Gleichungen beschreiben:

$$\begin{aligned} \rho(p) &= \rho_0 \cdot \exp(\beta_p \cdot dp - \beta_T \cdot dT) & a &= 7.4e^{-4} \\ &\text{und} & \text{; mit } b &= 3.6e^{-6} \\ \beta_p &= a - b \cdot dp + c \cdot dp^2 & c &= 1.2e^{-8} \end{aligned} \quad (5.2)$$

Der Wärmeausdehnungskoeffizient β_T kann für Temperaturen bis 200°C und somit für eine mittlere Kraftstofftemperatur von 50°C als konstant angesehen werden. Bei einer Wellenlänge λ von 0.456 m für die Rohrlänge des verwendeten Hochdruckspeichers ergibt sich eine druckabhängige Eigenfrequenz $\nu(p)$ von ca. 1500 Hz .

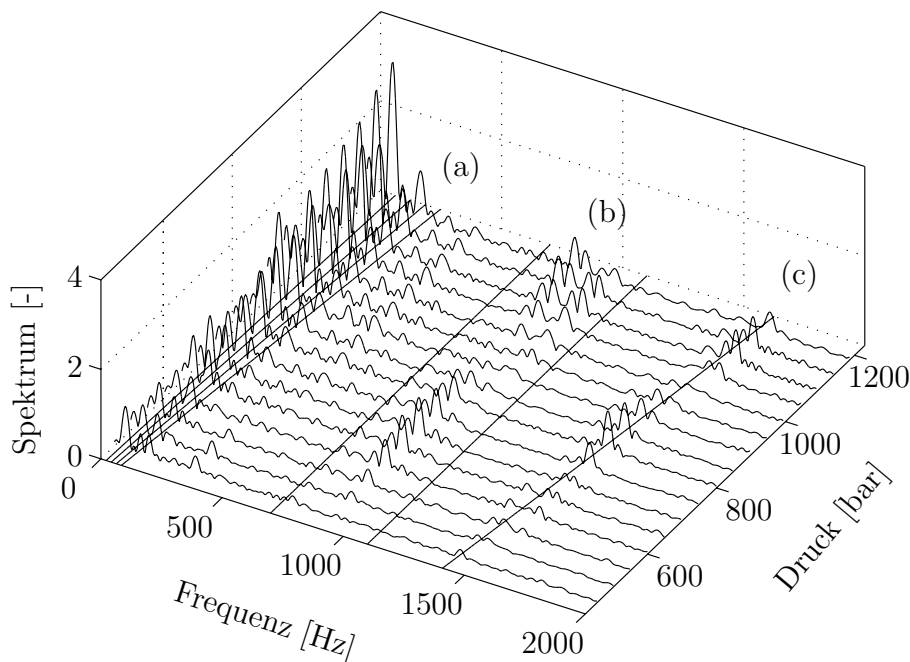


Abbildung 5.9: Normiertes Spektrum verschiedener Raildrucksignale.

In den Spektren diverser Raildrucksignale mit aufsteigenden Drücken und Drehzahlen sind 3 Bereiche auszumachen. Im Bereich (a) liegen die drehzahlabhängigen Frequenzen der Pumpe und der 6 Einspritzungen. Unter (b) befinden sich drehzahlunabhängige Frequenzen der Injektoren und der Bereich (c) läßt die Druckabhängigkeit der Frequenzen stehender Wellen erkennen.

Werden die Spektren von Raildrucksignalen diverser Arbeitspunkte gegenübergestellt, lassen sich im wesentlichen 3 Bereiche identifizieren. In Abbildung 5.9 sind dazu Spektren von Arbeitspunkten mit gleichzeitig aufsteigendem Druck ($400 - 1200\text{ bar}$) und aufsteigender Drehzahl ($500 - 3500\text{ upm}$) nebeneinander dargestellt.

Im unteren Frequenzbereich (a) lassen sich die drehzahlabhängigen Frequenzen des Pumpendrucksignals und der 6 Einspritzungen pro Arbeitsspiel identifizieren. Im mittleren Frequenzband (b) treten die drehzahlunabhängigen Frequenzen der gedämpften, harmonischen Schwingungen der Injektordrucksignale auf. Im oberen Frequenzbereich (c) ist die mit den Gleichungen 5.1 und 5.2 beschriebene Druckabhängigkeit stehender Wellen mit der von der Rohrlänge des Hochdruckspeichers abhängenden Eigenfrequenz zu erkennen.

Eine ausführliche Betrachtung der Druckabhängigkeit von Frequenzen, insbesondere der stehenden Wellen des *Common-Rail*-Systems, kann in [Fro96] und [CF96] gefunden werden.

Modellierung des Hochdruckspeichers:

Um die Vorgänge der Schwingungen im *Common-Rail*-System besser zu analysieren und das Einspritzverhalten zu simulieren, kann ein physikalisches Modell des Systems hilfreich sein. In einem ersten Ansatz ist ein solches Modell durch die Arbeit von [Ahl00] für den Hochdruckspeicher entstanden.

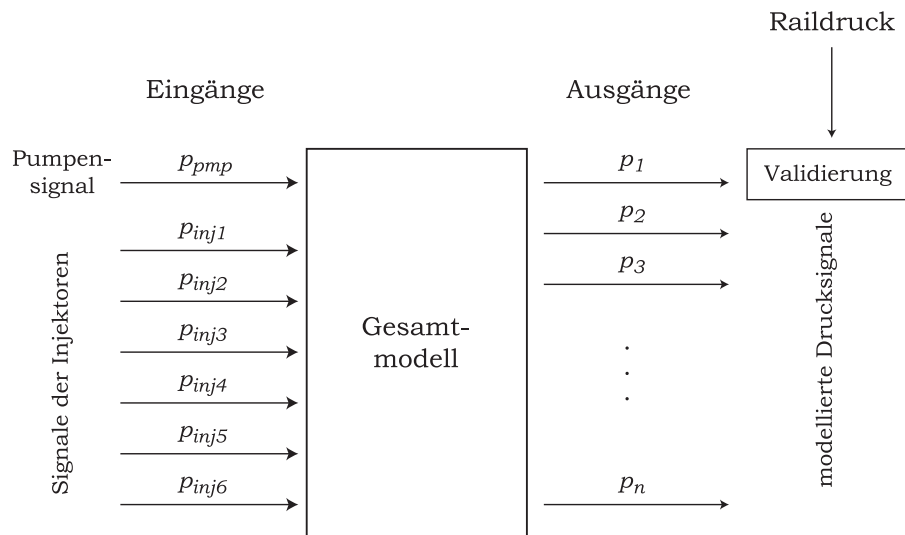


Abbildung 5.10: Gesamtmodell des Hochdruckspeichers.

Das Modell des Hochdruckspeichers soll die Schwingungen des Raildrucks aus den an den Injektoren gemessenen Einzelsignalen nachbilden. Durch Invertieren des Modells läßt sich dann aus gemessenen Raildrücken das Signal an den Injektoren gewinnen.

Abbildung 5.10 veranschaulicht den prinzipiellen Aufbau des Modells mit seinen Ein- und Ausgängen. Als Eingangssignale des Modells dienen die am *Common-Rail*-System aufgezeichneten Messungen des Pumpendrucksignals p_{pmp} und der Injektordrucksignale $p_{inj1} \dots p_{inj6}$. Die Modellausgänge sind über die Länge des Hochdruckspeichers verteil-

te Drucksignale $p_1 \dots p_n$, wobei auch das meßbare Raildrucksignal p_1 zu Validierungszwecken modelliert wird. Durch Invertieren eines solchen Modells läßt sich dann aus dem Raildrucksignal das Druckverhalten an den Injektoren modellieren und mit Hilfe eines Injektormodells auch die Einspritzung, sowie deren Menge.

Das Pumpendrucksignal entspricht dabei der Darstellung aus Abbildung 5.7 und spiegelt den Druckaufbau einer Dreistempelpumpe wider. Die ebenfalls festgestellte, gedämpfte, harmonische Schwingung des Injektordrucksignals ist eine Folge des schlagartigen Schließens der Injektoren am Ende jeder Einspritzung. Sie entsteht durch den damit verbundenen Druckrückschlag in der Injektordruckleitung.

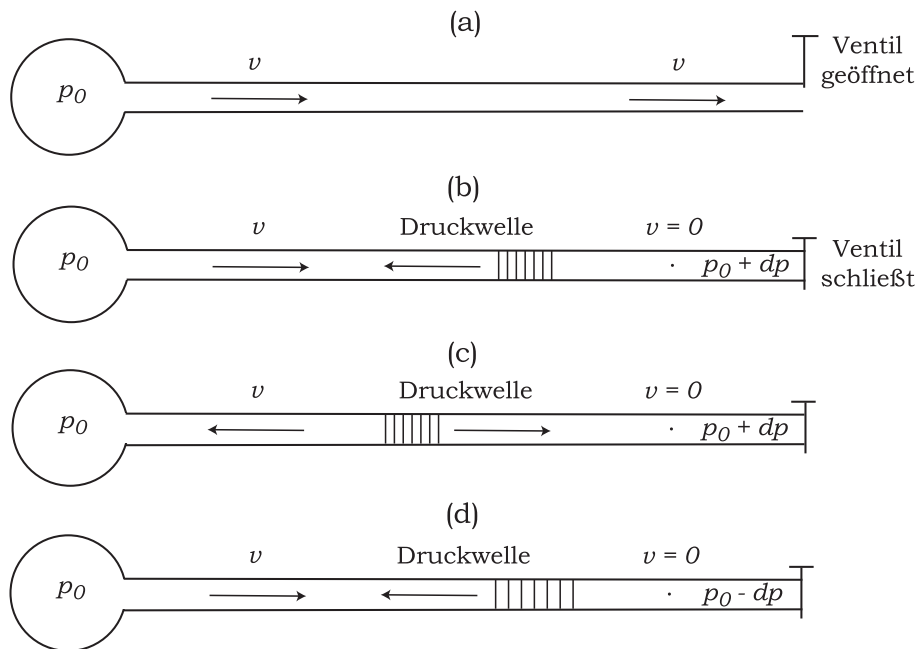


Abbildung 5.11: Druckrückschläge schließender Injektoren.

Bei geöffneten Injektorventilen (a) fließt Kraftstoff mit einer Geschwindigkeit v vom Hochdruckspeicher zum Injektor. Schließt nun der Injektor, erfährt der fließende Kraftstoff einen Druckrückschlag (b). Durch den weiterhin bestehenden Geschwindigkeitsunterschied in der Zuleitung wechselt die Welle wie unter (c) und (d) gedämpft zwischen Hochdruckspeicher und Injektor hin und her.

Abbildung 5.11 veranschaulicht die Entstehung von Schwingungen, die durch solche Druckrückschläge auftreten. Ist das Ventil des Injektors geöffnet, so herrscht in der Injektordruckleitung der gleiche Druck p_0 wie im Hochdruckspeicher und der Kraftstoff fließt entsprechend (a) mit einer Geschwindigkeit v . Schließt nun das Ventil schlagartig, so kommt der Kraftstoff vor dem Ventil zum Stillstand. Da sich die restliche Flüssigkeit in der Leitung weiterhin bewegt, kommt es wie unter (b) zu einer Druckwelle, ausgehend

vom Staudruck $p_0 + dp$ vor dem Ventil. Erreicht die Druckwelle den Hochdruckspeicher, so hat die Injektordruckleitung einem um dp höheren Druck und es kommt zu einem Rückfluß in den Druckspeicher, was auch wie in (c) zur Reflexion der Druckwelle führt. Erreicht die Druckwelle das Ventil, ist der Kraftstoff immer noch in Bewegung und der Druck $p_0 - dp$ ist nun geringer als der ursprüngliche Wert und die Druckwelle wird entsprechend (d) erneut reflektiert.

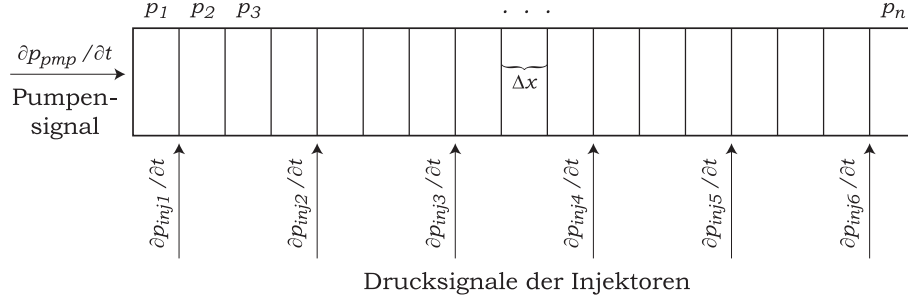


Abbildung 5.12: Modell des Hochdruckspeichers.

Zur Berechnung der erweiterten, allgemeinen Wellengleichung ist der Hochdruckspeicher in n Teilstücke der Breite Δx zu zerlegen.

Die Modellierung des Hochdruckspeichers kann entsprechend [Ahl00] über die eindimensionale, allgemeine Wellengleichung nach [Ing88] erfolgen. Sind Dichte ρ und dynamische Viskosität μ zu berücksichtigen, so kann folgende erweiterte Version der allgemeinen Wellengleichung angegeben werden:

$$\frac{\partial^2 p}{\partial t^2} = c^2 \cdot \frac{\partial^2 p}{\partial x^2} - \frac{4}{3} \frac{\mu}{\rho} \cdot \frac{\partial^3 p}{\partial x^2 \partial t} \quad (5.3)$$

Dabei ist p der Druck und c die Geschwindigkeit der Welle. Da Gleichung 5.3 den Druck zu jedem Zeitpunkt t und an jeder Position x des Hochdruckspeichers angibt, ist eine begrenzte Anzahl n an Positionen zu wählen, um die Wellengleichung zu repräsentieren. Durch Differenzieren in beide Richtungen des Raums mit $i = 2 \dots n - 1$ läßt sich die Gleichung beschreiben durch:

$$\frac{\partial^2 p_i}{\partial t^2} = c^2 \cdot \frac{p_{i+1} - 2p_i + p_{i-1}}{(\Delta x)^2} - \frac{4}{3} \frac{\mu}{\rho} \cdot \frac{\partial \left(\frac{p_{i+1} - 2p_i + p_{i-1}}{(\Delta x)^2} \right)}{\partial t} \quad (5.4)$$

Zur Überprüfung des Modells kann der Modellausgang p_1 mit dem am Hochdruckspeicher real gemessenen Signal p_{rail} verglichen werden. Ein entsprechender Vergleich ist durch Abbildung 5.13 gegeben. Anhand diverser Arbeitspunkte ist die Leistungsfähigkeit des physikalischen Modells zu erkennen.

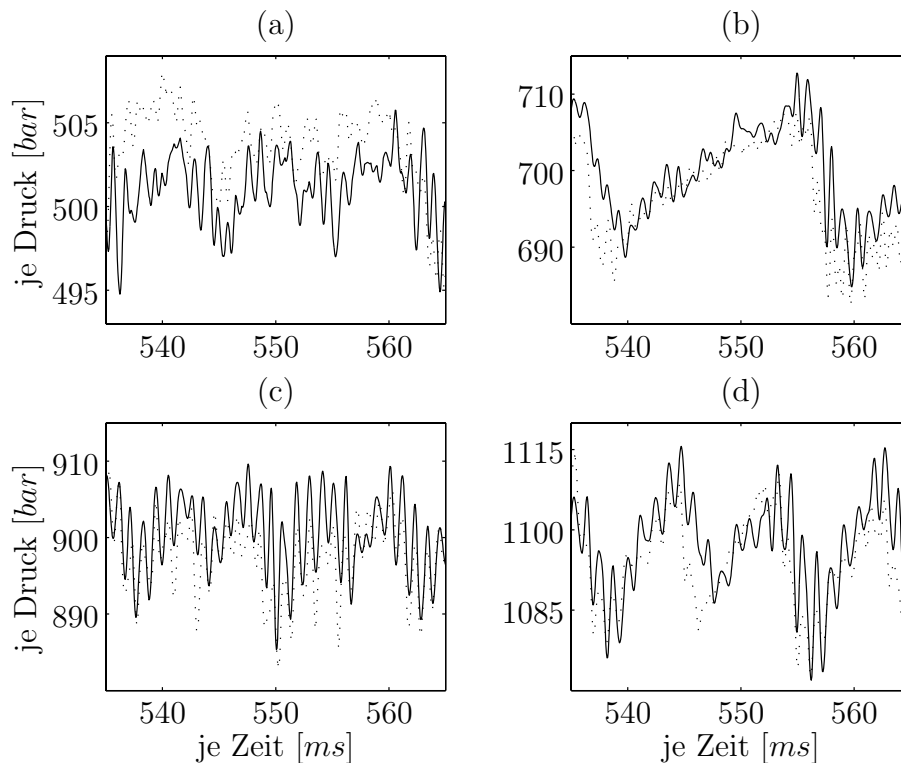


Abbildung 5.13: Vergleich von realen- und Modelldruckschwingungen.

Das reale Drucksignal p_{rail} (durchgezogen) im Vergleich zu den Modellsignalen p_1 (gestrichelt) für 4 verschiedene Arbeitspunkte mit:

- (a) Druck 500 bar, Drehzahl 1000 *upm*, Einspritzmenge 10 mm^3 .
- (b) Druck 700 bar, Drehzahl 1800 *upm*, Einspritzmenge 50 mm^3 .
- (c) Druck 900 bar, Drehzahl 2400 *upm*, Einspritzmenge 20 mm^3 .
- (d) Druck 1100 bar, Drehzahl 3000 *upm*, Einspritzmenge 70 mm^3 .

Das Modell erreicht eine Genauigkeit über alle Arbeitspunkte mit einer durchschnittlichen, absoluten Abweichung von 3.62 bar. Diese Abweichung entspricht, in Anbetracht der durch die Einspritzungen ausgelösten Schwingungsamplituden von 10 – 40 bar, einem Fehler von durchschnittlich 18%. Dieses Ergebnis ist für den gewählten, einfachen Modellansatz befriedigend.

Die bisherige Modellierung hat in jedem Fall zu einer weiteren, wichtigen Erkenntnis geführt. Die Ausprägung der Druckschwingungen ist an diversen Positionen des Hochdruckspeichers stark gedämpft. Diese Tatsache führt zu unterschiedlichen Ausprägungen der einzelnen Injektordrucksignale, sowohl am Injektor, als auch im Raildrucksignal. Dieses Verhalten erschwert sowohl die Injektordruck-, als auch die Raildruckauswertung.

Die vorgestellte, physikalische Modellierung des *Common-Rail*-Systems weist für den Einsatz zur Einspritzmengenbestimmung folgende Probleme auf:

- Analysen der modellierten Drucksignale haben ergeben, daß trotz Berücksichtigung von Reflexionen, die Ausprägung stehender Wellen des Systems nicht gelungen ist.
- Für eine Verwendung als Echtzeitanwendung in der Motordiagnose ist die Berechnung eines selbst einfachen Modells zu zeitaufwendig, um die Einspritzmenge zylinderindividuell und arbeitsspielaufgelöst zu bestimmen.
- Eine Modellierung des Injektors mit seinen elektrohydraulischen Komponenten ist als sehr komplex anzusehen und ein derartiges Modell besteht bisher noch nicht.

Die in diesem Abschnitt dargestellten Analysen haben einige wichtige Erkenntnisse zum *Common-Rail*-System geliefert. So ist mit Hilfe verschiedener Frequenzanalysen dargestellt worden, daß es im Raildrucksignal sowohl drehzahlabhängige, -unabhängige, sowie druckabhängige Frequenzen gibt. Filter zur Signalvorverarbeitung sind auf diese Frequenzen abzustimmen. Das im Abschluß diskutierte physikalische Modell des Hochdruckspeichers konnte die unterschiedliche Ausprägung von Druckschwingungen der einzelnen Injektoren bestätigen. Diese Unterschiede ergeben sich durch die geometrischen Begebenheiten des Systems und erschweren jede und somit auch eine neuronale Modellierung des Gesamtsystems.

5.3.3 Informationsoptimierte Datenanalyse

Künstliche neuronale Netze können ein System nur mit der Qualität abbilden, die durch die Trainingsmenge der Meßdaten gewährleistet ist. Um aus den Messungen eine geeignete Trainingsmenge zu generieren ist eine gründliche Analyse der Meßsignale erforderlich.

In Abschnitt 5.3.2 sind herkömmliche Methoden zur Analyse des *Common-Rail*-Systems dargestellt worden. Sie können als Basis zur Signalvorverarbeitung und somit auch für den Entwurf geeigneter Filter dienen. Auf Grund der gewonnenen Information stehen folgende Filter zur Diskussion:

- Tiefpaßfilter bis 700 Hz , um alle amplitudenstarken, jedoch drehzahlabhängigen Frequenzen zu erfassen.
- Filter, die Frequenzen bei 1000 Hz umfassen, damit drehzahlunabhängige Frequenzen eingeschlossen sind.
- Filter zur Erfassung der druckabhängigen Frequenzen um 1500 Hz .

Welche Frequenzbereiche den größten Anteil der Information zur Bestimmung der eingespritzten Kraftstoffmenge beinhalten, kann durch die bisherigen Untersuchungen nicht erkannt werden. Jedoch geben sie einen ersten Anhaltspunkt und schränken die Suche nach optimalen Filtern ein.

Die in Kapitel 4 neu entwickelte Methode der informationsoptimierten Frequenzanalyse kann gerade darüber Aufschluß geben, welche Frequenzbereiche einen großen Teil der Information zur Einspritzmengenbestimmung beinhalten. Dazu verarbeitet der Algorithmus 4.2.2 aus Abschnitt 4.2 zur Frequenzanalyse nicht nur Raildrucksignale, sondern auch die entsprechenden Einspritzverläufe. Abbildung 5.14 veranschaulicht zwei Signale, die bei der informationsoptimierten Frequenzanalyse als Systemeingang $x(t)$ und als Systemausgang $y(t)$ dienen können.

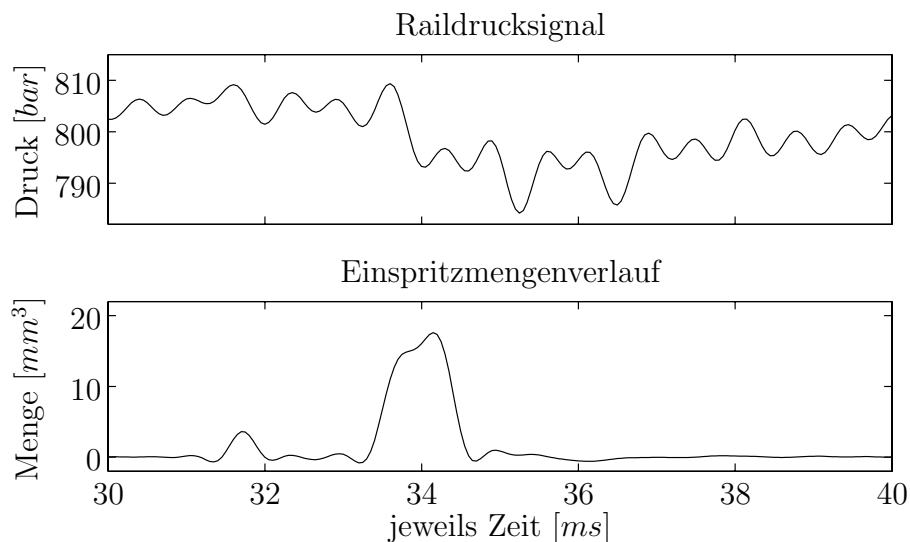


Abbildung 5.14: Signale zur informationsoptimierten Frequenzanalyse.

Bei der Raildruckanalyse durch Algorithmus 4.2.2 ist das Raildrucksignal als Systemeingang $x(t)$ und der Einspritzmengenverlauf als Systemausgang $y(t)$ zu verstehen.

Mit Hilfe des Algorithmus 4.2.2 und der in Abschnitt 4.2 angegebenen Formeln, läßt sich aus $x(t)$ und $y(t)$ bestimmen, welche Frequenzen des Raildrucksignals den größten Informationsanteil am Einspritzmengenverlauf und somit an der Einspritzmenge haben.

In Abbildung 5.15 sind die Fourier- und informationsoptimierte Frequenzanalyse gegenübergestellt. Die Auflösung der Frequenzen bei der Fourier-Analyse ist erheblich feiner. Dennoch werden wesentliche Frequenzen auch von der informationsoptimierten Frequenzanalyse erkannt und gewichtet. Die Gewichtung auf Basis der Transinformation erfolgt in Bezug auf die Einspritzverläufe der entsprechenden Raildruckschwingungen. Durch das in Kapitel 4.2 vorgestellte Bewertungskriterium 4.4 können alle

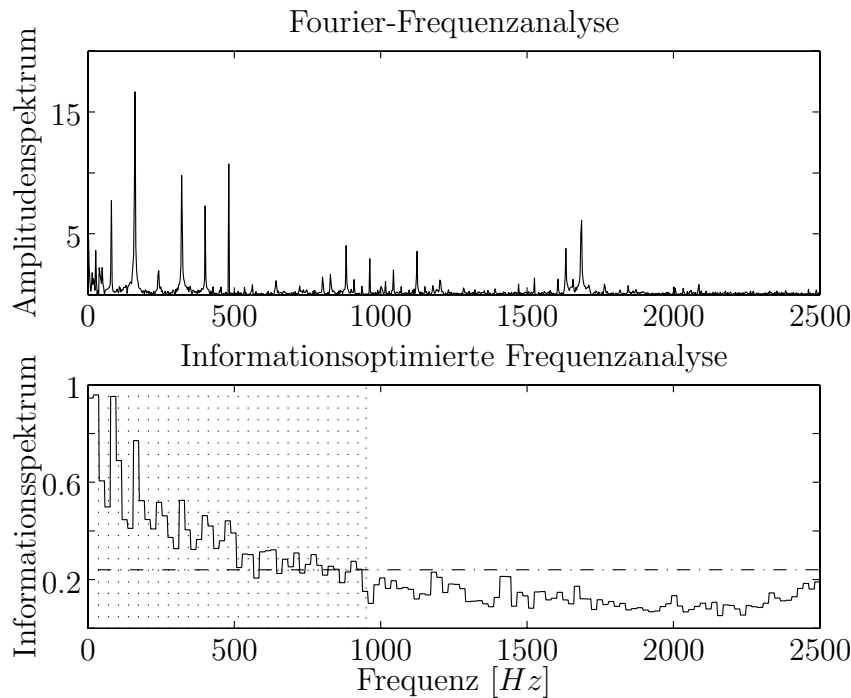


Abbildung 5.15: Gegenüberstellung Fourier- und informationsoptimierte Analyse.

Die Auflösung der Frequenzen bei der Fourier-Analyse ist erheblich feiner. Dennoch werden wesentliche Frequenzen auch von der informationsoptimierten Frequenzanalyse erkannt und gewichtet.

Frequenzen unterhalb 950 Hz als relevant erkannt werden. Diese Erkenntnisse der informationsoptimierten Frequenzanalyse liefert die Grenzfrequenz für einen an die Aufgabe angepaßten Tiefpaßfilter. Die aus der Datenanalyse gewonnenen Erkenntnisse sind durch folgende Vorverarbeitungsschritte umgesetzt:

1. **Y-Translation:** Die Amplituden der Druckschwingungen ($10 - 40\text{ bar}$) der einzelnen Einspritzungen sind im Verhältnis zum Raildruckniveau ($400 - 2000\text{ bar}$) sehr klein. Zur neuronalen Raildruckauswertung sind daher die Raildrucksignale durch eine Mittelwertbereinigung auf ein einheitliches Druckniveau zu bringen.
2. **X-Translation:** Mit steigender Drehzahl kommen die Druckschwingungen der einzelnen Einspritzungen immer dichter zu liegen und die Fenstergröße ist so zu wählen, daß sich die Druckschwingungen der zylinderindividuellen Einspritzungen nicht überlappen. Dies erfordert eine von den Einspritzungen getriggerte Fensterung des Raildrucksignals.
3. **Filterung:** Entsprechend der Frequenzanalyse tragen die drehzahlabhängigen, tiefen Frequenzen den größten Informationsanteil. Aber auch die drehzahlunabhängigen Frequenzen enthalten wichtige Informationsanteile, was zu einem Tiefpaßfilter mit einer Grenzfrequenz von 1000 Hz führt.

Bewertung der informationsoptimierten Datenanalyse:

Die Anwendung, der bei der Datenanalyse gewonnenen Erkenntnisse können die Abweichungen der Netzausgaben reduzieren. Eine Bewertung erfolgt anhand der Raildruckauswertung zur Bestimmung der Einspritzmenge, wobei einige Besonderheiten zu beachten sind.

Durch die Abtastfrequenz bei der Datenaufzeichnung von 20.000 Hz und die gewählte Fensterung zur Trennung der Einspritzungen, ist die Anzahl der Stützstellen mit 260 recht hoch. Für eine erste, einfache Reduzierung der Eingangsdimension wird nur jeder 4. Stützstelle verwendet, was einer Eingangsgröße von 65 Werten und einer Signalabtastung mit 5.000 Hz entspricht. Weitere Netzparameter können der Tabelle 5.3 entnommen werden.

Argument	Parameter
X-Translation	Einspritzgetriggerte Fensterung
Y-Translation	Mittelwertebereinigung
Simulator	MATLAB
Netztopologie	$65 \times 50 \times 1$ (feed-forward)
Transferfunktion	' logsig ' (logarithmisch sigmoid)
Initialisierung	' initnw ' (Nguyen-Widrow)
Lernregel	' trainrp ' (<i>Resilient Propagation</i>)
Trainingsparameter	$\Delta_{inc} = 1.01$, $\Delta_{dec} = 0.25$, $\Delta_0 = 0.07$, $\Delta_{max} = 50$
Abbruchbedingung	ansteigender Validierungsfehler

Tabelle 5.3: Trainingsparameter zur informationsoptimierten Datenanalyse.

Die Balkendiagramme der Fehlerauswertung in Abbildung 5.16 veranschaulichen den Einfluß der Filtergrenzen auf den Ausgabefehler des neuronalen Netzes. Dabei können die in Kapitel 2, Abschnitt 2.2.3 beschriebenen Fehlermaße verglichen werden. Ebenfalls wird zwischen Trainings-, Validierungs- und Testdaten unterschieden. In jedem Teildia-gramm sind vier verschiedene Datensätze mit unterschiedlichen Filtern gegenübergestellt. Die Auswahl der Filtergrenzen spiegelt dabei die Erkenntnisse der Datenanalyse wider (vgl. auch Abbildung 5.9):

- A** Tiefpaßfilter mit $0 - 1000\text{ Hz}$: Von der informationsoptimierten Frequenzanalyse als relevant erkannte Frequenzen.
- B** Tiefpaßfilter mit $0 - 1700\text{ Hz}$: Alle bei der Frequenz-Analyse untersuchten und interpretierten Frequenzen.
- C** Tiefpaßfilter mit $0 - 700\text{ Hz}$: Drehzahlabhängige Frequenzen, die Schwingungen der 6 Einspritzungen abdecken.
- D** Bandpaßfilter mit $700 - 1100\text{ Hz}$: Frequenzband der drehzahlunabhängigen, harmonischen Schwingungen der Injektoren.

Bei den verwendeten Filtern handelt es sich um Chebyshev Filter vom Typ II. Die Flanken der Filter erfahren nach 100 Hz eine Dämpfung um 20 dB . Die Dämpfung im Paßband darf maximal 0.3 dB betragen.

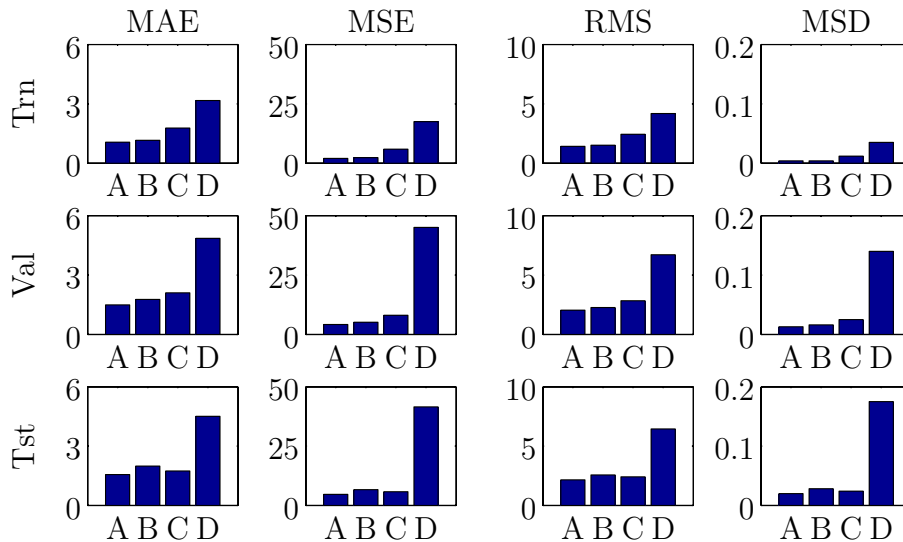


Abbildung 5.16: Auswertung unterschiedlicher Filtergrenzen.

Gegenüberstellung verschiedener Datensätze mit diversen Fehlermaßen für Trainings-, Validierungs- und Testdaten. Dabei unterscheidet sich die Vorverarbeitung der Datensätze nur in den Filtergrenzen:

- A: Tiefpaßfilter mit $0 - 1000\text{ Hz}$ B: Tiefpaßfilter mit $0 - 1700\text{ Hz}$
 C: Tiefpaßfilter mit $0 - 700\text{ Hz}$ D: Bandpaßfilter mit $700 - 1100\text{ Hz}$

Wie in allen Diagrammen zu erkennen sind, fallen die Fehler der Netzausgabe unter Verwendung des Datensatzes A am geringsten aus. Damit sind die durch die informationsoptimierte Frequenzanalyse gefundenen Frequenzgrenzen zum Netztraining am besten geeignet.

Neben den herkömmlichen Methoden zur Eingrenzung notwendiger Frequenzen hat die informationsoptimierte Frequenzanalyse einen wichtigen Beitrag zum Verständnis der Signalzusammenhänge geliefert. Sie konnte alle relevanten Frequenzen voll automatisch identifizieren und ist somit eine gute Grundlage zum Filterdesign. Die neue Methode der informationsoptimierten Frequenzanalyse hat sich bei der Raildruckauswertung als geeignetes Werkzeug zur Frequenzanalyse herausgestellt.

5.3.4 Reduzierung der Eingangsdimensionalität

Die Anzahl verfügbarer Stützstellen sprengt mit 260 den Rahmen als Eingangsdimensionalität neuronaler Netze im Hinblick auf eine Anwendung im Fahrzeug und in Echtzeit. Durch eine geeignete Stützstellenauswahl kann nicht nur die Eingangsdimensionalität neuronaler Netze reduziert werden, sondern auch die gesamte Netztopologie und somit auch der Berechnungsaufwand während der Anwendungsphase.

Abbildung 5.17 veranschaulicht die unterschiedlichen Verfahren zur Reduzierung der Eingangsdimensionalität. Aus der Abbildung können folgende Aussagen zu den Verfahren gezogen werden:

Äquidistante Abtastung:

- Nicht relevante Druckpositionen erhöhen die Eingangsdimensionalität unnötig und zwingen zu einer großen Netzstruktur.
- Selbst bei einer entsprechend der Datenanalyse minimal erforderlichen Abtastung von 2500 Hz (jeder 8. Abtastwert) sind 32 Stützstellen zu berücksichtigen.

Korrelationskoeffizienten:

- Bezogen auf lineare Abhängigkeiten erscheinen unverhältnismäßig viele Druckpositionen vor der Einspritzung als relevant.
- Durch die unscharfe Angabe von relevanten Bereichen ist wiederum eine äquidistante Abtastung erforderlich.

Transinformationsanalyse:

- Nur einzelne Druckpositionen sind ausgewählt, was keine äquidistante Abtastung erfordert.
- Kurz nach Beginn der Haupteinspritzung häufen sich Druckpositionen mit hohem Informationsgehalt.
- Zur Klassifikation von Druckniveau und Schwingungstendenz sind einige Druckpositionen vor der Einspritzung ausgewählt.

Diese Zusammenstellung macht deutlich, welche Vorteile die Transinformationsanalyse gegenüber den anderen Verfahren bietet. Zur Reduzierung der Eingangsdimensionalität durch Transinformation ist der Algorithmus 4.3.1 aus Kapitel 4 zum Einsatz gekommen. Dabei besteht die Eingangsmatrix $X(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}$ mit $t = \{1, 2, \dots, 260\}$ aus N Messungen von Raildruckschwingungen unterschiedlicher Einspritzmengen. Die zugehörigen gemessenen N Einspritzmengen bilden einen Zeilenvektor $Y = \{y_1, y_2, \dots, y_N\}$ als einzigen Eintrag der Ausgangsmatrix Y . Das Ergebnis der ersten 14 Iterationen und somit der durch den Algorithmus gefundenen Stützstellen ist in Abbildung 5.17 (c) zu sehen.

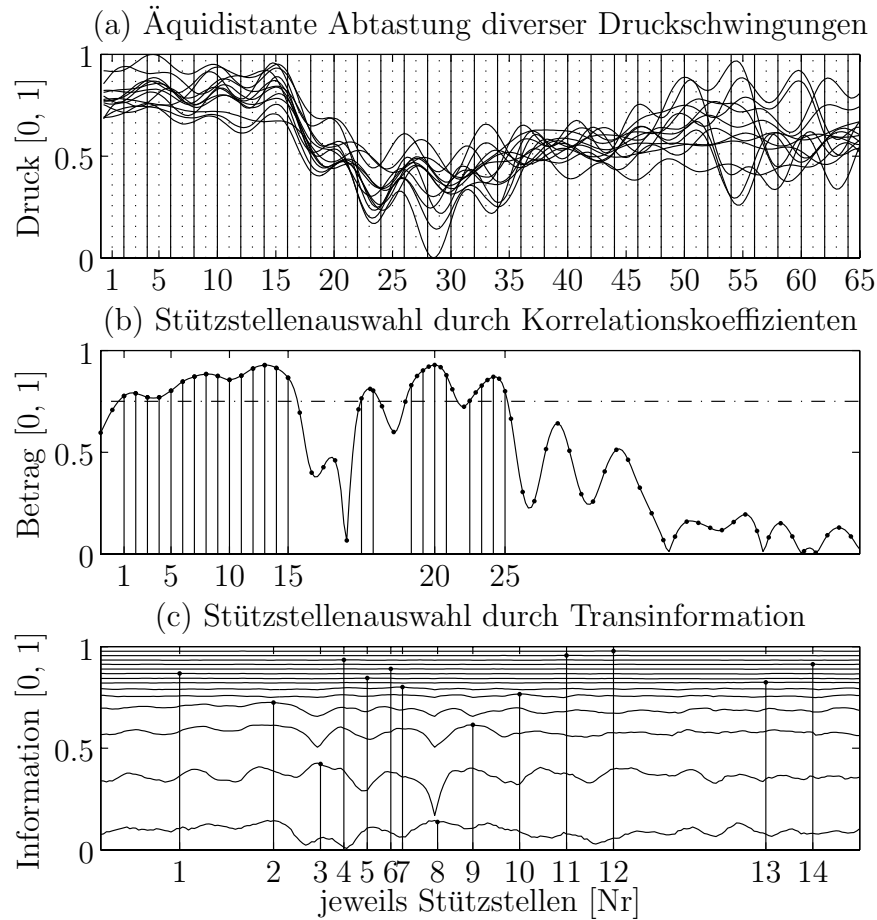


Abbildung 5.17: Methoden zur Reduzierung der Eingangsdimensionalität.

Reduzierung der Stützstellenanzahl durch unterschiedliche Methoden:

- (a) Verlauf von Raildruckschwingungen ausgewählter Arbeitspunkte. Andeutung der äquidistanten Abtastung jeder 4. (gestrichelt) bzw. jeder 8. (durchgezogen) Druckposition der Datenaufzeichnung.
- (b) Funktionsverlauf der Korrelationskoeffizienten bezogen auf Raildruckschwingungen und zugehörige Einspritzmengen. Darstellung einer äquidistanten Abtastung (jeder 4. Wert), sowie der durch den Schwellwert von 75 % ausgewählten Stützstellen.
- (c) Funktionsverläufe der mehrdimensionalen Transinformationswerte bezogen auf Raildruckschwingungen und zugehörige Einspritzmengen. Darstellung der ausgewählten Maximalwerte einer jeden Iteration. Die ersten 10 Stützstellen liefern ca. 75% der Information.

Neben den hier vorgestellten Verfahren ist auch die Verwendung von Hauptkomponenten gebräuchlich. Untersuchungen haben gezeigt, daß unter Verwendung der Hauptkomponentenanalyse ca. 98% der Information durch 10 Komponenten abgedeckt sind.

Da die Berechnung der Hauptkomponenten auch während der Anwendungsphase erfolgen muß, wurde dieses Verfahren im Hinblick auf den Einsatz im Fahrzeug nicht in Betracht gezogen.

Stützstellenauswahl durch ...	Dimension	Netztopologie	Gewichte
äquidistant jeder 4. Wert	65	65x50x1	3300
äquidistant jeder 8. Wert	32	32x50x1	1650
Korrelationskoeffizienten	25	25x50x1	1300
Transinformationsanalyse	14	14x50x1	750

Tabelle 5.4: Unterschiedliche Verfahren zur Reduktion der Eingangsdimensionalität. Die Zahlenwerte veranschaulichen den Einfluß auf die Gesamtanzahl der Gewichte.

Tabelle 5.4 faßt die Möglichkeiten zur Reduzierung der Eingangsdimensionalität zusammen, die bei der Einspritzmengenbestimmung durch neuronale Druckauswertung zur Anwendung gekommen sind. Es ist ersichtlich, welchen Einfluß die Eingangsdimensionalität auf die gesamte Netztopologie hat und wie sehr die Gesamtzahl der Gewichte reduziert werden kann. Da die Anzahl der Gewichte einen großen Einfluß auf den Berechnungsaufwand hat, ist eine Reduzierung der Eingangsdimensionalität im Hinblick auf eine Berechnung im Fahrzeug von hohem Stellenwert.

Bewertung der informationsoptimierten Eingangsdimension:

Zum direkten Vergleich sind den äquidistanten Abtastungen entsprechend Tabelle 5.4 die Trainingsergebnisse sowohl der Korreltaions- als auch der Transinformationsanalyse gegenübergestellt. Die verwendeten Netztopologien können der gleichen Tabelle entnommen werden.

Argument	Parameter
X-Translation	Einspritzgetriggerte Fensterung
Y-Translation	Mittelwertebereinigung
Tiefpaßfilter	0 – 1000 Hz
Simulator	MATLAB
Transferfunktion	'logsig' (logarithmisch sigmoid)
Initialisierung	'initnw' (Nguyen-Widrow)
Lernregel	'trainrp' (<i>Resilient Propagation</i>)
Trainingsparameter	$\Delta_{inc} = 1.01$, $\Delta_{dec} = 0.25$, $\Delta_0 = 0.07$, $\Delta_{max} = 50$
Abbruchbedingung	ansteigender Validierungsfehler

Tabelle 5.5: Beschreibung der Trainingsparameter zur Eingangsdimensionalität.

Die Vorverarbeitung entspricht dabei den Erkenntnissen aus Abschnitt 5.3.3 zur Datenanalyse. Tabelle 5.5 faßt die verwendeten Verarbeitungsschritte und alle Lernparameter zusammen. Um vergleichbare Ergebnisse zu erzielen, erfolgte das Training mit

identischen Parametern, jedoch unterschiedlichen Datensätzen. Dabei blieb die Vorverarbeitung der Daten identisch, lediglich die Druckpositionen wurden durch Transinformation bzw. Korrelation ausgewählt. Die besten Trainingsergebnisse sind durch Balkendiagramme in Abbildung 5.18 gegenübergestellt:

- A $65 \times 50 \times 1$ -Topologie mit 3300 Gewichten: Auswahl durch eine äquidistante Abtastung unter Verwendung jedes 4. Wertes.
- B $14 \times 50 \times 1$ -Topologie mit 750 Gewichten: Stützstellenauswahl unter Verwendung der mehrdimensionalen Transinformation.
- C $32 \times 50 \times 1$ -Topologie mit 1650 Gewichten: Wie A jedoch mit einer äquidistanten Abtastung unter Verwendung jedes 8. Wertes.
- D $25 \times 50 \times 1$ -Topologie mit 1300 Gewichten: Stützstellenauswahl unter Bewertung von Korrelationskoeffizienten mit äquidistanter Abtastung in Bereichen großer linearer Abhängigkeit.

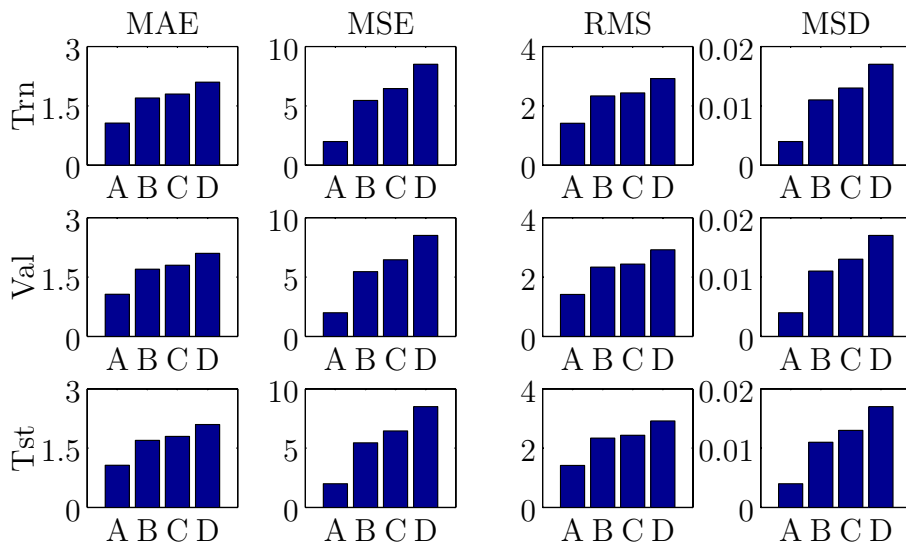


Abbildung 5.18: Auswertung unterschiedlicher Stützstellen.

Gegenüberstellung verschiedener Datensätze mit diversen Fehlermaßen für Trainings-, Validierungs- und Testdaten. Dabei unterscheidet sich die Auswahl der Stützstellen durch:

- A: Äquidistant 65 Werte (3300 Gewichte)
- B: Transinformation (750 Gewichte)
- C: Äquidistant 32 Werte (1650 Gewichte)
- D: Korrelation (1300 Gewichte)

Wie zu erkennen ist, sind die Ergebnisse für eine Reduzierung der Eingangsdimensionalität durch Transinformation besser, als bei der Auswahl aufgrund der Korrelationskoeffizienten, obwohl das Netzwerk weniger Gewichte enthält. Auch wenn sich die

Trainingsergebnisse gegenüber der äquidistanten Abtastung mit 65 Eingangsneuronen verschlechtert haben, so liegen sie im Bereich der äquidistanten Abtastung mit 32 Neuronen. Zudem ist die Verschlechterung weniger gravierend, unter Betrachtung der gewonnenen Reduzierung von 3300 zu 750 Gewichten.

Ein Nachteil des verwendeten Verfahrens soll jedoch nicht verschwiegen bleiben. Bei der mehrdimensionalen Transinformation besteht das Problem, daß schon ab 10 Komponenten jede weitere Auswahl zunehmend statistischen Grenzen unterliegt und die Informationsmaxima weniger deutlich heraustreten (siehe Abbildung 5.17 (c)). Somit ist die Anzahl an Stützstellen begrenzt, die durch das Verfahren ermittelt werden können.

Diesem Problem kann durch den Algorithmus 4.4.1 zur informationsoptimierten Netztopologie aus Kapitel 4, Abschnitt 4.4 begegnet werden (siehe Beispiel 4.4.1). Auch die im Folgenden Abschnitt 5.3.5 gewonnene Netztopologie mit 49 Neuronen in der Eingangsschicht spricht für die bessere Abdeckung des Eingangsvektors durch dieses weitere Verfahren. Zudem optimiert der im Folgenden Kapitel demonstrierte Algorithmus nicht nur die Eingangsdimension eines Netzes, sondern die gesamte Netztopologie.

5.3.5 Informationsoptimierte Netztopologie

Auch wenn die Reduzierung der Eingangsdimensionalität des letzten Abschnitts die Anzahl der Verbindungsgewichte von anfänglich 3300 auf 750 Verbindungen reduzieren konnte, bestehen weitere Möglichkeiten die Netztopologie zu verkleinern. Bezogen auf eine Umsetzung im Fahrzeug stellt der Speicherbedarf von 750 Fließkommawerten pro Zylinder für heutige Steuergeräte immer noch einen zu großen Umfang dar.

Zur Beeinflussung bzw. weiteren Reduzierung der Netztopologie stehen verschiedene Verfahren zu Verfügung. Tabelle 5.3.5 vergleicht die Ergebnisse unterschiedlicher Verfahren zur Optimierung der Netzwerktopologie, die bei der Bestimmung der Einspritzmenge zur Anwendung kamen.

Topologieoptimierung durch ...	Netztopologie	Gewichte
<i>Pruning</i> -Algorithmus	14x34x1	353
<i>Cascade Correlation</i>	14x23x1	262
Informationsoptimierung	49x13x1	111

Tabelle 5.6: Verfahren zur Optimierung der Netztopologie.

Dabei fügen *Cascade Correlation* und Informationsoptimierung einzelne Neuronen zu kleinen Netzstrukturen hinzu, wohingegen der *Pruning*-Algorithmus große Netze durch Entfernen einzelner Neuronen verkleinert. Wie zu erkennen ist, können die konstruierenden Algorithmen kleinere Netztopologien erzeugen. Die auf Transinformation basierende, neu entwickelte Methode zum Aufbau von informationsoptimierten Netztopologien bildet die kleinste Netzstruktur mit der geringsten Anzahl an Gewichten.

Die Besonderheiten der verschiedenen Methoden sind im Folgenden zusammengefaßt:

Pruning-Algorithmen:

- Als trainiertes Ausgangsnetzwerk kam das in der Eingangsschicht durch Transinformation optimierte 14x50x1-Netzwerk zum Einsatz (siehe Abschnitt 5.3.4).
- Zum *Node-Pruning* ist das Verfahren *Skeletonization* zur Anwendung gekommen. Allerdings sind dabei nur verdeckte Neuronen entfernt worden.
- Beim *Link-Pruning* konnte der *Optimal-Brain-Damage*-Algorithmus gute Dienste leisten.
- Als Abbruchkriterium ist ein *MSE* von 8.0 für den Validierungsfehler gewählt worden.

Cascade Correlation:

- Als Eingangsvektor sind die durch Transinformation ermittelten 14 Stützstellen der Raildruckschwingungen zum Einsatz gekommen (siehe Abschnitt 5.3.4).
- Nach jedem Einfügen einer neuen Zelle ist ein *Link-Pruning* durch *Optimal-Brain-Damage* erfolgt.
- Als Abbruchkriterium ist ein ansteigender Validierungsfehler gewählt worden.

Informationsoptimierte Netztopologie:

- In jedem Iterationsschritt (einfügen eines Neurons) stehen alle 260 Stützstellen der Raildruckschwingung zur Auswahl zur Verfügung.
- Die verdeckten Neuronen werden gezielt und informationsoptimiert verbunden, wodurch nur wenige Stützstellen bzw. verdeckte Neuronen ausgewählt werden.
- Als Abbruchkriterium ist wie bei *Cascade Correlation* ein ansteigender Validierungsfehler gewählt worden.

Werden beim Einsatz des *Pruning*-Algorithmus bzw. von *Cascade Correlation* ebenfalls alle 260 zur Verfügung stehenden Stützstellen der Raildruckschwingung verwendet, entstehen noch größere Netztopologien als die beschriebenen. In diesem Fall würde *Cascade Correlation* schon mit einer sehr großen Eingangsschicht starten und der *Pruning*-Algorithmus hätte eine sehr große Netzstruktur zu reduzieren, was leicht zu einem starken Anstieg des Fehlers der Netzausgabe führt.

Die erzeugte informationsoptimierte Netztopologie wird im Folgenden genauer untersucht und beschrieben. Abbildung 5.19 stellt die zur Reduzierung der Eingangsdimension ausgewählten Stützstellen des Algorithmus 4.3.1 den Stützstellen des Algorithmus 4.4.1 zur informationsoptimierten Netztopologie gegenüber.

Gut zu erkennen ist die verstärkte Auswahl von Stützstellen im Bereich des größten Druckabfalls (Bereich der Haupteinspritzung). Nur wenige Druckpositionen aus den Bereichen vor und nach der Haupteinspritzung konnten durch den Algorithmus zur Netzwerkoptimierung als relevant identifiziert werden.

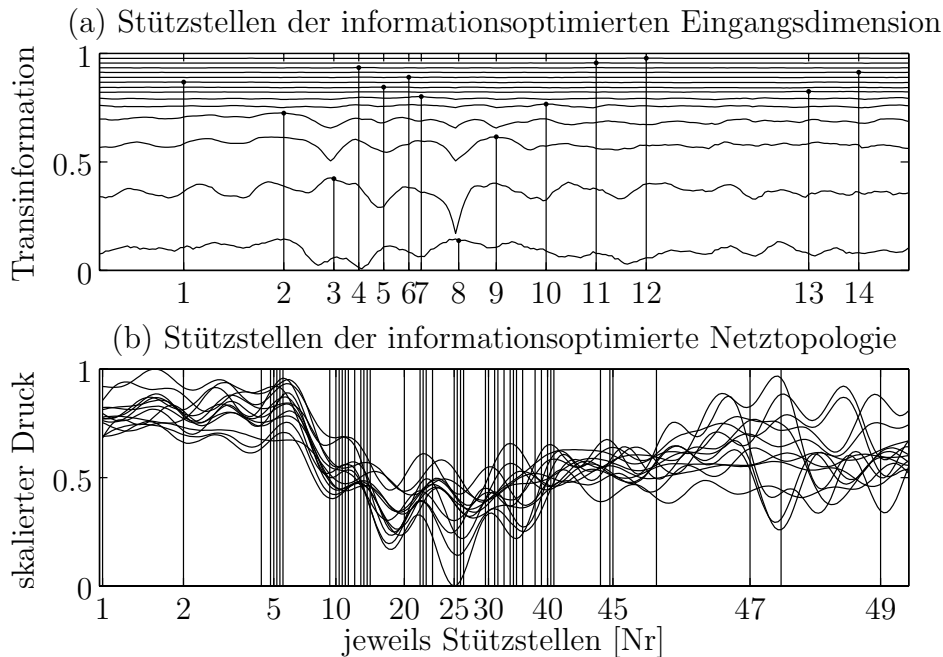


Abbildung 5.19: Methoden zur Reduzierung der Netztopologie.

Vergleich der ausgewählten Stützstellenanzahl durch unterschiedliche Methoden:

- (a) Darstellung der Funktionsverläufe und der ausgewählten Maximalwerte zur Reduzierung der Eingangsdimensionalität.
- (b) Verlauf von Raildruckschwingungen ausgewählter Arbeitspunkte. Von der informationsoptimierten Netztopologie ausgewählte Druckpositionen.

Wie ebenfalls zu erkennen ist, liegen die ausgewählten Stützstellen in den gleichen Bereichen der Druckschwingung, jedoch können beim Aufbau einer informationsoptimierten Netztopologie mehr optimierte Stützstellen gefunden werden. Dies ist gegenüber der reinen Reduzierung der Eingangsdimensionalität mit Hilfe der mehrdimensionalen Transinformation von Vorteil, da dem Verfahren, wie schon erwähnt, statistische Grenzen gesetzt sind.

Durch die Optimierung der gesamten Netztopologie ist nicht jedes verdeckte Neuron mit allen Eingängen verbunden, wodurch eine viel geringere Gesamtanzahl an Gewichten erzielt werden kann. Abbildung 5.20 veranschaulicht das durch den Algorithmus 4.4.1 aus Kapitel 4 Abschnitt 4.4 vollständig automatisch generierte Netzwerk. Die Ein-

gangsmatrix $X(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}$ mit $t = \{1, 2, \dots, 260\}$ des Algorithmus besteht dabei aus vollständigen Railldrucksignalen $x_n(t)$ aller N Trainingsdaten. Auch in diesem Fall besteht die Ausgangsmatrix nur aus dem Vektor $Y = \{y_1, y_2, \dots, y_N\}$ der zugehörigen, gemessenen Einspritzmengen.

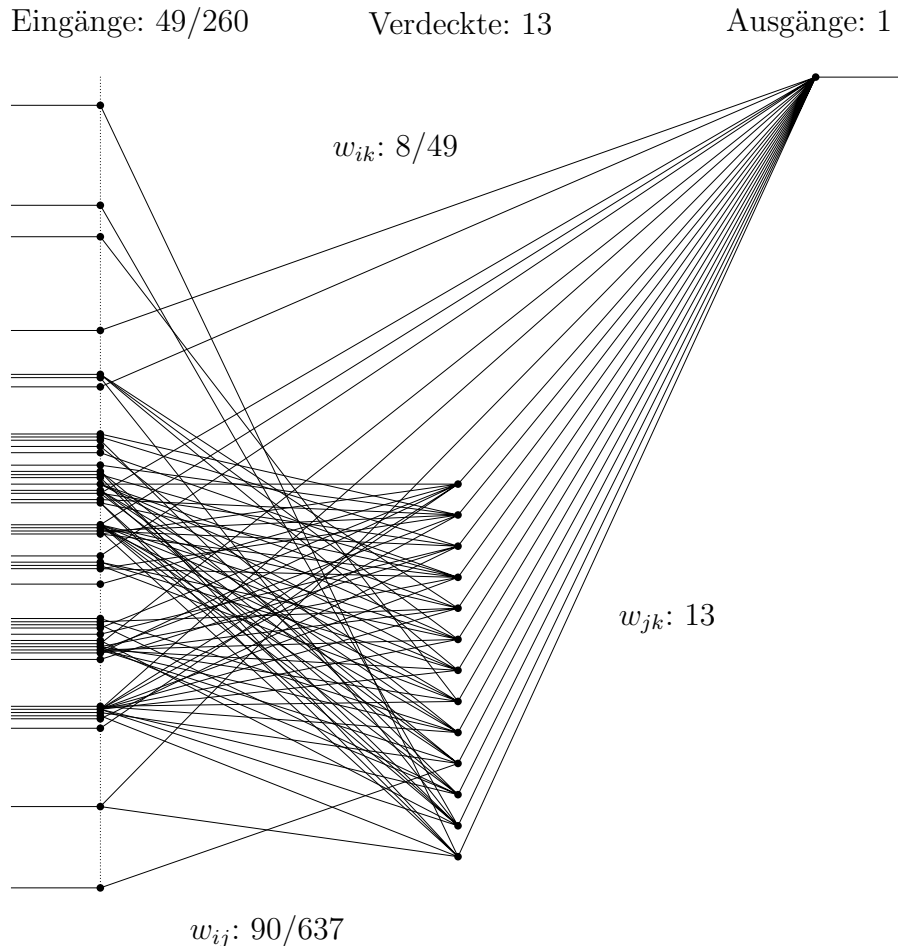


Abbildung 5.20: Informationsoptimierte Netztopologie.

Bei der Verwendung der verallgemeinerten Transinformation zur Optimierung der gesamten Netztopologie wird nicht nur die Eingangsdimensionalität bestimmt, sondern auch die Anzahl der verdeckten Neuronen und aller Verbindungen.

Der Abbildung 5.20 können alle Verbindungen und somit die Anzahl aller Gewichte entnommen werden. Von den 49 ausgewählten Druckpositionen des Eingangs führen nur 90 von 637 möglichen Verbindungen zu verdeckten Neuronen. Das Ausgangsneuron ist nur mit 8 von 49 möglichen Eingängen direkt verbunden. Zu der Gesamtanzahl von 111 Gewichten kommen noch die 13 Verbindungen der verdeckten Neuronen mit dem Neuron des Ausgangs.

Bewertung der informationsoptimierten Netztopologie:

Zur Bewertung dieser neu entwickelten Methode, werden die Restfehler der Ausgabe der informationsoptimierten Netztopologie mit den Ausgabefeldern der von den bekannten Verfahren *Cascade Correlation* und *Netzwerk-Pruning* erzeugten Netzwerken verglichen.

Argument	Parameter
X-Translation	Einspritzgetriggerte Fensterung
Y-Translation	Mittelwertebereinigung
Tiefpaßfilter	0 – 1000 Hz
Simulator	NEURONATOR ¹³
Netztopologie	Informationsoptimierte Netzwerktopologie
Optimierungsparameter	$n = 100$, $proz = 0.02$, $grad = 0.9$
Transferfunktion	'logsig' (logarithmisch sigmoid)
Initialisierung	'initnw' (Nguyen-Widrow)
Lernregel	'trainrp' (<i>Resilient Propagation</i>)
Trainingsparameter	$\Delta_{inc} = 1.01$, $\Delta_{dec} = 0.25$, $\Delta_0 = 0.07$, $\Delta_{max} = 50$
Abbruchbedingung	ansteigender Validierungsfehler

Tabelle 5.7: Trainingsparameter zum Training mit informationsoptimierter Topologie.

Tabelle 5.7 stellt die Parameter des Trainings zur informationsoptimierten Netzwerkstruktur zusammen. Sowohl die Vorverarbeitung, als auch die Trainingsparameter sind identisch mit dem Netztraining aus Abschnitt 5.3.4, was die Ergebnisse vergleichbarer macht. Dabei ist jedoch zu bemerken, daß für das informationsoptimierende Training der gesamte Vektor des Raildrucksignals zur Verfügung gestellt wird. Die Auswahl relevanter Eingangsneuronen erfolgt, ebenso wie die Generierung erforderlicher verdeckter Neuronen, während des Trainings. Zur Steuerung der informationstheoretischen Optimierung sind einige zusätzliche Parameter erforderlich:

- Die Anzahl n an Trainingsepochen, bis zum Einfügen eines neuen Neurons.
- Die Belegungsdichte $proz \in [0, 1]$, als indirektes Maß der Suchboxgröße.
- Die Abbruchbedingung $grad \in [0, 1]$, der zu erreichenden Gesamtinformation.

Mit der in Tabelle 5.7 eingestellten Parameter konnte ein sehr kleines und dennoch leistungsstarkes künstliches, neuronales Netz voll automatisch generiert werden. Damit ist die Methode ebenso bedienerfreundlich, wie die verglichen Verfahren *Cascade Correlation* und *Pruning*.

¹³NEURONATOR ist eine auf MATLAB basierende Programmbibliothek, die im Rahmen dieser Arbeit zum informationstheoretisch gestützten Netzwerktraining entwickelt wurde.

Die Trainingsergebnisse sind zum Vergleich in Abbildung 5.21 durch Balkendiagramme zusammengefaßt. Dabei entsprechen die einzelnen Balken den Ausgabebefehlern der Netzstrukturen mit folgenden Unterschieden:

- A $14 \times 50 \times 1$ -Topologie mit 750 Gewichten: Stützstellenauswahl unter Verwendung der mehrdimensionalen Transinformation (feste Topologie).
- B $14 \times 23 \times 1$ -Topologie mit 262 Gewichten: Stützstellenauswahl durch Transinformation und Topologieaufbau mit *Cascade Correlation*.
- C $49 \times 13 \times 1$ -Topologie mit 111 Gewichten: Automatische Generierung einer informationsoptimierten Netztopologie.
- D $14 \times 34 \times 1$ -Topologie mit 353 Gewichten: Stützstellenauswahl durch Transinformation und Topologieoptimierung unter Verwendung von *Pruning*-Algorithmen.

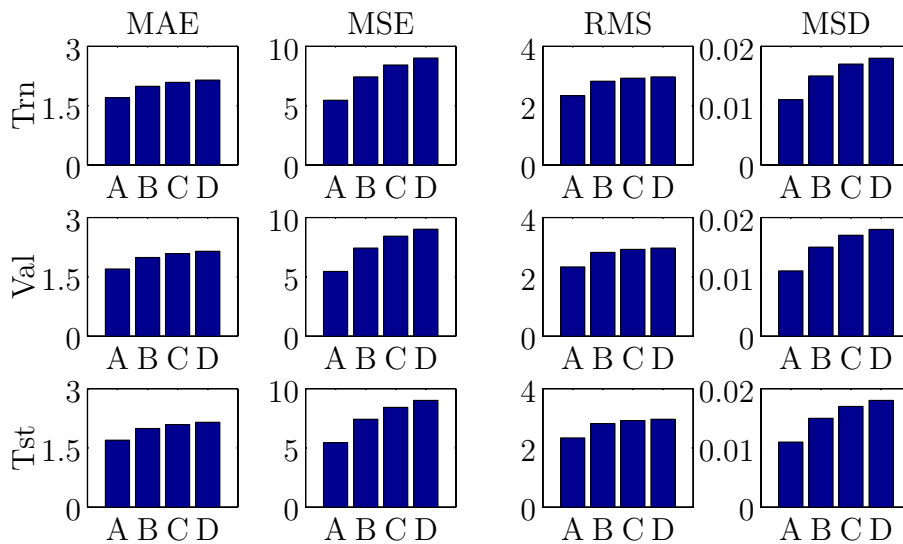


Abbildung 5.21: Auswertung unterschiedlicher Stützstellen.

Gegenüberstellung verschiedener Datensätze mit diversen Fehlermaßen für Trainings-, Validierungs- und Testdaten. Dabei unterscheidet sich die Auswahl der Stützstellen durch:

- A: opt. Eingangsdim. (750 Gewichte) B: *Cascade Correlation* (262 Gewichte)
- C: opt. Netztopologie (111 Gewichte) D: *Pruning*-Verfahren (353 Gewichte)

Wie gut zu erkennen ist, weichen die erzielten Ergebnisse nicht wesentlich von den bisher erzielten Trainings-, Validierungs- und Testfehlern ab. Jedoch konnte die Anzahl an notwendigen Verbindungen stark reduziert werden. Da der Berechnungsaufwand in der Anwendungsphase im Wesentlichen proportional zur Anzahl der Gewichte ist (vgl.

[Fro96]), erfüllt die kleinste Netzstruktur die Anforderungen einer Echtzeitanwendung am Besten. Somit ist die informationsoptimierte Netztopologie trotz des leicht angestiegenen Ausgabefehlers mit einem MAE von ca. 2 mm^2 am Besten für die Bestimmung der eingespritzten Kraftstoffmenge im Fahrzeug und in Echtzeit geeignet.

Ein großer Nachteil der neuen Methode zum Aufbau informationsoptimierter Netztopologien ist der hohe Berechnungsaufwand. Zusätzlich zum Aufwand für einen gewöhnlichen Trainingsalgorithmus wie z.B. *Backpropagation* kommt dabei, mit jedem eingefügten Neuron, der Berechnungsaufwand zur Schätzung der verallgemeinerten Transinformation hinzu. Dieser Aufwand ist um ein Vielfaches höher, als bei *Cascade Correlation* oder den verglichenen *Pruning*-Algorithmen. Allerdings fallen diese Berechnungen nur in der Trainingsphase an und haben keinen Einfluß auf die Berechnungsgeschwindigkeit in der Anwendungsphase.

5.4 Zusammenfassung der Ergebnisse

Dieses Kapitel konnte darstellen wie sich die im Kapitel 4 vorgestellten, neuen Methoden zur Optimierung von Netzstrukturen für den Echtzeiteinsatz im Motormanagement einsetzen lassen. Nach einer kurzen Einführung der Grundlagen des *Common-Rail*-Systems sind folgende Anwendungen beschrieben und analysiert worden:

Bestimmung des 50%-Umsatzpunktes:

Die Bestimmung des 50%-Umsatzpunktes erfolgt unter Verwendung von künstlichen neuronalen Netzen und anhand von Brennraumdruckverläufen. Durch den Einsatz des Algorithmus 4.3.1 zur Reduzierung der Eingangsdimensionalität ist es gelungen, mit nur 5 ausgewählten Druckpositionen ein neuronales Netz zur Bestimmung des 50%-Umsatzpunktes zu trainieren.

Datenanalyse von Raildrucksignalen:

Unterliegen Meßdaten, wie bei der Raildruckanalyse, starken Fremdeinflüssen, ist es schwer die Information der Nutzsignale von diesen Störungen zu trennen. Mit Hilfe des neu entwickelten Algorithmus 4.2.2 zur informationsoptimierten Frequenzanalyse können nicht nur Frequenzanteile, sondern auch der, auf das Nutzsignal bezogene, Informationsgehalt dargestellt werden. Dies erleichtert den Entwurf geeigneter Filter.

Bestimmung der eingespritzten Kraftstoffmenge:

Durch jede Einspritzung werden Druckschwingungen im *Common-Rail*-System ausgelöst. Diese Schwingungen sind in jedem Betriebspunkt des Motors unterschiedlich und können zur Bestimmung der eingespritzten Kraftstoffmenge genutzt werden. Da es schwer ist einen funktionalen Zusammenhang zu formulieren, sind künstliche, neuronale Netze zum Einsatz gekommen. Zur Modellierung sind, aufgrund der Komplexität der Approximationsaufgabe, zunächst große Netzstrukturen notwendig, die sich nicht für einen Echtzeiteinsatz im Fahrzeug eignen. Zur Reduzierung der Netzgröße konnten folgende neue Algorithmen erfolgreich eingesetzt werden:

- Algorithmus 4.3.1 zur Reduzierung der Eingangsdimensionalität und
- Algorithmus 4.4.1 für informationsoptimierte Netztopologien.

Mit Hilfe dieser Algorithmen ist es gelungen, kleine und robuste, künstliche, neuronale Netze zu trainieren, mit denen die eingespritzte Kraftstoffmenge im Fahrzeug und in Echtzeit bestimmt werden kann.

Zu jeder Teilanwendung wurde zunächst eine Beschreibung der Aufgabenstellung und Problematik gegeben, worauf dann ein Lösungsansatz mit Bewertung erfolgte. Im Folgenden sind die Ergebnisse der Anwendungen nochmals zusammengefaßt, wobei der Schwerpunkt bei den Ergebnissen der Hauptanwendung zur Bestimmung der eingespritzten Kraftstoffmenge liegt.

5.4.1 Bewertung der Ergebnisse zum 50%-Umsatzpunkt

Die Reduzierung der Eingangsdimensionalität am Beispiel der Bestimmung des 50%-Umsatzpunktes konnte eine erhebliche Verringerung der notwendigen Neuronen in der Eingangsschicht erzielen.

Durch die informationsoptimierte Auswahl von 5 Druckpositionen ist es gelungen, die Topologie von ursprünglich $33 \times 50 \times 1$ (äquidistanten Abtastung) auf $5 \times 50 \times 1$ (informationsoptimierte Abtastung) zu beschränken, ohne Einbußen in der Qualität bei der Bestimmung des 50% Umsatzpunktes. Damit konnte die Gesamtanzahl an Verbindungen von 1700 auf 300 reduziert werden. Dadurch ist eine Verbesserung des Berechnungsaufwands in der Anwendungsphase um den Faktor 5.6 gelungen.

Eine detaillierte Beschreibung der Ergebnisse und der Umsetzung des Regelalgorithmus im Fahrzeug kann dem Technischen Bericht [DMF01] zum BMBF-Projekt „Technische Anwendungen der Regelung chaotischer Systeme und der Zeitreihenanalyse“ entnommen werden.

5.4.2 Bewertung der Ergebnisse zur Datenanalyse

Neben den herkömmlichen Methoden zur Eingrenzung notwendiger Frequenzen hat die informationsoptimierte Frequenzanalyse einen wichtigen Beitrag zum Verständnis der Signalzusammenhänge geliefert. Durch die so gewonnenen Erkenntnisse konnten digitale Filter entworfen werden, die nicht relevante Frequenzanteile gut aus den Meßdaten beseitigen.

Künstliche, neuronale Netze, die mit entsprechend gefilterten Daten trainiert wurden, konnten die eingespritzte Kraftstoffmenge mit einer mittleren Abweichung von lediglich 1.6 mm^3 für die Testdatensätze ermitteln. Auf herkömmliches Systemverständnis aufbauende Filterentwürfe haben Abweichungen von bis zu 4.5 mm^3 gezeigt. Die Durchschnittliche Abweichung aller Filter lag mit 2.4 mm^3 weit über dem Ergebnis, der mit Hilfe der informationsoptimierte Frequenzanalyse entworfenen Filter.

Damit hat sich die neu entwickelte Methode als nützliches Werkzeug für das Verständnis der Systemzusammenhänge des *Common-Rail*-Systems und der darin auftretenden Frequenzen erwiesen.

5.4.3 Bewertung der Ergebnisse zur Raildruckanalyse

Für die Bestimmung der eingespritzten Kraftstoffmenge am *Common-Rail*-System konnten durch den Einsatz der verallgemeinerten Transinformation unterschiedliche Verbesserungen erzielt werden. Genau wie bei der Bestimmung des 50%-Umsatzpunktes führte die Reduzierung der Eingangsdimensionalität zu einer Verbesserung des Berechnungsaufwands.

Dabei wurden neuronale Netze mit $65 \times 50 \times 1$ bzw. $32 \times 50 \times 1$ (äquidistante Abtastung), $25 \times 50 \times 1$ (korrelationsbezogene Abtastung) und $14 \times 50 \times 1$ (informationsoptimierte Abtastung) Topologie verglichen. Dies führte zu einer Reduzierung des Berechnungsaufwands der Anwendungsphase um einen Faktor von bis zu 4.4. Gegenüber der Optimierung durch lineare Korrelation ist immerhin ein Faktor von 1.7 möglich, wobei der Ausgabefehler des informationsoptimierten Netzwerks wesentlich geringer ausfällt.

Durch die Generierung einer informationstheoretisch optimierten Netztopologie konnte der Berechnungsaufwand noch weiter reduziert werden. So ist es gelungen, die vollständig verbundene $14 \times 50 \times 1$ Topologie mit 750 Verbindungen auf eine ausgedünnte $49 \times 13 \times 1$ Topologie mit nur 111 Gewichten zu beschränken, was einem Faktor von 6.8 entspricht. Gegenüber der anfänglichen $65 \times 50 \times 1$ Topologie entspricht das einer Reduzierung um den Faktor 29.7 für den Berechnungsaufwand in der Anwendungsphase.

Was diese extreme Reduzierung der Netzgröße für die Abweichungen der Netzausgaben gegenüber der gemessenen Einspritzmenge bedeutet, soll folgende Betrachtung aufzeigen. Tabelle 5.8 stellt akzeptable Toleranzen der Abweichung unterschiedlicher Bereiche von realen Einspritzmengen zusammen.

Bereich	zulässige Abweichung	
	absolut	prozentual
$> 30 \text{ mm}^3$	$0.9 - 2.4 \text{ mm}^3$	3%
$20 \dots 30 \text{ mm}^3$	$0.8 - 1.2 \text{ mm}^3$	4%
$10 \dots 20 \text{ mm}^3$	$0.5 - 1.0 \text{ mm}^3$	5%
$04 \dots 10 \text{ mm}^3$	$0.3 - 0.6 \text{ mm}^3$	6%
$< 04 \text{ mm}^3$	$0.1 - 0.4 \text{ mm}^3$	7%

Tabelle 5.8: Toleranzen für Abweichungen von realen Einspritzmengen in mm^3 .

Es ist wünschenswert, daß die Netzausgabe für alle Arbeitspunkte des *Common-Rail*-Systems innerhalb dieser Grenzen liegt. Diese Vorgaben können nur erfüllt werden, wenn die Netzausgabe über mehrere Arbeitsspiele gemittelt wird. Dadurch kann die Einspritzmenge nicht für jede Einspritzung bestimmt werden, jedoch liegen die Abweichungen eher in den gewünschten Toleranzgrenzen. Die Mittelung bedeutet für einen Diagnoseeinsatz im Fahrzeug keine Nachteile, allerdings hat sie zur Folge, daß keine arbeitsspielaufgelöste Regelung der Einspritzmenge realisiert werden kann.

Abbildung 5.22 veranschaulicht, für welche Bereiche dies gelungen ist, wenn die Netzausgabe der Testdaten für jeweils 10 Arbeitsspiele gemittelt wird. Die gerasterte Fläche entspricht dabei den zulässigen Abweichungen aus Tabelle 5.8 für jede Position des Datenraums (vgl. Abschnitt 5.1.3). Die überhöhten Bereiche entsprechen den Arbeitspunkten, in denen die Netzausgabe die Vorgaben nicht erfüllen kann.

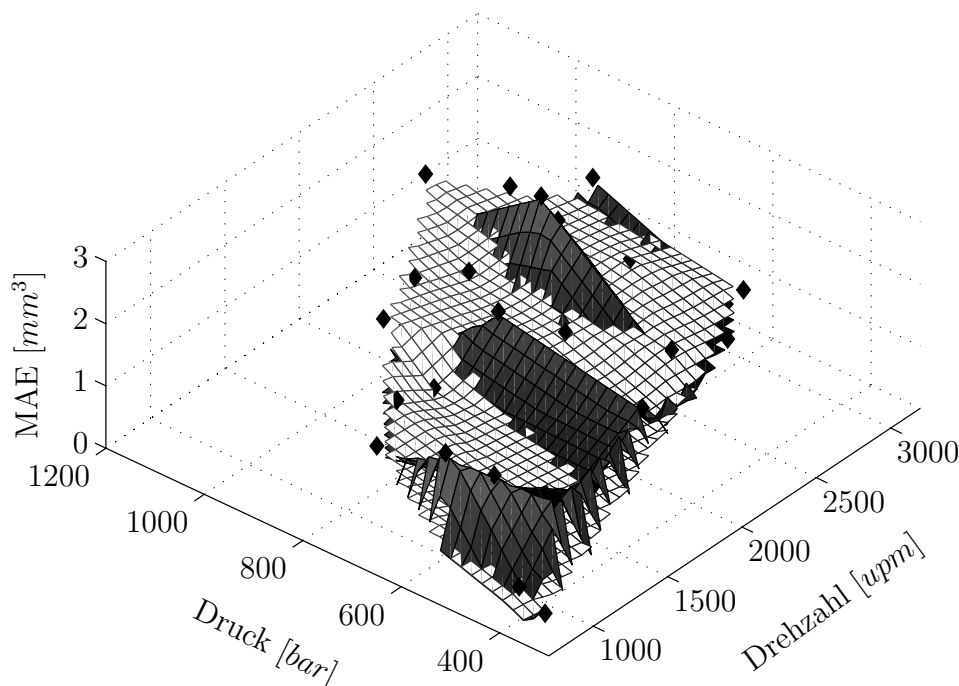


Abbildung 5.22: Darstellung von Arbeitspunkten mit großer Abweichung. Darstellung der mittleren, absoluten Abweichung der Netzausgaben von der tatsächlich eingespritzten Kraftstoffmenge für Raildruckmessungen aller Arbeitspunkte.

Zu erkennen sind Problembereiche einzelner Drehzahlen bzw. einzelner Arbeitspunkte. Um diese Bereiche besser abbilden zu können, kann z.B. eine angepaßte Abdeckung des Datenraums erfolgen, was auch im Ausblick vorgeschlagen wird. Diese Bereiche konnten allgemein als Problemzonen identifiziert werden, da eine Auswertung von Injektor-druckdaten für ähnliche Bereiche ebenfalls überdurchschnittlich schlechte Ergebnisse liefert. Dies läßt darauf schließen, daß die Abbildungsprobleme nicht von der Netz-topologie verursacht werden, sondern auf eine erhöhte Komplexität dieser Bereiche zurückzuführen ist.

5.5 Aufgetretene Probleme

Neben der bereits erwähnten Einschränkung, daß durch die vorgestellte, neuronale Bestimmung der eingespritzten Kraftstoffmenge keine arbeitsspielaufgelöste Regelung der Einspritzung erfolgen kann, bleiben zwei wesentliche Probleme bestehen:

- Die Schätzung der verallgemeinerten Transinformation, die als Basis aller vorgestellten Methoden dient, erfordert einen sehr großen Berechnungsaufwand.
- Zum Training der vorgestellten neuronalen Netze sind Meßdaten von sehr vielen Arbeitspunkten entsprechender Motoren aufzuzeichnen.

Da der Berechnungsaufwand nur vor und während der Trainingsphase der neuronalen Netze anfällt, lassen sich die gewonnenen, kleinen Netzstrukturen leicht im Fahrzeug einsetzen. Jedoch verursacht die Berechnung in der Entwicklungsphase genau wie das Aufzeichnen vieler Daten erhebliche Kosten. Einen Lösungsansatz soll folgender Ausblick liefern.

Kapitel 6

Zusammenfassung und Ausblick

Mit der vorliegenden Arbeit konnten Verfahren vorgestellt werden, die es ermöglichen, Diagnosefunktionen zur Überwachung von schadstoffrelevanten Größen in Fahrzeugsteuergeräten umzusetzen. Eine derartige Umsetzung ist insbesondere zur Erfüllung der gesetzlichen Forderungen notwendig. Sie vergrößert die Wartungsintervalle und kann zudem die Lebensdauer des Motors erhöhen, was neben den verbesserten Abgaswerten auch zu Kosteneinsparungen führt. Für verbrennungsspezifische Motorfunktionen haben sich Algorithmen auf der Basis von künstlichen neuronalen Netzen zur Echtzeitüberwachung bewährt. Für den direkten Einsatz in Motorsteuergeräten ist die Berechnungszeit und somit die Größe der verwendeten Netztopologie eine kritische Kennzahl.

Zur Reduzierung neuronaler Strukturen ist eine Analyse der Informationspfade innerhalb neuronaler Netze von großer Bedeutung. Die schon von SHANNON eingeführte Transinformation in Form eines Informationskanals, erscheint in diesem Zusammenhang als geeignetes Maß derartiger Informationsflüsse. Die Geschwindigkeit heutiger Computer, aber auch ein geschicktes Schätzverfahren zur verallgemeinerten Transinformation, ermöglichen eine effiziente Berechnung von Informationsflüssen in Systemen. Solche Informationsflüsse können dazu genutzt werden, die Relevanz von Netzwerkverbindungen künstlicher, neuronaler Netze zu bestimmen.

Bisherige Methoden zur Beeinflussung der Netztopologie versuchen die Informationsanteile der Netzverbindungen ohne Berechnung von Transinformationen zu schätzen oder verzichten sogar vollständig auf ihre Bestimmung. Durch das Entfernen einzelner Verbindungen oder Neuronen kann zwar die Netzgröße reduziert werden, jedoch besteht die Gefahr, daß mit der entfernten Einheit auch ein Informationspfad gestört wird. Der negative Einfluß kleiner Veränderungen kann erst nach mehreren Iterationen bemerkt werden.

Auch gezieltes Hinzufügen von Neuronen erfordert Kenntnis über die Relevanz der Netzverbindungen für das neue Neuron. Bisherige Verfahren verbinden neue Neuronen

eher willkürlich und erhöhen somit die Anzahl der Verbindungen unnötig. Auch wenn solche Verbindungen nachträglich entfernt werden können, wäre für gezieltes Löschen ein informationstheoretischer Ansatz erforderlich.

Auf Basis der verallgemeinerten Transinformation sind im Rahmen dieser Arbeit Algorithmen entwickelt worden, die eine Analyse des Informationsgehalts von Rohsignalen, sowie Signalübertragungen innerhalb neuronaler Netze ermöglichen. Der in diesem Zusammenhang eingeführte Begriff *informationsoptimiert*, bezieht sich auf eine informationstheoretische Optimierung bestehender Methoden zur Frequenzanalyse, zur Reduzierung der Eingangsdimension, sowie der gesamten Netztopologie. Dabei erfolgt die aufwendige Berechnung zur Schätzung der verallgemeinerten Transinformation vor bzw. während des Trainings der Netzstrukturen, jedoch nicht in der Anwendungsphase des neuronalen Netzes.

Mit Hilfe der neuen Methoden zur informationstheoretischen Optimierung ist es gelungen, Netzstrukturen neuronal basierter Ansätze derart zu reduzieren, daß eine Umsetzung in Fahrzeugsteuergeräten möglich wird. Durch eine neuronale Brennraumdruckauswertung ist es möglich den 50%-Umsatzpunkt durch Verschieben des Verbrennungszeitpunktes zu optimieren. Diese Regelung kann in Echtzeit am laufenden Motor erfolgen und verbessert den Wirkungsgrad der Verbrennung.

Durch eine neuronale Analyse von Druckschwingungen am *Common-Rail*-System ist eine Diagnose der tatsächlich eingespritzten Kraftstoffmenge möglich. Die Überwachung dieser Größe ist für eine saubere Verbrennung von größter Wichtigkeit. Sie dient der Kontrolle der Schadstoffe und ist zur Erfüllung gesetzlicher Vorschriften erforderlich.

Als Fazit ist die *informationsoptimierte* Analyse von Verbindungen künstlicher, neuronaler Netze als geeignetes Werkzeug zur Reduzierung der Netzgröße anzusehen. Der Vorteil gegenüber bestehenden Methoden ist darin zu sehen, daß nicht nur einzelne Verbindungen betrachtet werden, sondern mit Hilfe der Transinformation die Gesamtheit des Informationsflusses zur Relevanzbetrachtung herangezogen wird. So entstehen ganzheitlich optimierte, künstliche, neuronale Netze.

Ausblick

Neben den Geschwindigkeitsvorteilen in der Anwendungsphase *informationsoptimierter* Topologien künstlicher, neuronaler Netze bestehen bei ihrer Entwicklung zwei bereits genannte Probleme:

- Die Schätzung der verallgemeinerten Transinformation, die als Basis aller vorgestellten Methoden dient, erfordert einen sehr großen Berechnungsaufwand.
- Zum Training der vorgestellten neuronalen Netze sind Meßdaten von sehr vielen Arbeitspunkten entsprechender Motoren aufzuzeichnen.

Zwar bestehen diese Probleme nicht in der Anwendungsphase, jedoch verursachen lange Berechnungen in der Entwicklungsphase genau wie das Aufzeichnen vieler Daten erhebliche Kosten für das entwickelnde Unternehmen. Eine Reduzierung der zur Approximation notwendigen Anzahl an Datensätzen führt nicht nur zu kürzeren Entwicklungszeiten während ihrer Aufzeichnung, sondern auch bei allen weiteren Berechnungen. Ähnliche Probleme sind bereits für vielseitige Anwendungen aufgetreten und haben die Entwicklung diverser Lösungsansätze initiiert.

So konnten bereits generische Algorithmen der Evolutionsstrategie erfolgreich für diverse kombinatorische Optimierungsaufgaben zur Datenerfassung für Motorkennfelder eingesetzt werden. Durch eine Optimierung von Prüfablaufplänen [KPMZ01] können deutlich kürzere Meßzeiten und Robustheitsgrade am Prüfstand erreicht werden. Werden aus mehreren potentiellen Kennfeldkandidaten optimale Kandidaten ausgewählt, läßt sich eine Kennfeldglättung [PKM⁺01] erzielen, was eine Approximation vereinfacht. Durch eine optimale Auswahl des Kennfeldrasters [PKZ01] bzw. eine Komprimierung von Kennfeldern [KPMZ02] kann die Anzahl an Stützstellen und somit an notwendigen Meßpunkten reduziert werden.

Eine informationstheoretisch gestützte Optimierung von Motorkennfeldern und somit der Anzahl an erforderlichen Meßdaten ist nach Vorbild der generischen Ansätze denkbar. In jedem Fall läßt sich der absolute Berechnungsaufwand durch eine geschickte Kombination aus optimierter Aufzeichnung vom Motorkenngrößen und *informationsoptimierten* Netztopologien reduzieren.

Anhang A

Nomenklatur

Tabelle A.0.1 (Elemente, Mengen, Zeitreihen, sowie Formelzeichen)

Formelzeichen	Beschreibung
Kapitel 2.1 Künstliche neuronale Netze:	
η, η^+, η^-	Lernraten der Lernregeln neuronaler Netze
δ_{pk}	Restfehler für Trainingsmuster p der Zelle k
Δ_{ij}	Schrittweite für <i>Resilient Propagation</i>
σ_k	Vorzeichen der Korrelation zwischen o_j und Ausgang k
a_j	Aktivierung des j 'ten Neurons zum Zeitschritt t
b_j	Gewicht des Bias-Eingangs für Neuron j
D	Diagonalmatrix aus Elementen der $J^T W J$ -Matrix
E	Fehlerfunktion
i, j, k	Indizes Eingangs-, verdeckter und Ausgangsneuronen
H	Hesse-Matrix (1. Ableitung, hier der Fehlerquadrate)
J	Jakobi-Matrix (2. Ableitung, hier der Fehlerquadrate)
net	Netzeingabe eines Neurons (Propagierungsfunktion)
m, n	Eingangs- bzw. Ausgangsdimension neuronales Netz
o	Ausgabe eines Neurons
p	Abhängige Parameter der Approximationsfunktion
r_{ij}	Relative Abweichung der Approximationsfunktion
R	Matrix der relativen Abweichungen r_{ij}
S	Korrelationssumme
t	Index für Zeit- oder Iterationsschritt
Δw_{ij}	Änderung des Gewichts w_{ji}
w_{ij}	Gewicht der Eingangszelle i zum verdeckten Neuron j
w_{jk}	Gewicht des verdeckten Neurons j zur Ausgangszelle k
W	Gewichtsmatrix eines künstlichen neuronalen Netzes
x	Eingabewerte des neuronalen Netzes
y	Zielwerte oder Trainingsvektor des neuronalen Netzes
\tilde{y}	Ausgabe des neuronalen Netzes (Approximation)

Formelzeichen	Beschreibung
Kapitel 2.2 Statistische Maße und Wahrscheinlichkeiten:	
μ	Erwartungswert der Elemente x_i einer Stichprobe
σ_x, σ_x^2	Standardabweichung und Varianz der Elemente x_i
σ_{xy}	Kovarianz der Elemente x_i und y_i
Σ	Kovarianzmatrix eines Zufallsvektors
ξ, η	Zufallsvariablen der Mengen $\{x_i\}_{i=1}^n$ und $\{y_k\}_{k=1}^m$
$\tilde{F}_{x^*}(\tilde{x})$	empirische Verteilungsfunktion von Merkmalswerten x^*
$F_\xi(\tilde{x})$	Verteilungsfunktion der Zufallsvariablen ξ
h_k, r_k	absolute bzw. relative Häufigkeit von x_k^*
H_k, R_k	absolute bzw. relative Summenhäufigkeit von x_k^*
$P(A)$	Wahrscheinlichkeit für das Auftreten des Ereignisses A
$\{p_i\}, \{p_{ik}\}$	Wahrscheinlichkeit und Verbundwahrscheinlichkeit
s_x, s_x^2, s_{xy}	empirische Standardabweichung, Varianz und Kovarianz
S	empirische Kovarianzmatrix einer Beobachtungsmatrix
$x = \{x_i\}_{i=1}^n$	Beobachtungsmenge oder Eingangswerte eines Systems
$x^* = \{x_k^*\}_{k=1}^N$	Merkmalswerte einer Beobachtungsmenge x
$\bar{x}, \hat{x}, d\bar{x}, d\hat{x}$	Mittelwert, Median und mittlere Abweichungen
x	Zeit- oder Meßreihe
$X = \{x_i\}_{i=0}^n$	Menge von n Zeitreihen (meist Eingangssignale)
$X_{Trn} = \{x_{i,Trn}\}_{i=0}^n$	Teilmenge der Zeitreihen X als Trainingsmenge
$X_{Tst} = \{x_{i,Tst}\}_{i=0}^n$	Teilmenge der Zeitreihen X als Testmenge
$X_{Val} = \{x_{i,Val}\}_{i=0}^n$	Teilmenge der Zeitreihen X als Validierungsmenge
$y = \{y_k\}_{k=1}^m$	Sollwerte (Trainingsmenge neuronaler Netzen)
$\tilde{y} = \{\tilde{y}_k\}_{k=1}^m$	Istwerte (Ausgabewerte neuronaler Netzen)
Kapitel 2.3 Grundlagen der Signalanalyse:	
$\delta(t)$	Dirac-Stoß zum Zeitpunkt t
$\theta = t - t_0$	Zeitverschiebung um t_0
$\varphi = f - f_0$	Frequenzverschiebung um f_0
a, b	Amplituden- und Zeitfaktor
f, f_0, F	Frequenzbereich, Frequenz und Frequenzbandgrenze
fs	Abtastfrequenz (das Abtasttheorem fordert $fs = 2 \cdot F$)
$g(t), G(f)$	Ausgangssignal eines Systems im Zeit-, Frequenzbereich
$g_a(t), s_a(t)$	Approximation der Signale $g(t)$ und $s(t)$
$H_0(f) = \text{hann}(\frac{f-f_0}{k})$	Hanning-Fenster über Frequenz f_0 mit Breite k
$h(t), H(f)$	Stoßantwort eines Systems auf Dirac-Stoß
$k = \frac{2}{\pi} \cdot \frac{F}{u}$	Fensterbreite in Abhängigkeit der Unterteilungen u
$s(t), S(f)$	Eingangssignal eines Systems im Zeit-, Frequenzbereich
$S_E(t) = e^{j2\pi ft}$	Eigen- Exponentialfunktion der Fourier-Transformation
t, t_0, T	(diskreter) Zeitvektor, Zeitpunkt und Zeitreihenlänge

Formelzeichen	Beschreibung
Kapitel 2.4 Verallgemeinerte Transinformation:	
$\alpha \in \mathbb{R}$	Ordnung der Rényi-Entropie
$\beta_{d,TT}, \mathcal{B}_{\xi_T}$	Binärelement und -matrix der Zufallsgröße ξ für Dimension d
$\delta_{d,TT}, \Delta_{\xi_d}$	Rang-Abstände und Matrix der Zufallsgröße ξ für Dimension d
ϵ, ϵ_d und \mathcal{E}_d	Epsilonumgebungen für Suchboxen der Dimension d
ξ, η	Zufallsgrößen
$\Xi \equiv \{\xi_1 \dots \xi_d\}$	Matrix mehrdimensionaler Zufallsvektoren
C, C^2	Korrelationsintegral und quadrierter Korrelationskoeffizient
$d = \{1 \dots D\}$	Dimensionszähler
$H_\alpha(\xi)$	Rényi-Entropie einer Zufallsgröße ξ
$H_\alpha(\xi, \eta)$	Gemeinsame Rényi-Entropie der Zufallsgrößen ξ und η
$I_\alpha(\xi, \eta)$	Maß gemeinsamer Information von ξ und η (Transinformation)
n, m, N, M	Indizes und Größe von Wahrscheinlichkeitsverteilungen
$\{p_m\}_{m=1}^M, \{q_n\}_{n=1}^N$	Wahrscheinlichkeitsverteilungen
$P_m \equiv \{p_{m_1} \dots p_{m_d}\}$	Matrix von Wahrscheinlichkeitsverteilungen
$\{s_{mn}\}_{m,n=1}^{M,N}$	Verbundwahrscheinlichkeitsverteilung
S_{mn}	Matrix von Verbundwahrscheinlichkeitsverteilungen
$r(t) = \frac{R(t)}{T}$	Relative Rangfolge einer Zeitreihe der Länge T
t, T	Zeitvektor und Dauer oder Länge der Zeitreihe
$x = \{x_i\}_{i=0}^n$	Menge x von n Elementen
$x(t) = \{x(t)\}_{t=0}^T$	Zeit- oder Meßreihe der Dauer T
$X(t) = \{x_i(t)\}_{i=0}^n$	Menge X von n Zeitreihen (meist Eingangssignale)
$Y(t) = \{y_j(t)\}_{j=0}^m$	Menge Y von m Zeitreihen (meist Ausgangssignale)
$\tilde{Y}(t) = \{\tilde{y}_j(t)\}_{j=0}^m$	Approximation der Menge y (meist Netzantwort)

Tabelle A.0.2 (Verwendete Operatoren, Funktionen und Signale)

Operator	Beschreibung
\mathbb{N}	Menge der natürlichen Zahlen
\mathbb{R}	Menge der realen Zahlen
$j = \sqrt{-1}$	Komplexes Element
$\delta(t)$	Dirac-Stoß
$\text{chirp}(t) = \cos(\frac{f_s}{2T} \cdot \pi \cdot t^2)$	Chirp-Signal mit Frequenzen $]0 \dots \frac{f_s}{2}]$ für $t = \{0 \dots T\}$
$\text{hann}(\frac{f-f_0}{k})$	Hanning-Fenster über Frequenz f_0 mit Breite k
$\text{rect}(\frac{t-t_0}{T})$	Rechteck-Impuls zum Zeitpunkt t_0 mit Breite T
$f_{\text{ein}}(o_i(t), w_{ij})$	Eingangsfunktion eines Neurons
$f_{\text{akt}}(a_j(t-1), \text{net}_j(t), b_j)$	Aktivierungsfunktion eines Neurons
$f'_{\text{akt}}(a_j(t-1), \text{net}_j(t), b_j)$	Ableitung der Aktivierungsfunktion eines Neurons
$f_{\text{aus}}(a_j(t))$	Ausgangsfunktion eines Neurons
$f_{x,p}$	Approximationsfunktion für x mit Parameter p
$E(\delta_{pk})$	Fehlerfunktion als Summe über p und k
$E_{MAE}(\delta_{pk})$	Mittlerer absoluter Fehler
$E_{MSE}(\delta_{pk})$	Mittlerer quadratischer Fehler
$E_{MSD}(\delta_{pk})$	Bereinigter quadratischer Fehler
$E_{RMS}(\delta_{pk})$	Wurzel des quadratischen Fehlers
$F(p)$	Gewichtete Summe der Fehlerquadrate
$S_j(o_{pj}, \delta_{pk})$	Korrelationssumme der Kandidatenzelle j
$\ \cdot\ $	2-Norm oder Betragswert
$\langle x/y \rangle$	Integer-Anteil des Quotienten x/y
$\text{conj}(\cdot)$	Konjunktion einer komplexen Zahl
$\text{exp}(\cdot)$	Exponentialfunktion
$\text{imag}(\cdot)$	Imaginäranteil einer komplexen Zahl
$\log_n(\cdot)$	Logarithmus zur Basis n
$\max\{X\}$	Maximalelement einer Menge X
$\min\{Y\}$	Minimalelement einer Menge Y
$\text{mod}(x, y)$	x Modulo y
$\text{real}(\cdot)$	Realanteil einer komplexen Zahl
$\text{sign}(x)$	Vorzeichen des Elements x
$\#\{n: \text{mit } n < 5; \text{ für } n \in \mathbb{N}\}$	Anzahl an Elementen n mit $n < 5$ für $n \in \mathbb{N}$
$G(f) \bullet \circ g(f)$	Übergang vom Frequenz- in den Zeitbereich
$S_E(t) = e^{j2\pi f t}$	Exponentialfunktion der Fourier-Transformation
$s(t) \circ \bullet S(f)$	Übergang vom Zeit- in den Frequenzbereich
$s(t) * g(t)$	Faltungsprodukt zwischen $s(t)$ und $g(t)$
$H(\xi \eta)$	Entropie der Zufallsgröße ξ , ohne η
$\beta_1 \wedge \beta_2$	UND-Verknüpfung zwischen Binärelementen
$\bigwedge_{d=1}^D \mathcal{B}_d$	UND-Verknüpfung von Binärmatrizen

Anhang B

Algorithmen

Algorithmus B.0.1 (Ranking)

Die Transformation einer beliebig verteilten Zufallsfolge in eine gleichverteilte Folge kann als Ranking bezeichnet werden. Im wesentlichen besteht der hier beschriebene Ranking-Algorithmus aus einem beliebigen Sortieralgorithmus, von welchem lediglich die Indizes der Umverteilung verwertet werden. Bei dem dargestellten Algorithmus kommt eine Variation von Quicksort [PTVF97] zur Anwendung. Die hier dargestellte Version kann von MATLAB compiliert und ausgeführt werden:

```
/*=====
 * ranking.c
 * C-code of Matlab function: [rank] = ranking(series)
 *=====*/

/* Include matlab library. */
#include "mex.h" #include <math.h>

#define SWAP(a,b) swap=(a); (a)=(b); (b)=swap;

void indexsort(double *series, int *index, int left, int right) {
    int i,j,swap;
    double value;

    if(right > left)
    {
        value = series[index[(left+right) >> 1]]; i = left; j = right;
        do
        {
            while((series[index[i]] < value)) i++;
            while((series[index[j]] > value)) j--;
            if(i <= j)
            {
                SWAP(index[i], index[j]);
                i++; j--;
            }
        }
        while(i <= j);
        indexsort(series, index, left, j);
        indexsort(series, index, i, right);
    }
}
```

```

void ranking(double *rank, double *series, int length) {
    int *index;
    int t;

    /* Allocated memory for index list*/
    index = (int*) malloc(length*sizeof(int));
    if (index == NULL)
    {
        mexErrMsgTxt("Couldn't allocate memory!\n");
    }

    /*Use index list to determin rank*/
    for (t=0; t<length; t++) index[t] = t;
    indexsort(series, index, 0, length-1);
    for (t=0; t<length; t++) rank[index[t]] = t;

    /* free allocated memory */
    free(index);
}

/* Interface for Matlab function: [rank] = ranking(series) */
void mexFunction(int nlhs, mxArray *plhs[], int nrhs, const
mxArray *prhs[]) {
    unsigned int numSeries, lenSeries, length;
    double *series, *rank;

    /* Examine input argument. */
    if (nrhs != 1 || nlhs != 1) mexErrMsgTxt("One input and output argument required!\n");
    numSeries = mxGetN(prhs[0]);
    lenSeries = mxGetM(prhs[0]);
    length = numSeries*lenSeries;
    if (lenSeries != 1 && numSeries !=1) mexErrMsgTxt("Input is not a vector!\n");
    if (length <= 2) mexErrMsgTxt("Input vector too short!\n");

    /* Create pointers for input and memory for output argument. */
    series = mxGetPr(prhs[0]);
    plhs[0] = mxCreateDoubleMatrix(lenSeries, numSeries, mxREAL);
    rank = mxGetPr(plhs[0]);

    /* Call ranking. */
    ranking(rank,series,length);
}

```

Der Algorithmus liefert für Zeitreihen $\{x_t\}_{t=0}^T$ die absolute Rangfolge $\{R_t\}_{t=0}^T \in \{0, 1, \dots, T\}$, was eine Integerrepräsentation erlauben würde, jedoch verlangt MATLAB den Typ 'double' zur Übergabe. Durch einfache Division mit der Signallänge T läßt sich die reelle, relative Rangfolge $\{r_t\} = \{\frac{R_t}{T}\}$ erzeugen, die auf dem Interval $[0, 1]$ gleichverteilt ist.

Algorithmus B.0.2 (Entropie)

Der Algorithmus berechnet zur Begrenzung des Speicheraufwands auf $2 * D * T$, die Korrelationssummen aller Nebendiagonalen der oberen Matrixdreiecke nacheinander. Die Rückgabewerte sind bereits die Entropien zur Berechnung der verallgemeinerten Transinformation.

```

/*=====
 * entropie.c
 * C-Code of Matlab function: [Hx, Hy, Hxy] = entropie(X, Y, epsilon)
 *=====*/
#include "mex.h" #include <math.h>

void entropie(double *Hx, double *Hy, double *Hxy,
              double *xMat, int xNum, int xLen,
              double *yMat, int yNum, int yLen, double eps)
{
    int m, i, n;
    int Cx = 0;
    int Cy = 0;
    int Cxy = 0;
    bool xBin, yBin;
    double **pVecX, **pVecY;

    /* create array of pointer to input X */
    pVecX = (double**) malloc(xNum*sizeof(double*));
    if (pVecX == NULL) mexErrMsgTxt("Couldn't allocate memory!\n");
    for (n=0; n < xNum; n++) pVecX[n] = &(xMat[n*xLen]);

    /* create array of pointer to input Y */
    pVecY = (double**) malloc(yNum*sizeof(double*));
    if (pVecY == NULL) mexErrMsgTxt("Couldn't allocate memory!\n");
    for (n=0; n < yNum; n++) pVecY[n] = &(yMat[n*yLen]);

    /* determine correlation integrals */
    for (m=1; m<xLen; m++)
    {
        for(i=0; i<(xLen-m); i++)
        {
            xBin = (fabs(pVecX[0][i]-pVecX[0][m+i])<eps);
            yBin = (fabs(pVecY[0][i]-pVecY[0][m+i])<eps);

            for(n=1; n<xNum; n++)
                if (!(xBin = (xBin & (fabs(pVecX[n][i]-pVecX[n][m+i])<eps))))
                    n=xNum;
            for(n=1; n<yNum; n++)
                if (!(yBin = (yBin & (fabs(pVecY[n][i]-pVecY[n][m+i])<eps))))
                    n=yNum;

            Cx = Cx + xBin;
            Cy = Cy + yBin;
            Cxy = Cxy + (xBin & yBin);
        }
    }

    /* Determine entropies from correlation integrals */
    Hx[0] = -log((1.0/xLen)*(1.0+((2.0/xLen)*Cx)))/log(2.0);
    Hy[0] = -log((1.0/yLen)*(1.0+((2.0/yLen)*Cy)))/log(2.0);
    Hxy[0] = -log((1.0/xLen)*(1.0+((2.0/xLen)*Cxy)))/log(2.0);

    /* free allocated memory */
    free(pVecX);
    free(pVecY);
}

```

```

/* Interface for Matlab function: [Hx, Hy, Hxy] = entropie(X, Y,epsilon) */
void mexFunction(int nlhs, mxArray *plhs[], int nrhs,
const mxArray *prhs[]) {
    int xNum, xLen, yNum, yLen;
    double eps, *xMat, *yMat, *Hx, *Hy, *Hxy;

    /* Examine input argument. */
    if (nrhs < 2) mexErrMsgTxt("Not enough input arguments specified!\n");
    xNum = mxGetN(prhs[0]);
    xLen = mxGetM(prhs[0]);
    yNum = mxGetN(prhs[1]);
    yLen = mxGetM(prhs[1]);
    if (xLen != yLen) mexErrMsgTxt("Input vectors should have the same length!\n");
    if (xLen <= 2) mexErrMsgTxt("Input vectors too short!\n");
    if (nrhs == 2) eps = sqrt(10.0/xLen);
    else eps = mxGetScalar(prhs[2]);

    /* Create pointers for input and memory for output argument. */
    xMat = mxGetPr(prhs[0]);
    yMat = mxGetPr(prhs[1]);
    plhs[0] = mxCreateDoubleMatrix(1, 1, mxREAL);
    plhs[1] = mxCreateDoubleMatrix(1, 1, mxREAL);
    plhs[2] = mxCreateDoubleMatrix(1, 1, mxREAL);
    Hx = mxGetPr(plhs[0]);
    Hy = mxGetPr(plhs[1]);
    Hxy = mxGetPr(plhs[2]);

    /* Call ranking. */
    entropie(Hx, Hy, Hxy, xMat, xNum, xLen, yMat, yNum, yLen, eps);
}

```

Zur Bestimmung der verallgemeinerten Transinformation $I(X, Y)$ zwischen zwei gleichverteilten, mehrdimensionalen Zeitreihen $X(t)$ und $Y(t)$ erfolgt somit durch die einfache Matlab-Sequenz:

```

[Hx, Hy, Hxy] = entropie(X,Y,epsilon);
Ixy = (Hx+Hy-Hxy)/min(Hx,Hy);

```

Die Wahl des Epsilons sollte so erfolgen, daß $H(X, Y)$, unabhängig von der gewählten Dimensionen für $X(t)$ und $Y(t)$, konstant ist. Dabei ist zu beachten, daß das Korrelationsintegral Cxy einen kleinen Wert ungleich null annimmt.

Algorithmus B.0.3 (Zufallszahlengenerator)

Zur Generierung von Zufallsfolgen steht eine große Anzahl an Algorithmen zur Verfügung. Aufbauend auf dem hier beschriebenen Basis-Algorithmus können sehr leistungsfähige Algorithmen implementiert werden [PTVF97].

```

/*****
 * random.c
 * C-Code of Matlab function: noise = random(seed)
 *****/
#include "mex.h" #include <math.h>

#define A 16807
#define M 2147483647
#define Q 127773
#define R 2836

void random(double *noise, double *seed){
    long k;

    /* create integer value of seed */
    k = (*seed)/Q;

    /* compute seed = modulo(A*seed, M) */
    *seed = A*(*seed - k*Q) - R*k;
    if (*seed < 0) *seed += M;

    /* convert seed to floating result in [0, 1] */
    noise = (*seed)/M;
}

/* Interface for Matlab function: noise = random(seed) */
void mexFunction(int nlhs, mxArray *plhs[], int nrhs,
const mxArray *prhs[]){
    int xNum, xLen, yNum, yLen;
    double eps, *xMat, *yMat, *Hx, *Hy, *Hxy;

    /* Examine input argument. */
    if (nrhs < 1) mexErrMsgTxt("Not enough input arguments specified!\n");

    /* Create pointers for input and memory for output argument. */
    plhs[0] = mxCreateDoubleMatrix(1, 1, mxREAL);
    noise = mxGetPr(plhs[0]);
    seed = mxGetPr(prhs[0]);
    if (*seed == 0) mexErrMsgTxt("Seed should not be equal to zero!\n");

    /* Call random. */
    random(noise, seed);
}

```

Der vorliegende Zufallszahlengenerator erzeugt für $seed > 0$ mit jedem Aufruf einen Wert einer gleichverteilten Zufallsfolge im Intervall $[0, 1]$. Dabei darf $seed$ nur zur Initialisierung verändert werden.

Literaturverzeichnis

- [AARH92] ALI, N. ARKANSU, RICHARD und A. HADDAD: *Multiresolution Signal Decomposition, Transforms, Subbands and Wavelets*. Academic Press, 1992.
- [AD75] ACZÉL, J. und J.Z. DARÓCZY: *On measures of Information and their Characterizations*. Academic Press [u.a.], New York, 1975.
- [Ahl00] AHLIN, K.: *Modeling of pressure waves in the Common Rail Diesel Injection System*. Diplomarbeit, Techniska Högskolan i Linköping, 23. November 2000. Reg Nr.: LiTH-ISY-EX-3547.
- [AMM⁺97] AMARI, S., N. MURATA, K.-R. MÜLLER, M. FINKE und H.H. YANG: *Asymptotic Statistical Theory of Overtraining and Cross Validation*. IEEE Transactions on Neural Networks, 8(5):985–996, September 1997.
- [And97] ANDERS, U.: *Statistische neuronale Netze*. Doktorarbeit, Universität Fidericiana Karlsruhe, 1997.
- [Bar94] BARTLETT, E.: *Dynamic Node Architecture Learning: An Information Theoretic Approach*. Neural Networks, 7(1):129–140, 1994.
- [Bar97] BARTLETT, P.L.: *For valid generalization, the size of the weights is more important than the size of the network*. In: 9 (Herausgeber): *Advances in Neural Information Processing Systems*, Seiten 134–140, Cambridge, MA, 1997. The MIT Press.
- [BD84] BRONSTEIN, I. und D.SEMENDJAJEW: *Taschenbuch der Mathematik*. Verlag Harri Deutsch, Frankfurt am Main, 24. Auflage, 1984. ISBN 3-87144-429-8.
- [BHPT95] BEYER, O., H.HACKEL, V. PIEPER und J. TIEDGE: *Wahrscheinlichkeitsrechnung und mathematische Statistik*. B.G. Teubner Verlagsgesellschaft, Stuttgart, 7. Auflage, 1995. ISBN 3-8154-2075-X.
- [Bis95] BISHOP, C.M.: *Neural Networks for Pattern Recognition*. Clarendon Press, 1995.

- [Boh91] BOHL, W.: *Technische Strömungslehre: Stoffeigenschaften von Flüssigkeiten und Gasen, Hydrostatik, Aerostatik, inkompressible Strömungen, kompressible Strömungen, Strömungsmesstechnik*. Vogel Buchverlag, Würzburg, 1991.
- [Bos84] BOSCH, K.: *Elementare Einführung in die Wahrscheinlichkeitsrechnung*. Vieweg Verlag, Braunschweig, 4. Auflage, 1984. ISBN 3-528-37225-7.
- [Bos94] BOSCH, K.: *Elementare Einführung in die angewandte Statistik*. Vieweg Verlag, Braunschweig, 5. Auflage, 1994. ISBN 3-528-47227-8.
- [BR93] BRAUN, H. und M. RIEDMILLER: *A direct adaptive method for faster Back-propagation learning: The RProp algorithm*. In: *Proceedings of the IEEE International Conference on Neural Networks*, Seiten 586–591, San Francisco, CA, 1993.
- [Bur94] BURGESS, N.: *A Constructive Algorithm that Converges for Real-Valued Input Patterns*. *Neural Information Processing Systems*, 51(1):59–66, März 1994.
- [Cec97] CECHIN, A.L.: *The Extraction of Fuzzy Rules form Neural Networks*. Doktorarbeit, Eberhard-Karls-Universität Tübingen, Auf dem Sand 13, 1997.
- [CF96] CARABIAS, J. L. H. und M. FROELICH: *Hydraulische Vorgänge des Common-Rail-Einspritzsystems*. Technischer Bericht F1M/EA 0000-96, Daimler-Benz AG, Abt. F1M/EA, 70546 Stuttgart, 14. Oktober 1996.
- [DMF01] DRESSLER, U., R. MÜLLER und M. FROELICH: *Analyse von Motorprozeßdaten mit Methoden der Nichtlinearen Dynamik zur Optimierung des Motormanagements*. Technischer Bericht TN-FT2/S-2000.013, Daimler-Chrysler AG, Abt. FT2/S und FT2/E, 70546 Stuttgart, 29. Januar 2001. BMBF-Projekt *Technische Anwendungen der Regelung chaotischer Systeme und der Zeitreihenanalyse* Teilprojekt II *Motormanagement*.
- [Fah88] FAHLMAN, S.E.: *Faster-learning variations on back-propagation: an empirical study*. In: *Proceedings of 1988 connectionist models summer school*, Seiten 38–51, San Mateo, CA, 1988. Morgan Kaufmann.
- [FL90] FAHLMAN, S.E. und CH. LEBIÈRE: *The Cascade-Correlation Learning Architecture*. In: TOURETZKY, D.S. (Herausgeber): *Advances in Neuronal Information Processing Systems*, Band 2, Seiten 524–532. Morgan Kaufmann, 1990.
- [Fli93] FLIEGE, N.: *Multiraten-Signalverarbeitung*. Teuber, 1993.

- [Fro96] FROEHLICH, M.: *Einspritzmengenbestimmung am Common-Rail-Einspritzsystem durch Druckverlaufsanalyse mittels neuronaler Netze*. Diplomarbeit, Fachbereich technische Informatik, Fakultät für Informatik, Eberhard-Karls-Universität Tübingen, Sand 13, 72076 Tübingen, 15. November 1996.
- [Fro99] FROEHLICH, M.: *Extraktion von Fuzzy-Regeln aus trainierten Neuronalen Netzen*. Technischer Bericht FT2/EA 0000-99, DaimlerChrysler AG, Abtl. FT2/EA, 70546 Stuttgart, 1999.
- [Fro01] FROEHLICH, M.: *Einspritzmengenbestimmung am Common-Rail-System durch Druckverlaufsanalyse mittels Neuronaler Netze*. Technischer Bericht FT2/EA 0012-01, DaimlerChrysler AG, Abtl. FT2/EA, 70546 Stuttgart, 2001.
- [Göp97] GÖPPERT, J.: *Die topologisch interpolierende selbstorganisierende Karte in der Funktionsapproximation*. Doktorarbeit, Eberhard-Karls-Universität Tübingen, Auf dem Sand 13, 1997.
- [HD93] HASSIBI, B. und D.G.STORK: *Second Order Derivates for Network Pruning: Optimal Brain Surgeon*. In: HANSON, S.J., J.D. COWAN und C.L. GILES (Herausgeber): *Advances in Neuronal Information Processing Systems*, Band 5, Seiten 164–171. Morgan Kaufmann, 1993.
- [HF99] HEISTER, F. und M. FROEHLICH: *Nonlinear Time Series Analysis of Combustion-Pressure Data for Neural Network Training with the Concept of Mutual Information*. Society of Automotive Engineers, Inc., 1999.
- [Hof95] HOFFMANN, J.: *Matlab und Simulink in Signalverarbeitung und Kommunikationstechnik*. Scientific Computing. Addison-Wesley, München, 1995.
- [Ing88] INGARD, K.U.: *Fundamentals of Waves and Oscillations*. Cambridge University Press, Cambridge, Great Britain, 1988. ISBN: 0-521-32734-2.
- [Jan92] JANG, J.-S.R.: *Self-Learning Fuzzy Controllers Based on Temporal Back Propagation*. IEEE Transactions on Neural Networks, III:714–723, 1992.
- [JW92] JOHNSON, R.A. und D.W. WICHERN: *Applied Multivariate Statistical Analysis*. Prentice-Hall Inc., Third Auflage, 1992.
- [Köh96] KÖHLER, J.: *Bestimmung der eingespritzten Kraftstoffmasse aus dem Brennraumdruckverlauf eines Dieselmotors mit neuronalen Netzen*. Diplomarbeit, Eberhard-Karls-Universität Tübingen, Sand 13, 72076 Tübingen, 1996.

- [Köh97] KÖHLER, J.: *Ermittlung der zylinderindividuell eingespritzten Kraftstoffmenge für die OBD bei Dieselmotoren mit Common-Rail-Einspritzsystem*. Technischer Bericht TB 96-0051, DaimlerChrysler AG, Abtl. F1M/E, 70546 Stuttgart, 1997.
- [Köh98] KÖHLER, J.: *Merkmalsselektion zur Echtzeitdiagnose im Fahrzeug mittels neuronaler Netze*. Doktorarbeit, Eberhard-Karls-Universität Tübingen, Sand 13, 72076 Tübingen, 1998.
- [Koh82] KOHONEN, T.: *Self-organized formation of topologically correct feature maps*. Biological Cybernetics, XMIII:59–69, 1982.
- [KPMZ01] KNÖDLER, K., J. POLAND, A. MITTERER und A. ZELL: *Optimizing Data Measurements at Test Beds using Multi-Step Generic Algorithms*. In: *Advances in Fuzzy Systems and Evolutionary Computation*, Seiten 277–282, 2001.
- [KPMZ02] KNÖDLER, K., J. POLAND, A. MITTERER und A. ZELL: *Memory Economy for Electronic Control Units: Compression of Conventional Look-up Tables*. In: *Recent Advances in Simulation, Computational Methods and Soft Computing*, Seiten 234–239, 2002.
- [Kun77] KUNZMAN, R.: *Thermodynamische Tabellen*. VDI-Verlags GmbH, Düsseldorf Auflage, 1977.
- [LDS90] LECUN, Y., J.S. DENKER und S.A. SOLLA: *Optimal Brain Damage*. In: TOURETZKY, D.S. (Herausgeber): *Advances in Neuronal Information Processing Systems*, Band 2, Denver, 1990. Morgan Kaufmann.
- [Lev44] LEVENBERG, K.: *A method of the solution of certain nonlinear problems in least squares*. Quarterly of Applied Mathematics, (2):164–168, 1944.
- [Lük92] LÜKE, H.D.: *Signalübertragung: Grundlagen der digitalen und analogen Nachrichtenübertragungssysteme*. Springer-Verlag, Berlin, 5. Auflage, 1992.
- [Lud01] LUDWIG, L.: *Automatisierte neuronale Netze zur Analyse technischer Daten mit dem Ziel der Qualitätssicherung*. Doktorarbeit, Eberhard-Karls-Universität Tübingen, Auf dem Sand 13, 2001.
- [Mar63] MARQUARDT, D.W.: *An algorithm for least squares estimation of nonlinear parameters*. Journal of the society of industrial and applied mathematics, (11):431–441, 1963.
- [Mas95] MASTERS, T.: *Advanced algorithms for neural networks - a C++ source-book*. Wiley & Sons, New York, 1995.

- [MBGH96] MÜLLER, R., K.H. BAIER, T. GERN und H.H. HEMBERGER: *Engine Control using Neural Networks*. Band 1, Genova, Italy, 3-4 October 1996. 1st International Conference on Control and Diagnostic in Automotive Applications.
- [MMJ94] MAO, J., K. MOHIUDDIN und A. JAIN: *Parsimonious network design and feature selection through node pruning*. In: *12th International Conference on Pattern Recognition*, Seiten 622–624, 1994.
- [MS89] MOZER, M.C. und P. SMOLENSKY: *Skeletonization: A Technique for Trimming the Fat from a Network via Relevance Assessment*. In: TOUTRETZKY, D.S. (Herausgeber): *Advances in Neuronal Information Processing Systems*, Band 1, Seiten 107–115, San Mateo, 1989. Morgan Kaufmann.
- [PKM⁺01] POLAND, J., K. KNÖDLER, A. MITTERER, T. FLEISCHHAUER, F. ZUBER-GOOS und A. ZELL: *Evolutionary Search for Smooth Maps in Motor Control Unit Calibration*. In: *Stochastic Algorithms: Foundations and Applications*, Seiten 107–116, 2001.
- [PKZ01] POLAND, J., K. KNÖDLER und A. ZELL: *On the Efficient Construction of Rectangular Grids from Given Data Points*. In: *Applications of Evolutionary Computation*, Seiten 110–119, 2001.
- [PM88] PARK, S.K. und K.W. MILLER. *Communication of the ACM*, 31:1192–1201, 1988.
- [Pom97] POMPE, B.: *Die Messung von Informationsflüssen mit einer verallgemeinerten Transinformation*. Doktorarbeit, Ernst-Moritz-Arndt-Universität Greifswald, F.-L.-Jahn-Straße 16, 1997.
- [PTVF97] PRESS, W.H., S.A. TEUKOLSKY, W.T. VETTERLING und B.P. FLANNERY: *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, 2. Auflage, 1997.
- [RL93] ROMANUIK, S. und H. LAWRENCE: *Devide and Conquer Neural Networks*. *Neural Networks*, 6:1105–1116, 1993.
- [RM86] RUMELHART, D.E. und J.L. MCCLELLAND: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. The MIT Press, I, 1986.
- [Rén77] RÉNYI, A.: *Wahrscheinlichkeitsrechnung mit einem Anhang über Informationstheorie*. Deutscher Verlag der Wissenschaften, Berlin, 1977.
- [Roj93] ROJAS, R.: *Theorie der Neuronalen Netze*. Springer, Berlin, 1993.

- [Sar95] SARLE, W.S.: *Stopped Training and other Remedies for Overfitting*. In: *Proceedings of the 27th Symposium on the Interface of Computing Science and Statistics*, Seiten 352–360, 1995.
- [Sch79] SCHRAGE, L. *ACM Transaction on Mathematical Software*, 5:132–138, 1979.
- [Sch96] SCHOCK, G.: *Einzel-Lambda-Schätzung mittels Künstlicher Neuronaler Netze*. Diplomarbeit, Institut für Mechanik, Universität Stuttgart, September 1996.
- [Sha48] SHANNON, C.E.: *A mathematical theory of communication*. *The Bell System Technical Journal*, XXVII:379–423, 623–656, 1948.
- [SJW92] SCHIFFMANN, W., M. JOOST und R. WERNER: *Synthesis and performance analysis of multilayer neural network architecture*. Technischer Bericht, Universität Koblenz, Koblenz, 1992.
- [Sob99] SOBOLL, P.: *Zeitreihenanalyse von Prozeßdaten eines Ottomotors*. Diplomarbeit, Physikalisches Institut der Johann Wolfgang Goethe-Universität Frankfurt am Main, 1999.
- [Spe95] SPECKMANN, H.: *Analyse von fraktalen Dimensionen und Parallelisierung von Kohonens Selbstorganisierender Karte*. Doktorarbeit, Eberhard-Karls-Universität Tübingen, Auf dem Sand 13, 1995.
- [Tac99] TACKE, S.: *Injektordruckanalyse mittels rekurrenter neuronaler Netze - Bestimmung der Einspritzmenge für den OM613 CDI Motor*. Diplomarbeit, Fachbereich technische Informatik, Fakultät für Informatik, Eberhard-Karls-Universität Tübingen, Sand 13, 72076 Tübingen, 31. Dezember 1999.
- [Tol90] TOLLENAERE, T.: *Supersab: Fast adaptive Backpropagation with good scaling properties*. *Neural Networks*, (3):561–573, 1990.
- [Wed99] WEDLER, E.: *Abschätzung der Gültigkeitsdomäne eines neuronalen Netzes*. Diplomarbeit, Fachhochschule Stuttgart, Hochschule für Technik, Fachbereich Mathematik, 1999.
- [Wei99] WEISSKOPF, D.: *Untersuchung, Implementierung und Einsatz des LEVENBERG-MARQUARDT-Algorithmus zur Analyse von Chip-Fertigungsdaten*. Diplomarbeit, Eberhard-Karls-Universität Tübingen, 1999.
- [Zel93] ZELL, A.: *Simulation Neuronaler Netze*. Addison Wesley, 1993.

