

Früherkennung sich anbahnender Störungen mit Hilfe neuronaler Netze

Dissertation
zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

von Dipl.-Ing. Sascha Grünbeck, M.Sc.
geboren am 07. 06. 1977
in Magdeburg

genehmigt durch die Fakultät für Verfahrens- und Systemtechnik der Otto-von-Guericke-Universität Magdeburg

Gutachter: Prof. Dr.-Ing. Dr. h.c. Ulrich Hauptmanns
Prof. Dr.-Ing. habil. Frank Palis

eingereicht am: 01. 03. 2010
Promotionskolloquium am: 31. 10. 2010

Zusammenfassung

Durch gezielte Erkennung sich anbahnender Fehlerzustände können das Ansprechen von Sicherheitseinrichtungen vermieden und angepasste Wartungsstrategien eingesetzt werden. Auf Grundlage einer technischen Systemanalyse können die Korrelationen der gemessenen Systemparameter aufgedeckt werden und unter Zuhilfenahme protokollierter historischer Messwertverläufe eine Mustererkennung mittels neuronaler Netze durchgeführt werden.

Neuronale Netze stellen eine Variante statistischer Verfahren dar, die auch, oder gerade bei Fehlen eines Prozessmodells, den Prozess als nichtlineare Funktion über ihre Netzwerkarchitektur abbilden können. Entscheidender Vorteil neuronaler Netze ist die Fähigkeit, ähnliche jedoch nicht bekannte Eingangsmuster dem entsprechenden gelernten Muster zuzuordnen und somit auf den korrekten Ausgabevektor abzubilden. Hierbei findet der Wissenstransfer durch einen Lernprozess statt, bei dem einem Eingabevektor ein entsprechender Ausgabevektor zugeordnet wird.

Problematisch stellt sich die Anwendung auf reale Systeme dar. Fehler sind seltene Ereignisse und werden, wenn sie aufgetreten sind im Idealfall so behoben, dass sie nicht wieder auftreten können. Es herrscht somit ein Informationsdefizit.

An vier Referenzfällen wurde die Anwendung des entwickelten Verfahrens erprobt. Gezeigt werden die Ergebnisse der technischen Systemanalyse, der Vergleich unterschiedlicher Netzwerkarchitekturen, Probleme und Lösungsansätze zum Ausgleich des Informationsdefizits sowie eine alternative Behandlungsmethode.

Inhaltsverzeichnis

1	Einleitung	1
2	Problemstellung und Zielsetzung.....	3
3	Stand der Technik	5
3.1	Methoden der Fehlererkennung	5
3.2	Kontrolle direkt messbarer Signale.....	7
3.3	Analyse von Signalmodellen	8
3.4	Analyse von Prozessmodellen	9
3.5	Berechnung von Kennwerten	10
3.6	Bewertung der Methoden	11
4	Neuronale Netze.....	12
4.1	Training des Netzes – überwachtes Lernen	17
4.2	Netzanwendung	22
4.3	Einfluss der Anzahl der Trainingsdurchläufe – Übertraining.....	22
4.4	Anzahl Trainingssätze	23
4.5	Einfluss Anzahl der verdeckten Schichten auf Effizienz	25
4.6	Einfluss Neuronenanzahl.....	29
4.7	Lernrate	29
4.8	Einfluss der Wahl des Lernverfahrens.....	30
4.9	Gütemaße	32
5	Übersicht über bisherige Arbeiten zur Fehlerfrüherkennung	36
6	Entwicklung des Verfahrens zur Fehlerfrüherkennung	41
6.1	Technische Systemanalyse.....	42
6.2	Merkmalsextraktion	46
6.3	Datenvorverarbeitung	48
6.4	Musterklassifikation – Netzwerkerstellung.....	51
7	Anwendungen.....	53
7.1	Referenzfall „Zwischenüberhitzer“	53
7.1.1	Verfahrenstechnische Beschreibung der Zwischenüberhitzer	53
7.1.2	Fehlerbeschreibung	54
7.1.3	Merkmalsextraktion.....	57
7.1.4	Anwendung des neuronalen Netzes	59

7.2	Referenzfall „Kompensatorschaden an einer Mitteldruckturbine“	66
7.2.1	Technische Systemanalyse	66
7.2.2	Auswertung der Messdaten	68
7.3	Referenzfall „Lagerschaden“	73
7.3.1	Technische Systemanalyse	73
7.3.2	Anwendung des neuronalen Netzes	74
7.4	Referenzfall „Wärmeübertrager“	83
7.4.1	Systemanalyse regenerativer Luftvorwärmer Bauart „Rothemühle“ ..	83
7.4.2	Merkmalsextraktion.....	86
7.4.3	Anwendung neuronales Netz.....	92
8	Programmtechnische Umsetzung.....	99
8.1	Untersuchungen mit dem Programm „Chem“	99
8.2	Programmbeschreibungen	105
8.3	Adaption des Programms	107
8.4	Trainingsalgorithmen	107
8.5	Ergebnisse	109
9	Vergleichsmethode	112
9.1	Fuzzy Regelung – Ansatz von Mamdani	112
9.2	Referenzfall „Lagerschaden“	118
9.3	Referenzfall „Zwischenüberhitzer“	120
10	Zusammenfassung und Ausblick.....	122
	Literaturverzeichnis.....	126

Abbildungsverzeichnis

Abbildung 3-1: Schematische Darstellung eines Prozesses, der durch einen Fehler beeinflusst wird /Ise94/.....	7
Abbildung 4-1: Schematische Darstellung eines neuronalen Netzes mit einer verdeckten Schicht.....	13
Abbildung 4-2: Lineare Separierbarkeit und das XOR- Problem.....	26
Abbildung 4-3: Beispiele für Abbildungsfähigkeiten vorwärtsgerichteter Netzwerke/	28
Abbildung 6-1: Messgrößenidentifizierung durch FMEA und PAAG	43
Abbildung 6-2: Regelkreis	44
Abbildung 6-3: Vorgehensweise zur Erstellung der Netzwerkarchitektur	52
Abbildung 7-1: R+I Fließbild der Zwischenüberhitzerkreisläufe	54
Abbildung 7-2: Parameteränderungen in Abhängigkeit vom Lastzustand	56
Abbildung 7-3: Differenzmassenstrom auf Grund der Leckage.....	61
Abbildung 7-4: Netzausgabe Zwischenüberhitzer mit Befund, gesamter Fehlerbereich	62
Abbildung 7-5: Netzausgabe Zwischenüberhitzer ohne Befund gesamter Fehlerbereich	63
Abbildung 7-6: Netzausgabe ZÜ ohne Befund, selektierter Fehlerbereich.....	64
Abbildung 7-7: Netzausgabe ZÜ mit Befund und selektiertem Fehlerbereich ..	64
Abbildung 7-8: Schematische Darstellung Axialkompensator	68
Abbildung 7-9: Änderungen der Temperaturdifferenzen	71
Abbildung 7-10: Änderungen der Druckdifferenzen	72
Abbildung 7-11 Ergebnis Verifizierung Lager SAC03.....	77
Abbildung 7-12 Ergebnis der Verifizierung Lager SAC03 mit Berücksichtigung der Messgröße ΔdB	79
Abbildung 7-13 Ergebnis der Verifizierung unter Berücksichtigung der Übergangszustände	80
Abbildung 7-14: Anwendung des Datensatz aus SAC 02 auf trainiertes Netz .	81
Abbildung 7-15: Regenerativer Luftvorwärmer „Rothemühle“	84
Abbildung 7-16: Schematische Darstellung des Luftvorwärmers	85
Abbildung 7-17: Zusammenstellung der als relevant erachteten Parameter....	88

Abbildungsverzeichnis

Abbildung 7-18: Auswahl der verfügbaren und relevanten Parameteränderungen	89
Abbildung 7-19 Netzausgabe für Validierungssatz Netzkonfiguration (5/3)	94
Abbildung 7-20 Netzausgabe für Validierungssatz Netzkonfiguration (5/7)	95
Abbildung 7-21 Netzausgabe für Validierungssatz Netzkonfiguration (7/3)	97
Abbildung 7-22 Netzausgabe für Validierungssatz Netzkonfiguration (7/9)	98
Abbildung 8-1 Programmablaufplan Programm „Chem“	100
Abbildung 8-2 Anzahl der Trainingsdurchläufe in Abhängigkeit von der Lernrate	102
Abbildung 8-3 Generalisierungsfähigkeit in Abhängigkeit der Netzgröße	103
Abbildung 8-4: Trefferquoten unterschiedlicher Netzkonfigurationen.....	104
Abbildung 8-5 Programmablaufplan für NeuroTrain.....	105
Abbildung 8-6 Programmablaufplan für NeuroApp.....	106
Abbildung 8-7: Verlauf des Trainings mit RPROP	110
Abbildung 8-8: Verlauf des Trainings mit Levenberg-Marquardt	110
Abbildung 9-1: monoton steigende lineare Zugehörigkeitsfunktion.....	114
Abbildung 9-2: monoton fallend lineare Zugehörigkeitsfunktion.....	114
Abbildung 9-3 Programmablaufplan Fuzzy Anwendung	117
Abbildung 9-4: Ergebnis der Fuzzy Anwendung für Lager SAC02.....	118
Abbildung 9-5: Ergebnis der Fuzzy Anwendung für Lager SAC03.....	119
Abbildung 9-6: Ergebnis der Fuzzy Anwendung für RB20	121

Tabellenverzeichnis

Tabelle 6-1: Mittelwertbestimmung für zeitliche Verläufe von Dampfdrücken ..	46
Tabelle 7-1: Ergebnisse Kruskal-Wallis Test.....	59
Tabelle 7-2 Übersicht über die Temperaturmessstellen für die Fehlerfrüherkennung.....	69
Tabelle 7-3: Übersicht über die Druckmessstellen für die Fehlerfrüherkennung	70
Tabelle 7-4: Erwartete Änderung der Differenzen zwischen den Parametern..	70
Tabelle 7-5: Übersicht über vorhandene Messwerte.....	75
Tabelle 7-6: Trainingssatz Lager SAC03	76
Tabelle 7-7: Validierungssatz für Training SAC 03.....	76
Tabelle 7-8: Trainingssatz für SAC03 unter Berücksichtigung der Different Δ B	78
Tabelle 7-9: Validierungssatz für Training SAC 03 mit Δ B	78
Tabelle 7-10: Trainingssatz mit Berücksichtigung von Übergangszuständen ..	80
Tabelle 7-11 Mittelwerte für Prozessparameter bei fehlerbehaftetem und fehlerfreiem Nennlastbetrieb	90
Tabelle 7-12: Trainingsdatensatz (5 Eingangsparameter)	92
Tabelle 7-13 Validierungsdatensatz	93
Tabelle 7-14: Trainingsdatensatz für 7 Eingabeneuronen.....	96
Tabelle 7-15: Validierungsdatensatz für 7 Eingangsneuronen.....	96
Tabelle 8-1: Netzausgabe im Verifizierungsschritt	111

Symbolverzeichnis

U		Messgröße
t	[s]	Zeit
Θ		Prozessgröße
Δ		Differenz
X		Zustandsgröße
Y		Signalgröße
R		Autokorrelationsfunktion
τ		Zeitschritt
C		Autokovarianzfunktion
S		Leistungsdichte (Kapitel 3)
X		Eingangsschicht neuronales Netz
H		Verdeckte Schicht neuronales Netz
I		Ausgabeschicht neuronales Netz
o		Ausgangswert Neuron
w		Gewicht (neuronales Netz)
a		Aktivierungsfunktion (neuronales Netz)
E		Fehler (neuronales Netz)
p		Muster (neuronales Netz)
t		Lerngröße (neuronales Netz)
δ		Fehlersignal (neuronales Netz)
ε		Fehlerschranke (neuronales Netz)
E_g		Verallgemeinerungsfehler (neuronales Netz)
sgn		Vorzeichen
η		Lernrate
Z		Volumen des Gewichtsraumes
I_s		Shannon Entropie
V		Wert der Information
p,p'		Wahrscheinlichkeiten
H		Teststatistik
S		Rangsummen (Kapitel 6)
C		Korrekturfaktor (Kapitel 6)

Symbolverzeichnis

r		Bindungen (Kapitel 6)
T		Rangsummen (Kapitel 6)
H'		Korrigierte Teststatistik
x		Merkmal (Kapitel 6)
μ		Erwartungswert (Kapitel 6)
σ		Varianz (Kapitel 6)
LR	[dB]	Messgröße
HR	[dB]	Messgröße
RB		Rohrbündel
SAC		Lagerbezeichnung
Vib_		Schwingung horizontal
H		
Vib_		Schwingung vertikal
A		
μ		Zugehörigkeitsfunktion Fuzzy
ξ		Messgröße (Kapitel 9)
θ		Fuzzy Ausgabe
y		„Stellgröße“ (Kapitel 9)
min		Minimum
max		Maximum
i,j,k,r		Laufvariablen

1 Einleitung

Der sichere Betrieb industrieller Anlagen ist die Grundlage einer erfolgreichen Volkswirtschaft. Basierend auf einer ausgewogenen Planung und Konstruktion der Anlagen stellt die Erhaltung der Betriebsfähigkeit sowie die Vermeidung von Störungen über die Laufzeit der Anlagen eine der Hauptherausforderungen für den Ingenieur dar.

Technische Systeme zeichnen sich durch einen hohen Grad an Zuverlässigkeit, aber auch an Komplexität aus. Mit Hilfe von Prozessleitsystemen wird eine Vielzahl von Prozessparametern aufgenommen und so der Gesamtzustand des Systems überwacht. Die nach wie vor häufigste Art der Prozessüberwachung erfolgt hierbei durch die Kontrolle der Prozessparameter, ihrer Änderungsgeschwindigkeit oder ihrer Trends etc. Wird der zulässige Bereich verlassen, so werden Regeleinrichtungen aktiviert, um die entsprechenden Größen auf zulässige Werte zurückzuführen. Reicht die Regelfähigkeit dieser Einrichtungen nicht aus, so werden Schutzeinrichtungen angesprochen, die zur Abschaltung der Anlage oder von Anlagenteilen führen.

Existieren Prozessmodelle, so können Residuen, also die Abweichungen zwischen den gemessenen Prozessgrößen und den durch das Modell berechneten Werten, bestimmt werden. Damit kann im Rahmen der Modellgüte eine Gesamtschau der Prozessparameter einschließlich ihrer Abhängigkeiten voneinander erfolgen und auf dieser Grundlage korrigierend in den Prozess eingegriffen werden. Da die notwendige Modellierung zumeist zu kompliziert und aufwendig und in einigen Fällen gar nicht realisierbar ist, wird diese Vorgehensweise selten angewandt. Sie ist aber Gegenstand aktueller Forschung.

Deshalb fällt in der Regel dem Bedienpersonal die Aufgabe zu, aus den Werten unterschiedlicher Prozessparameter den Anlagenzustand zu erkennen und, falls notwendig, einzugreifen. Dies erfordert ein hohes Maß an Erfahrung, die vor allem aus bereits aufgetretenen Abweichungen vom Nominalzustand der Anlagen resultiert. Allerdings ist es für das Betriebspersonal schwierig, eine

über längere Betriebszeiten auftretende Charakteristik von Abweichungen zu analysieren und Ähnlichkeiten bei erneutem Auftreten zu erkennen. Eine Übertragung von Erfahrungen auf neue Mitarbeiter ist ein weiteres Problem. Deshalb wird die Erfahrung aus dem Anlagenbetrieb derzeit nur unzureichend genutzt. In der vorliegenden Arbeit wird dem für das Gebiet betrieblicher Störungen dadurch abgeholfen, dass das Wissen des Anlagenpersonals mit Hilfe von Rechnerprogrammen „konserviert“ wird.

Gegenstand der Arbeit ist es, die Überwachung vieler Prozessparameter und deren Korrelationen durch die Entwicklung einer Methode, die die Elemente „Parameterüberwachung“, „Nutzung von Informationen aus der Vergangenheit“ sowie „modellunabhängige Analyse“ enthält, zu ermöglichen. Dabei ist Randbedingung, dass keine zusätzlichen Messstellen benötigt und deshalb keinerlei Veränderungen an der Anlage erforderlich werden. Die Methode unterstützt das Betriebspersonal dabei, Störungen frühzeitig zu erkennen.

Dies gelingt mit Hilfe der Mustererkennung. Sich anbahnenden Fehler werden aufgrund von Veränderungen in den Signalmustern detektiert und dann dem Operateur signalisiert.

2 Problemstellung und Zielsetzung

Auf Grund des hohen Überwachungsgrades technischer Anlagen wird eine Vielzahl von Prozessgrößen messtechnisch erfasst. Sich anbahnende Störungen führen, wie gezeigt wird, zu einer Veränderung der Prozessparameter. Diese werden mit Fortschreiten der Störungsentwicklung im Allgemeinen größer. Das zur Interpretation erforderliche Systemverständnis wird mittels einer technischen Systemanalyse strukturiert aufgebaut. Durch diese werden die Auswirkungen auf den Prozess und die relevanten Messgrößen, in denen sich die Auswirkungen widerspiegeln, herausgearbeitet. Unter Berücksichtigung des bestimmungsgemäßen Betriebs als Referenz stellen Auswirkungen der Störungen Signalmuster dar, die für jeden Fehler signifikant sind und vom Muster des Normalbetriebs abweichen. Durch den Einsatz einer geeigneten statistischen Methode werden diese Muster aus den Anlagendaten herausgelesen. Im nächsten Schritt wird das Wissen über die Fehlermuster dahin gehend genutzt, den laufenden Anlagenbetrieb überwachend zu begleiten. Hierzu sind neuronale Netze zum Einsatz gekommen, die bereits in anderen Gebieten erfolgreich zur Mustererkennung eingesetzt wurden.

Die Entwicklung der Vorgehensweise zur frühzeitigen Entdeckung sich anbahnender Störungen soll die Operateure technischer Anlagen wie z. B. Kernkraftwerke dahin gehend unterstützen, Veränderungen im technischen System so rechtzeitig zu erkennen, dass eine Ausweitung zu Störung oder Störfall und somit eine Anforderung der Sicherheitssysteme, die bei ihrem weiterem ungehinderten Verlauf notwendig würden, nicht erfolgt. Damit wird ein wirkungsvoller Beitrag zur Sicherheit des Reaktorbetriebs geleistet, da an der Schnittstelle Mensch Maschine, deren Optimierung im Rahmen der Reaktorsicherheitsforschung vom BMWi gefördert wird, die Kommunikation zwischen beiden Ebenen verbessert wird. Die Verknüpfung von Systemanalyse und Beurteilung prozessleitetechnischer Messwerte ermöglicht hierbei eine umfassendere Sicherheitsbeurteilung als sie bei getrenntem Vorgehen möglich wäre. Durch diesen Synergieeffekt wird ein wesentlicher Beitrag zur Erhöhung der Sicherheit geleistet.

Im Rahmen der Arbeit wird eine Übersicht über die gängigen Methoden der Fehlererkennung und ihre Anwendbarkeit zur Fehlerfrüherkennung gegeben. Hieran schließt sich ein Überblick über neuronale Netze, der das Verständnis ihrer Anwendung in der Fehlerfrüherkennung ermöglicht und die Grundlagen für die entwickelte Methode vermittelt. Anschließend wird eine Übersicht über Forschungsergebnisse auf dem Gebiet der Fehlerfrüherkennung gegeben und der Nutzen der vorliegenden Arbeit aus den Defiziten bisheriger Forschungsergebnisse abgeleitet. Auf Grundlage der technischen Systemanalyse, deren Vorgehensweise und Anforderungen vorgestellt werden, wird die Merkmalsextraktion als Eingangsvektor für die Fehlerfrüherkennungsmethode aufgezeigt. Hierauf aufbauend wird die Methode der Fehlerfrüherkennung vorgestellt und ihre programmtechnische Umsetzung erläutert. Die Anwendung der entwickelten Methode auf Referenzfälle und eine Bewertung der Ergebnisse folgen.

Um die Umsetzung und effektive Anwendung der Methode einschätzen zu können, wird ein Vergleich mit der Anwendung unscharfer Mengen („Fuzzy sets“) vorgenommen. Somit kann eingeschätzt werden, welche der beiden Methoden effizienter ist.

Ziele der Arbeit sind die Entwicklung einer übertragbaren Methodik zur Erkennung sich anbahnender Störungen in komplexen technischen Anlagen und der Nachweis, dass der Einsatz neuronaler Netze mit realen Daten aus den Anlagen möglich ist. Weiterhin wird gezeigt, dass die Verwendung korrelierter Parameter Vorteile gegenüber der Nutzung von Grenzwerten einzelner Parameter hat.

3 Stand der Technik

3.1 Methoden der Fehlererkennung

Um die Zahl von Ausfällen technischer Anlagen so gering wie möglich zu halten, bedarf es eines Systems zur Fehlererkennung. Hierzu muss zunächst definiert werden, wann von einem Fehler und wann von einem Ausfall zu sprechen ist. Isermann leitet die Definition des Fehlers aus verschiedenen Normen wie folgt ab: „Unzulässige Abweichung mindestens eines Merkmals einer Betrachtungseinheit“ [Ise94/]. Dabei stellt die unzulässige Abweichung einen über den Toleranzbereich hinausgehenden Unterschied, zwischen Ist- und Sollwert dar. Der Toleranzbereich ist dabei durch die Randbedingungen des jeweiligen Prozesses gegeben. Isermann unterscheidet weiterhin verschiedene Fehlerarten wie z.B. Entwurfsfehler, Konstruktionsfehler, Bedienfehler, etc. Gleichzeitig wird betont, dass ein Fehler nicht zwangsläufig die Funktion der Betrachtungseinheit beeinträchtigen muss. Allerdings kann ein Fehler, wenn er eine Funktionsbeeinträchtigung verursacht, eine Störung oder einen Ausfall der Anlage zur Folge haben.

Für die in der vorliegenden Arbeit behandelte Fragestellung sind Fehler, die einen Ausfall der Betrachtungseinheit zur Folge haben, relevant. Der Ausfall wird von Isermann als „nach Beanspruchungsbeginn entstandenes Aussetzen der Ausführung einer Aufgabe einer Betrachtungseinheit aufgrund einer in ihr selbst liegenden Ursache und im Rahmen der zulässigen Beanspruchung“ definiert. Die verschiedenen Arten von Ausfällen unterscheidet Isermann z.B. aufgrund der Vorhersehbarkeit des Ausfalls: „Zufallsausfall und systematischer Ausfall“, „zeitliches Ausfallverhalten“ sowie „Sprung- und Driftausfall“.

Beim Sprungausfall bewirkt das Auftreten eines Fehlers spontan den Ausfall der Betrachtungseinheit. Diese Art von Fehler kann man mit einem Fehlerfrüherkennungssystem, das den Fehler, bevor es zu einem Ausfall kommt, erkennen soll, nicht beikommen; denn die Identifizierung eines Fehlers und die Durchführung notwendiger Gegenmaßnahmen benötigen Zeit, die naturgemäß beim

Sprungausfall nicht zur Verfügung steht. Der Sprungausfall muss durch die vorhandenen Sicherheitssysteme beherrscht werden.

Beim Driftausfall liegt zwischen dem Auftreten eines Fehlers und dem Ausfall der Betrachtungseinheit eine Zeitspanne, die dafür genutzt werden kann, den Fehler zu identifizieren und zu beheben, bevor es zum Ausfall kommt.

Auf die Frage, ab welcher Zeitspanne von einem Driftausfall zu sprechen ist, wird in der gesichteten Literatur nicht eingegangen. Letztendlich sind jedoch die Identifizierungszeit und die Zeitspanne, die für geeignete Gegenmaßnahmen nötig sind, entscheidend dafür, ob ein Sprung- oder Driftausfall vorliegt.

Um Fehler zu erkennen, muss der zu Grunde liegende Prozess überwacht werden. Bei einem Prozess können die Eingangs- und Ausgangssignale gemessen werden. Abbildung 3-1 zeigt schematisch einen Prozess, der durch einen internen oder externen Fehler beeinflusst wird. Durch einen Fehler werden sich zunächst die internen Prozessparameter und Zustandsgrößen verändern. Nur wenn diese Änderungen Einfluss auf das Ausgangssignal haben bzw. eine signifikante Abweichung der Regelparameter auftritt, kann der Fehler festgestellt werden. Zusätzlich zu den auftretenden Fehlern beeinflussen meist nicht messbare Störsignale die Ausgangssignale.

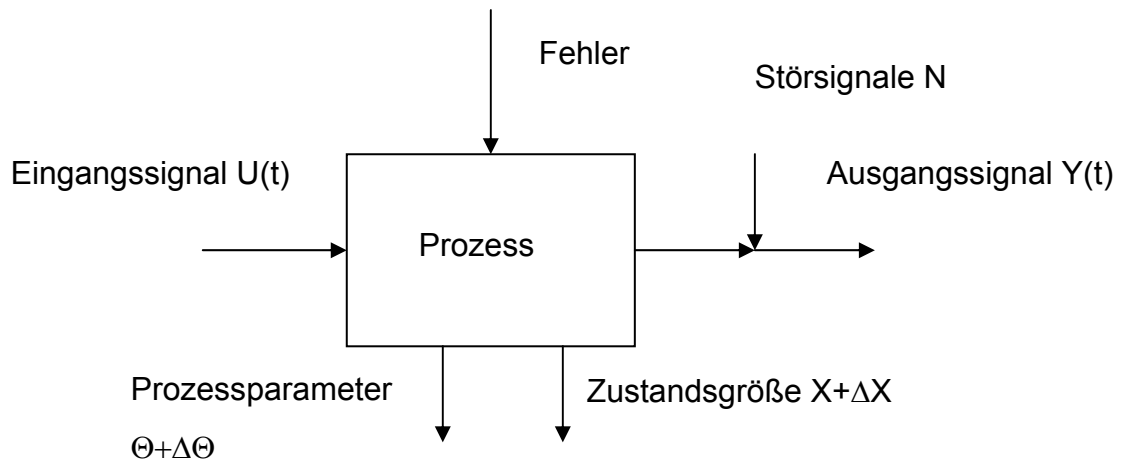


Abbildung 3-1: Schematische Darstellung eines Prozesses, der durch einen Fehler beeinflusst wird /Ise94/

Im Folgenden werden verschiedene Methoden der Fehlererkennung vorgestellt und ihre Eignung für die zu entwickelnde Methode bewertet. Für die Bewertung ist dabei ausschlaggebend, ob die Art der Prozessüberwachung es erlaubt, einen sich anbahnenden Fehler frühzeitig zu erkennen.

3.2 Kontrolle direkt messbarer Signale

Zu den direkt messbaren Signalen zählen alle physikalisch messbaren Größen, die durch die Messtechnik der Anlage aufgenommen werden können. Zur Fehlererkennung kann die Grenzwertüberwachung eingesetzt werden, die zu den Absolutwertkontrollen zählt. Sobald ein festgelegter Bereich nach oben oder nach unten verlassen wird, erfolgt eine Meldung. Entscheidend ist, dass Bereiche eingestellt werden, die groß genug sind, um die stochastischen und betriebsbedingten Schwankungen der Prozessparameter zu tolerieren, deren Grenzen aber weit genug vom Auftreten einer Gefährdung entfernt sind. Absolutwertkontrollen sind für statische, nur durch Prozessschwankungen variierende jedoch nicht für dynamische Vorgänge, wie z. B. An- und Abfahrvorgängen, geeignet. Hier sind dynamische Anpassungen der Grenzwerte nötig.

Bei der Trendkontrolle wird die Änderungsgeschwindigkeit überwacht, also die zeitliche Änderung des Merkmals. Sie wird meist bei sich langsam ändernden Signalen angewendet. Durch eine genügend klein gewählte Grenzwertsetzung - die zulässige Änderungsgeschwindigkeit wird nach oben und / oder unten begrenzt - kann eine Störungsmeldung früher als bei der Absolutwertkontrolle erfolgen /Ise94/.

Die Kombination von Absolutwert- und Trendkontrolle soll eine möglichst frühe Störungserkennung bei einer geringen Anzahl an Fehlalarmen ermöglichen.

Anhand bereits gemessener Signale kann mit Hilfe der Signalvorhersage der weitere Verlauf des Signals bestimmt werden. Hierzu wird ein mathematisches Modell des Signals benötigt, das beispielsweise durch Regressionsanalyse für deterministische Signale oder durch Parameterschätzung für stochastische Signale erstellt wird /Ise94/.

Messwerte müssen im Allgemeinen erst aufbereitet werden, bevor sie in der Fehlerfrüherkennung genutzt werden. Dies ist notwendig, um Messrauschen herauszufiltern oder Ausreißer zu erkennen und zu eliminieren.

3.3 Analyse von Signalmodellen

Aus dem zeitlichen Verlauf der Signale kann man bei Annahme stationärer Signale den Mittelwert und die Varianz des Signals bestimmen. Mit Hilfe dieser beiden Größen können verschiedene Funktionen abgeleitet werden, die dann das Signalmodell darstellen. Hier ist z. B. die Autokorrelationsfunktion zu nennen, die sich aus dem Produkt eines Signals zum Zeitpunkt t , $Y(t)$, und seinem Wert zum Zeitpunkt $Y(t+\tau)$ ergibt:

$$R(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T Y(t) \cdot Y(t + \tau) dt = E\{Y(t) \cdot Y(t + \tau)\} \quad (3-1)$$

Die Autokorrelationsfunktion beschreibt den inneren Zusammenhang des Signals und stellt ein (statistisches) Modell für den Zeitbereich dar /Ise94/. Ein wei-

teres häufig angewandtes Signalmodell im Zeitbereich, die Autokovarianzfunktion, betrachtet die Abweichungen vom Mittelwert:

$$C(\tau) = E\{(Y(t) - \bar{Y}) \cdot (Y(t + \tau) - \bar{Y})\} \quad (3-2)$$

Die fourier-transformierte Autokovarianzfunktion stellt ein Signalmodell im Frequenzbereich dar und wird auch als Leistungsdichte bezeichnet:

$$S(i\omega) = \int_{-\infty}^{\infty} C(\tau) \cdot e^{-i\omega\tau} d\tau \quad (3-3)$$

Die Bewertung der Signalmodelle kann nun wieder durch eine Absolutwert- oder Trendkontrolle erfolgen.

3.4 Analyse von Prozessmodellen

Die geschilderten Methoden erlauben unterschiedliche Untersuchungen der Ausgangssignale auf Veränderungen. Woher diese Veränderungen im Einzelnen stammen, kann jedoch mit den genannten Methoden nicht ermittelt werden. Um dies zu erreichen, werden mathematische Prozessmodelle benötigt, die den Zusammenhang von Eingangssignalen, Prozessparametern und Ausgangssignalen nachbilden und zum Teil die Störsignale N (vgl. Abbildung 3-1) von den Ausgangssignalen trennen können /Ise94/. Prozessmodelle nutzen die Abhängigkeiten der Signale voneinander, um die daraus resultierende analytische Redundanz zum realen Prozess zur Identifizierung und Lokalisierung des Fehlers zu nutzen. Sie ermöglichen außerdem die indirekte Berechnung der internen Prozessparameter Θ und der Zustandsgrößen X , wodurch eine frühere Fehlererkennung möglich wird /Ise94/.

In Prozessmodellen werden dazu die physikalischen und chemischen Zusammenhänge, durch die Beziehungen zwischen Ein- und Ausgangsgrößen, beispielsweise durch Differentialgleichungssysteme dargestellt.

Die Prozessmodelle stellen damit eine Möglichkeit dar, Fehlerzustände zu erkennen. Die Modellierung ist hierbei jedoch, wie bereits erwähnt, sehr aufwendig und in vielen Fällen derzeit nicht möglich. Ein weiterer Nachteil ist, dass meist nur das ideale Systemverhalten modelliert werden kann und somit nur Abweichungen von diesem erkannt werden. Die Zuordnung zu einzelnen Fehlerarten kann, ohne das Vorhandensein einzelner Fehlermodelle, durch das Modell nicht realisiert werden.

Mit Hilfe einfacher Prozessmodelle können jedoch Eingangsgrößen für neuronale Netze geschaffen werden, wenn beispielsweise signifikante Parameter nicht in Form von Messgrößen vorliegen.

3.5 **Berechnung von Kennwerten**

Die Berechnung von Kennwerten, wie z. B. Wirkungsgraden, aus Ein- und Ausgangssignalen bietet oft wichtige Hinweise auf den Zustand eines Prozesses. Sind die Einflüsse von Fehlerquellen auf die Kennwerte bekannt, so kann aus Änderungen der Kennwerte auf den Fehler geschlossen werden. Da die Kennwerte jedoch aus den gemessenen Signalen berechnet werden, stellt die Kennwertkontrolle nur eine Erweiterung der Signalkontrolle dar.

Interessante Untersuchungen zur Steuerung chemischer Prozesse an Hand der Auswertung von Kennzahlen mit Hilfe neuronaler Netze wurden von Neumann /Neu02/ angestellt. Die Idee ist, die bekannten charakteristischen Muster der Prozesse im Zustandsraum zur Mustererkennung zu nutzen. Der Zustandsraum soll so beschaffen sein, dass man in der Lage ist, die Netze mit einfachen Prozessbeispielen im Labormaßstab anzulernen und sie dann ohne Bedarf des Nachlernens am realen Prozess im Produktionsbetrieb einzusetzen. Die dimensionslose Vorgehensweise bewirkt eine Invarianz gegenüber Skalierungsproblemen und bezüglich der Reaktionsparameter. Ein auf dieser Basis arbeitendes neuronales Netz benötigt eine intensive Vorverarbeitung der Daten, da eine eindeutige Transformation in den dimensionslosen Zustand notwendig ist. Problematisch sind hierbei sowohl die Übertragbarkeit der Labordaten auf den realen Prozess als auch die Gewinnung der Kennzahlen.

3.6 **Bewertung der Methoden**

Die vorgestellten Methoden lassen entweder keine effektive Fehlerfrüherkennung zu oder sind, wie im Fall der Prozessmodellierung, aufwendig oder nicht realisierbar. Es musste daher eine Methode entwickelt werden, die es ermöglicht, eine datenbasierte Fehlerfrüherkennung durchzuführen. Wie bereits erwähnt, stellen neuronale Netze eine Möglichkeit dar, Muster zu identifizieren. Entsprechend wird nachfolgend ein einführender Überblick über Aufbau und Arbeitsweise neuronaler Netze gegeben.

4 Neuronale Netze

Ein neuronales Netzwerk (NN) besteht aus einer Vielzahl einfacher Verarbeitungseinheiten, den Neuronen, und aus den gewichteten Verbindungen zwischen ihnen. Das Vorbild technischer neuronaler Netzwerke bildet das Nervensystem der Lebewesen. Trotz der relativ langsamen Weitergabe elektrischer Impulse durch biochemische Reaktionen ist das Nervensystem in der Lage, durch seine stark parallele Arbeitsweise, höchst komplexe Operationen in kürzester Zeit durchzuführen. Seriell arbeitende Computer sind trotz ihrer um Zehnerpotenzen schnelleren Verarbeitungszeit einzelner Rechenschritte langsamer.

Die Vielfalt der Arten künstlicher neuronaler Netze beruht zum einen auf den unterschiedlichen Verarbeitungsstrategien, die durch die Neuronen verwirklicht werden können und zum anderen auf den verschiedenen Verknüpfungsmöglichkeiten der Neuronen untereinander. Neuronale Netze basieren auf der Weiterleitung von Signalen, die aus den Eingängen der Neuronen berechnet werden. Dies wird im Folgenden näher erklärt.

Neuronale Netze sind schichtenweise aufgebaut. Es wird im Allgemeinen zwischen der Eingabeschicht, den verdeckten Schichten und der Ausgabeschicht unterschieden. Abbildung 4-1 zeigt den schematischen Aufbau eines künstlichen neuronalen Netzes.

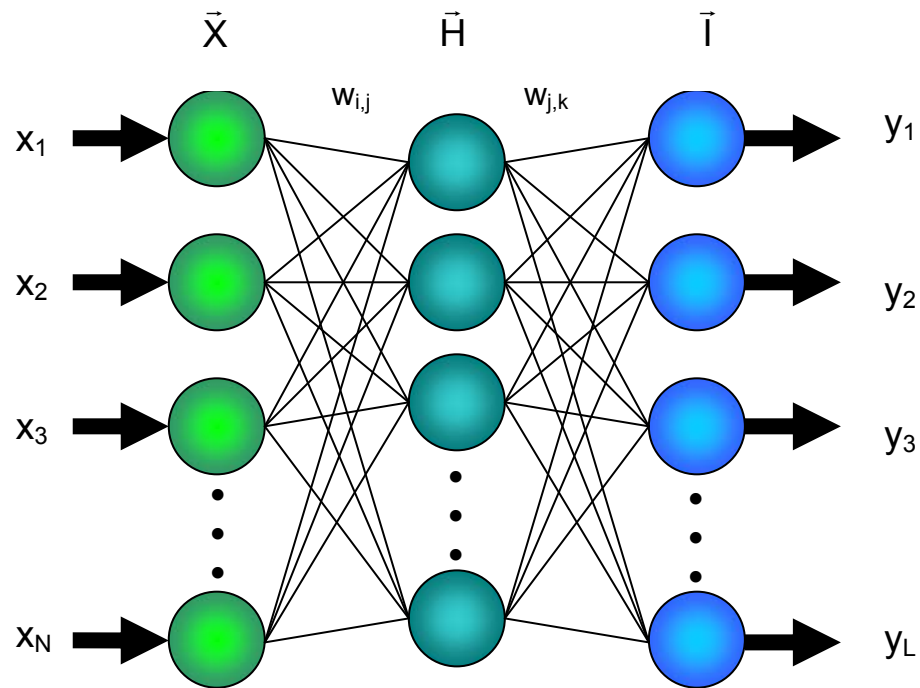


Abbildung 4-1: Schematische Darstellung eines neuronalen Netzes mit einer verdeckten Schicht

Es gelten die folgenden Bezeichnungen:

- Eingangsgrößen $\bar{X} = \{x_1, x_2, \dots, x_N\}$
- Ausgabe der Neuronen der Eingangsschicht $o_i = f(x_i) \quad i = 1, \dots, N$
- Für die verdeckte Schicht $H_j = \sum_i w_{i,j} \cdot o_i = \sum_i w_{i,j} \cdot f(x_i)$ und
 $o_j = f(H_j) \quad j = 1, \dots, P$
- Für die Ausgabeschicht: $l_k = \sum_j w_{j,k} \cdot o_j = \sum_j w_{j,k} \cdot f(H_j)$ und
 $y_k = o_k = f(l_k) \quad k = 1, \dots, L$

Neuronen der Eingabeschicht \bar{X} verfügen in der Regel nur über einen Eingang z. B. die Messwerte der Prozessparameter einer zu untersuchenden Anlage (x_1, \dots, x_N) . Dabei steht jedes Eingabeneuron für einen Parametermesswert, Sie verarbeiten ihr Eingangssignal wie folgt:

$$o_i = f(x_i) \quad (4-1)$$

Hier ist f eine Funktion mit bestimmten Eigenschaften, auf die noch eingegangen wird.

Neuronen der weiteren Schichten verfügen über so viele Eingänge, wie sie Eingangsverbindungen mit anderen Neuronen haben.

Die Neuronen der verdeckten Schicht besitzen keine Verbindung nach Außen und erhalten ihren Eingang von allen Neuronen, mit deren Ausgabe sie verbunden sind.

Aus der Summe aller Eingänge berechnet sich die Netzeingabe eines Neurons der verdeckten Schicht H :

$$H_j = \sum_{i=1}^N o_i \cdot w_{ij} \quad (4-2)$$

Hierbei stellt o_i den Output des Neurons aus der Schicht X entsprechend Gleichung 4-1 dar. Die Größe w_{ij} bezeichnet die Gewichtung des Outputs des Neurons i zum verbundenen Neuron j . Die Neuroneneingabe ergibt sich also aus der Summe der mit dem Neuron j verbundenen Neuronen multipliziert mit den Verbindungsgewichten. Die Ausgabe eines Neurons der verdeckten Schicht erhält man somit zu:

$$o_j = f(H_j) \quad (4-3)$$

Der Eingang eines Neurons der Ausgabeschicht errechnet sich nach:

$$I_k = \sum_j w_{j,k} \cdot o_j \quad (4-4)$$

Als Ausgabewert eines Neurons der Ausgabeschicht erhält man entsprechend:

$$y_k = o_k = f(l_k) \quad (4-5)$$

Daraus ergibt sich die Darstellung des Netzes zu:

$$y_k = f(l_k) = f\left(\sum_j w_{jk} o_j\right) = f\left(\sum_j w_{jk} f\left(\sum_i w_{ij} o_i\right)\right) = f\left(\sum_j w_{jk} f\left(\sum_i w_{ij} f(x_i)\right)\right) \quad (4-6)$$

Der von Neuronen der Ausgabeschicht berechnete Ausgang stellt die Netzausgabe (y_1, \dots, y_L) und damit das Ergebnis des neuronalen Netzes dar. Die aus den Eingaben über die einzelnen Neuronen berechnete Ausgabe stellt die Arbeitsleistung des neuronalen Netzes dar.

Aus der Eingabe berechnen die Neuronen den Wert ihrer Aktivierungsfunktion, also ob und in welchem Maße es auf die Netzeingabe reagiert. Als Aktivierungsfunktion kann theoretisch jede Funktion gewählt werden. Da jedoch im Lernprozess, auf den im nächsten Kapitel eingegangen wird, die Ableitungen der Aktivierungsfunktion benötigt werden, muss die Aktivierungsfunktion stetig sein. Die verschiedenen Arten von Aktivierungsfunktionen werden z. B. in /Zel97/ vorgestellt. Die sigmoide Aktivierungsfunktion a_j , die in der vorliegenden Arbeit verwendet wird, lautet:

$$a_j = \frac{1}{1 + e^{-H_j}} = f(H_j) \quad (4-7)$$

Die Ausgabefunktion o_j des Neurons j berechnet sich aus der Aktivierungsfunktion und ist oftmals identisch mit a_j :

$$o_j = f(a_j) = f(f(H_j)) = a_j \quad (4-8)$$

Bei speziellen Anwendungen und, um dem biologischen Vorbild nachzukommen, wird mit einem Schwellenwert gearbeitet, der bewirkt, dass ein Neuron erst von einer bestimmten Aktivierung an eine von Null verschiedene Ausgabe hat.

Um die Ausgabe des neuronalen Netzes beeinflussen zu können, stehen verschiedene Möglichkeiten zur Verfügung. Man spricht dabei vom Lernen des Netzes. Die am häufigsten verwendete Lernmethode ist die Modifikation der Verbindungsgewichte des Netzwerkes (also der Größen w_{ij} , w_{jk}).

Die Lernmethoden werden in drei Gruppen unterteilt, das überwachte Lernen, das bestärkende Lernen und das nicht überwachte Lernen. Für das Vorhaben wurde ausschließlich die Methoden des überwachten Lernens genutzt.

Beim überwachten Lernen muss die Ausgabe, die das neuronale Netz bei einem bestimmten Eingabevektor liefern soll, bekannt sein. Es müssen also Informationen darüber vorliegen, wie das Netz auf bestimmte Eingaben reagieren soll. Um die gewünschte Reaktion zu erreichen, werden die Gewichte der Verbindungen modifiziert. Aufgabe des Lernalgorithmus ist es, dass das Netz nach wiederholter Präsentation der Muster, die Assoziation zwischen Eingabemuster und korrekter Ausgabe, selbstständig durchführt.

Ziel des Lernens ist es, dass das neuronale Netz neue unbekannte Eingangsdaten richtig klassifizieren kann. Das Netz muss deshalb in der Lage sein, das gelernte Wissen zu generalisieren. Wie im Folgenden noch gezeigt wird, bestimmt die Netzwerkarchitektur die Arbeitsleistung und damit die Anwendungsmöglichkeit des neuronalen Netzes. Eine Festlegung der Architektur kann also nur an Hand einer konkreten Problemstellung erfolgen. Die benötigte Anzahl an Neuronen und Neuronenverbindungen kann im Voraus nicht bestimmt werden, sondern muss mit Hilfe von Netzwerkspezifikationen ermittelt werden. Hierauf wird noch näher eingegangen.

4.1 Training des Netzes – überwachtes Lernen

Beim überwachten Training eines neuronalen Netzes werden die Gewichte der einzelnen Verbindungen so angepasst, dass die Eingabedaten x_i die gewünschte Ausgabe o_k der Ausgabeschicht möglichst genau approximieren. Es werden daher Trainingsdaten benötigt, für die die Sollausgabe des Netzes bekannt ist. Man spricht hier auch von der Trainingsmenge, wobei ein Eingabesatz ein Muster (t_p) darstellt.

Nachfolgend wird die Backpropagation-Regel hergeleitet, da sie Grundlage der in der entwickelten Methode verwendeten Lernalgorithmen ist. Sie stellt die Verallgemeinerung der Delta-Regel für Netze mit mehreren Ebenen (Schichten). Die Backpropagation-Regel ist das mit Abstand am häufigsten verwendete Lernverfahren. Die meisten anderen Lernverfahren leiten sich aus der Backpropagation-Regel ab oder sind Modifikationen davon. Die grundlegende Idee ist es, eine Fehlerfunktion, die aus der Abweichung von gewünschter und tatsächlicher Netzausgabe besteht, zu minimieren. Die Fehlerfunktion kann dabei auf verschiedene Art und Weise definiert werden, z. B. als absoluten Fehler, als euklidischer Fehler und, wie meist verwendet, als quadratischer Fehler. Der quadratische Fehler für ein Muster, also der Netzeingabe und der zugehörigen Netzausgabe, ergibt sich wie folgt:

$$E_p = \frac{1}{2} \sum_{k=1}^l (t_{pk} - o_{pk})^2 \quad p = 1, \dots, P \quad (4-9)$$

wobei t_{pk} die Lernvorgabe, also die Sollausgabe ist. Da es keinen Unterschied macht, wenn man den halben Fehler minimiert, wird der Faktor $\frac{1}{2}$ hinzugefügt. Dies hat den Vorteil, dass sich der, durch die folgende Ableitung der Fehlerfunktion, gegen den entstehenden Faktor Zwei verkürzt. Für den Gesamtfehler über alle Lernmuster P ergibt sich:

$$E = \sum_{p=1}^P E_p \quad (4-10)$$

Wie aus Gleichungen (4-6) und (4-9) ersichtlich, ist der Fehler eine Funktion der Gewichte \vec{w} . Ziel ist es, die Gewichte \vec{w} zu finden, die die Fehlerfunktion minimal werden lassen, also:

$$E \rightarrow \min, \text{ wenn } \frac{\partial E}{\partial w_{jk}} = 0 \quad (4-11)$$

wobei davon ausgegangen wird, dass die Determinante der zweiten Ableitung gleich Null ist.

Es handelt sich also um ein Gradientenverfahren, das, da das Minimum der Funktion gesucht ist, auch Gradientenabstiegsverfahren genannt wird. Ausgehend von der Netzeingabe des Neurons j für ein Muster p

$$H_{pj} = \sum_{i=1}^N o_{pi} w_{ij} \quad (4-12)$$

erhält man die Aktivierungsfunktion

$$o_j = f_{\text{act}}(H_{pj}) \quad (4-13)$$

Die Änderung der Gewichte erfolgt unter der Berücksichtigung der Lernrate η

$$\Delta w_{jk} = \sum_p -\eta \cdot \frac{\partial E}{\partial w_{jk}} \quad (4-14)$$

Die Lernrate ist ein Maß dafür, in welcher Schrittweite die Änderung der Gewichte stattfindet. Zu große Werte der Lernrate führen dazu, dass die Minima der Fehlerfunktion übersprungen werden können, zu kleine Werte führen zu einem inakzeptablen Anstieg der Rechenzeit. Oftmals wird die Rechnung mit relativ großen Werten begonnen z.B. 0,9 und die Lernrate dann in Richtung 0,1 verringert. Weitere Einflüsse der Lernrate auf das Training werden nachfolgend noch dargestellt.

Die Anwendung der Kettenregel für die Differentiation ergibt,

$$\frac{\partial E_p}{\partial w_{jk}} = \frac{\partial E_p}{\partial H_{pk}} \cdot \frac{\partial H_{pk}}{\partial w_{jk}} \quad (4-15)$$

wobei

$$\frac{\partial H_{pk}}{\partial w_{jk}} = \frac{\partial}{\partial w_{jk}} \cdot \sum_i o_{pj} w_{jk} = o_{pk} \quad (4-16)$$

und sich der erste Term der Gleichung (4-15) als Fehlersignal schreiben lässt

$$\delta_{pk} = -\frac{\partial E_p}{\partial H_{pk}} \quad (4-17)$$

Dies führt beim Einsetzen in Gleichung (4-14) zu:

$$\Delta w_{jk} = \eta \sum_{k=1}^L o_{pj} \delta_{pk} \quad (4-18)$$

Hier ist δ_{pk} :

$$\delta_{pk} = \frac{\partial E_p}{\partial H_{pk}} = -\frac{\partial E_p}{\partial o_{pk}} \cdot \frac{\partial o_{pk}}{\partial H_{pk}} \quad (4-19)$$

Man erhält für den zweiten Term:

$$\frac{\partial o_{pk}}{\partial H_{pk}} = \frac{\partial}{\partial H_{pk}} f_{\text{act}}(H_{pk}) = f'_{\text{act}}(H_{pk}) \quad (4-20)$$

So dass:

$$\Delta_p w_{jk} = \eta o_{pi} \delta_{pk} \quad (4-21)$$

Handelt es sich um ein Neuron der Ausgabeschicht so wird aus dem ersten Term der Gleichung (4-19):

$$-\left(\frac{\partial E_p}{\partial o_{pk}}\right) = (t_{pk} - o_{pk}) \quad (4-22)$$

Ist das Neuron Teil der verdeckten Schicht, so ergibt sich:

$$-\frac{\partial E_p}{\partial o_{pk}} = -\sum_k \frac{\partial E_p}{\partial H_{pk}} \cdot \frac{\partial H_{pk}}{\partial o_{pk}} = \sum_k (\delta_{pk} \cdot \frac{\partial}{\partial o_{pk}} \sum_i o_{pi} w_{ik}) = \sum_k \delta_{pk} w_{jk} \quad (4-23)$$

Wie man sieht, berechnet sich der Gesamtfehler des Neurons j für das Muster p aus den gewichteten Fehlern δ_{pk} aller folgenden Neuronen k und den Gewichten der Verbindungen von j zu k.

Eingesetzt in Gleichung (4-19) ergibt sich für ein Ausgabeneuron j:

$$\delta_{pj} = f'_{act}(H_{pj}) \cdot (t_{pj} - o_{pk}) \quad (4-24)$$

Und im Falle eines Neurons der verdeckten Schicht:

$$\delta_{pj} = f'_{act}(H_{pj}) \cdot \sum_k \delta_{pk} w_{jk} \quad (4-25)$$

Die beiden letzten Gleichungen ermöglichen die rekursive Berechnung von δ_{pj} für alle Neuronen, beginnend mit der Ausgabeschicht rückwärts bis zur Eingabeschicht. Die Schritte werden so lange durchgeführt, bis eine vorgegebene Schranke für den Fehler unterschritten wird.

Das Backpropagation Lernverfahren ist mit einer Reihe von Problemen behaftet, die kurz vorgestellt werden.

Es handelt sich bei dem Verfahren um ein lokales Verfahren, das nur in der Lage ist, ein lokales Minimum zu finden. Dies ist nicht notwendigerweise das globale Minimum. Die Fehlerfläche wird mit zunehmender Komplexität des Netzes immer zerklüfteter, so dass die Wahrscheinlichkeit, dass der Algorithmus in einem lokalen Minimum endet, zunimmt.

Zu Beginn des Trainingsprozesses müssen die Gewichte auf einen von Null abweichenden Wert gesetzt werden. Es kann bewiesen werden, dass das Setzen derselben Startwerte für die Gewichte in jeder Schicht nicht zum Lernerfolg führt /Zel97/, da sich danach keine unterschiedlichen Gewichte herausbilden können. Als gängige Methode werden die Anfangswerte mit Hilfe eines Zufalls-generators gesetzt. Die zufälligen Werte werden im Allgemeinen aus dem Intervall $[-1, 1]$ ausgewählt /Zel97/.

Ein weiteres Problem ergibt sich bei der sigmoiden Aktivierungsfunktion und der Verwendung binärer Eingabedaten. Die Änderung der Gewichte Δw_{ij} ist, wie Gleichung (4-17) zeigt, abhängig von o_i . Im Mittel liefern die Hälfte aller Eingabeneuronen die Ausgabe $o_i=0$, so dass sich die Gewichte für diese Verbindungen nicht ändern. Als Lösung wird in /Zel97/ die Änderung des Eingabebereiches auf $[-\frac{1}{2}; \frac{1}{2}]$ vorgeschlagen und das Hinzufügen eines Schwellenwertes, der den Wertebereich der sigmoiden Aktivierungsfunktion auf $[-\frac{1}{2}; \frac{1}{2}]$ senkt:

$$o_j = \frac{1}{(1 + e^{-H_j})} - \frac{1}{2} \quad (4-26)$$

Alternativ kann auch die Tangens Hyperbolicus als Aktivierungsfunktion verwendet werden.

Um die erwähnten Probleme des Backpropagation Lernverfahrens zu lösen, wurde eine Vielzahl von Modifikationen des Verfahrens entwickelt, die im Abschnitt 4.8 vorgestellt werden.

Das Lernen erfolgt, indem eine Anzahl von Trainingszyklen durchlaufen wird. Dadurch soll der Fehler, den das Netzwerk verursacht, verringert werden. Da es in der Regel nicht möglich ist, die Fehlerfunktion auf Null zu führen, wird eine Fehlerschranke festgelegt, bei deren Erreichen das Training abgebrochen wird. Dies ist notwendig, um bei der rechentechnischen Umsetzung zu lange Rechenzeiten und auch Übertraining (vgl. Kapitel 4.4) zu vermeiden. Entsprechend der Fehlerschranke wird bei der programmtechnischen Umsetzung auch mit einer maximalen Anzahl an Zyklen gearbeitet, deren Erreichen zum Abbruch des Trainings führt, wenn bis dahin die Fehlerschranke nicht unterschritten wird. Auf den Einfluss der Trainingsdauer auf das Lernverhalten wird im Abschnitt 4.3 eingegangen.

4.2 Netzanwendung

Einem erfolgreichen Netztraining, also dem Unterschreiten der festgelegten Fehlerschranke, folgt die Erprobung des trainierten Netzes mit unbekanntem Daten. Damit wird überprüft, ob das Netz in der Lage ist, durch das Training unbekannte Daten richtig zuzuordnen. Man spricht hier von der Generalisierungsfähigkeit des Netzes. Diese stellt eine wichtige Größe für die Bewertung der Arbeitsleistung des Netzes dar. Auf diese und weitere Arten zur Bewertung der Leistung neuronaler Netze wird im Abschnitt 4.9 eingegangen.

Die Generalisierungsfähigkeit eines neuronalen Netzes ist die für die Anwendung relevante Größe. In Kapitel 8 wird gezeigt, dass ein durch das Training minimierter Fehler nicht immer zur optimalen Generalisierungsfähigkeit führt. Damit hat die Steigerung der Generalisierung eine Rückwirkung auf die Netzwerkarchitektur und das Trainingsverfahren.

4.3 Einfluss der Anzahl der Trainingsdurchläufe – Übertraining

Die Anzahl der Durchläufe, in denen der Trainingsdatensatz vom Netz verarbeitet wird, beeinflusst die Genauigkeit, mit der die Lernaufgabe durch das Netz gelöst wird. Voraussetzung für eine Verbesserung der Genauigkeit ist jedoch die Berücksichtigung einer Reihe von Einflussgrößen, so dass nicht automatisch mit steigender Durchlaufanzahl die Lernaufgabe besser gelöst wird. Auf die wesentlichen Größen wird in den folgenden Abschnitten eingegangen.

Ziel des Lernens ist es, die unterschiedlichen Muster der jeweiligen Sollnetzausgabe zuzuordnen. Es werden also Klassenzugehörigkeiten gebildet. Erfolgen zu viele Durchläufe mit der begrenzten Trainingsmenge, so verändern sich die Klassengrenzen, damit sie den einzelnen Trainingsmustern besser entsprechen. Man spricht hier vom „Auswendiglernen“ durch das Netz. Die Verschiebung der Grenzen führt dazu, dass eine Anzahl an nicht im Trainingssatz verwendeten Mustern nicht mehr dieser Klasse zugeordnet wird, wodurch die Generalisierungsfähigkeit des Netzes sinkt /Pat97/.

Im Allgemeinen wird ein Abbruchkriterium für die Anzahl der Trainingsdurchläufe festgelegt. Das Abbruchkriterium ist ein Maß für den Fehler, den man für das Netz beim Lernprozess toleriert. Diese Schranke sollte dabei nicht willkürlich gewählt werden, da zum einen der Lernalgorithmus konvergieren soll und zum anderen aber ein Übertraining vermieden werden muss.

Bildet man den so genannten Generalisierungsfehler, also ein Fehlermaß für die Generalisierungsleistung des Netzes, so zeigt sich, dass die Optima von Generalisierungsfehler und Lernfehler nicht gemeinsam auftreten. Der Generalisierungsfehler durchläuft ein Minimum bei einer bestimmten Anzahl von Trainingsdurchläufen und steigt dann wieder an, wogegen der Lernfehler, Konvergenz vorausgesetzt, weiter sinkt. Hier ist auch der Grund dafür zu sehen, dass sich eine zu niedrige Schranke negativ auf die Generalisierungsleistung des Netzes auswirken kann /Pat97/.

4.4 Anzahl Trainingssätze

Die Anzahl der vorhandenen Trainingssätze ist eine weitere Einflussgröße auf die erforderliche Anzahl an Spielen und die Generalisierungsleistung des Netzes. Anders /And97/ führt dazu aus, dass die verfügbare Trainingsmenge in zwei Teilmengen unterteilt werden muss. Zum einen in die zum Training verwendete und zum anderen in die Testmenge, mit deren Hilfe das trainierte Netz seine Generalisierungsleistung an unbekanntem Daten zeigen muss. Diese Teilung bewirkt jedoch, dass zum Training nicht alle Muster genutzt werden können. Insbesondere vor dem Hintergrund, dass ohnehin zumeist nur geringe Mengen von Eingangsdaten zur Verfügung stehen, ist dies ein Nachteil.

Dieser Informationsverlust führt zu einer Ineffizienz, die dadurch größer wird, dass mehr Validierungssätze zurückgehalten werden /And97/.

Patterson /Pat97/ erwartet, dass sich aus einem guten Lernalgorithmus mit einer hohen Wahrscheinlichkeit eine geringe Fehlerrate für die Klassifizierung unbekannter Daten ergibt. Die Trainingsbeispiele stellen dabei Punkte bzw.

Elemente der gesuchten Zielmenge (Klasse) dar. Das Netz muss nun eine Verallgemeinerung finden, die es ermöglicht, anhand der Trainingsbeispiele eine Abbildung auf der Zielmenge vorzunehmen und die Wahrscheinlichkeit zu minimieren, ein zufällig gewähltes Element falsch zuzuordnen. Hierzu ist eine gewisse Beispielmenge notwendig, die eine Abbildung ermöglicht und dabei eine festzulegende Fehlerwahrscheinlichkeit nicht übersteigt. Man kann daher fordern, dass die Wahrscheinlichkeit, dass der Verallgemeinerungsfehler unterhalb der Schranke bleibt, größer ist als die Wahrscheinlichkeit einer Fehlklassifizierung. Ausgedrückt über eine Ungleichung folgt nach /Pat97/:

$$P(E_g \leq \varepsilon) \geq (1 - \delta) \quad (4-27)$$

mit ε als tolerierbare Fehlerschranke des Verallgemeinerungsfehlers E_g und $(1 - \delta)$ der Wahrscheinlichkeit, dass die Musterklassifizierung höchstens einen Fehler von ε hat.

Patterson /Pat97/ gibt Gleichungen an, die aus Untersuchungen von Baum und Haussler /Bau89/ stammen, mit denen Grenzen für die Mindestanzahl von notwendigen Trainingssätzen bestimmt werden können. Beispielhaft soll hier:

$$P \geq \Omega\left(\frac{W}{\varepsilon}\right) \quad (4-28)$$

genannt sein. Hierbei ist W die Anzahl der Gewichte und ε die Fehlerschranke. Einschränkend ist zu sagen, dass dies nur für Netze mit linearer Aktivierungsfunktion und mit Schwellenwert zutreffend ist. Es wird jedoch die Vermutung geäußert, dass die Ungleichung auch für sigmoide Funktionen zutreffend sei. Nimmt man beispielhaft eine Fehlerschranke von 0,1 an, so ergibt sich für die Mindestanzahl der Trainingssätze das Zehnfache der Anzahl der Gewichtungen. Daraus resultiert bereits für kleine Netze ein immenser Bedarf an Trainingssätzen, was den allgemeinen Erfahrungen widerspricht. Z.B. benutzten Himmelblau et al /Wat89/ 20 Trainingssätze in Netzen mit über 100 Gewichten und erhielten zufrieden stellende Ergebnisse.

Es bleibt festzuhalten, dass keine belastbaren Aussagen zu der notwendigen Anzahl von Trainingsdaten vorhanden sind.

4.5 Einfluss Anzahl der verdeckten Schichten auf Effizienz

Wie beschrieben wurde, haben der Lernprozess, die Anzahl der Trainingsvorgänge und die Anzahl der Datensätze einen Einfluss auf die Leistung neuronaler Netze. Hier wird eine weitere wichtige Eigenschaft neuronaler Netze betrachtet. Da am Anfang ihrer Entwicklung noch kein Lernalgorithmus vorhanden war, der es ermöglichte, die Gewichte der Verbindungen der verdeckten Neuronen anzupassen und auch nicht bekannt war, dass einige Probleme nicht durch Netze ohne verdeckte Schichten (einschichtige) gelöst werden können, rückten die einschichtigen Netze in den Blickpunkt der Untersuchungen. Da das Konvergenztheorem für einschichtige Perzeptrons besagt, dass, wenn es eine Menge von Gewichten gibt, die die Trainingsmuster korrekt klassifizieren, durch den Trainingsalgorithmus diese Menge, in endlichen Iterationsschritten, gefunden werden kann. Daher führte die Erkenntnis, dass eine Vielzahl von Problemen zu keiner Lösung kam, zu einem Stillstand der Entwicklung neuronaler Netze. Das aufgetretene Problem lässt sich sehr anschaulich an der Auswertung des ausschließenden „Oder“ („XOR“)- Gatters zeigen.

Ein einschichtiges Netz mit zwei Eingangs- und einem Ausgangsneuron soll die möglichen Eingaben den korrekten Ausgaben zuordnen. Die Ausgabe soll dabei, entsprechend dem „XOR“- Gatter, Null sein, wenn beide Eingaben gleich und Eins, wenn die Eingaben verschieden sind. Daraus ergibt sich die Netzeingabe des Ausgabeneurons j zu:

$$H_j = o_1 w_{1j} + o_2 w_{2j} \quad (4-29)$$

Verwendet man eine binäre Schwellenwertfunktion als Aktivierungsfunktion des Ausgabeneurons, so ist die Ausgabe des Neurons Null bis der Schwellenwert erreicht ist, sonst Eins. Anschaulich erklären lässt sich das Problem, in dem man versucht, die gegenüberliegenden Ecken eines Rechteckes mit einer Ge-

raden von den anderen Eckpunkten zu trennen. Dies ist offensichtlich nicht möglich. Abbildung 4-2 zeigt das Problem grafisch.

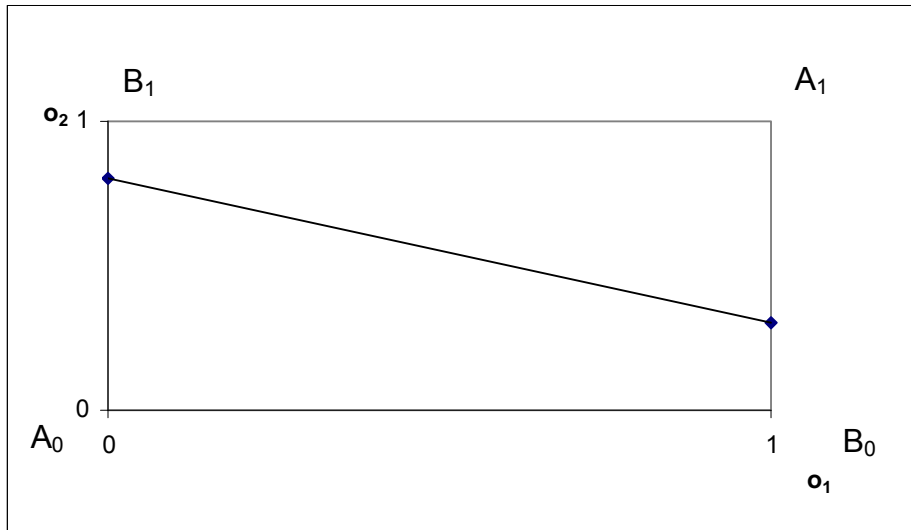


Abbildung 4-2: Lineare Separierbarkeit und das XOR- Problem /Zel97/.

Ist das vorliegende Problem nicht linear separierbar, so benötigt das neuronale Netz verdeckte Schichten. In der Literatur werden bei den Untersuchungen der Eigenschaften zumeist so genannte Multi-Layer-Perceptrons verwendet. Es wird in den meisten Fällen nur ein Ausgabeneuron eingesetzt, so dass sich das vollständige Netzwerk wie folgt schreiben lässt /Pat97/:

$$y_k = f(I_k) = f\left(\sum_j w_{jk} o_j\right) = f\left(\sum_j w_{jk} f\left(\sum_i w_{ij} o_i\right)\right) = f\left(\sum_j w_{jk} f\left(\sum_i w_{ij} f(x_i)\right)\right) \quad (4-30)$$

mit; $o_j = f(H_j)$ für Neuronen der verdeckten Schicht, $o_k = f(I_k)$ für das Neuron der Ausgabeschicht und x_i die Werte der Eingabeneuronen. Die Erweiterung auf drei oder mehr Schichten ist dabei leicht möglich.

Widersprüchlich werden in der Literatur die zulässigen Verbindungen in den Multi-Layer-Perceptrons festgelegt. In /Nau96/ sind Schichten übergreifende Verbindungen zwischen Neuronen nicht zulässig, in /And97/ wird mit Schichten übergreifenden Verbindungen, so genannten „short-cuts“, gearbeitet.

Zweistufige Perzeptrons können konvexe Polygone klassifizieren /Zel97/. Dies bedeutet anschaulich für den zweidimensionalen Fall, dass zwei Geraden eine Lösungsfläche bilden, so dass das „XOR“- Problem gelöst werden kann.


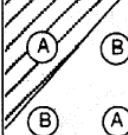



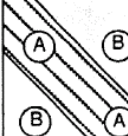
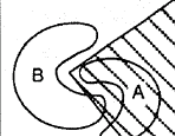


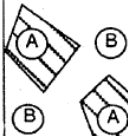
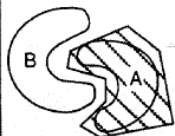
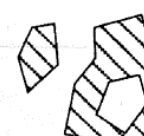
Dreistufige Perzeptrons sind durch Überlagerung und Schnitt konvexer Polygone in der Lage, beliebige Formen zu repräsentieren. Weitere Schichten verdeckter Neuronen besitzen in diesem Modell keine weiteren Fähigkeiten mehr /Zel97/.

In /Pat97/ wird auf die Tatsache hingewiesen, dass ein zweischichtiges vorwärtsgerichtetes Netzwerk in der Lage ist, eine beliebige stetige Abbildung g aus einem n -dimensionalen Würfel $[0,1]^n$ in \mathbb{R}^m vorzunehmen. Hierzu ist ein Netzwerk mit n Eingabe- und m Ausgabeneinheiten vonnöten, wobei in der verborgenen Schicht mindestens $2n+1$ Neuronen erforderlich sind. Diese als Satz von Kolmogorov bekannte Funktion kann wie folgt verallgemeinert dargestellt werden:

$$g(x_1, x_2, \dots, x_n) = \sum_{j=1}^{2n+1} \Psi_j \left(\sum_{i=1}^n \Phi_{ij}(x_i) \right) \quad (4-31)$$

wobei Ψ_j und Φ_{ij} stetige Funktionen der Variablen x_i sind. Die Einsatzmöglichkeit beschränkt sich, da für die Darstellung der stetigen Funktion g eine andere unbekannte Funktion Ψ_j erforderlich ist.

Die Abbildung 4-3 zeigt die verschiedenen Abbildungsmöglichkeiten der Netzstrukturen.

Struktur	Entscheidungsbereichstypen	Exklusiv-ODER-Problem	Klassen mit ineinander verflochtenen Bereichen	Allgemeine Bereichsumrisse
Eine Schicht 	Halbebene, begrenzt durch Hyperebene			
Zwei Schichten 	Konvex geöffnete oder geschlossene Bereiche			
Drei Schichten 	Zufällig (Komplexität nur begrenzt durch die Anzahl der Knoten)			

Entscheidungsbereiche, die durch einfache und MLFF-Netzwerke gebildet werden können.

Die Entscheidungsbereiche werden durch die schattierten Bereiche A und B dargestellt.

Abbildung 4-3: Beispiele für Abbildungsfähigkeiten vorwärtsgerichteter Netzwerke /Pat97/

Ein Nachteil von Netzen mit mehreren Schichten ist, dass die Gewichtsänderungen bei Backpropagation von der Ausgangsschicht zur Eingangsschicht immer kleinere Werte annehmen, woraus sich sehr lange Lerndauern ergeben /Zel97/. Um dies zu vermeiden, wurden z.B. Trainingsverfahren wie Backpercolation, das in Abschnitt 8.4 noch erläutert wird, entwickelt /Jur91/.

Wie gezeigt wurde, definiert die Anzahl der Schichten eines neuronalen Netzes die Funktionen, die durch das Netz abgebildet werden können. Welche Anzahl von Neuronen für die jeweilige Schicht notwendig ist, soll im Folgenden betrachtet werden.

4.6 Einfluss Neuronenanzahl

In der Literatur gibt es verschiedene Empfehlungen für die Anzahl der Neuronen in den verdeckten Schichten. Diese stammen jedoch aus empirischen Untersuchungen und sind somit stets an das untersuchte Problem gekoppelt. Im Allgemeinen wird davon ausgegangen, dass mit steigender Komplexität der abzubildenden Funktion auch die notwendige Anzahl der Neuronen steigt /Nau96/. Es existieren derzeit keine Regeln mit denen sich bei praktischen Problemen die Anzahl der verdeckten Neuronen pro Schicht eindeutig bestimmen lässt.

Bei Versuchsreihen zur Bestimmung der optimalen Anzahl der Neuronen in den verdeckten Schichten ist unter Berücksichtigung der Anmerkungen der Abschnitte 4.3 und 4.4 darauf zu achten, dass Wechselbeziehungen zwischen der Anzahl der Trainingssätze, der Zahl der Trainingsdurchläufe und der Neuronenanzahl bestehen.

4.7 Lernrate

In der Literatur werden zum Teil recht widersprüchliche Angaben zur Wahl der Lernrate gemacht. Die Lernrate hat einen direkten Einfluss auf den für das Training benötigten Zeitaufwand; sie sollte also nicht zu klein sein. Bei einer zu großen Lernrate können jedoch Minima der Fehlerfläche übersprungen werden oder es kann zu Oszillationen in steilen Schluchten der Fehlerfläche kommen. Zu kleine Werte können z.B. auf Plateaus der Fehlerfunktion zu extrem langsamen Lernzeiten führen.

Zell /Zel97/ stellte in eigenen Versuchen fest, dass Lernraten größer als 1 zu optimalen Trainingszeiten führten. Dies lässt sich jedoch nicht verallgemeinern. Wie bereits erwähnt, wird oft mit einer variablen Lernrate gearbeitet, die mit relativ großen Werten (>1) beginnt, die dann reduziert werden.

Auch hier bleibt festzuhalten, dass durch Versuchsreihen problemangepasste Lösungen gefunden werden müssen, und es keine allgemeingültige Vorgehensweise gibt.

Eine weitere Möglichkeit besteht darin, die Lernrate nach jedem Iterationsschritt für jedes Gewicht neu festzulegen. Erfolgte die Gewichtsänderung im Vorgängerzyklus unterhalb einer festgelegten Schranke, so wird die Lernrate erhöht, andernfalls belassen oder verringert.

4.8 Einfluss der Wahl des Lernverfahrens

Neben dem Backpropagation Verfahren, das ausführlich im Abschnitt 4.1 vorgestellt wurde, sind eine Vielzahl weiterer Verfahren entwickelt wurden. An dieser Stelle wird auf die Verfahren „Quickprop“ und „Backpercolation“ eingegangen, da sie bei den programmtechnischen Anwendungen der Arbeit auftauchen, und dem Leser so ein leichter Einblick in die besprochenen Probleme gegeben wird.

Das Quickprop Verfahren wurde als schnellere Alternative zum Backpropagation Verfahren entwickelt /Fah89/. Es stellt ein iteratives Verfahren zweiter Ordnung zur Bestimmung des Minimums der Fehlerfunktion dar, das sich an das Newton-Verfahren anlehnt. Die Fehlerfunktion wird dabei als eine nach oben geöffnete lokal quadratische Funktion angesehen, und ihr Minimum wird gesucht. Die Änderung der Gewichte erfolgt dabei nach folgender Regel

$$\Delta w_{ij}(t) = G(t) + P(t)$$

wobei $G(t)$ und $P(t)$ sich wie folgt berechnen

$$G(t) \begin{cases} \eta \sum_P o_{pi} \delta_{pj} - dw_{ij}(t-1) & \text{falls } (t=0) \vee (\Delta w_{ij}(t-1) = 0) \\ & \text{oder } \text{sgn}(S(t-1)) = \text{sgn}(S(t)) \\ 0 & \text{falls } \text{sgn}(S(t-1)) \neq \text{sgn}(S(t)) \end{cases} \quad (4-32)$$

$$P(t) \begin{cases} = 0 & \text{falls } (t = 0) \vee (\Delta w_{ij}(t-1) = 0) \\ \frac{S(t)}{S(t-1) - S(t)} \cdot \Delta w_{ij}(t-1) & \text{falls } \left| \frac{S(t)}{S(t-1) - S(t)} \right| \leq \mu \\ \mu \cdot \Delta w_{ij}(t-1) & \text{falls } \left| \frac{S(t)}{S(t-1) - S(t)} \right| > \mu \end{cases} \quad (4-33)$$

Hier ist μ der maximale Wachstumsfaktor, der die mögliche Gewichtsänderung begrenzt und im Intervall zwischen 1,75 und 2,25 liegen soll /Zel97/.

Quickprop zeigte in vielen Versuchen ein deutlich schnelleres Lernverhalten als Backpropagation, ist jedoch auch mit einigen Problemen wie z. B. dem „flache Plateau“ Problem behaftet.

Ein Problem der Backpropagation ist, dass bei vielschichtigen Netzen die Gewichtsänderungen der tieferen Ebenen nur sehr langsam ablaufen. Dies kann anschaulich dadurch begründet werden, dass von Schicht zu Schicht die treibende Kraft, also die Fehlerabweichung δ , immer geringer wird. Beim Verfahren Backpercolation wird für jedes Neuron ein eigener Aktivierungsfehler und damit eine eigene Fehlerfläche berechnet und minimiert /Jur91/. Hierzu wird das Trainingsmuster an die Eingabeneuronen angelegt und durch das Netz propagiert. Anschließend wird der globale Fehler des Netzes berechnet und der Gradient des Aktivierungsfehlers rückwärts zu allen Schichten verdeckter Neuronen mit Ausnahme der ersten verdeckten Schicht zurückpropagiert. Der Aktivierungsfehler:

$$\Delta a_j = a_j(t+1) - a_j(t) \quad (4-34)$$

mit:

$$a_j = \sum_i w_{ij} o_i \quad (4-35)$$

wird zu allen verdeckten Schichten zurückgerechnet, und entsprechend werden die Gewichte verändert.

Um die Aktivierung eines Neurons zu ändern, können die Gewichte, die von den Vorgängerneuronen eingehen oder deren Ausgaben modifiziert werden. Sind diese erwünschten Änderungen an jede relevante Zelle des Netzes geleitet worden, so hat jedes Gewicht jeder Zelle eine eigene Fehlerfläche, und das Neuron kann seine Gewichte nach dieser Fehlerfläche minimieren [Zel97]. Die Änderung des als bekannt vorausgesetzten Aktivierungsfehlers kann demnach auf die zwei folgenden Weisen erfolgen:

$$\Delta w_{ij} = -\eta_1 \frac{\partial}{\partial w_{ij}} \Delta a_j \quad (4-36)$$

und

$$\Delta o_i = -\eta_2 \frac{\partial}{\partial o_i} \Delta a_j \quad (4-37)$$

Backpercolation stellt ein Verfahren dar, das in vielen Fällen eine wesentlich höhere Lerngeschwindigkeit als Backpropagation aufweist. Da bei der vorliegenden Untersuchung Netze mit nur einer verdeckten Schicht verwendet werden und die Verbesserung der Lerngeschwindigkeit kein vorrangiges Ziel ist, wird auf eine vertiefende Betrachtung verzichtet.

Zusammenfassend stellen die unterschiedlichen Lernverfahren verschiedene Wege dar, um den Ausgabefehler eines neuronalen Netzes zu minimieren. Wie beschrieben wurde, stellt das Minimum der Fehlerfunktion nicht das Optimum der Generalisierungsleistung dar, so dass es sich bei den Arbeitsschritten, „Aufbau Netzwerk“, „Training und Bestimmung der Generalisierungsleistung“ um ein iteratives Verfahren handelt. Um die Netzleistung zu bewerten, werden unterschiedlich Gütemaße angewandt, die nachfolgend vorgestellt werden.

4.9 Gütemaße

Zur Bestimmung der Leistungsfähigkeit wird eine Leistungsmessung des neuronalen Netzes durchgeführt. Dabei ist zu unterscheiden, ob auf der algorithmi-

schen Ebene, der Implementierungsebene oder der Hardwareebene verglichen werden soll /Zel97/. Diese Unterteilung, die sich in der Informatik bewährt hat, kann auch im vorliegenden Zusammenhang dazu dienen, die Vergleichsverfahren zu unterscheiden. Es sollte jedoch beachtet werden, dass die Ebenen nicht unabhängig voneinander sind.

Um die Leistungsfähigkeit bestimmen zu können, wurde eine Vielzahl von Maßeinheiten entwickelt, die die Vergleichbarkeit ermöglichen sollen. Zell /Zel97/ gibt eine gute Übersicht.

In dem in der Arbeit betrachteten Zusammenhang ist die Leistungsbewertung des Lernverfahrens maßgebend. Daher wird auf die relevanten Leistungskenngrößen eingegangen.

Zyklen und Epochen sind Maßeinheiten für Lernverfahren. Als Zyklus wird dabei die Präsentation eines Trainingsmusters bezeichnet, als Epoche die Präsentation aller Trainingsmuster. Der Vergleich der Anzahl der Zyklen / Epochen, die benötigt werden, um das Konvergenzkriterium zu erfüllen, gibt Aufschluss darüber, welche Lernvariante schneller trainiert.

Die Konvergenzwahrscheinlichkeit bietet eine andere Vergleichsmöglichkeit. Bei einer Vielzahl von Problemen kann nicht davon ausgegangen werden, dass ein Lernverfahren, wenn es in einem Fall konvergierte, auch in allen anderen Fällen erfolgreich ist. So kann die Eignung des Lernverfahrens für das zu bearbeitende Problem ermittelt werden.

Die Generalisierungsleistung ist das wohl wichtigste Vergleichskriterium neuronaler Netze. Die Trainingsprozedur ist zwar meist der aufwendigste Teil bei der Implementierung neuronaler Netze, jedoch ist die Fähigkeit, unbekannte Daten korrekt zu klassifizieren, das eigentliche Ziel des neuronalen Netzes.

Die einfachste Variante ist die Bestimmung der Trefferquote bei der Präsentation unbekannter Daten. Im Allgemeinen wird der vorhandene Trainingsdatensatz in die eigentliche Trainingsmenge und die Validierungsmenge aufgeteilt.

Da die Validierungsmenge meist recht klein ist, sind für die Trefferquoten keine sicheren und belastbaren Daten zu erwarten. Können die Fehler simuliert werden, so sind durch Monte-Carlo Simulationen belastbare Ergebnisse erzielbar.

Eine weitere Möglichkeit besteht darin, die Wahrscheinlichkeit einer korrekten Klassifizierung zu bestimmen. Hierbei soll $P_0(g)$ die Wahrscheinlichkeit dafür sein, dass das Netzwerk ein zufällig ausgewähltes Muster \mathbf{x} aus der unbekannt-ten Verteilung des Prozesses $\pi(x)$ korrekt klassifiziert /Pat97/:

$$P_0(g) = \frac{Z(g)}{Z_0} \quad (4-38)$$

wobei $Z_0 = \int \rho(W) dW$ das effektive Gesamtvolumen darstellt, das sich aus der Dichtefunktion $\rho(W)$ des Gewichtsraumes W ergibt. $Z(g)$ stellt das Volumen dar, für das das Netzwerk eine Zielfunktion abbilden kann. Mit anderen Worten, die Konstellation der Gewichte, für die die Zielfunktion dargestellt werden kann. Während des Trainings wird das effektive Gesamtvolumen Z_0 reduziert, so dass die Wahrscheinlichkeit der korrekten Klassifizierung erhöht wird:

$$P_k(g) = \frac{Z(g)}{Z_k} \quad \text{und} \quad Z_k < Z_0 \quad (4-39)$$

wobei Z_k das Restvolumen ist, das nach dem k -ten erfolgreichen Training verbleibt /Pat97/.

Aus der Informationstheorie kann die Shannon-Entropie als Vergleichsmaß herangezogen werden. Sie ist definiert als:

$$I_S = -\sum_{i=1}^M p_i \log_2 p_i \quad (4-40)$$

wobei p_i die relative Häufigkeit der korrekten Fehlerklassifizierung darstellt /Wolk90/.

Definiert man den Wert einer Information als das Verhältnis der Wahrscheinlichkeiten ein Ziel zu erreichen bevor und nachdem man eine Information erhalten hat, so erhält man /Wol90/:

$$V = \log_2 \left(\frac{p'}{p} \right) \quad (4-41)$$

mit p als der Wahrscheinlichkeit das Ziel vor und p' der Wahrscheinlichkeit das Ziel nach Erhalt der Information zu erreichen.

Aus den in der Literatur ausführlich beschriebenen, hier vorgestellten Methoden zur Festlegung und Optimierung von Netzwerkstrukturen, wurden einige ausgewählte im Rahmen der Arbeit empirisch untersucht (vgl. Abschnitt 8).

5 Übersicht über bisherige Arbeiten zur Fehlerfrüherkennung

Anwendungsbeispiele für neuronale Netze in der Fehlerfrüherkennung gibt es, gemessen an der Gesamtanzahl der Anwendungen, nur wenige. Oft werden neuronale Netze dazu eingesetzt, einen Fehler zu diagnostizieren oder den Zustand eines Prozesses zu beurteilen.

In /Kra00/ wurde ein neuronales Netz dazu verwendet, über die Brauchbarkeit eines geschweißten Bauteils zu entscheiden. Hierzu wurden sechs Fehler identifiziert, die zur Unbrauchbarkeit des Bauteils beitragen können. Für jeden der Fehler wurde ein separates Netz erstellt. Dies ist zum einen von Vorteil, da das Netz dann nur den einen Fehler und den Normalzustand erlernen muss und zum anderen als Eingänge nur Größen vorgesehen werden; die für den betrachteten Fehler relevant sind. Wie im Kapitel 7 noch gezeigt wird, führen Netzeingänge mit nicht relevanten Größen zu falsche Klassifizierungen. Die Ausgänge der sechs Einzelnetze bilden die Eingänge eines übergeordneten Netzes, das auf Grund von Expertenwissen trainiert wurde und aus den Einzel Fehlern auf den Zustand des Bauteils schließt. Besonders hervorgehoben wurde die Notwendigkeit der Datenvorverarbeitung, um geeignete Eingangsgrößen für die neuronalen Netze zu erhalten. Hier sind die durch Transformation der Messsignale erreichte Rauschunterdrückung und die Verstärkung der Prägnanz der Fehlercharakteristik zu nennen.

Die Verwendung dynamischer neuronaler Netze zur Fehlerdetektion und Fehlerisolierung wurde von Patan et al. /Pat05/ am Beispiel eines Prozesses zur Zuckerherstellung untersucht. Dabei werden dynamische Strukturen durch rekurrente Netzarchitekturen erreicht. In /Pat05/ wird ein vorwärtsgerichtetes Netz durch die Wahl des Neuronenmodells dynamisiert. Dazu wird ein lineares dynamisches System eingeführt – der „infinite impulse response“ Filter. Dabei reproduziert jedes dynamische Neuron sein letztes Signal. Der Ausgang des Neurons lautet wie folgt:

$$y(k) = F(g \cdot \tilde{y}(k) + c) \quad (5-1)$$

wobei

$$\tilde{y}(k) = \sum_{i=1}^n (b_i x(k-i)) - \sum_{i=1}^n a_i \tilde{y}(k-i)$$

Somit erhält das Neuron eine Information aus der „Vergangenheit“ und kann diese bei der Berechnung berücksichtigen.

Das neuronale Netz wurde am Beispiel eines Prozesses zur Herstellung von Zucker getestet. Wegen Fehlens realer Fehlerdaten mussten die Fehler mit einem mathematischen Modell simuliert werden. Es erfolgte eine Datenvorverarbeitung, bei der die Prozessvariablen skaliert wurden. In der Arbeit wurden zwei Gruppen von Experimenten durchgeführt. Zum einen wurden neuronale Netze dazu verwendet, Sensorfehler zu entdecken und zum anderen dazu, Fehler im Prozess zu entdecken und zu isolieren /Pat05/. Das Training erfolgte hierbei durch stochastische Näherungsmethoden, die als Alternative mit kürzeren Rechenzeiten zu den Gradientenabstiegsverfahren eingestuft werden. Die besondere Eignung des verwendeten Verfahrens für diesen Fall wurde mit dem Vorhandensein von Störsignalen begründet.

Power et al. /Pow04/ untersuchten einen Prozess zur Aluminiumgewinnung aus Bauxit mit Hilfe von Petri-Netzen. Diese repräsentieren ein dynamisches Modell des Systems. Als Netzwerkarchitektur wurden Radiale-Basisfunktionen-Netze (RBF Netze) gewählt. Die Besonderheit besteht in der Aktivierungsfunktion der Neuronen, die die Basisfunktionen eines Funktionssystems zur Approximation mehrdimensionaler Funktionen anhand von Werten an einer Reihe von Stützstellen sind /Zel97/. Da es sich bei den RBF-Netzen um Netze mit nur einer Schicht handelt, sind die Gewichte direkt (nichtiterativ) berechenbar.

Das untersuchte technische System wurde in zwei Teilbereiche gegliedert, für die jeweils drei Fehler zu identifizieren waren. Es wurde dabei ebenfalls mit simulierten Fehlern gearbeitet, die aus Abweichungen des modellierten Normalbetriebes gewonnen wurden. Als Ergebnis wurde die Vermutung aufgestellt, dass mit kleineren Netzen für einzelne Teilbereiche der Anlage eine akkuratere

Fehlerdiagnose möglich ist als mit einem einzigen großen Netz für die gesamte Anlage.

Eine Untersuchung sich anbahnender Fehlerzustände führte Himmelblau et al. in /Wat89/ durch. Es wurde der chemische Prozess zur Gewinnung von Toluol aus Heptan dynamisch modelliert. Dabei wurden fünf Fehler ausgewählt, die mit Hilfe dreier Messgrößen erkannt werden sollen. Die Fehlerursachen sind dabei entweder nicht messbare bzw. nicht gemessene Prozessgrößen. Die Änderung dieser Prozessgrößen führen zu Änderungen der gemessenen Werte.

Die simulierten Fehler wurden in vier unterschiedlichen Ausprägungsgraden verwendet. Das Training des Netzes wurde mit dem Backpropagation Verfahren durchgeführt. Als Abbruchkriterium wurde eine Schranke von 10% der Differenz aus Lerneingabe und Netzantwort gewählt.

Das Training wurde rauschfrei und mit simuliertem Messrauschen durchgeführt. Das Messrauschen wurde über eine festgelegte Standardabweichung der Signale nach einer Normalverteilung simuliert. In einem zweiten Schritt erfolgte die simultane Bestimmung der Fehlerart sowie der Schwere der Abweichung. Hierzu wurden zwei Netze gekoppelt, wobei das erste Netz die Fehlerursache identifiziert und das zweite den Grad der Abweichung.

Die Arbeit von Himmelblau et al. hat gezeigt, dass die Fehlerfrüherkennung in technischen Systemen durch neuronale Netze im Prinzip realisierbar ist. Aufgrund des Vorhandenseins eines Prozessmodells konnten die für die Fehlererkennung notwendigen Messgrößen eindeutig identifiziert und so das Netz ohne überflüssige Eingangswerte aufgebaut werden. Durch die Verwendung von Daten aus einem Modell ist ein Rückschluss zur Übertragbarkeit auf Messdaten aus realen Anlagen allerdings nicht möglich..

Attieh et al. /Att96/ führten Rechnungen zur Mustererkennung von Transienten in Kernkraftwerken durch. Ziel war es, dem Kraftwerksoperator ein Hilfsmittel in die Hand zu geben, das es ihm ermöglicht, die Ursache einer Störung zu identifizieren. Schwerpunkt der Arbeit war es, die Fehler nicht nur im stationä-

ren Betrieb zu erkennen, sondern auch im Teillastbetrieb. Es wurde für jeden identifizierten Fehler ein separates Netz erstellt, das mit Datensätzen zu allen Fehlern und zum Normalzustand trainiert wurde.

Aus der Arbeit gehen allerdings weder die Art der verwendeten Netzwerkarchitektur hervor noch das benutzte Trainingsverfahren und seine Randbedingungen. Die gezeigten Ergebnisse sind viel versprechend; es wurden alle Fehler erfolgreich trainiert und die Validierung war erfolgreich. Lediglich die Unterscheidung zweier Fehler konnte nicht erfolgreich durchgeführt werden, da die zu Grunde liegenden Messwertverläufe zu ähnlich waren. Dem wurde abgeholfen, indem die ähnlichen Fehler als ein und derselbe Fehler gewertet wurden.

Das entwickelte System wurde bislang nicht mit Daten unterschiedlicher Leistungen und Grade der Fehler getestet.

Die Arbeit von Liangyu Ma et al. /Lia06/ befasst sich u.a. mit der Vorverarbeitung von Messdaten. Auch hier werden der Prozess und die auftretenden Fehler durch ein Prozessmodell simuliert. Als Netzwerkstruktur wird ein Radiale-Basisfunktionen Netz eingesetzt. Wie bereits bei Power wird in der Arbeit darauf hingewiesen, dass es ein prinzipieller Vorteil dieser Netze ist, dass durch den einfachen Aufbau mit nur einer Schicht eine direkte Berechnung der Netzwerkgewichte möglich ist. Die Änderungen der Parameter werden über eine Fuzzyfizierung der Größen realisiert. Hier ist ähnlich den im Verlauf der Arbeit gezeigten eigenen Überlegungen, eine Festlegung spezieller Werte notwendig.

In der Arbeit von M. Chow und S, Yee /Cho90/ wurde mittels unterschiedlicher Netzwerkarchitekturen eine Realtime Anwendung für die Fehlerfrüherkennung entwickelt. Es wurden mit Hilfe einer simulierten Fehlerbasis verschiedene Netzwerke mit und ohne Berücksichtigung von Messrauschen trainiert. Die Aussage, dass die Genauigkeit mit steigender Anzahl Neuronen der verdeckten Schicht steigt, widerspricht eigenen Untersuchungen (vgl. Kapitel 8). Die Verwendung von Messwerten und berechneten Größen als Eingangsgrößen wird ungenügend diskutiert. Die erwähnte Realtime Erprobung wird nicht beschrie-

ben; es werden ausschließlich die Ergebnisse unterschiedlicher Trainingsarten verglichen.

Rengaswamy et al. /Ren00/ entwickelten eine Vorgehensweise, ein Nachtraining durch Integrierung eines Bayes Klassifizierers vorzunehmen. Wesentliche Punkte der Arbeit sind ein schnelles Training, die Berücksichtigung des zeitlichen Verlaufes im Klassifizierer sowie die Möglichkeit des „online“ Nachtrainings. Der Lösungsansatz wird mittels eines Bayes-Klassifizierers entwickelt. Somit ergibt sich die Wahrscheinlichkeit einer Klassenzugehörigkeit aus dem normierten Produkt von a-priori Wahrscheinlichkeit und der Wahrscheinlichkeitsdichtefunktion für die entsprechende Klasse.

Die vorgestellten Arbeiten zeigen, dass die Anwendung neuronaler Netze in der Fehlererkennung ein Erfolg versprechender Ansatz ist. Es wurde jedoch deutlich, dass die Beschaffung realer Fehlerdaten stets ein Problem darstellt, und man die benötigten Trainingsdaten durch die Simulation von Fehlern erzeugen musste. Diese Simulation erfolgte meist aus der Abweichung vom Normalzustand; es wurden keine individuellen Fehlermodelle entwickelt. Hier ist die Schwierigkeit beim Übertragen auf reale Situationen zu sehen, da sich die Abweichungen der gemessenen Größen im realen System in der Regel anders darstellen und die Simulationen komplexer Systeme im Allgemeinen keinen gangbaren Weg darstellt.

Deshalb wurde im Rahmen der vorliegenden Arbeit durch Anwendung realer Fehlerdaten die Brücke zwischen Machbarkeitsstudie der Fehlerfrüherkennung und realem Einsatz zur Unterstützung der Kraftwerksoperateure geschlagen.

6 Entwicklung des Verfahrens zur Fehlerfrüherkennung

Wie im voran stehenden Kapitel dargestellt, wurden bislang eine Reihe technischer Systemen mittels neuronaler Netze bewertet. Dies wurde jedoch stets für Einzelfälle durchgeführt. Der Nachweis fehlte, dass die Untersuchungsschritte übertragbar und strukturiert, ähnlich der Vorgehensweise bei Sicherheitsanalysen, gestaltbar sind. Im Folgenden wird eine übertragbare Vorgehensweise erarbeitet. Dazu werden nachfolgend die einzelnen Arbeitsschritte und die Umsetzung der Methode in ein Rechnerprogramm vorgestellt.

Ziel ist es, eine möglichst allgemeine Vorgehensweise zu entwickeln, die es erlaubt, eine Vielzahl von unterschiedlichen Anwendungsfeldern abzudecken. Hierzu ist es notwendig, die einzelnen Entwicklungsschritte klar zu trennen, um jeweils die speziellen Anforderungen aus den Anwendungen berücksichtigen zu können. Daher ist die Methode modular aufgebaut. Es wird, ausgehend von der technischen Systemanalyse, der Systemzustand anhand der Systemmerkmale (gemessene Parameter) beschrieben. Dazu werden durch Gegenüberstellung von Gut- und Fehlerzuständen die zur Ermittlung der Fehlerzustände notwendigen Systemmerkmale extrahiert. Anschließend werden die Messwerte aufgearbeitet, um die Trainingsdatensätze zusammenstellen zu können. Danach werden die Systemzustände durch Eingabe der sie charakterisierenden Daten in das neuronale Netz klassifiziert.

Die Methode kann ihre Anwendung durch Nutzung der entsprechenden Messstellenparameter im Betrieb finden. Dabei werden Messdaten aus dem Prozessleitsystem (PLS) verwendet, um eine Zustandsüberwachung des Systems durchzuführen und sich anbahnende Fehler dem Bedienpersonal anzuzeigen.

Nachfolgend werden die einzelnen Schritte ausführlich dargestellt.

6.1 Technische Systemanalyse

Ausgangspunkt der Methode ist die technische Systemanalyse. Diese gliedert sich in die Schritte:

- Systembeschreibung
- Fehleridentifikation
- Beschreibung der Systemzustände „Gutzustand“ – „Fehlerzustand“
- Ermittlung der notwendigen Messparameter

Auf der Grundlage von Informationen über die Anlage wird die Abgrenzung des für die Entdeckung einzubeziehenden Anlagenbereichs sowie das Verständnis der verfahrenstechnischen Gegebenheiten erarbeitet. Dazu werden Informationen aus

- R+I Fließbildern
- Anlagenbeschreibungen
- Betriebsbedingungen und –abläufen

ausgewertet.

Die Fehleridentifikation erfolgt an Hand der Messwertverläufe der Parameter bei auftretendem Fehler sowie der Beschreibungen und Erfahrungen des Anlagenpersonals. Darüber hinaus müssen die unterschiedlichen Betriebsfahrweisen berücksichtigt werden.

Die Beschreibung der Systemzustände basiert auf dem Gutzustand und der konsequenten Abfrage aller Auswirkungen des Fehlers auf das System. Hierdurch erhält man eine Übersicht über die Anlagenteile, die vom jeweiligen Fehler betroffen sind. Daraus lassen sich die Messparameter ableiten, die vom Fehler beeinflusst werden. Angelehnt an die Ausfalleffektanalyse /FME06/ werden unter Berücksichtigung der Ausfallart die Auswirkungen auf andere Systemkomponenten ermittelt. Darauf aufbauend werden analog zur PAAG Analyse /PAA90/ die Auswirkungen auf das System untersucht. Vorteilhaft ist hierbei,

dass bei konsequenter Durchführung auch andere Ursachen für Abweichungen erkannt werden und so mehrdeutige Zustände identifiziert werden. Dies ist notwendig, um falsche Zuordnungen zu vermeiden.

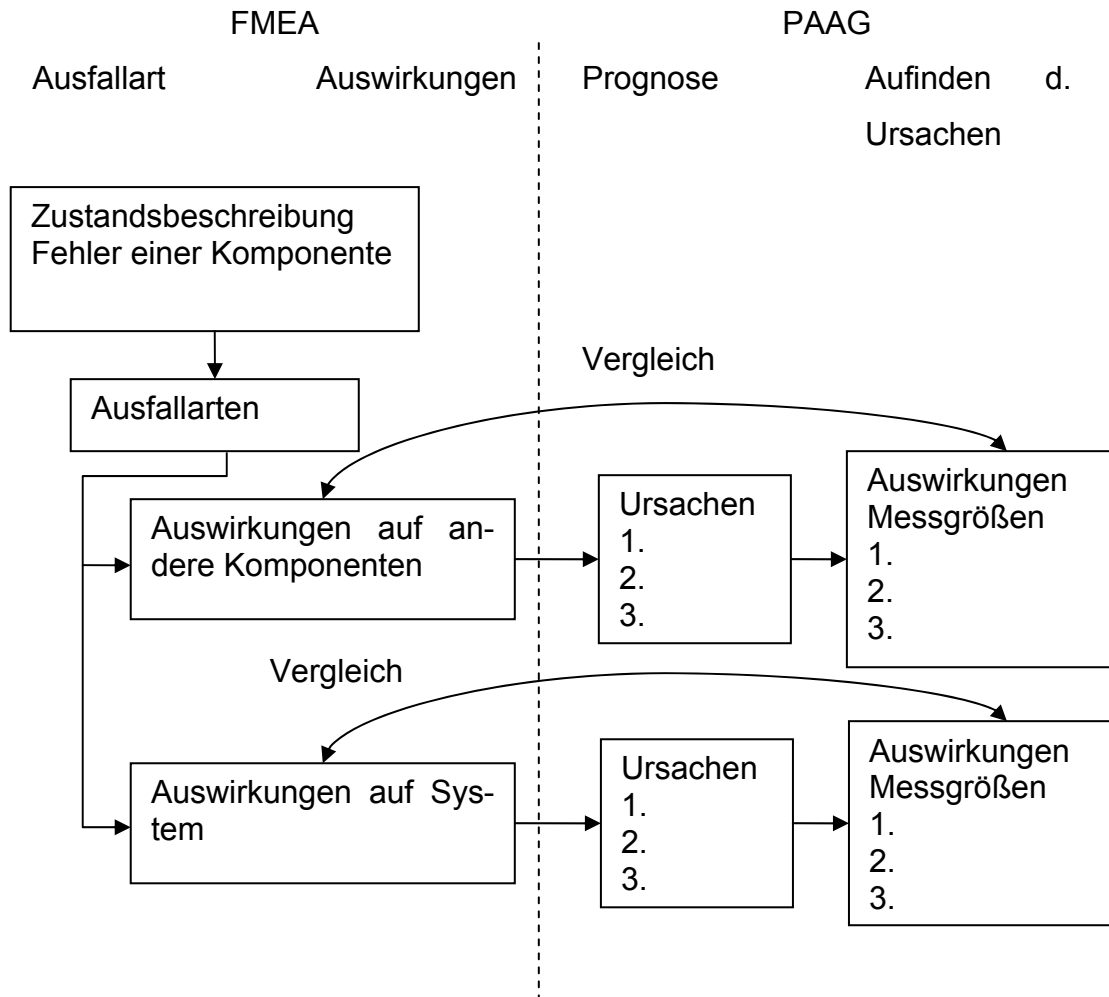


Abbildung 6-1: Messgrößenidentifizierung durch FMEA und PAAG

Abbildung 6-1 zeigt schematisch die entsprechende Vorgehensweise. Durch den Vergleich wird erreicht, dass alle für den jeweiligen Fehler relevanten Messgrößen identifiziert werden und gleichzeitig Zustände erkannt werden, die sich in denselben Messparametern auswirken.

Bei der Bewertung des Systemzustandes und der anschließenden Identifizierung der notwendigen Messgrößen muss berücksichtigt werden, ob es sich um ein geregeltes oder ungeregeltes System handelt. Dies wird nachfolgend behandelt.

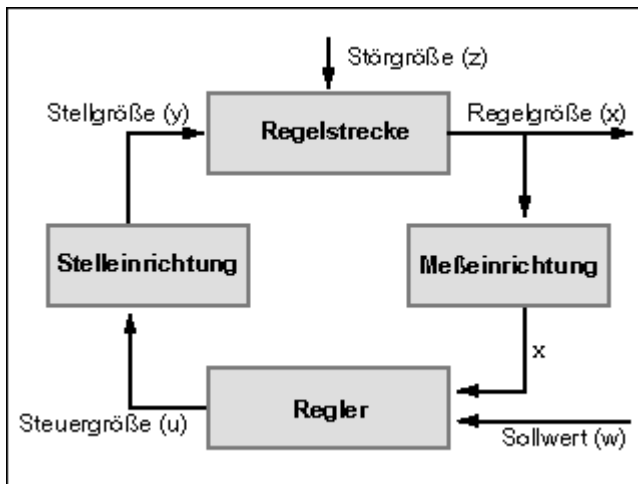


Abbildung 6-2: Regelkreis

Abbildung 6-2 zeigt einen Regelkreis, der die Regelgröße x über die Stellgröße y auf dem Sollwert w halten soll. Abweichungen vom Sollwert sind Folge der Störgröße z .

Tritt ein Fehler im geregelten System auf, so wird, so lange die Regelstrecke es ermöglicht, der Fehler nicht durch eine Veränderung der Regelgröße x erkennbar sein sondern durch die Veränderungen der Stellgröße y , die einen permanenten Impuls auf die Regelstrecke aufgibt.

In einem unregelmäßigem System wird die Störgröße z direkte Auswirkung auf die Größe x , nun keine Regelgröße mehr, haben. Der Fehler wird sich dementsprechend über eine Veränderung der Größe x manifestieren.

Um den Einfluss auf die Methode der Fehlerfrüherkennung zu zeigen, soll ein Beispiel die unterschiedlichen Auswirkungen zeigen.

Ein unregelmäßiger Wärmeübertrager soll untersucht werden. Als Fehler, der zum Ausfall der Einheit führt, wird die Verschmutzung der Wärmeübertragerflächen angenommen, ein häufiges Problem bei Wärmeübertragern. Da im unregelmäßigem System die Eingangsparameter konstant bleiben, werden sich entsprechend die Ausgangsgrößen verändern. Aus dem Fehler „unzulässige Verschmutzung“ der Wärmeübertragerflächen resultiert ein geringerer Wärmetransport zwischen

den beiden Fluiden. Das Wärme abgebende Fluid wird eine erhöhte Austrittstemperatur aufweisen und das aufnehmende Medium wird weniger stark erhitzt werden. Nimmt man betrieblich bedingt veränderliche Eingangsparameter an, so lässt sich der Fehler aufgrund der sich ändernden Verhältnisse der Ein- und Ausgangstemperaturen identifizieren.

Der gleiche Wärmeübertrager wird nun als geregeltes System betrachtet. Die Regelgröße x ist hierbei sinnvoller Weise entweder die Rücklaufstemperatur des Wärme abgebenden Fluids, im Falle eines Kühlkreises, oder die Vorlaufstemperatur des Wärme aufnehmenden Fluids, wenn es sich um eine Heizeinheit handelt. Hier soll es sich um eine Kühleinrichtung handeln, so dass die Regelgröße durch die Rücklaufstemperatur festgelegt ist. Tritt nun die Störgröße „Verschmutzung des Wärmeübertragers“ auf, so ergibt sich eine Abweichung zwischen dem Sollwert und der Regelgröße. Durch die Regeleinrichtung kann nun beispielsweise der Massenstrom des Kühlmediums erhöht werden, um die erforderliche Rücklaufstemperatur einzuhalten. Infolge dessen wird sich der Fehler nicht mehr am veränderten Verhältnis von Eintritts- zur Austrittstemperatur feststellen lassen. Treten nun zusätzlich wiederum veränderliche Eingangsparameter auf, so lässt sich der Fehler nicht mehr über das Verhältnis der Ein- und Ausgangstemperaturen erkennen. Der Fehler lässt sich hier über die Veränderung der Massenstromverhältnisse feststellen, die durch die Erhöhung des Massenstroms des Kühlmediums bewirkt wurde. Eine Früherkennung kann hier nur durch die Berücksichtigung der Störeinrichtung und der Regelstrecke erkannt werden. In der betrieblichen Praxis bedarf es also genauer Kenntnis und einer breiten Datenbasis, um einen systematischen Fehler und seine Auswirkungen auf den Regelkreis von „normalen“ betrieblichen Störungen zu unterscheiden.

Dies zeigt, dass für die Methode der Fehlerfrüherkennung im Schritt „Systemanalyse“ darauf geachtet werden muss, ob es sich um ein geregeltes oder unreguliertes System handelt, um die entscheidenden Parameter zu identifizieren.

6.2 Merkmalsextraktion

Aus der technischen Systemanalyse ergeben sich eine Reihe von Messgrößen, deren Verläufe als geeignet eingeschätzt werden, um einen Fehler zu identifizieren. Aus den zugrunde liegenden Messwertverläufen müssen nun Werte extrahiert werden, die als Trainingssatz und Validierungssatz für den Einsatz im neuronalen Netzen dienen sollen (vgl. Kapitel 4). Sie werden dazu in Gruppen eingeteilt, wobei eine Gruppe den Gutzustand die anderen Gruppen die jeweiligen Fehlerzustände darstellen. Um zu untersuchen, ob die Messwerte sich unterscheiden, wird ihr Mittelwert bestimmt und es wird bewertet, ob eine vermutete Abweichung vorhanden ist oder nicht. Da es um die Früherkennung von Störungen geht, werden die Abweichung im Allgemeinen gering sein, weil es ja Voraussetzung ist, dass die Früherkennung eine Anzeige macht, bevor es bei einzelnen Messwerten zu Überschreitungen von Grenzwerten kommt. Deshalb ist zumeist zu erwarten, dass sich die Mittelwerte oft nicht wesentlich unterscheiden. Tabelle 6-1 zeigt dies am Beispiel des Dampfdruckes aus einem Referenzfall (vgl. Abschnitt 7.1).

Messwert Dampfdruck	
Gutzustand [bar]	Fehler [bar]
9,60	9,57
9,56	9,48
9,60	9,56
9,64	9,57
9,62	9,55
9,59	9,63
9,73	9,58
9,73	9,57
9,73	9,59
9,70	9,51
Mittelwert: 9,65	Mittelwert: 9,56

Tabelle 6-1: Mittelwertbestimmung für zeitliche Verläufe von Dampfdrücken

Es ist ersichtlich, dass die Mittelwerte sich nur geringfügig unterscheiden. Zur Bewertung der statistischen Signifikanz des Unterschiedes zwischen beiden Gruppen muss ein statistischer Test herangezogen werden. Für unabhängige Stichproben bietet sich der Kruskal-Wallis Test /Kru52/ an.

Hierfür wird die Nullhypothese aufgestellt, dass es keinen Unterschied zwischen beiden Messgruppen, den Klassen, gibt. Den vereinigten Messwerten aus beiden Gruppen werden nun Ränge zugewiesen, wobei der kleinste Messwert den Rang 1 erhält der nächst größere den Rang 2 usw. Aus den Rängen werden Rangsummen für die einzelnen Klassen gebildet und die Teststatistik ermittelt:

$$H = \frac{12}{n(n+1)} \sum_h \left[\frac{T_i^2}{n_i} - 3 \cdot (n+1) \right] \quad (6-1)$$

mit T_i für die Werte der Rangsummen

Kommt es dazu, dass Stichproben denselben Wert haben, werden Zwischenränge eingeführt. Es erhalten dann alle gleichen Messwerte den mittleren Rang. Sind z. B. zwei kleinste Werte vorhanden, so erhalten beide Werte den Rang 1,5. Um diese Bindung zu berücksichtigen wird ein Korrekturfaktor eingeführt.

$$C = 1 - \sum \frac{(r_i^3 - r_i)}{(N^3 - N)} \quad (6-2)$$

mit r_i der Anzahl der gebundenen Beobachtungen.

Damit ergibt sich die korrigierte Teststatistik zu:

$$H' = \frac{H}{C} \quad (6-3)$$

Mittels der Chi-Quadrat Verteilung kann für verschiedene Vertrauensbereiche und dem Freiheitsgrad, der sich auf Grund der 2 Klassen zu eins ergibt, der

Wert der Chi-Quadrat Verteilung bestimmen lassen und mit dem berechneten verglichen werden. Ist der berechnete Wert größer als der der Chi-Quadrat Verteilung so muss die Nullhypothese verworfen werden. Ist dies der Fall, kann davon ausgegangen werden, dass sich beide Klassen signifikant unterscheiden.

Somit erhält man auch für Messreihen, deren Messgrößen scheinbar nicht ausreichend verschieden sind, eine statistisch belastbare Aussage bezüglich des signifikanten Unterschieds zwischen ihnen.

6.3 Datenvorverarbeitung

Die Datenvorverarbeitung stellt einen wichtigen Bestandteil für den effizienten Einsatz neuronaler Netze dar. Messrauschen, fehlerhafte Messsignale sowie große Absolutwertunterschiede sollen hier stellvertretend als Gründe genannt sein. Fehlerhafte Messsignale müssen nicht nur bei neuronalen Netzen vor der eigentlichen Fehlererkennung identifiziert und eliminiert werden, um die Bewertung zu ermöglichen, sondern auch bei allen anderen Methoden.

Dies kann durch einfache Plausibilitätskontrollen geschehen. Große Unterschiede bei den Absolutwerten können dazu führen, dass relevante Änderungen kleiner Absolutwerte nicht berücksichtigt werden, da sie von den Einflüssen der großen Absolutwerte überdeckt werden. Weiterhin liegen die Daten oft in Rohform vor, so dass mit der Vorverarbeitung ihre Glättung erreicht wird.

Zur Vorverarbeitung kommt z. B. die Skalierung in Frage

$$\bar{x}_i = \frac{(x_i - \mu(x))}{\sigma(x)} \quad (6-4)$$

mit μ als Erwartungswert und σ der Varianz des Eingangswertes x_i . Ein weiteres Verfahren ist die lineare Skalierung

$$x_s = \frac{(x-a)}{(b-a)} \quad (6-5)$$

wobei a die untere und b die obere Grenze des Eingangssignals darstellen.

Mit der so genannten z-Transformation oder auch Autoskalierung werden alle Parameter auf den Mittelwert 0 und die Standardabweichung 1 normiert /Kann04/. Die Normierung der Daten erfolgt dabei in zwei Schritten, der Zentrierung um den Mittelwert und der nachfolgenden Normierung auf die Standardabweichung 1.

Schritt 1:

$$y_{ij}^* = x_{ij} - \bar{x}_j \quad (6-6)$$

$$\text{wobei } \bar{x}_j = \frac{1}{n} \sum_{i=1}^n x_{ij} \quad (6-7)$$

das arithmetische Mittel ist.

Schritt 2:

$$\sigma_j = \sqrt{\frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2}{n-1}} \quad (6-8)$$

die Standardabweichung.

Mit Hilfe von Formel (6-6) und (6-8) ergibt sich:

$$y_{ij} = \frac{y_{ij}^*}{\sigma_j} \quad (6-9)$$

Die genannten Verfahren arbeiten mit den tatsächlichen Signalwerten. Vorstellbar ist jedoch auch eine Bewertung des Zustandes eines Merkmales. Es erfolgt

also eine Verknüpfung von Signalwert und Expertenwissen, bevor das Merkmal als Eingangsgröße in das neuronale Netz gelangt. Kann man unterstellen, dass die Änderungsrichtung der signifikante Einfluss für einen Fehler ist, so kann das Signal wie folgt transformiert werden.

Wird der Signalwert größer als der Mittelwert, so wird der Eingangswert auf +1 gesetzt. Stimmt er mit dem Mittelwert überein, so ergibt sich der Wert Null. Sinkt der Signalwert, so folgt als Eingangswert -1. Als Referenzwerte können auch unterschiedliche Werte genutzt werden, z. B. die obere und untere Grenze des definierten Normalbereiches des Signals. Dann wäre die Netzeingabe, solange sich das Signal im Normbereich befindet Null ansonsten +1 bzw. -1. Können alle Eingabegrößen nur drei Werte annehmen, sollten sowohl der Trainingsaufwand minimiert als auch die Generalisierungsleistung des Netzes verbessert werden können, da eindeutige Muster eingegeben werden.

Spielt die Änderungsgeschwindigkeit des Merkmals eine Rolle für die Fehlerentstehung, so kann das Verfahren entsprechend modifiziert werden, indem man Bereiche von Änderungsgeschwindigkeiten als Zustände auffasst, die dann mit +1, 0 oder -1 belegt werden.

Basierend auf den dargestellten Schritten zur Vorverarbeitung der Rohdaten werden nun die Trainingsdatensätze erstellt. Für jeden identifizierten Zustand werden die Messwerte zusammengestellt. Mit Hilfe der Trainingssätze kann dann die Erstellung des neuronalen Netzes erfolgen.

6.4 Musterklassifikation – Netzwerkerstellung

Wie im Kapitel 4 beschrieben, gibt es für die Festlegung der Netzwerkarchitektur keine Regeln. Es bedarf vielmehr der Erfahrung des Bearbeiters, um den iterativen Prozess der Architekturfindung möglichst effizient zu gestalten.

Abbildung 6-3 zeigt die verschiedenen Ansatzpunkte zur Aufstellung einer geeigneten Netzwerkarchitektur. Hieraus ist ersichtlich, dass eine allgemeingültige Vorgehensweise nicht angegeben werden kann. Ausgehend von der Problemspezifizierung führt der erste Schritt zur Ermittlung der Eingangsgrößen, die entsprechend der vorangehend beschriebenen Arbeitsschritte verarbeitet werden. Aus dem sich ergebenden Eingangsvektor wird eine Netzarchitektur gewählt und das Netz trainiert. Bei erfolgreichem Training wird das Netz mit nicht für das Training benutzten Daten validiert. Ist dieser Schritt erfolgreich, so ist die Netzwerkerstellung abgeschlossen. In den meisten Fällen wird es eine Schleife zwischen der Auswahl des Netzmodells und dem Netztraining geben, bis eine für das Training geeignete Architektur gefunden wurde. Führt die Validierung des Netzes nicht zu der erwünschten Bestätigung des Trainingserfolges, so muss durch Modifizierung der Architektur ein passenderes Netz gesucht werden. Es kann jedoch auch dazu kommen, dass die Eingangsgrößen neu bewertet werden müssen.

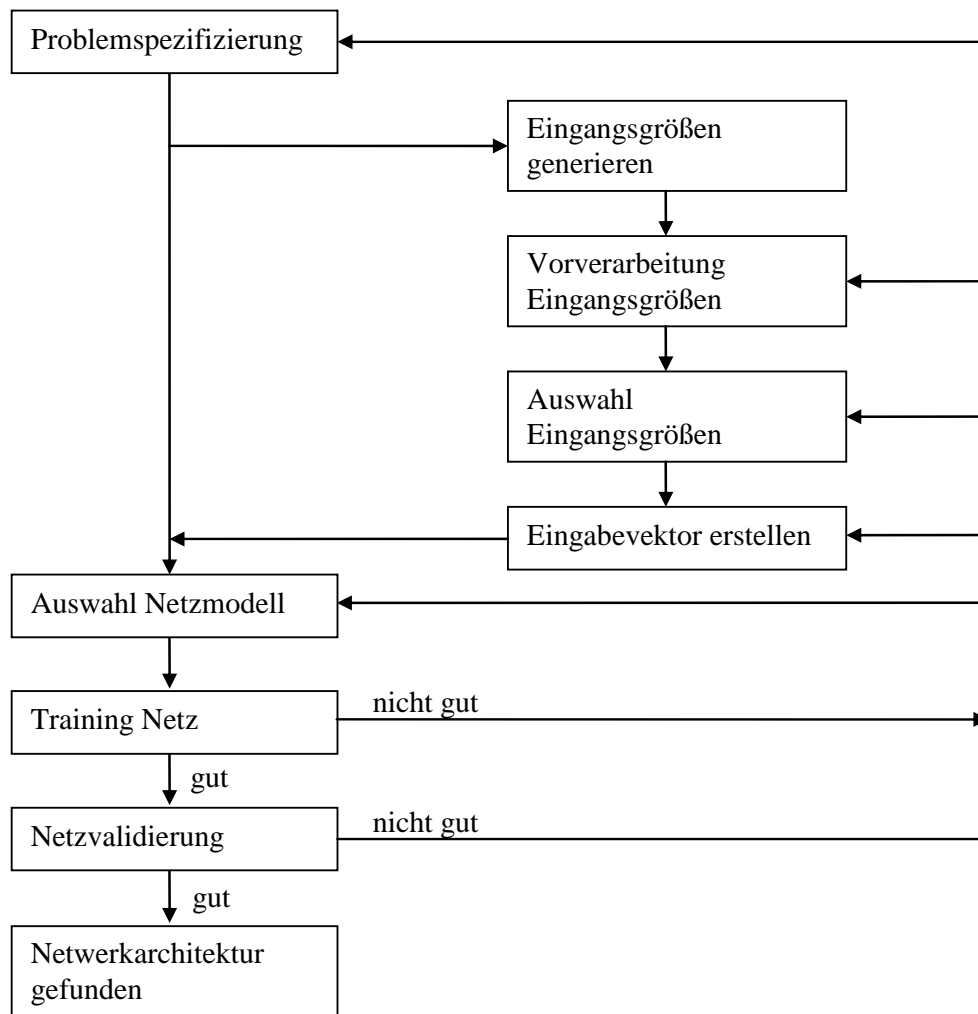


Abbildung 6-3: Vorgehensweise zur Erstellung der Netzwerkarchitektur

Wie im Kapitel 8 gezeigt, haben die Untersuchungen an unterschiedlichen Netzwerkarchitekturen ergeben, dass es sinnvoll erscheint, kleinere Netze zu bevorzugen. Dies ist jedoch nicht allgemeingültig und führt daher eher zu der Empfehlung, die Entwicklung mit kleineren Netzen zu beginnen.

7 Anwendungen

Nachfolgend wird an vier Referenzfällen die Anwendung und Durchführung des Verfahrens zur Fehlerfrüherkennung gezeigt. Dem liegt ein Entwicklungsprozess zugrunde, bei dem, ausgehend von der ersten Referenzanlage, die einzelnen Schritte der Methode weiterentwickelt wurden. Insbesondere die Änderungen, die sich aus der Anforderung einer möglichst breiten Anwendbarkeit der Vorgehensweise ergaben, konnten nur schrittweise, entsprechend dem Erfahrungsgewinn, vorgenommen werden. Dies folgt daraus, dass bei jeder Anwendung andere Randbedingungen vorliegen, die es zu berücksichtigen galt.

Ein weiterer Entwicklungsschritt erfolgte bei der Festlegung der Voraussetzungen, die für die Auswahl eines geeigneten Anlagenteils notwendig sind. Neben der grundsätzlichen Anforderung, dass es sich um einen Driftausfall handeln muss, sind hier die Qualität der Fehlerdokumentation und die Messdaten hervorzuheben.

Letzteres wird im ersten Beispiel deutlich, bei dem Messungen von Zweiphasenströmungen ausgewertet wurden, die mit erheblichen Messunsicherheiten behaftet sind.

7.1 Referenzfall „Zwischenüberhitzer“

7.1.1 Verfahrenstechnische Beschreibung der Zwischenüberhitzer

Das Teilsystem „Zwischenüberhitzer“ der Referenzanlage wurde als erstes Beispiel zur Entwicklung und Erprobung des Verfahrens ausgewählt. Das Teilsystem wurde zunächst der technischen Analyse unterzogen, um die Signalmuster, die für die Früherkennung geeignet sind, herauszuarbeiten.

Die Zwischenüberhitzer dienen der Überhitzung des Dampfes hinter der Hochdruckturbine, bevor er in der Niederdruckturbine weiter entspannt wird.

Der Dampf aus dem Reaktor wird vor der Hochdruckturbine abgezweigt und hinter dem Zwischenüberhitzer über den Kondensatkühler wieder in den Speisewasserkreislauf rückgeführt. Je nach Betriebszustand kann der Kondensatstrom über die Stränge „Speisewasserbehälter“ oder „Verweilbehälter“ umgeleitet werden (siehe Abbildung 7-1).

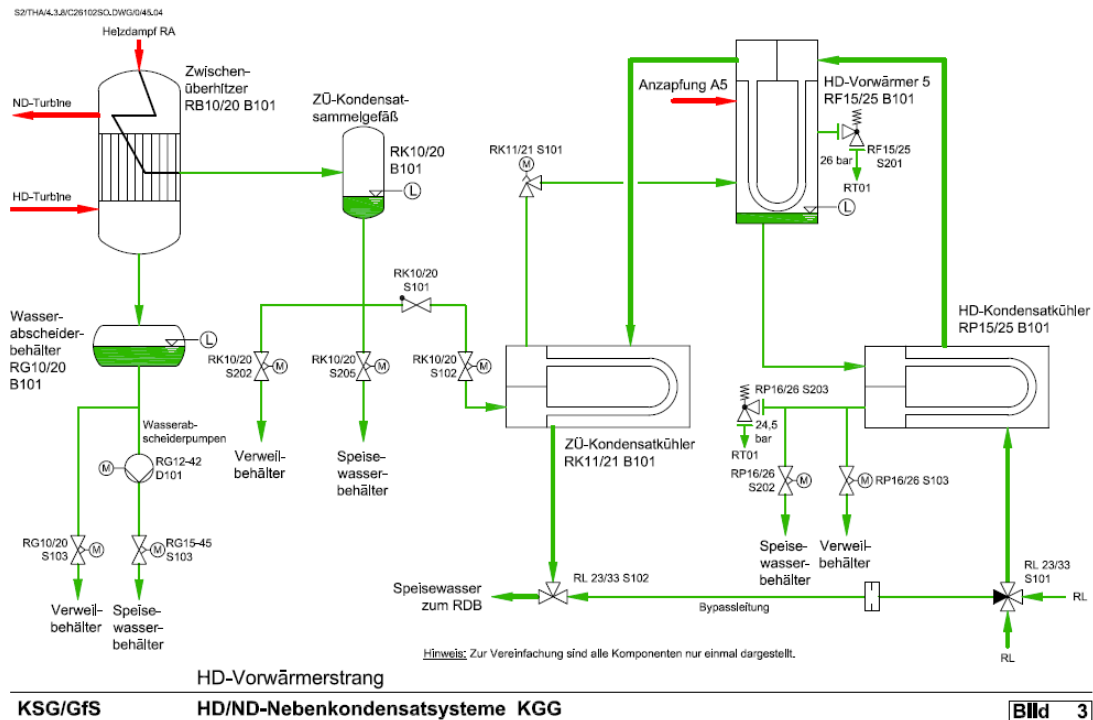


Abbildung 7-1: R+I Fließbild der Zwischenüberhitzerkreisläufe Quelle: Betreiber

7.1.2 Fehlerbeschreibung

Bei der Auswertung der Betriebsdaten wurde eine Leckage im Zwischenüberhitzer als Fehler identifiziert. Hierbei kam es zur Leckbildung bei einzelnen Rohren im Rohrbündel des Wärmeübertragers. Der Fehler wurde durch einen über mehrere Monate ansteigenden Betrag der Differenz des Massenstroms vor und hinter dem Zwischenüberhitzer vom Kraftwerkspersonal vermutet. Im Rahmen von Revisionsarbeiten konnte diese Vermutung bestätigt werden.

Auf Grund von Lastwechseln, den beschriebenen Kondensatumleitungen sowie der dynamischen Funktion des Vorratsbehälters handelt es sich um ein instationäres System. Hieraus resultiert eine Laufzeitdifferenz. Diese ist bedingt durch

die Zeit, die das Medium braucht, um von der ersten Messstelle zur zweiten Messstelle zu strömen. Im stationären Zustand ist dies unerheblich, da $dm/dt=0$. Im instationären Fall resultiert hieraus die Berechnung eines fehlerbehafteten Differenzmassenstromes, der zu einer Anzeige „Leckage“ führen kann. Eine einfache Überwachung des Differenzmassenstroms ist somit nicht ausreichend, um einen Fehler eindeutig zu identifizieren; es müssen weitere Parameter gefunden werden, mit deren Hilfe eine eindeutige Zustandsbeschreibung möglich ist. Hier sind die Berücksichtigung der Laufzeitdifferenz sowie der Stellungen der Armaturen der Umleitungsmöglichkeiten zu nennen.

Wie bereits erwähnt, trat an einem Zwischenüberhitzerkreislauf ein Fehler auf, der bei Revisionsarbeiten bestätigt (Block B RB20) wurde. Der andere Kreislauf zeigte im Beobachtungszeitraum keinen Befund. Im Block C weisen die Datensätze für den Zwischenüberhitzer – Kreislauf RB 10 eine ähnliche Charakteristik auf wie im Block B RB20, so dass ein ähnlicher Fehler vermutet wird. Da jedoch im Zeitraum der Bearbeitung keine Revision stattgefunden hatte, fehlt die Bestätigung oder Widerlegung der Vermutung. Die durchgeführten Arbeiten beschränkten sich daher auf den Block B. Als mögliche Ursachen für die aufgetretenen Leckagen sind Materialfehler oder fehlerhafte Schweißnähte der Rohrleitungen aufzuführen. Unter dem Einfluss thermischer und mechanischer Belastungen im Betrieb sind die aufgetretenen Leckagen erklärbar. Die genannten Ursachen können nicht direkt durch die vorhandenen Messstellen detektiert werden. Die Fehlerfrüherkennung musste daher über die messtechnisch erfassbaren Auswirkungen auf die Systemparameter erfolgen. Neben den durch die Leckagen resultierenden Massenstromänderungen ergeben sich in Abhängigkeit von der Lage des Lecks (oberhalb bzw. unterhalb der Heizrohrbündel) geringfügige Unterschiede der Dampftemperaturen. Darüber hinaus haben die Stellungen der Bypassarmaturen und die Füllstände der Sammelgefäße einen Einfluss auf die Mustererkennung. Daher wurden die entsprechenden Messgrößen, soweit vorhanden, als Eingangsgrößen für das neuronale Netz aufgenommen. Aufgrund der Messwertverläufe wurde vom Kraftwerkspersonal eine Leckage vermutet, was sich dann bei den Revisionsarbeiten bestätigte. Allerdings kann der Zeitpunkt des Leckagebeginns nicht genau angegeben werden.

Er wurde deshalb in Ansehung der Messwertverläufe festgelegt und muss folglich nicht unbedingt dem wahren Zeitpunkt entsprechen. Es ist davon auszugehen, dass im Laufe der Zeit weitere Rohre von Leckagen betroffen waren, wodurch sich der Einfluss auf die Messwerte verstärkte. Somit ist die Erstellung des Trainingsdatensatzes mit Unsicherheiten behaftet. Der Messwert „Massenstrom vor Zwischenüberhitzer“ resultiert überdies aus der Messung einer Zweiphasenströmung, die mit Messungenauigkeiten behaftet ist, welche in der Größenordnung der vermutlich aufgetretenen Leckmassenströme liegen können. Somit ist mit einer Messunsicherheit umzugehen, die so groß ist, dass eine erfolgreiche Fehlerfrüherkennung nur durch Hinzuziehen der erwähnten zusätzlichen Messgrößen möglich ist. Aus den zur Verfügung stehenden Messdaten lassen sich die in Abbildung 7-2 angegebenen Änderungen der Messwerte in Abhängigkeit vom Lastzustand ableiten.

Parameter Änderung der Größe	Reaktor- leistung	Heizdampf- massen- strom	Dampf- druck hinter ZÜ	Dampf- temp. Hinter ZÜ	Stellung Bypass	Massen- strom hin- ter ZÜ Kond.- Kühler
Nominalzustand	↔	↔	↔	↔	↔	↔
Anfahrvorgang	↑	↑	↑	↑	↓	↑
Abfahrvorgang	↓	↓	↓	↓	↑	↓
Fehler (bei Nennleistung)	↔	↔	↔	↑/↓	↔	↓

Abbildung 7-2: Parameteränderungen in Abhängigkeit vom Lastzustand

Es handelt sich bei den zur Verfügung gestellten Messdaten um Viertelstundenmittelwerte, mit denen die Bestimmung der Laufzeitdifferenz, die vermutlich im Bereich einiger Sekunden liegt, nicht möglich ist. Die Laufzeitdifferenz kann daher entgegen der ursprünglichen Annahme nicht zur Fehlerentdeckung herangezogen werden.

Da es sich um ein Kraftwerk handelt, das fast ständig im Grundlastbereich betrieben wird, war der überwiegende Betriebszustand im Betrachtungszeitraum die Nennlast. Im Nennlastbetrieb treten die Einflüsse der Stellungen der By-

passarmaturen und Zwischenbehälter nicht auf, da diese ausschließlich bei An- und Abfahrvorgängen verfahren werden. Dadurch spielt die Laufzeitdifferenz im quasi-stationären Anlagenbetrieb für Nennlast keine Rolle und die vorhandenen Daten konnten verwendet werden.

7.1.3 Merkmalsextraktion

Die Merkmalsextraktion hat das Ziel, diejenigen Größen aus dem vorhandenen Datensatz zu identifizieren, die von dem zu diagnostizierenden Fehler beeinflusst werden. Aus dem Rohdatensatz wurde ein Nennlastdatensatz extrahiert. Die Merkmalsextraktion erfolgte mit Hilfe der Signifikanzanalyse für alle Messwerte. Es wurde hierbei untersucht, bei welchen Messgrößen ein signifikanter Unterschied zwischen fehlerfreiem und fehlerbehaftetem Betrieb feststellbar war. Dazu wurde, wie im Abschnitt Systemanalyse beschrieben, ein Zeitpunkt für den Eintritt des Fehlers festgelegt. Dann wurden die Messwerte vor und nach dem Zeitpunkt mit Hilfe des Kruskal-Wallis Tests /Har94/ miteinander verglichen.

Die Ergebnisse der Signifikanzanalyse sind in Tabelle 7-1 wiedergegeben. Die Reaktorleistung stellt die Führungsgröße dar. Damit gleiche Betriebsverhältnisse vor und nach dem Fehler zu Grunde gelegt werden, darf die Reaktorleistung in diesem Zeitraum keine signifikant unterschiedlichen Werte aufweisen, Wie der Tabelle entnommen werden kann, ist dies nicht der Fall. Die Nullhypothese ist dabei die Annahme, dass die verglichenen Parameter signifikant verschieden sind. Die Annahme wird somit bestätigt.

7 Anwendungen

Messgröße	Ergebnis der Nullhypothese / χ^2 - Wert	Ergebnis
Dampfdruck hinter ZÜ	8,99 / 3,84	Stichproben sind signifikant verschieden bei Vertrauensniveau 95%
Temperatur hinter ZÜ	0,82 / 3,84	Stichproben sind <u>nicht</u> signifikant verschieden
Heizdampfmassenstrom	14,29 / 6,63	Stichproben sind signifikant verschieden bei Vertrauensniveau 99%
Reaktorleistung	0,00	Stichproben sind <u>nicht</u> signifikant verschieden
Massenstrom hinter Kondensatkühler	3,86 / 3,84	Stichproben sind signifikant verschieden bei Vertrauensniveau 95%
Druck im Sammelbehälter	2,08 / 3,84	Stichproben sind <u>nicht</u> signifikant verschieden
Regelgefäß	13,72 / 6,63	Stichproben sind signifikant verschieden bei Vertrauensniveau 99%
Kondensatablauf	5,85 / 3,84	Stichproben sind signifikant verschieden bei Vertrauensniveau 95%
Dampfdruck hinter ZÜ	8,99	Stichproben sind signifikant verschieden bei Vertrauensniveau 95%
Temperatur hinter ZÜ	0,82	Stichproben sind <u>nicht</u> signifikant verschieden
Heizdampfmassenstrom	14,29	Stichproben sind signifikant verschieden bei Vertrauensniveau 99%
Reaktorleistung	0,00	Stichproben sind <u>nicht</u> signifikant verschieden

Messgröße	Ergebnis der Nullhypothese / χ^2 - Wert	Ergebnis
Massenstrom hinter Kondensatkühler	3,86	Stichproben sind signifikant verschieden bei Vertrauensniveau 95%
Druck Sammelbehälter	2,08	Stichproben sind <u>nicht</u> signifikant verschieden
Regelgefäß	13,72	Stichproben sind signifikant verschieden bei Vertrauensniveau 99%
Kondensatablauf	5,85	Stichproben sind signifikant verschieden bei Vertrauensniveau 95%

Tabelle 7-1: Ergebnisse Kruskal-Wallis Test

Damit sind aufgrund von technischer Systemanalyse und Merkmalsextraktion die folgenden Größen für die Fehlerfrüherkennung heranzuziehen:

- Massenstrom vor Zwischenüberhitzer
- Massenstrom hinter Zwischenüberhitzer
- Dampfdruck hinter ZÜ
- Kondensatablauf
- Differenzmassenstrom
- Notablauf Regelgefäß

Die Messgröße „Notablauf Regelgefäß“ spiegelt eine Stellungsanzeige für eine Armatur wider. Da diese Größe nach Aussagen des Betreibers im Nennlastbetrieb nur auf Grund des Messprinzips Werte liefert, die jedoch keine Stellungsänderung der Armatur bedeuten, also kein eigentliches Stellsignal beinhaltet, wurde diese Größe nicht weiter berücksichtigt.

7.1.4 Anwendung des neuronalen Netzes

Gehen Daten unterschiedlicher Größenordnungen in ein neuronales Netz ein, so kann gezeigt werden, dass relativ gleiche Änderungen von Parametern mit

großem numerischen Wert, wie z.B. die Reaktorleistung (ca. 3800MW thermisch) höheren Einfluss haben als die Parameter mit kleinen Absolutwerten (z. B. Temperatur mit ca. 240°C). Dies ergibt sich daraus, dass eine prozentual gleiche Änderung einer großen Grundgröße zu einer größeren Änderung führt als bei einer kleineren. Dies verursacht eine stärkere Anregung der Aktivierungsfunktion der Neuronen des neuronalen Netzes. Daraus folgt die Notwendigkeit, die vorhandenen Rohdaten der Messgrößen vorzuverarbeiten. In einer Versuchsreihe wurden, um der Tatsache der unterschiedlichen Größenordnungen Rechnung zu tragen, die Rohdaten auf den jeweils größten auftretenden Wert bezogen. Sie werden somit auf einen Bereich zwischen Null und Eins abgebildet. In einem ersten Schritt wurde ausschließlich der Betrieb bei Nennlast untersucht. Es wurden sowohl Netze mit Rohdaten als auch Netze mit vorverarbeiteten Daten trainiert, um an Hand der Generalisierungsfähigkeit den Einfluss der Vorverarbeitung bewerten zu können.

Bei den Versuchsreihen mit den dazu entwickelten Programmen „NeuroTrain“ und „NeuroApp“ (siehe Kapitel 8) trat folgendes Problem auf. Wie bereits erwähnt, kann der exakte Zeitpunkt des Fehlers nicht bestimmt werden. Die messbaren Auswirkungen des Fehlers nehmen aber mit der Zeit zu. Dies ist in Abbildung 7-3 anhand der zunehmenden Differenz der Massenströme zu erkennen.

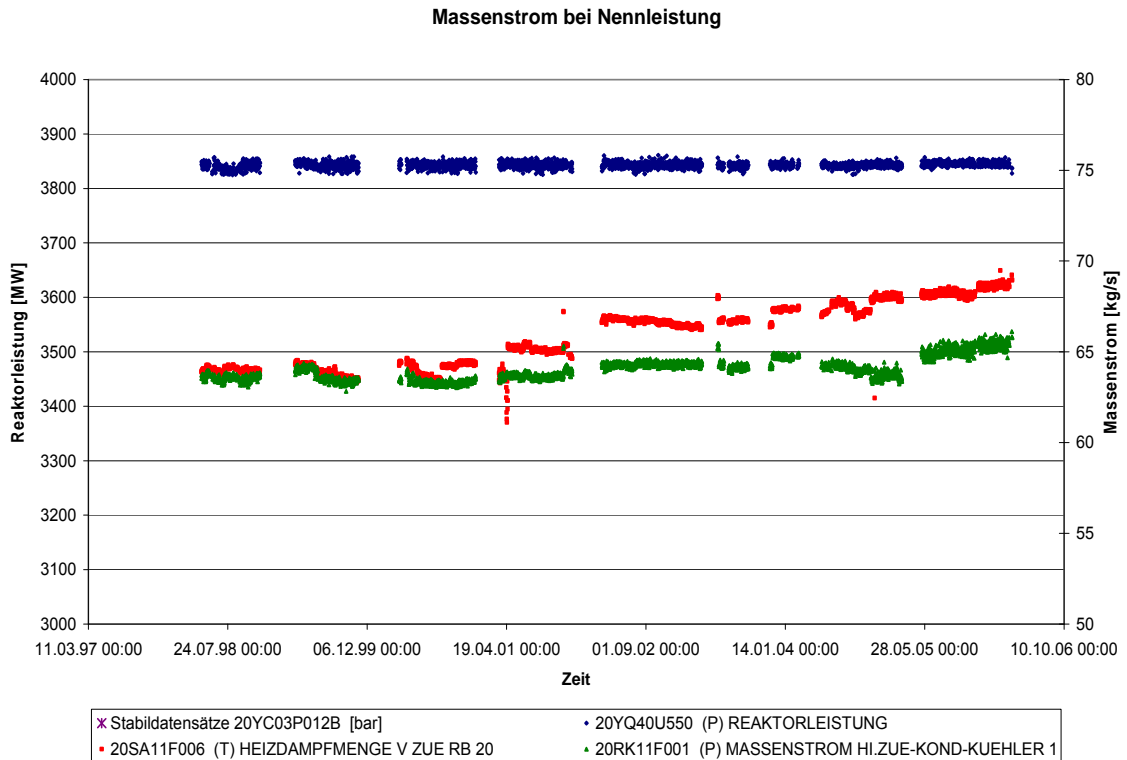


Abbildung 7-3: Differenzmassenstrom auf Grund der Leckage

Die Ergebnisse zeigen, dass mit dem neuronalen Netz der Fehler im fehlerbehafteten Zwischenüberhitzer erkannt wurde. Jedoch zeigte sich bei Validierungsversuchen am nicht fehlerbehafteten Zwischenüberhitzer eine nicht zutreffende Fehlermeldung. Dabei wurden zum Training des Netzes Daten aus allen Stufen der Fehlerausprägung eingesetzt. Da anzunehmen ist, dass hierdurch Übergangsbereiche vom Netz bereits als Fehler ausgelegt werden, wurde ein modifizierter Trainingssatz erstellt, in dem nur Daten zum Einsatz kommen, bei denen der Fehler bereits in ausgeprägter Form vorliegt (Abbildung 7-4, Abbildung 7-7). Es wurde untersucht, in wie weit hierdurch die Wahrscheinlichkeit von fehlerhaften Zuordnungen reduziert werden kann, ohne die Qualität der Trennschärfe zu negativ zu beeinflussen.

Neben den bei der Systemanalyse gezeigten Ergebnissen, stellt die Ausgabe des Programms „NeuroApp“ den entscheidenden Punkt der Arbeiten dar (vgl. Kapitel 8). Die Ausgabe des Netzsimulators kann als ein Maß für die Wahrscheinlichkeit der Fehlererkennung interpretiert werden. Wie unter Kapitel 8 beschrieben wird der Verlauf der Wahrscheinlichkeit des Fehlers bzw. des Gut-

zustandes über der Zeit dargestellt. Nachfolgend werden die Ergebnisse der Untersuchungen mit unterschiedlichen Ausprägungen der Fehler zur Verwendung für das Training des neuronalen Netzes vorgestellt.

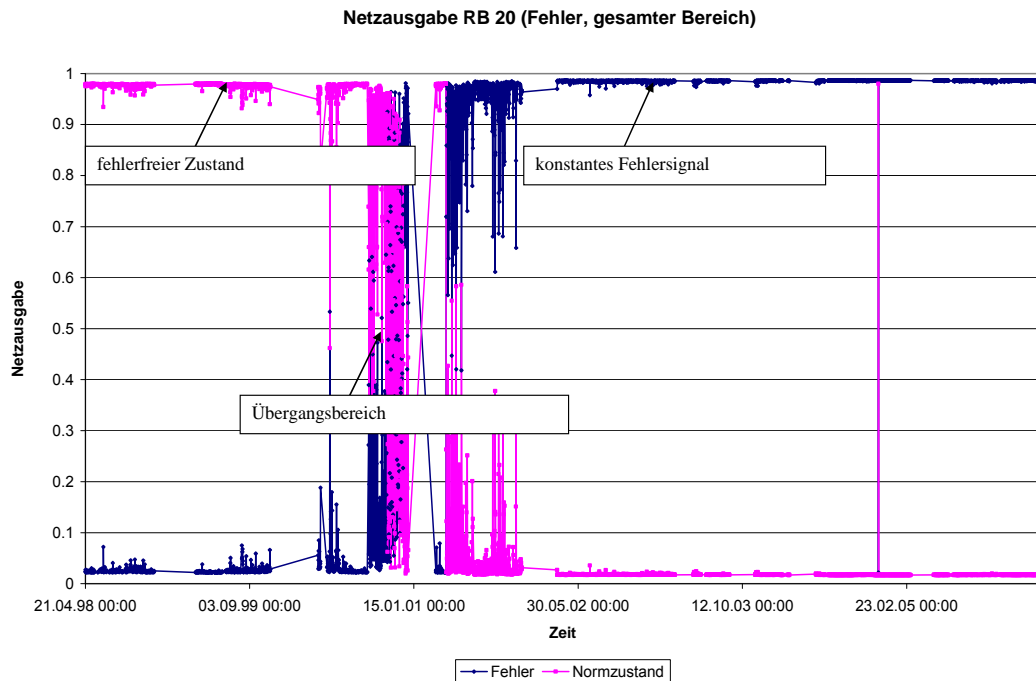


Abbildung 7-4: Netzausgabe Zwischenüberhitzer mit Befund, gesamter Fehlerbereich

Abbildung 7-4 zeigt die Netzausgabe bei Verwendung des gesamten Fehlersatzes beim Zwischenüberhitzer mit Fehler. Der Fehler wird eindeutig erkannt. Dies erfolgt bereits sehr früh, was durch den unter Kapitel 7.2 diskutierten Übergangsbereich erklärbar ist. Es sei hier nochmals darauf hingewiesen, dass der tatsächliche Zeitpunkt des Auftretens der Leckage unbekannt ist. Es ist zu vermuten, dass auf Grund der Verwendung des gesamten Fehlerspektrums, also auch Verwendung der Datensätze, die zeitlich nah bei dem angenommenen Zeitpunkt des Leckagebeginns liegen, auch statistische Fluktuationen als Fehler gewertet werden.

Die obige Annahme wird durch die Anwendung auf den fehlerfreien Kreislauf bestätigt. Abbildung 7-5 zeigt die Netzausgabe bei Verwendung des gesamten Fehlersatzes beim Zwischenüberhitzer ohne Fehler. Es zeigt sich, dass das

Netz relativ häufig einen Fehler diagnostiziert, obwohl kein Fehler vorliegt. Das neuronale Netz wertet offenbar statistische Abweichungen als Fehler. Auch wenn ein Fehler erst bei einem zeitlich länger andauernden Signal unterstellt wird, waren die Ergebnisse nicht zufrieden stellend. Es wurde daher nach einer Möglichkeit gesucht, die fehlerhaften Zuordnungen zu reduzieren, ohne die Sensitivität der Fehlerfrüherkennung negativ zu beeinflussen.

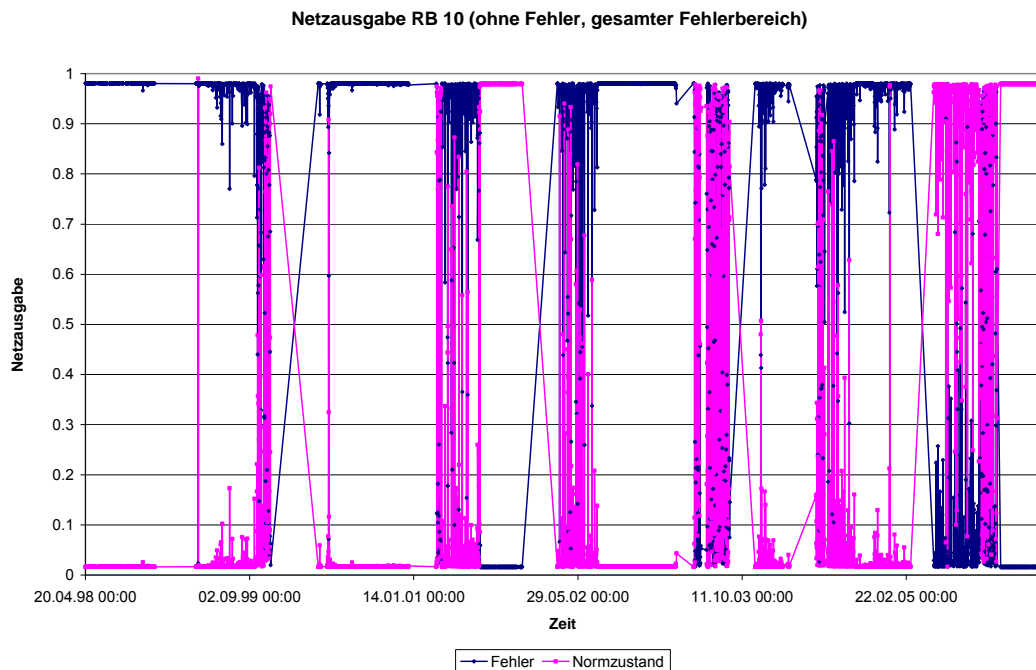


Abbildung 7-5: Netzausgabe Zwischenüberhitzer ohne Befund gesamter Fehlerbereich

Abbildung 7-6 zeigt die Netzausgabe bei Verwendung des modifiziertem Fehlerspektrums beim Zwischenüberhitzer ohne Fehler. Im Vergleich zu Abbildung 7-5 ist eine deutlich verbesserte Zustandserkennung zu verzeichnen, was für die Wahl des modifizierten Fehlerspektrums spricht.

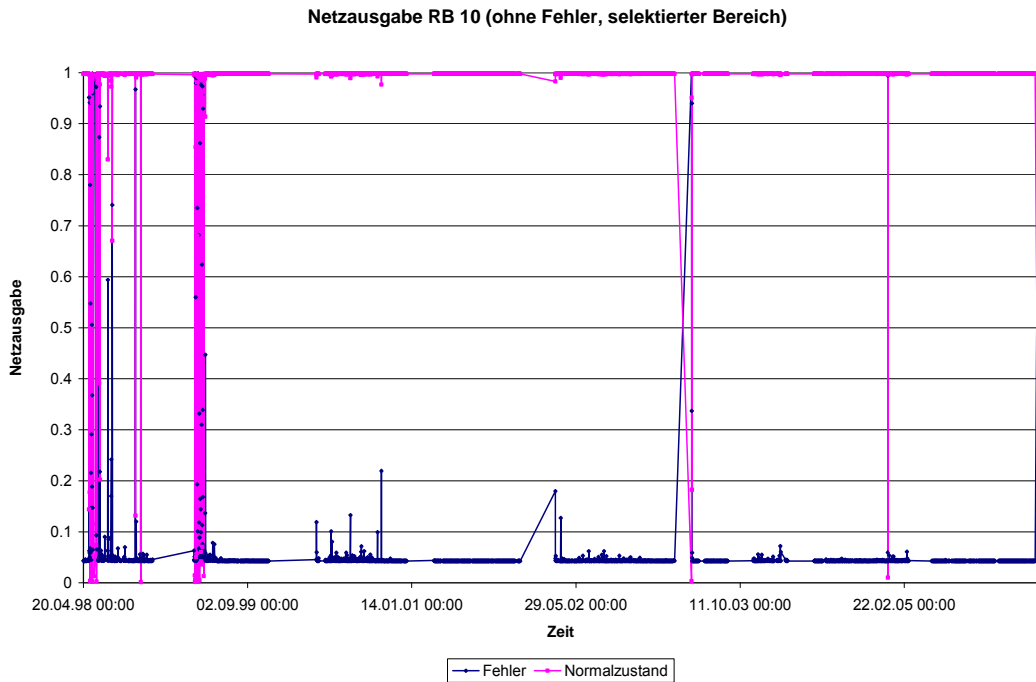


Abbildung 7-6: Netzausgabe ZÜ ohne Befund, selektierter Fehlerbereich

Abbildung 7-7 zeigt die Netzausgabe bei Verwendung des modifizierten Fehlerpektrums beim Zwischenüberhitzer mit Fehler. Im Gegensatz zu Bild 2 wird der Fehler erkennbar später, aber rechtzeitig, angezeigt, fällt jedoch besser mit dem angenommenen

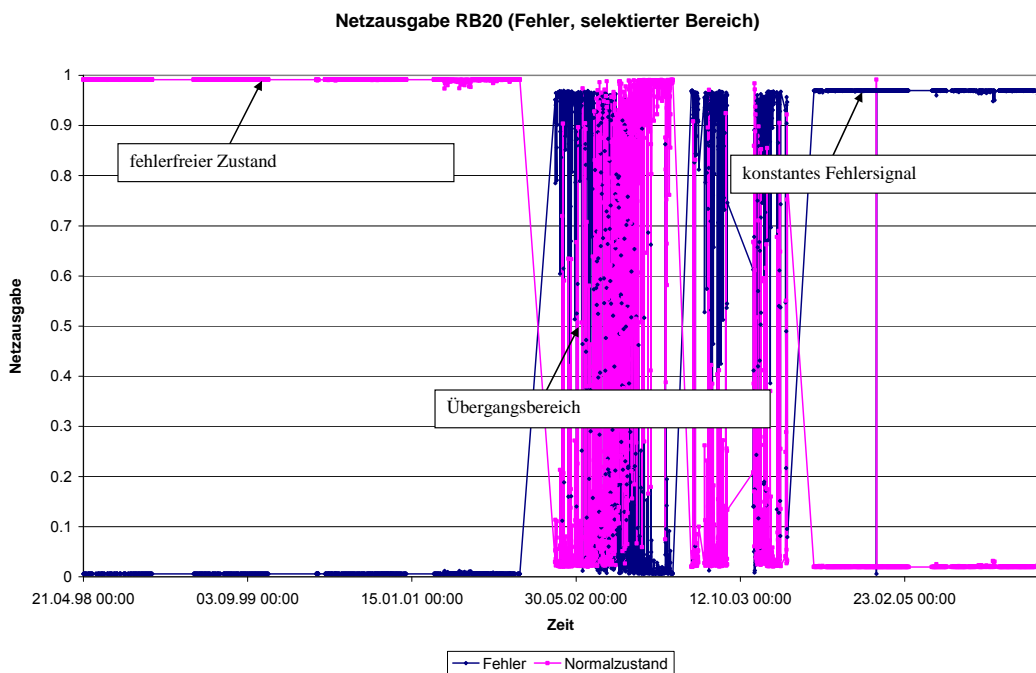


Abbildung 7-7: Netzausgabe ZÜ mit Befund und selektiertem Fehlerbereich

Zeitpunkt zusammen. Der Übergangsbereich ist breiter ausgeprägt, was den Kompromiss in der Trennschärfe widerspiegelt.

Die gezeigten Ergebnisse spiegeln auch die erfolgreiche Gestaltung der Programme „NeroTrain“ und „NeuroApp“ wider.

7.2 Referenzfall „Kompensatorschaden an einer Mitteldruckturbine“

Die Auswahl eines geeigneten Teilsystems fand in Zusammenarbeit mit den entsprechenden Verantwortlichen des Kraftwerkes statt. Im Rahmen eines vier-tägigen Arbeitsbesuchs im Kraftwerk wurden die für die Systemanalyse relevanten Messstellen identifiziert und diese für den entsprechenden Zeitraum aus dem Systemspeicher ausgelesen.

Ein Kompensatorschaden stellt ein sicherheitstechnisch relevantes Schadensereignis dar, das nicht über einen direkt messbaren Parameter detektiert werden kann. Es bietet sich daher das Verfahren an, mit dem auf Grund von Mustern relevanter Parameter, der Fehler nachgewiesen werden kann. Im Folgenden wurde durch die technische Systemanalyse und die Auswertung der bereitgestellten Daten die Möglichkeit untersucht, die Methode erfolgreich anzuwenden.

7.2.1 Technische Systemanalyse

Kompensatoren werden überall dort eingesetzt, wo es nicht möglich ist, über eine entsprechende Rohrleitungsführung Längen- und Winkeländerungen aufgrund von Temperaturschwankungen auszugleichen. Sie fungieren als Dehnungsausgleicher und sind infolge konstruktiver Randbedingungen und Werkstoffauswahl die Schwachstellen in Rohrsystemen.

Aufgrund ihrer physikalischen Eigenschaften dehnen sich die verwendeten Materialien bei Temperaturerhöhung aus und ziehen sich mit abnehmender Temperatur zusammen. In Rohrleitungssystemen macht sich dieser Vorgang im Wesentlichen durch Längenänderungen in den einzelnen Rohrleitungssträngen bemerkbar. Da Rohrleitungen im Allgemeinen an mindestens zwei Anschlusspunkten fixiert sind, treten zwangsläufig Spannungserhöhungen aus behinderter Wärmedehnung und Reaktionskräfte an den Anschlusspunkten auf. Soweit es die Randbedingungen zulassen, wird versucht, die Rohrleitungsdehnung durch eine elastische Verlegung im System aufzunehmen, d. h. natürlich zu kompen-

sieren. Wenn alle Möglichkeiten des natürlichen Dehnungsausgleichs ausgeschöpft sind, werden zum Ausgleich von Längenänderungen Kompensatoren verwendet. Darüber hinaus sind beim Betrieb von Kompressoren, Pumpen, Turbinen oder Motoren die entstehenden mechanischen Schwingungen so zu dämpfen, dass an den angeschlossenen Rohrleitungen, Armaturen und Unterstützungskonstruktionen keine Schäden verursacht werden. Außerdem dürfen an den Stützen von Verdichtern, Turbinen, Behältern und Kolonnen die zulässigen Kräfte und Momente nicht überschritten werden. Für die vorgenannten Anwendungsfälle werden Kompensatoren eingesetzt.

Kernstück eines jeden Kompensators ist der Metallbalg, der durch seine Wellengeometrie und dünnwandige Ausführung als Feder wirkt, wobei er nachfolgende Basisbedingungen erfüllen muss, um als Dehnelement Verwendung zu finden. Er muss den Betriebs- und Probebedingungen (Druck, Temperatur) des Rohrleitungssystems standhalten, gegenüber inneren und äußeren Einflüssen korrosionsbeständig sein, flexibel Dehnungen und ggf. auch Schwingungen aufnehmen können und dabei eine geforderte Lebensdauer erreichen sowie eine ausreichende Knickstabilität aufweisen.

Das Versagen eines Kompensators an der Turbine stellt eine erhebliche Gefahr für das Personal sowie für die Sicherheit des Kraftwerkes dar. Eine Früherkennung eines sich anbahnenden Schaden am Kompensator stellt somit einen Beitrag zur Erhöhung der Sicherheit dar.

In diesem Fall wird die Fehlerfrüherkennung allerdings dadurch erschwert, dass der Fehlerursprung, die mikroskopische Rissbildung auf Grund von Materialermüdung bzw. Materialfehlern, keine direkte Auswirkung auf üblicherweise gemessenen Parameter hat. Erst bei Fortschreiten des Fehlers und einem Aus- bzw. Übertritt des Fluids ist mit einer messbaren Auswirkung zu rechnen.

Im vorliegenden Fall sind die Kompensatoren der Einströmung aus der Mitteldruckturbine in die zweiflutige Niederdruckturbine durch mechanische Belastung gerissen. Es handelt sich um zweilagige Axialkompensatoren, deren prinzipieller Aufbau in Abbildung 7-8 gezeigt wird.

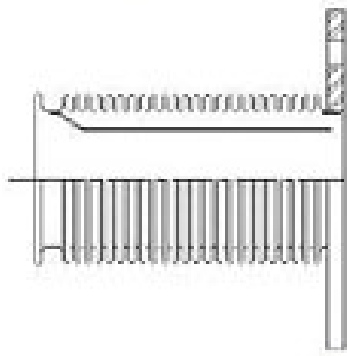


Abbildung 7-8: Schematische Darstellung eines Axialkompensators

7.2.2 Auswertung der Messdaten

Da eine direkte Messung der Integrität der Kompensatoren bislang technisch nicht umgesetzt wird, muss die Zustandsdiagnose über die gemessenen Systemparameter erfolgen. Es war dementsprechend die Aufgabe der Systemanalyse, die relevanten Messstellen zu identifizieren, um eine Zustandsdiagnose zu ermöglichen.

Die technische Systemanalyse zeigte eine Vielzahl von Messparametern auf, aus deren Verlauf eine Zustandsidentifizierung möglich erscheint. Hier seien neben Temperatur und Druckmessstellen vor allem die Schwingungsüberwachung der Turbine genannt, da im Falle eines Kompensatorschadens eine Verstärkung der Schwingungen der Turbinenwelle zu erwarten ist. Diese Vermutung wurde durch die Auswertung des dokumentierten Schadensereignisses bestätigt, da ein Schaden auf Grund einer Überschreitung der Grenzwerte der Turbinenschwingung festgestellt wurde. In wie weit dies sich als Indikation für eine Fehlerfrüherkennung als geeignet herausstellt, musste durch die weiteren Arbeiten geklärt werden. Um eine möglichst große Auflösung der Messdaten zu erhalten und so auch kurzfristigere Schwankungen berücksichtigen zu können, wurden Minutenwerte zur Analyse herangezogen. Hieraus folgte eine sehr große Datenmenge, die aufgrund des vorliegenden Datenformates, vorab bearbeitet werden musste, um eine Auswertung zu ermöglichen. Die entsprechenden Routinen werden im Kapitel 8 vorgestellt.

Die aus dem Kraftwerksarchiv extrahierten Daten wurden mittels einer Visual Basic Routine in ein zur Weiterverarbeitung durch Excel geeignetes Format überführt. Es handelt sich hierbei um eine Sortieroutine, die die Blockdatensätze im Abspeicherungsformat der Kraftwerkswerte in Zeitreihen umwandelt. Anschließend wurde anhand einer grafischen Auswertung versucht, den Zeitpunkt des Kompensatorschadens nachzuvollziehen. Der Schaden wurde durch die Überschreitung der Schwingungsgrenzwerte der Turbine und die daraus folgende Turbinenschnellabschaltung identifiziert.

Aufgrund der technischen Analyse wurde folgendes Systemverhalten erwartet. Vor dem Abschaltzeitpunkt sollten die gemessenen Parameter (Druck Temperatur und Schwingungsverhalten Turbine) einen, gegenüber vorangegangenen ähnlichen Betriebsfahrweisen, geänderten Verlauf aufweisen. Hierbei sollten vor allem die Temperaturen und Drücke hinter der Niederdruckturbine 2 (ND2) absinken, da durch eine Leckage ein Druckverlust folgt, der auch ein Absinken der Temperatur zur Folge hat. Im Gegensatz dazu sollten die Werte hinter Niederdruckturbine 1 (ND1) konstant bleiben. Demzufolge lässt sich die Abweichung an einer Änderung aus der Differenz zwischen ND1 und ND 2 am besten ablesen.

Messwerte Temperaturen:
ND1 Abdampf 1 SA13T036A
ND1 Abdampf 2 SA13T066A
ND2 Abdampf SA14T036A
ND2 Abdampf SA14T066A
ND1 Anzapfung A1 SA13T021A
ND1 Anzapfung A2 SA13T022A
ND1 Anzapfung A3 SA13T023A
ND2 Anzapfung A1 SA14T021A
ND2 Anzapfung A2 SA14T022A
ND2 Anzapfung A3 SA14T023A

Tabelle 7-2 Übersicht über die Temperaturmessstellen für die Fehlerfrüherkennung

Beim Auftreten von Abweichungen ist noch zu klären, ob an entsprechender Stelle Maßnahmen getroffen wurden, welche anstelle des Schadens als Erklärung für das beobachtete Verhalten dienen könnten.

Tabelle 7-2 und Tabelle 7-3 geben einen Überblick über die als relevant identifizierten Messstellen.

Messwerte Drücke
ND1 Anzapfung A1 SA13P021
ND1 Anzapfung A2 SA13P022
ND1 Anzapfung A3 SA13P023
ND2 Anzapfung A1 SA14P021
ND2 Anzapfung A2 SA14P022
ND2 Anzapfung A3 SA14P023

Tabelle 7-3: Übersicht über die Druckmessstellen für die Fehlerfrüherkennung

Im Allgemeinen werden die beiden Niederdruckturbinen mit denselben Dampfmen gen bei gleichen Dampfparametern beaufschlagt. Daraus resultiert, dass eine Abweichung aus einer steigenden Differenz zwischen den jeweiligen Parametern beider Turbinen ablesbar sein sollte. Dies sollte sowohl bei den Temperaturen als auch bei den Druckmesswerten der Fall sein.

Messwert	Prognostizierte Änderung bei Fehler
Temperaturen	
Differenz ND1 & ND2 T036	steigend
Differenz ND1 & ND2 T066	steigend
Differenz ND1 & ND2 T022	steigend
Differenz ND1 & ND2 T023	steigend
Drücke	
Differenz ND1 & ND2 P022	steigend
Differenz ND1 & ND2 P023	steigend

Tabelle 7-4: Erwartete Änderung der Differenzen zwischen den Parametern

Tabelle 7-4 zeigt die Differenzen zwischen den Messstellen von Niederdruckturbine 1 und 2. Da für Niederdruckturbine 1 keine Veränderungen zu erwarten sind, sind steigende Differenzen für alle Differenzwerte zu unterstellen.

Abbildung 7-9 zeigt den Verlauf der Temperaturdifferenzen. Bei den Differenzen T036, T066 sowie T023 (auf der linken Achse skaliert) ergeben sich keine signifikanten Änderungen. Die Änderung der Temperatur bei der Messung T036 liegt im Bereich von ca. 4 K, der bei Dampf-Kondensat-Mischungen keinen signifikanten Unterschied darstellt. Für die Temperaturdifferenz T022, skaliert auf der rechten Achse, ergibt sich eine sehr hohe Differenz, die vordergründig nicht zu erklären ist. Da es sich jeweils um dieselben Anzapfstellen der Niederdruckturbinen handelt, sind große Abweichungen thermodynamisch nicht zu erklären. Die stark erhöhte und breit schwankende Differenz deutet jedoch darauf hin, dass sie für den aufgetretenen Fehler nicht relevant ist. Man würde stattdessen eine stetig steigende Differenzänderung erwarten, die in Folge des Kompensatorschadens dann stark ansteigt.

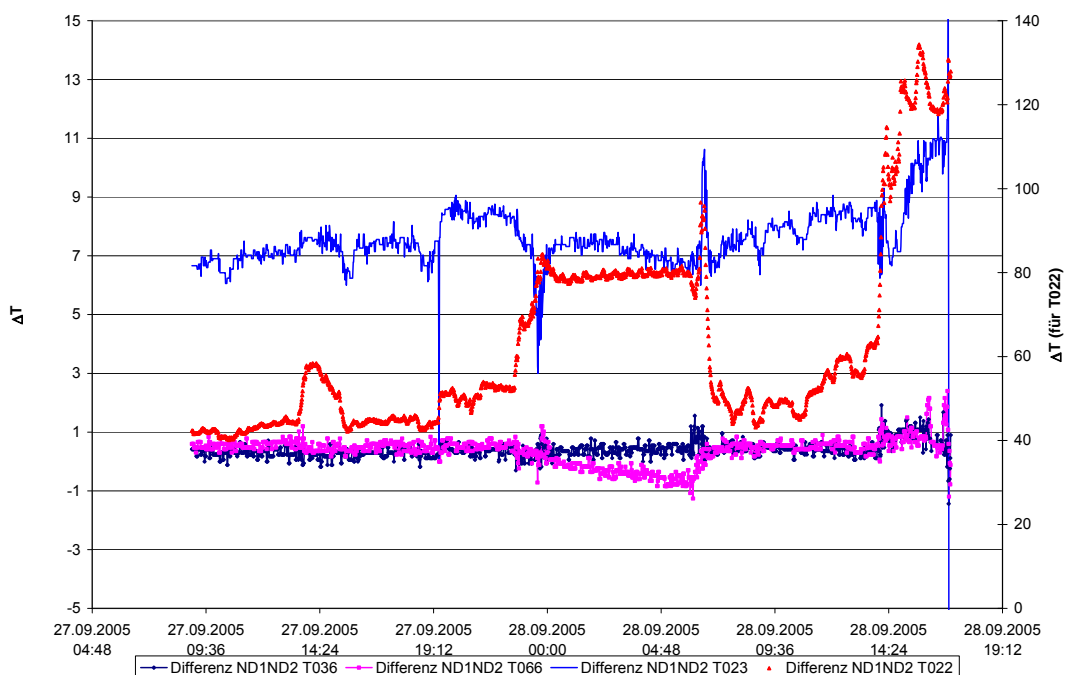


Abbildung 7-9: Änderungen der Temperaturdifferenzen

Wie aus Abbildung 7-10 ersichtlich, unterliegen die Änderungen beim Differenzdruck ausschließlich statistischen Schwankungen und lassen keinen Rückschluss auf einen sich anbahnenden Fehler zu.

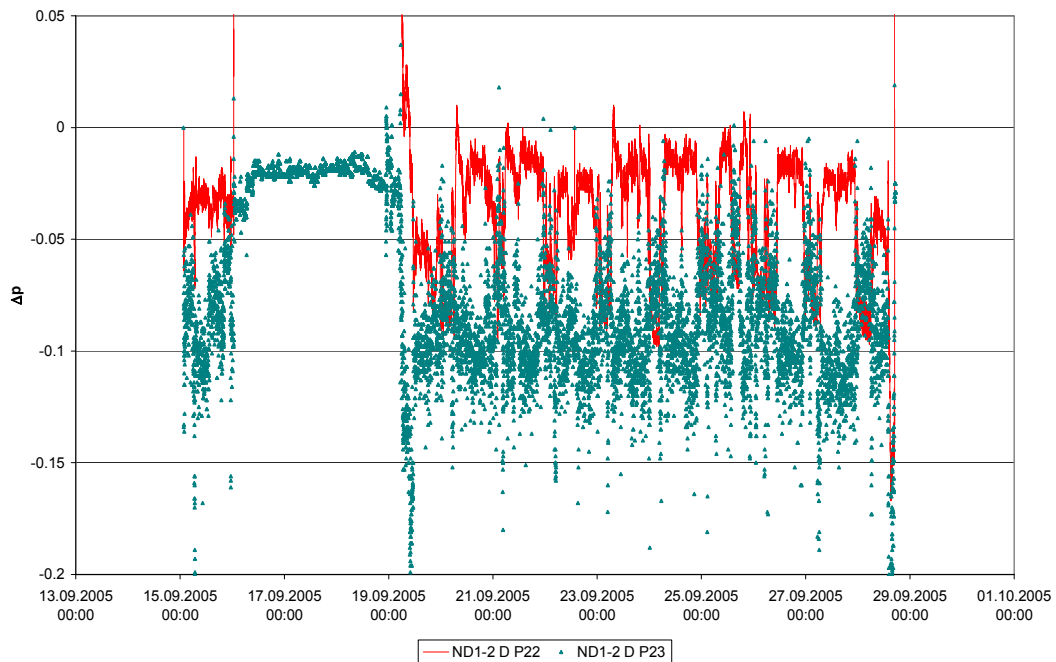


Abbildung 7-10: Änderungen der Druckdifferenzen

Die Auswertung der Messungen zeigte, dass eine Früherkennung des aufgetretenen Fehlers nicht möglich ist. Als Erklärung hierfür kann die Tatsache dienen, dass Messungen von Dampfparametern relativ starke Messunsicherheiten aufweisen und somit kleine Abweichungen, wie sie aufgrund des beschriebenen Fehlers auftreten würden, im Messrauschen „verschwinden“.

Es bleibt festzuhalten, dass es sich bei dem aufgetretenen Fehler um einen Spontausfall gehandelt hat, der sich im Vorfeld in den gemessenen Daten nicht angekündigt hat. Es sind keine signifikanten Muster aus den vorhandenen Messdaten zu extrahieren, die es ermöglichen, den sich anbahnenden Fehler innerhalb einer Zeitspanne, die die Einleitung von Gegenmaßnahmen ohne Schnellabschaltung der Turbine ermöglicht, zu detektieren.

7.3 Referenzfall „Lagerschaden“

Rotierende Maschinen stellen einen Großteil der im Kraftwerk genutzten Maschinen dar. Alle rotierenden Maschinen besitzen Lager, deren Funktion maßgeblich für ihr Funktionieren ist.

Bei großen Maschinen wie z.B. Turbinen ist eine Lagerüberwachung bereits seit langem etabliert. Hierbei erfolgt meist eine Schwingungsüberwachung. Da dies nicht für alle Maschinen ökonomisch sinnvoll ist, werden mobile Messgeräte für die Messung an Maschinen eingesetzt. Die Auswertung der Daten erfolgt über Software Tools, wobei die Zustandsermittlung für die Lager durch Überwachung von Grenzwerten erfolgt. Basierend auf der entwickelten Methode soll hier versucht werden, sich anbahnende Schäden unter Ausnutzung der zur Verfügung stehenden Messdaten zu identifizieren, bevor die festgelegten Grenzwerte erreicht werden.

7.3.1 Technische Systemanalyse

Im vorliegenden Fall werden die Lager durch eine nicht kontinuierliche Stoßimpulsüberwachung überwacht, die mit mobilen Handgeräten durchgeführt wird.

Lager erzeugen im Betrieb kontinuierliche und transiente Schwingungen, deren Größe und Frequenz von Betriebsbedingungen und Lagerzustand abhängen. Die kontinuierlichen Schwingungen enthalten alle Eigenfrequenzen des Systems mit den zugehörigen Harmonischen. Sie dominieren im Leistungsspektrum. Hierbei ist zu beachten, dass die Amplituden der vom Lager an sich erzeugten Schwingungen klein sind. Der Großteil der erzeugten Schwingungen werden durch die Last und von außen einwirkende Einflüsse erzeugt. Die kontinuierlichen Schwingungen des Lagers stehen dabei im festen Bezug zur Drehzahl, werden aber zum Teil von den Schwingungen des Gesamtssystems überlagert.

Die transienten Schwingungen werden durch Unregelmäßigkeiten beim Abrollen der Lagerelemente in den Laufflächen hervorgerufen. Hierdurch werden elastische Wellen im Ultraschallbereich erzeugt. Die im Lager erzeugten Stoßimpulse haben viele Ursprünge und führen zu einer Vielzahl von Signalmustern in den auftretenden Frequenzen. Zufällige Wechselwirkungen von Unregelmäßigkeiten (keine Schäden sondern materialbedingte Unregelmäßigkeiten in der Lagerstruktur) erzeugen die so genannte Grundlinie (Grundrauschen) schwacher Stoßimpulse. Abriebteilchen oder Verschmutzungen verursachen zufällige Stoßimpulse größerer Stärke und niedriger Frequenzen. Diese werden über die so genannte „High Rate“ erfasst.

Lagerschäden wie z. B. Abplatzungen, Vertiefungen oder Lagerbrüche verursachen anfangs periodische Impulse mit festen Intervallen, die jedoch beim Fortschreiten des Schadens zunehmend unregelmäßiger werden. Die zugehörigen Frequenzen werden über die „Low Rate“ erfasst.

Da eine nicht kontinuierliche Messung durchgeführt wird, können die periodischen Impulse nicht erfasst und somit nicht ausgewertet werden. Dies stellt eine Schwierigkeit für die Früherkennung dar, da nur eine begrenzte Anzahl von Werten zu Verfügung steht. Hier sollte sich jedoch das entwickelte Verfahren bewähren, weil es die Messwerte als Muster identifiziert und so sich anbahnende Störungen anzeigen kann, bevor die standardmäßig verwendete Software eine Grenzwertüberschreitung meldet.

7.3.2 Anwendung des neuronalen Netzes

Die Lager werden in regelmäßigen Abständen nachgeschmiert oder, wenn aufgrund der Messung eine unzureichende Schmierung vermutet wird. Unzureichende Schmierung kann Lagerschäden hervorrufen. Dabei können die Messwerte einen Lagerschaden nahe legen, da hierbei die „High Rate“ ansteigt. Dies galt es zu berücksichtigen. Im Fall unzureichender Schmierung wird sich nach erfolgter Schmierung der „high rate“ Wert wieder normalisieren.

Für die Früherkennung stehen als Ergebnis der technischen Systemanalyse die in Tabelle 7-5 aufgeführten Messwerte zu Verfügung.

Messwerte	Beschreibung
Low Rate (dBLR)	Messwerte mit kleiner Anzahl hoher Impulse
High Rate (dBHR)	Messwerte mit hoher Anzahl schwacher Impulse
Gesamtschwingung horizontal	Schwingungsmessung nach ISO 2372 in horizontaler Achsrichtung
Gesamtschwingung vertikal	Schwingungsmessung nach ISO 2372 in vertikaler Achsrichtung
Differenz ΔbB	Verhältnis von „High Rate“ zu „Low Rate“

Tabelle 7-5: Übersicht über vorhandene Messwerte

Da die Differenz $\Delta db(HR/LR)$ keine zusätzliche Information darstellt, wird in unterschiedlichen Versuchen untersucht, ob sich eine Verbesserung der Früherkennung feststellen lässt, wenn sie einbezogen wird. In Ergänzung der oben angeführten Messwerte wurden die Wartungsprotokolle der jeweiligen Lager ausgewertet. Damit können den jeweiligen Messwerten die tatsächlichen Schadensformen sowie die Beobachtungen des Personals zugeordnet werden. Dies ermöglicht es, bei der Zusammenstellung der Trainingssätze für die Früherkennungsmethode, die Erfahrung des Personals einzubeziehen. Im Gegensatz dazu wird durch die üblicherweise verwendete Software ausschließlich der „high rate“ Wert und der Differenzwert auf Grenzwertüberschreitung überwacht sowie in Zwischenbereichen über eine Ampelfunktion eine Warnfunktion realisiert.

Der Trainingssatz ohne Berücksichtigung von ΔdB wird in Tabelle 7-6 gezeigt. Es wurde einer der drei aufgetretenen Schäden zum Training verwendet. Da

hier mehrere Messungen durchgeführt wurden, ergibt sich ein Fehlermuster unterschiedlicher Ausprägung, was mit den Messpunkten „intakt/Übergang“ zu einer sauberen Abgrenzung des Fehlerzustandes führt und so eine Erkennung des sich anbahnenden Fehlers ermöglicht.

Lager SAC03	Datum	Zeit	LR	HR	Vib_H	Vib_V
defekt	27.08.2008	08:09:47	48	42	2,79	2,61
	27.08.2008	09:18:40	47	41	2,53	2,37
	27.08.2008	10:59:54	47	42	3,16	3,14
	17.09.2008	09:09:19	45	41	4,31	6,68
	29.09.2008	09:19:59	43	40	2,87	4,42
intakt/Übergang	13.12.2007	09:42:31	32	26	2,09	2,2
Übergang	12.06.2008	13:09:29	38	33	2,78	2,80
intakt	09.12.2004	13:20:00	27	23	2,50	2,80
	09.12.2004	10:02:00	29	25	2,64	2,51
	09.12.2004	10:02:00	29	25	2,64	2,51
	06.10.2004	13:21:00	26	21	3,68	2,95

Tabelle 7-6: Trainingssatz Lager SAC03

Anhand des Trainingssatzes wurden die Eingangsgrößen des neuronalen Netzes festgelegt. Somit besteht die Eingabeschicht des Netzwerkes aus vier Neuronen.

Validierungssatz SAC03	Datum	Zeit	LR	HR	Vib_H	Vib_V
defekt	12.02.2009	07:50:37	48	42	3,51	5,73
	11.02.2009	11:31:06	48	42	3,30	3,58
	11.02.2009	10:01:06	45	41	3,34	4,54
Übergang	17.06.2008	09:59:13	40	36	3,19	4,01
intakt	09.12.2004	10:02:00	29	25	2,64	2,51

Tabelle 7-7: Validierungssatz für Training SAC 03

In den Versuchsdurchläufen wurden verschiedene Netzwerkstrukturen (Anzahl der Neuronen der verdeckten Schichten, Anzahl der verdeckten Schichten) sowie verschiedene Trainingsalgorithmen (siehe Kapitel 8.3) verwendet und die Ergebnisse mit Hilfe des Validierungsdatensatzes (Tabelle 7-7) verglichen.

Als Validierungssatz wurden Messwerte zu einem weiteren aufgetretenen Fehler sowie Messwerte ohne Befund eingesetzt. Die Messwerte, die mit „Übergang“ gekennzeichnet sind, wurden im Trainingssatz als „gut“-Werte eingesetzt, um die Wahrscheinlichkeit für eine fehlerhafte Zuordnung herabzusetzen.

Zur Verifizierung des Netzwerkes wurde die Gesamtheit der Daten für das Lager SAC03 eingesetzt. Abbildung 7-11 zeigt die Ausgabe des trainierten Netzes. Es zeigt klar die drei aufgetretenen Schäden. Deutlich zu erkennen sind die ansteigende Fehlerwahrscheinlichkeit für einen Fehler im Zeitraum Mitte 2008. Hier sind jedoch durch die vorhandene Software keine Fehlermeldungen dokumentiert worden.

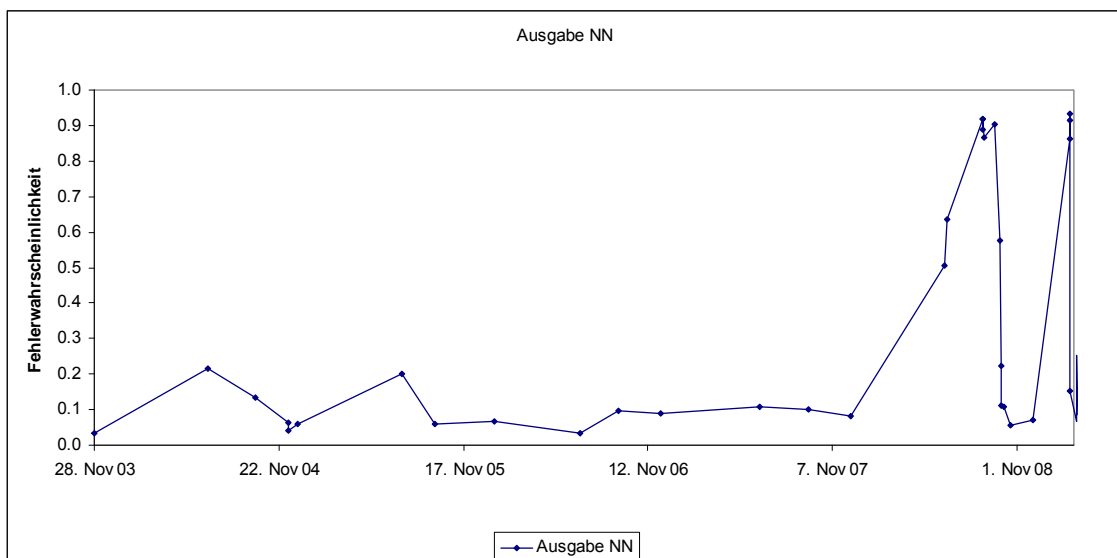


Abbildung 7-11 Ergebnis Verifizierung Lager SAC03

In einem weiteren Rechenlauf wurde die Differenz ΔdB als zusätzliche Eingangsgröße für das neuronale Netz implementiert. Dementsprechend wurde die Anzahl der Eingangsneuronen auf fünf erhöht und äquivalent zur vorangegan-

genen Versuchsreihe eine geeignete Netzstruktur und erfolgreicher Trainingsalgorithmus iterativ ermittelt. Der Trainingssatz ist in Tabelle 7-8 aufgeführt.

Lager	Datum	Zeit	LR	HR	Δ LR/HR	Vib_H	Vib_A
defekt	27.08.2008	08:09:47	48	42	6	2,79	2,61
	27.08.2008	09:18:40	47	41	6	2,53	2,37
	27.08.2008	10:59:54	47	42	5	3,16	3,14
	17.09.2008	09:09:19	45	41	4	4,31	6,68
	29.09.2008	09:19:59	43	40	3	2,87	4,42
intakt/ Übergang	13.12.2007	09:42:31	32	26	6	2,09	2,20
Übergang	12.06.2008	13:09:29	38	33	5	2,78	2,80
intakt	09.12.2004	13:20:00	27	23	4	2,50	2,80
	09.12.2004	10:02:00	29	25	4	2,64	2,51
	09.12.2004	10:02:00	29	25	4	2,64	2,51
	06.10.2004	13:21:00	26	21	5	3,68	2,95

Tabelle 7-8: Trainingssatz für SAC03 unter Berücksichtigung der Different Δ dB
Entsprechend ergibt sich als Validierungssatz; Tabelle 7-9:

Validierungs- satz	Datum	Zeit	LR	HR	Δ LR/HR	Vib_H	Vib_A
defekt	12.02.2009	07:50:37	48	42	6	3,51	5,73
	11.02.2009	11:31:06	48	42	6	3,30	3,58
	11.02.2009	10:01:06	45	41	4	3,34	4,54
Übergang	17.06.2008	09:59:13	40	36	4	3,19	4,01
intakt	09.12.2004	10:02:00	29	25	4	2,64	2,51

Tabelle 7-9: Validierungssatz für Training SAC 03 mit Δ dB

Die Datensätze sind bis auf die Berücksichtigung der Größe Δ dB identisch. So kann der Einfluss der zusätzlichen Größe am besten eingeschätzt werden. Aus Abbildung 7-12 wird ersichtlich, dass die Berücksichtigung der Differenzgröße die Ausgabe des Netzes deutlich beeinflusst. Kleinere Anstiege im Netzausgang werden gedämpft, wohingegen stärker werdende Anzeichen auf einen

Fehler besonders verstärkt werden. Dies kann sich durchaus positiv auswirken, da somit sich anbahnende Fehler frühzeitig angezeigt werden. Es besteht jedoch auch die Gefahr, dass sich fehlerhafte Zuordnungen häufen, da z. B. wegen der angesprochenen Nachschmierung fehlerähnliche Muster als Fehler interpretiert werden.

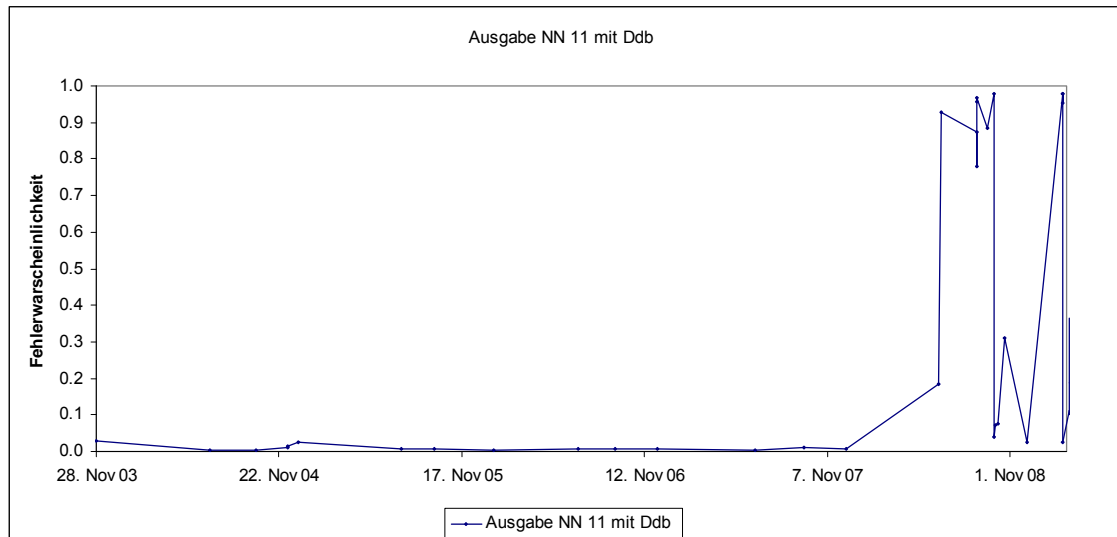


Abbildung 7-12 Ergebnis der Verifizierung Lager SAC03 mit Berücksichtigung der Messgröße ΔdB

In den bislang dargestellten Rechnungen wurden Übergangszustände dem Gutzustand zu geordnet. Um den Einfluss der Daten der Übergangszustände zu untersuchen, wurden in der folgenden Analyse die Übergangszustände als „Teilfehler“ wie in Tabelle 7-10 gezeigt.

7 Anwendungen

Lager SAC03	Datum	Zeit	LR	HR	Vib_H	Vib_V	Zustand
defekt	27.08.2008	08:09:47	48	42	2,79	2,61	1
defekt	27.08.2008	09:18:40	47	41	2,53	2,37	1
defekt	27.08.2008	10:59:54	47	42	3,16	3,14	1
defekt	17.09.2008	09:09:19	45	41	4,31	6,68	1
defekt	29.09.2008	09:19:59	43	40	2,87	4,42	1
intakt/Übergang	13.12.2007	09:42:31	32	26	2,09	2,2	0,6
Übergang	12.06.2008	13:09:29	38	33	2,78	2,8	0,8
intakt	09.12.2004	13:20:00	27	23	2,5	2,8	0
intakt	09.12.2004	10:02:00	29	25	2,64	2,51	0
intakt	09.12.2004	10:02:00	29	25	2,64	2,51	0
intakt	06.10.2004	13:21:00	26	21	3,68	2,95	0

Tabelle 7-10: Trainingssatz mit Berücksichtigung von Übergangszuständen

Hierbei wurden im Gegensatz zu den vorangegangenen Rechnungen die Übergangszustände nicht dem Gutzustand, also Sollgröße 0, zugeordnet, sondern erhielten eine Abstufung 60 bzw. 80% Fehler.

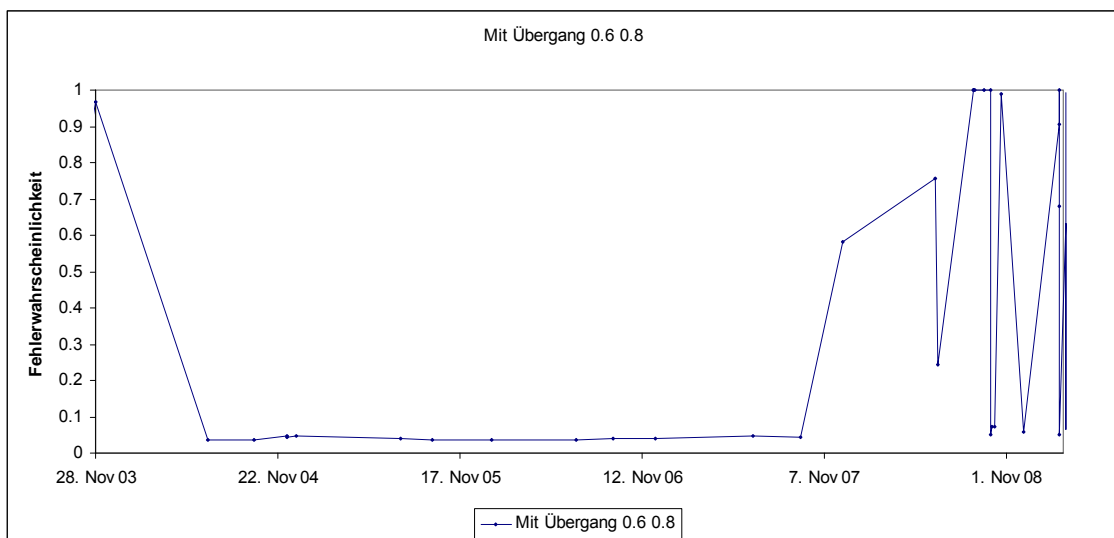


Abbildung 7-13 Ergebnis der Verifizierung unter Berücksichtigung der Übergangszustände

Wie in der Abbildung 7-13 zu erkennen, findet auch unter Berücksichtigung von Übergangszuständen eine starke Überzeichnung statt. Kleinere Ausschläge werden fast vollständig unterdrückt, wohingegen fehlerähnliche Muster schnell und ausgeprägt dem Fehler zugeordnet werden. Es empfiehlt sich daher Zwi-

schenzustände nicht zu implementieren, um unzutreffende Fehlermeldungen zu vermeiden und so die Trennschärfe zu erhöhen.

Ein wichtiger Gesichtspunkt einer erfolgreichen betrieblichen Anwendung der Fehlerfrüherkennung liegt in der Übertragbarkeit der Ergebnisse auf baugleiche Anlagenteile. Eine Vielzahl von Komponenten und Anlagenbereichen in Kraftwerken sind redundant ausgelegt oder mehrfach vorhanden. Ein wichtiger Schritt ist daher, die Ergebnisse übertragen zu können und so die Fehlerfrüherkennung auch in den Anlagenteilen anwenden zu können, in denen bislang keine Fehler gleicher Art aufgetreten sind.

Im vorliegenden Beispiel ist ein baugleiches Anlagenteil vorhanden, und die entsprechenden Messdaten liegen vor. Auf diesen Datensatz wurde das trainierte neuronale Netz angewendet. Somit stellt dies den Anwendungsfall dar, bei dem in das Netz unbekannte, aber ähnliche Daten eingegeben werden. Abbildung 7-14 zeigt die Ergebnisse.

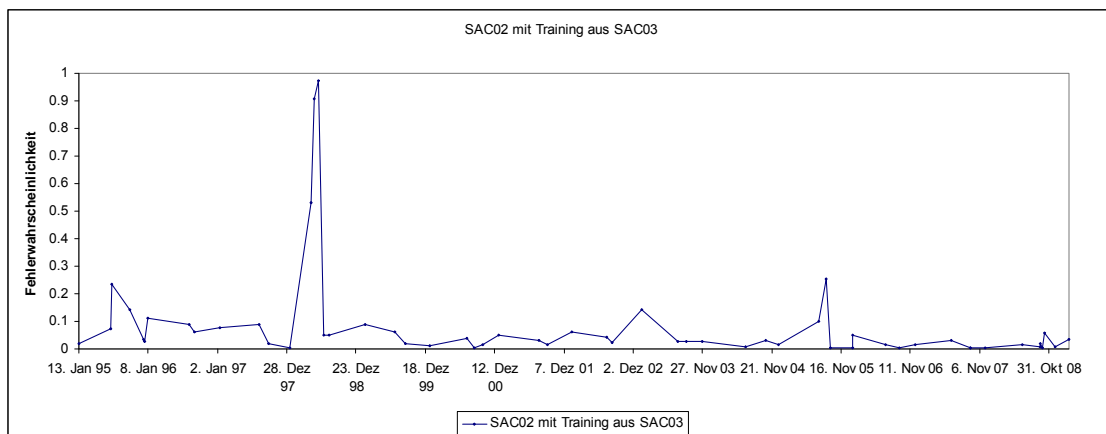


Abbildung 7-14: Anwendung des Datensatz aus SAC 02 auf trainiertes Netz

Ein Fehler im Frühjahr 1998 wird eindeutig angezeigt. Aus der vom Kraftwerk übergebenen Dokumentation ergab sich für den angegebenen Zeitraum jedoch kein Schaden; lediglich eine Sondermessung auf Grund einer Störungsmeldung wurde vermerkt.

Bei der Suche nach Ursachen für diesen durch das Netz angezeigten Fehler, der wie die Abbildung 7-14 zeigt, über drei Messungen anhält, erfolgte eine Rücksprache mit den Verantwortlichen des Kraftwerkes. Hierbei stellte sich heraus, dass in diesem Zeitraum die Anlage unter besonderer Beobachtung stand und daher mehrere Messungen durchgeführt wurden. Letztendlich wurde im Zuge eines geplanten Wechsels das Lager ausgetauscht, ohne dass der Befund des aufgetretenen Lagerschadens dokumentiert wurde.

Damit konnte nicht nur die Netzausgabe erklärt werden, sondern es zeigt sich, dass mit Hilfe der Früherkennung und unter Berücksichtigung der Erfahrung aus den anderen Lagern der entsprechende Fehler deutlich früher hätte diagnostiziert werden können. Das Verfahren zeigt somit klar den sich anbahnenden Fehler an. Gleichzeitig können kleinere, auf Grund mangelnder Schmierung auftretende Ausschläge eindeutig zugeordnet und von tatsächlich auftretenden Fehlern unterschieden werden.

7.4 Referenzfall „Wärmeübertrager“

Wärmeübertrager sind die Bauteile, welche die Energieübertragung im Kraftwerk gewährleisten. Ihr fehlerfreies Funktionieren ist unabdingbar, damit Kraftwerke ihre Aufgabe erfüllen können. Wärmeübertrager sind den unterschiedlichsten Umgebungsbedingungen ausgesetzt. Neben Druck und Temperatur sind hier die Belastung durch radioaktive bzw. aggressive Medien zu nennen. In verfahrenstechnischen Anlagen ist die Anzahl der Einsatzstoffe noch breiter, was das Beanspruchungsprofil der Bauteile erhöhen kann.

Häufiger Grund für den Ausfall des Wärmeübertragers ist hierbei ein Zusetzen der Wärmeübertragungsflächen. Dadurch wird der Übergang der Wärme gemindert. Dies wird zu Störungen führen, falls zu spät bemerkt wird.

7.4.1 Systemanalyse regenerativer Luftvorwärmer Bauart „Rothemühle“

Bei dem betrachteten Aggregat handelt es sich um einen regenerativen Luftvorwärmer (LuVo) der Bauart „Rothemühle“. Dieser Zweistrom-Luftvorwärmer nutzt die im Rauchgas enthaltene Wärmeenergie, um die im Gegenstrom strömende Luft zu erwärmen (vgl. Abbildung 7-15). Der Wärmeübergang erfolgt nach dem regenerativen Wärmeaustauschprinzip. Das Bauartprinzip „Rothemühle“ heißt hier, dass es sich um eine feststehende Speichermasse handelt. Die alternierende Beaufschlagung mit heißem Rauchgas und Frischluft des Speichermediums wird über sich drehende Hauben realisiert. Das Rauchgas gibt hierbei die Wärmeenergie an ein Heizblechpaket (Speichermasse) ab, nachdem die Haube weitergedreht ist, strömt die Frischluft über das Speichermedium und wird aufgewärmt. Dabei ist das Ziel, bei geringem Druckverlust einen hohen Wärmeübergang zu gewährleisten. Gleichzeitig wird der kontinuierliche Strom beider Medien gewährleistet.

Die benötigte Luft wird einerseits aus dem Freien und andererseits aus dem Kesselhaus durch ein Axialgebläse angesaugt. Zum Anfahren und im Betrieb wird

die Luft vor dem Einströmen in den regenerativen Luftvorwärmer durch einen Dampfvorwärmer so weit vorgewärmt, dass eine Unterschreitung des Taupunktes auf der Kaltheizfläche vermieden wird. Anschließend wird der Luftstrom im Stator in Primär- und vom Sekundärstrom geteilt. Hiermit kann eine Temperaturstufung

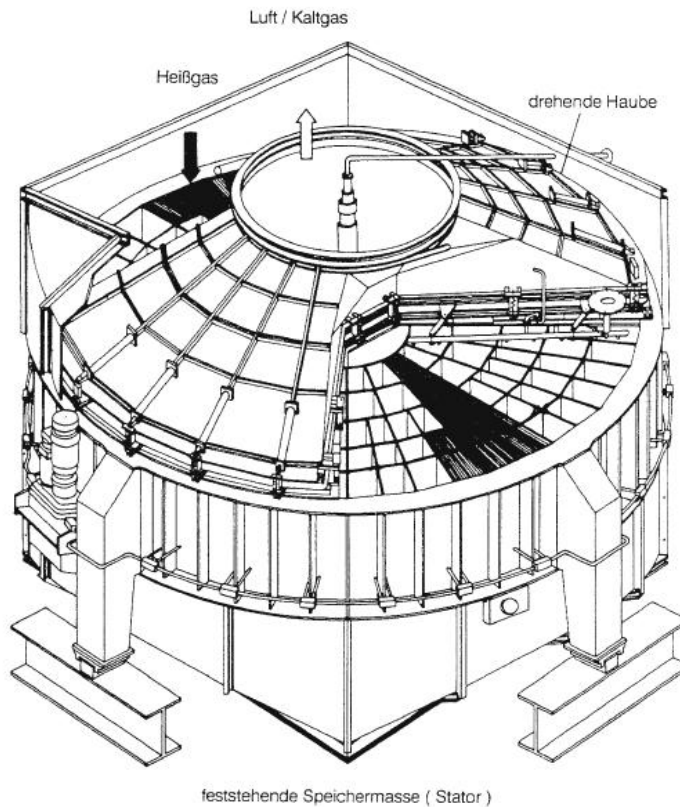


Abbildung 7-15: Regenerativer Luftvorwärmer „Rothemühle“

zwischen dem Primärstrom, der den Kohlemühlen zugeführt wird, und dem Sekundärstrom, der den Brennern zugeführt wird, erreicht werden. Über einen Bypass wird ein Teilstrom der Luft am Dampf- und Rauchgas-LuVo vorbeigeführt. Dieser Teilstrom fördert die (Mühlen-)Kaltluft zur Regelung der Sichter-temperatur (vgl. Abbildung 7-16).

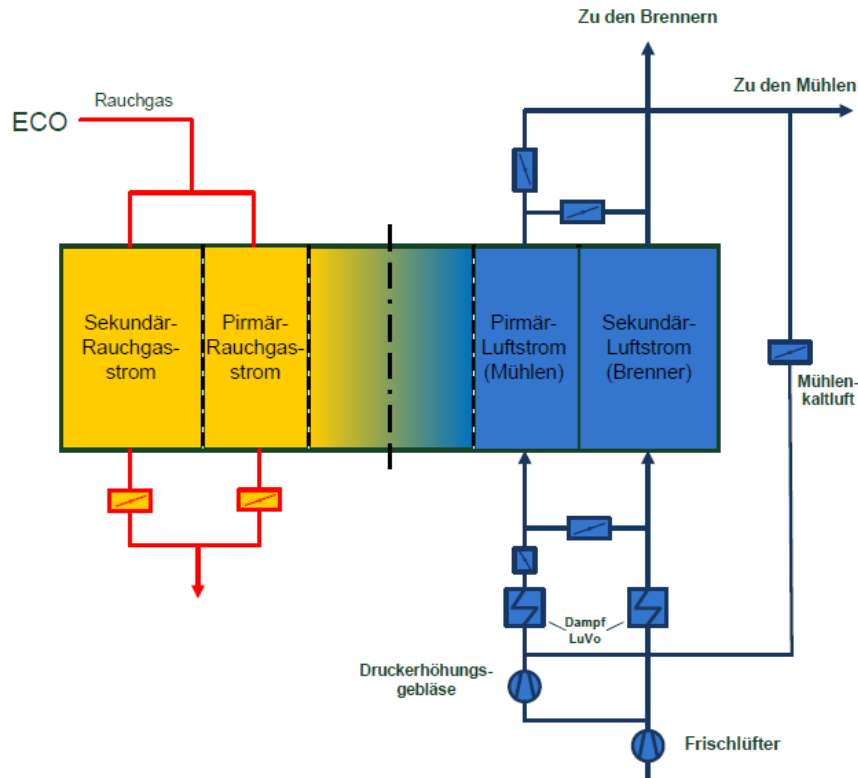


Abbildung 7-16: Schematische Darstellung des Luftvorwärmers

Der Primärluftstrom wird zur Trocknung der Rohkohle und gleichzeitig zum Transport des Kohlenstaubs eingesetzt. Zur Überwindung der Strömungswiderstände in den Mühlen und den Staubleitungen dient ein laufschaufelgeregeltes Axialgebläse (Primärlüfter). Aufgrund des vom Primärlüfter erzeugten Drucks liegen die Mühlen und die Zuteiler im Überdruckbereich. Nachdem die Luft im Luvo aufgeheizt wurde, wird sie den 4 Mühlen zugeführt.

Der Sekundärluftstrom wird über 2 Verteilerkanäle zu den 16 Brennergruppen und zu 4 „over fire air“ Leitungen gefördert.

Auf der Rauchgasseite ist der Luvo als letzte Heizfläche direkt hinter dem Economiser angesiedelt. Der Rauchgasstrom wird ebenfalls in Primär- und Sekundärstrom aufgeteilt, um verschiedene Temperaturen auf der Luftseite zu erreichen.

Die Zustandsüberwachung des LuVos erfolgt über die luftseitige Differenzdruckmessung Δp . Aufgrund unterschiedlicher Fahrweisen der Anlage, bedingt

durch Veränderungen in den Lastanforderungen und unterschiedliche Kohlequalität, lässt sich der Zustand des Wärmeübertragers mit Hilfe eines einzelnen Parameters nicht eindeutig zuordnen. Es erfolgen zum Teil Überschreitungen des normalen Differenzdruckes, ohne dass eine Verschmutzung vorliegt, und es kann vorkommen, dass eine Verschmutzung vorliegt, obwohl kein erhöhter Differenzdruck festgestellt wird. Es ist daher notwendig, durch die Berücksichtigung mehrerer Parameter eine eindeutige Zustandsbestimmung für den LuVo zu erhalten. Hier bietet sich das Verfahren der Früherkennung an, da durch Erstellen von Trainingssätzen eine eindeutige Zuordnung von Anlagenparametern und den Zuständen „intakt“ bzw. „Fehler“ des Luvos möglich ist. So kann aus den vorhandenen Messdaten ein entsprechendes Fehlermuster extrahiert werden und durch das neuronale Netzwerk erlernt werden.

7.4.2 Merkmalsextraktion

Mit Hilfe der nachfolgend aufgeführten Parameter soll die Zustandsüberwachung aufgrund der technischen Systemanalyse eindeutig möglich sein. Sicherheitsgefährdende Anlagenzustände, wie zum Beispiel der bereits aufgetretene Fall, dass sämtliche Regelapparaturen maximal aufgefahren waren, da ein unzulässig hoher Verschmutzungsgrad des Wärmeübertragers nicht erkannt wurde, sollen so vermieden werden. Für die Untersuchung berücksichtigt wurden die folgenden Zustände:

- normaler Differenzdruck als Referenzzustand (Zustand 1)
- erhöhter Differenzdruck bei Verschmutzung (Zustand 2)
- erhöhter Differenzdruck ohne Verschmutzung (Zustand 3)

Der Referenzzustand wird bei Nennlast definiert. Aufgrund der Erfahrungen aus dem Referenzfall „Zwischenüberhitzer“ werden Teillastzustände nicht berücksichtigt, da es hierbei zu nicht eindeutig zuzuordnenden Signalen kommt. Nach Aussagen des verantwortlichen Kraftwerkspersonals wird bis zu einem Differenzdruck von 35 mbar über dem luftseitigen Primärteil des LuVos vom verschmutzungsfreien Zustand ausgegangen.

Bei verschmutztem LuVo muss gegen den erhöhten Wärmedurchgangswiderstand geregelt werden, um die Führungsgröße Sichtertemperatur halten zu können. Es wird daher der Luftvolumenstrom durch den Primärteil des LuVos erhöht, was durch eine Stellungsänderung der Trimmklappen erreicht wird. Um eine Erhöhung des Volumenstroms zu erreichen, muss die Leistung der Gebläse gesteigert werden. Durch den verringerten Wärmeübergang im Wärmeübertrager ist von Änderungen der Rauchgastemperaturen auszugehen.

Zustand 3 tritt auf wenn die Kohle, aufgrund eines hohen Feuchtegehalts stärker getrocknet werden muss. Dafür muss der Primärluftstrom zur Mühle erhöht werden. Um dies zu erreichen, müssen Trimmklappen und Gebläse entsprechend geregelt werden. Dies führt zu einem erhöhten Volumenstrom im LuVo, woraus ein größerer Differenzdruck resultiert.

Die Fehlerfrüherkennung muss daher in der Lage sein, zwischen den Zuständen „2“ und „3“ differenzieren zu können. Da es nicht sinnvoll erscheint, die Zustände 1 und 3 zusammenzufassen, war zu klären, wie das Netz und das Netzwerktraining entsprechend gestaltet werden.

Aus der Systemanalyse ergab sich eine Vielzahl von gemessenen Parametern, die als geeignet erachtet wurden. Abbildung 7-17 zeigt eine Übersicht. Anhand dieser Zusammenstellung wurde für jeden Parameter in Zusammenarbeit mit dem Kraftwerkspersonal ihre prognostizierte Änderung sowie deren Verfügbarkeit (wird der Parameter permanent gemessen) zusammengestellt. Die vorhandenen und sich unterscheidenden Parameter sind in Abbildung 7-18 zusammengestellt.

7 Anwendungen

Allgemein:		Betriebszustände		
		1	2	3
1	Lastzustand des Blockes	-	-	-
2	Feuchte der Kohle (Eigenfeuchte/ Gesamtwasseranteil)	-	-	↑
3	Außentemperatur	-	-	↓
Luvo:				
4	Mittlere Blechtemperatur (Sek)	-	-	-
5	Mittlere Blechtemperatur (Pri)	-	-	↓
6	Δp Primär Luft	-	↑	↑
7	Δp Sekundär Luft	-	↑	-
8	Temperatur (Pri) Luft vor Luvo	-	-	-
9	Temperatur (Pri) Luft nach Luvo	-	-	-
10	Temperatur (Sek) Luft vor Luvo	-	-	-
11	Temperatur (Sek) Luft nach Luvo	-	-	-
12	Temperatur (Pri) Rauch vor Luvo	-	-	-
13	Temperatur (Pri) Rauch nach Luvo	-	↑	-
14	Temperatur (Sek) Rauch vor Luvo	-	-	-
15	Temperatur (Sek) Rauch nach Luvo	-	↑	-
16	Trimmklappenstellung Rauch (Sek) nach Luvo	-	↓	↓
Mühlen:				
17	Sichtertemperatur	-	-	-
18	Δp über die Mühlen	-	-	↑
19	Zuteilerdrehzahl	-	-	-
20	Mühlenlüfter Leistung	-	↑	↑
21	Mühlenlüfter Volumenstrom	-	-	↑
22	Volumenstrom Heißluft	-	↑	↑
23	Volumenstrom Kaltluft	-	↓	↓
Frischlüfter:				
24	Frischlüfter-Volumenstrom	-	-	-
25	Frischlüfterleistung	-	-	-
Saugzug:				
26	Saugzug-Volumenstrom	-	↓	↓
27	Saugzugleistung	-	↑	↑

Abbildung 7-17: Zusammenstellung der als relevant erachteten Parameter

7 Anwendungen

	Allgemein:	Betriebszustände		
		1	2	3
2	Feuchte der Kohle (Eigenfeuchte/ Gesamtwasseranteil)	-	-	↑
3	Außentemperatur	-	-	↓
	Luvo:			
5	Mittlere Blechtemperatur (Pri)	-	-	↓
6	Δp Primär Luft	-	↑	↑
7	Δp Sekundär Luft	-	↑	-
9	Temperatur (Pri) Luft nach Luvo	-	-	-
13	Temperatur (Pri) Rauch nach Luvo	-	↑	-
15	Temperatur (Sek) Rauch nach Luvo	-	↑	-
16	Trimmklappenstellung Rauch (Sek) nach Luvo	-	↓	↓
	Mühlen:			
17	Sichtertemperatur	-	-	-
18	Δp über die Mühlen	-	-	↑
19	Zuteilerdrehzahl	-	-	-
20	Mühlenlüfter Leistung	-	↑	↑
21	Mühlenlüfter Volumenstrom	-	-	↑
22	Volumenstrom Heißluft	-	↑	↑
23	Volumenstrom Kaltluft	-	↓	↓
	Frischlüfter:			
24	Frischlüfter-Volumenstrom	-	-	-
25	Frischlüfterleistung	-	-	-
	Saugzug:			
26	Saugzug-Volumenstrom	-	↑	↑
27	Saugzugleistung	-	↑	↑

Abbildung 7-18: Auswahl der verfügbaren und relevanten Parameteränderungen

Zur Kontrolle der theoretischen Überlegungen wurden die Messwertverläufe der Parameter vom Kraftwerk angefordert, um die prognostizierten Änderungen entsprechend überprüfen zu können. Hierfür wurden für zwei Zeiträume vor dem Fehler und zwei Zeiträume während des Fehlers durch Abgleich der Entwicklung der Mittelwerte die tatsächlich aufgetretenen Änderungen unter Nennlast untersucht. Die Entwicklung der Mittelwerte für die einzelnen Parameter sind in Tabelle 7-11 dargestellt.

7 Anwendungen

	Parameter					
	6	7	9	13	15	Durchs.16
	PD Luft über Prim.-Luvo 10NG06P950. XQ01	PD Luft über Sek.-Luvo 10NG00P950. XQ01	DEA T LFT H LUVO PRIM-STR 10NG06T006. XQ01	DEA T RG HI LUVO PRIM-STR 10NR06T002. XQ01	DEA T RG HI LUVO SEK-STR 10NR05T001. XQ01	Kl. 2h RG-Luvo sek. 10NR05S002. XQ01
Mittelwert Gutbereich 28.08.08	32,94	36,54	304,14	132,93	145,23	36,40
Mittelwert Gutbereich 29.08.08	31,64	36,09	317,31	131,86	147,09	18,84
Mittelwert Fehler 21.10.08	37,64	46,06	320,57	127,70	153,16	35,29
Mittelwert Fehler 22.10.08	41,55	46,27	317,71	122,27	147,92	31,62
	Durchs. 17	Durchs. 18	Durchs. 19	20	21	24
	T Ko-Stb Mühle-Sichter 10NL40T001. XQ01	DEA PD MUE-LFT/SICHT 10NG09P003. XQ01	Zuteiler-Drehzahl 10NL29C011. XQ06	Wirkleistung Mühlen.-Geb. 10NG20E112. XQ01	F Luft Mue-Lft-Geb. 10NG20F901. ZQ01	F Luft FrischLüfter 10NG10F901. XQ01
	Mittelwert Gutbereich 28.08.08	86,77	54,47	78,47	2340,47	517,74
Mittelwert Gutbereich 29.08.08	88,56	53,74	78,74	2308,38	512,73	3991,86
Mittelwert Fehler 21.10.08	84,95	59,17	81,56	2189,65	503,94	4050,13
Mittelwert Fehler 22.10.08	86,19	60,02	82,12	2168,11	499,10	4007,53
	25	26	27			
	Wirkleistung Frischlüfter 10NG10E112. XQ01	F RG hi E-Filter1+2 10NR10F902. XQ01	Wirkleistung Saugzug 10NR10E112. XQ01			
	Mittelwert Gutbereich 28.08.08	6530,42	2752,43	7796,85		
Mittelwert Gutbereich 29.08.08	6507,78	2749,67	8149,71			
Mittelwert Fehler 21.10.08	7480,13	2735,34	8767,22			
Mittelwert Fehler 22.10.08	7280,64	2720,42	8828,61			

Tabelle 7-11 Mittelwerte für Prozessparameter bei fehlerbehaftetem und fehlerfreiem Nennlastbetrieb

Durch den Abgleich zwischen prognostizierten Parameteränderungen und tatsächlich aufgetretenen Änderungen der Messwerte können die für die Fehlerfrüherkennung geeigneten Größen ausgewählt werden.

Die Größe „Feuchte der Kohle“ ist ein Laborwert, der dementsprechend nicht kontinuierlich gemessen wird. Die Außentemperatur ist auf Grund der Tatsache, dass ein Großteil der Luft über das Kesselhaus angesaugt wird und somit eine von der Außentemperatur relativ unabhängige Mischtemperatur hat, zu vernachlässigen. Die mittlere Blechtemperatur stellt eine berechnete Größe dar, die aus den Rauchgastemperaturen ermittelt wird. Da die Rauchgastemperaturen in die Rechnung eingehen, wird die mittlere Blechtemperatur nicht herangezogen. Die Differenzdrücke am LuVo stellen das relevante Merkmal dar, an Hand dessen der Zustand des LuVos üblicherweise bewertet wird. Es wurde hier zuerst eine Betrachtung ohne diese Werte durchgeführt, um den Zustand ausschließlich an der Korrelation der Parameter ablesen zu können. Vergleichend wurde dann mit den Werten der Differenzdrücke gearbeitet.

Als fehlerspezifisch wurden die folgenden Größen identifiziert:

- Δp primär luftseitig
- Δp sekundär luftseitig
- Temperatur primärseitig der Luft hinter LuVo
- Temperatur primärseitig des Rauchgases hinter LuVo
- Δp der Mühlen
- Zuteilerdrehzahl
- Mühlenlüfterleistung

Für diese Daten wurden Trainings- und Verifizierungssätze zusammengestellt und die Früherkennung durchgeführt.

7.4.3 Anwendung des neuronalen Netzes

Wie bereits erläutert, wurde im ersten Schritt ein Trainingsdatensatz ohne Berücksichtigung der Differenzdrücke erstellt, Tabelle 7-12.

Zustand	Parameter				
	9	13	18	19	20
gut	302,74	129,06	55,27	79,09	2341,65
	303,62	132,72	54,49	79,06	2334,70
	304,51	132,97	54,04	79,03	2345,01
	305,39	133,22	55,03	79,00	2367,90
	306,28	133,47	54,36	78,98	2352,99
	307,16	133,72	54,73	78,95	2335,03
	308,05	133,96	54,43	78,92	2336,91
	315,88	133,59	53,66	78,56	2334,19
	314,64	134,21	53,88	78,59	2337,66
	314,64	134,82	54,89	78,64	2351,51
	314,65	134,35	55,46	78,69	2331,07
	314,65	133,76	55,72	78,75	2315,80
314,66	133,17	55,63	78,80	2319,95	
schlecht	316,82	124,71	54,96	79,65	2191,95
	322,55	129,47	58,56	82,34	2289,83
	321,41	129,17	60,83	82,49	2252,89
	320,27	128,87	60,15	82,55	2312,18
	319,13	128,64	61,55	82,39	2291,49
	317,99	128,63	60,46	82,37	2267,62
	318,67	128,63	60,03	82,62	2250,89
	327,23	124,99	60,27	82,11	2054,24
	328,87	126,99	61,75	82,58	2073,11
	326,28	126,33	62,98	82,45	2151,95
	323,68	125,67	62,59	82,08	2259,39
	321,09	124,81	60,73	81,69	2264,30
318,50	123,87	60,63	81,61	2285,99	

Tabelle 7-12: Trainingsdatensatz (5 Eingangsparameter)

Entsprechend wurde ein zweiter Satz zur Verifizierung zusammengestellt (Tabelle 7-13). Es wurde jeweils aus beiden ausgewählten Zeiträumen für Gut- und Fehlerzustand ein Teil der Daten für das Training, der andere Teil für die Validierung verwendet.

7 Anwendungen

Zustand	Parameter				
	9	13	18	19	20
gut	308,93	134,21	53,92	78,84	2344,60
	306,53	134,46	54,63	78,70	2340,20
	303,98	133,82	55,06	78,56	2322,88
	301,42	132,99	53,80	78,43	2322,58
	298,87	132,16	54,36	78,29	2323,02
	296,32	131,33	53,96	74,30	2358,59
	326,82	125,84	53,06	78,78	2280,38
	325,26	128,55	53,45	78,75	2279,44
	323,70	130,53	53,79	78,72	2265,19
	322,13	131,14	53,39	78,69	2296,68
	320,57	131,76	53,70	78,66	2311,35
	319,00	132,37	52,88	78,62	2295,86
	317,44	132,98	53,55	78,59	2307,99
schlecht	319,61	128,62	59,75	82,54	2261,00
	320,54	128,62	59,70	82,49	2232,72
	321,48	128,61	59,36	82,45	2179,12
	322,41	128,60	61,02	82,45	2138,51
	323,35	128,60	59,30	82,49	2097,54
	324,29	128,30	58,21	82,22	2090,61
	317,12	122,94	59,75	81,76	2200,73
	316,99	122,21	60,03	82,17	2156,72
	316,86	121,62	60,49	81,90	2114,57
	316,73	121,03	59,20	81,87	2177,33
	316,60	120,45	57,78	82,00	2179,87
	315,83	119,86	58,10	82,02	2189,09
	313,26	119,46	58,75	82,51	2145,14

Tabelle 7-13 Validierungsdatensatz

Die Erstellung des neuronalen Netzes wurde, wie in Abschnitt 6.4 beschrieben, durch Versuchsreihen vorgenommen. Auf spezielle Anforderungen an das Netz und die verwendeten Trainingsalgorithmen wird in Kapitel 8 näher eingegangen. Abbildung 7-19 zeigt die Netzantwort bei einem Netz mit fünf Eingangsneuronen und drei Neuronen in der verdeckten Schicht sowie den beiden Ausgabeneuronen. Die vorangegangenen Untersuchungen haben gezeigt, dass es vorteilhaft ist, sowohl eine Ausgabeneuron für den Gutzustand als auch ein Neuron für den fehlerbehafteten Zustand zu nutzen.

Die im Folgenden präsentierten Ergebnisse zeigen die Netzantwort auf den vollständigen Datensatz bei Nennlast. Es werden die Ergebnisse präsentiert, bei denen Training und Verifizierung erfolgreich waren. Der Verlauf zeigt, dass der Fehler tendenziell gut erkannt wird. Aufgrund häufiger Lastwechsel, die durch die Fahrweise des Kraftwerkes bedingt sind, kommt es jedoch des öfte-

ren zu Zuständen, die die Ausgabe des Netzes stark beeinflussen. So wird zum Beispiel in regelmäßigen Abständen der Luftvorwärmer durch ein integriertes Reinigungssystem gereinigt. Dies führt trotz Fehler zu kurzfristigen Verbesserungen des Zustandes, die das Netz durch eine entsprechende Rückkehr zum Gutzustand deutet.

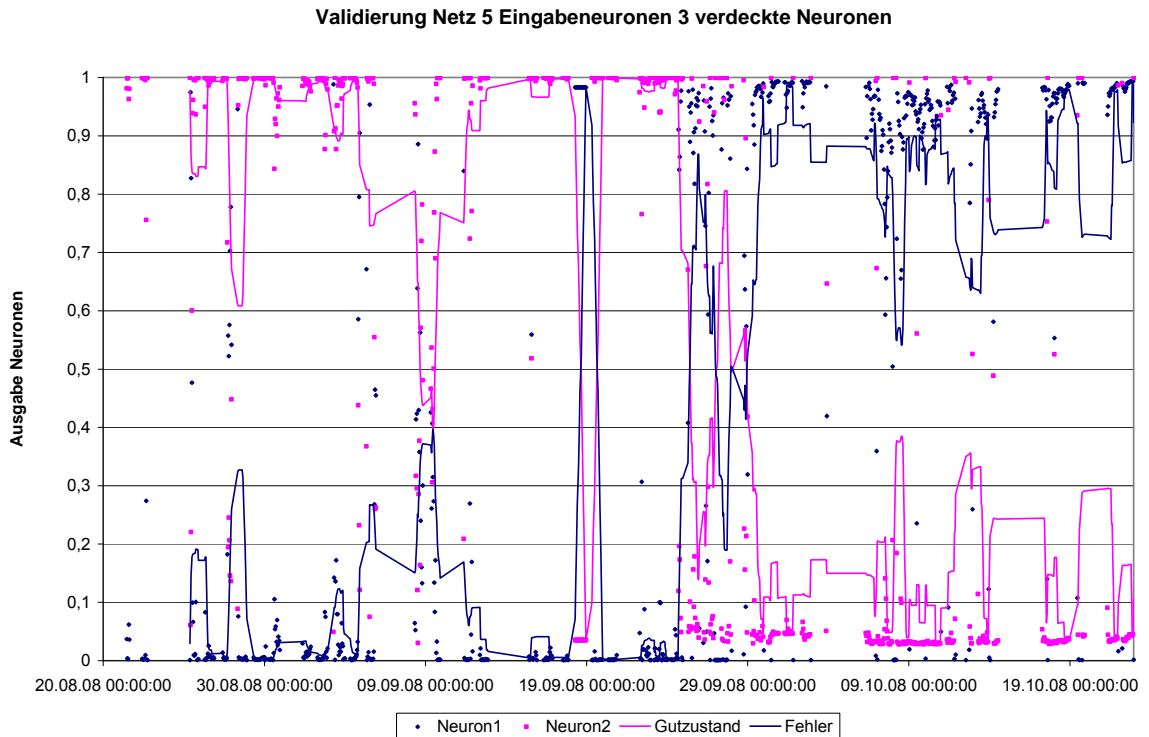


Abbildung 7-19 Netzausgabe für Validierungssatz Netzkonfiguration (5/3)

In einem Netz mit sieben Neuronen in der verdeckten Schicht wird tendenziell ein ähnliches Resultat erzielt (vgl. Abbildung 7-20). Die Netzantwort ist jedoch deutlich inkonsistenter. Ob dies an einer höheren Empfindlichkeit des Netzes auf Zustandsänderungen liegt, ist nicht zweifelsfrei festzustellen. Hier zeigt sich eine der Schwächen neuronaler Netze, nämlich dass Wissensverarbeitung und -speicherung nicht nachvollziehbar sind und daher kaum Rückschlüsse über die Gründe unterschiedlichen Netzverhaltens gezogen werden können.

Ein weiterer möglicher Grund liegt in der Erkenntnis, dass oftmals kleinere Netze bessere Ergebnisse erzielen als größere. Hierzu wird auf die Ausführungen in Kapitel 4 verwiesen.

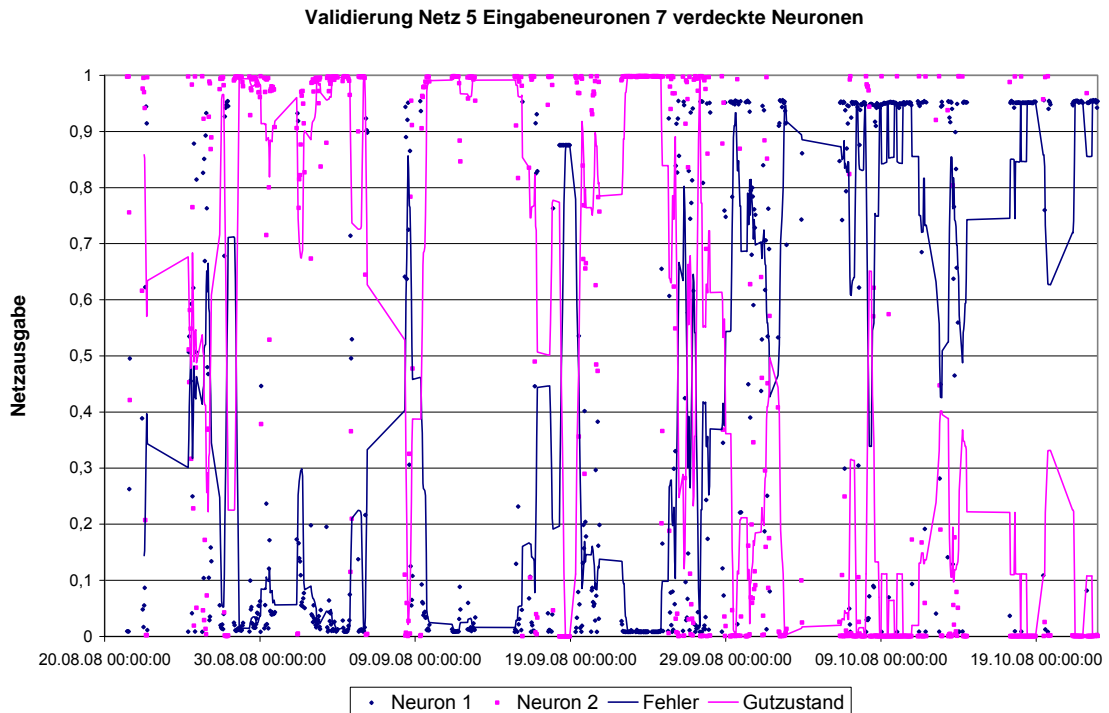


Abbildung 7-20 Netzausgabe für Validierungssatz Netzkonfiguration (5/7)

Eine wichtige Erkenntnis aus den Ergebnissen ist, dass die Identifizierung des Fehlers eindeutig aus den herausgearbeiteten Parametern ohne Verwendung der eigentliche Kontrollgröße durchgeführt werden konnte. Somit ist der Ansatz, die Korrelation von Parametern untereinander zu berücksichtigen, als zielführend herauszustellen.

Um vergleichend zu überprüfen, in wie weit die Berücksichtigung der Kontrollgröße die Performance des Netzes verbessert oder verändert, wurde im Folgenden der Trainings- und Validierungssatz um die beiden Größen „Differenzdruck primär luftseitig“ und „Differenzdruck sekundär luftseitig“ erweitert.

Die übrigen Parameter wurden beibehalten. Somit erhält man für den Trainingsdatensatz Tabelle 7-14

7 Anwendungen

Zustand	Parameter						
	6	7	9	13	18	19	20
gut	36,80	36,86	302,74	129,06	55,27	79,09	2341,65
	34,00	37,20	303,62	132,72	54,49	79,06	2334,70
	34,20	36,20	304,51	132,97	54,04	79,03	2345,01
	33,97	36,88	305,39	133,22	55,03	79,00	2367,90
	34,48	36,14	306,28	133,47	54,36	78,98	2352,99
	32,80	37,00	307,16	133,72	54,73	78,95	2335,03
	33,76	36,49	308,05	133,96	54,43	78,92	2336,91
	29,36	35,18	315,88	133,59	53,66	78,56	2334,19
	32,63	35,90	314,64	134,21	53,88	78,59	2337,66
	32,33	36,18	314,64	134,82	54,89	78,64	2351,51
	32,73	36,12	314,65	134,35	55,46	78,69	2331,07
	31,97	36,53	314,65	133,76	55,72	78,75	2315,80
	29,84	35,08	314,66	133,17	55,63	78,80	2319,95
schlecht	34,46	47,25	316,82	124,71	54,96	79,65	2191,95
	39,05	46,76	322,55	129,47	58,56	82,34	2289,83
	36,56	47,57	321,41	129,17	60,83	82,49	2252,89
	37,84	44,67	320,27	128,87	60,15	82,55	2312,18
	38,04	46,14	319,13	128,64	61,55	82,39	2291,49
	39,05	44,39	317,99	128,63	60,46	82,37	2267,62
	41,16	46,42	318,67	128,63	60,03	82,62	2250,89
	39,70	44,14	327,23	124,99	60,27	82,11	2054,24
	33,12	46,58	328,87	126,99	61,75	82,58	2073,11
	35,05	45,84	326,28	126,33	62,98	82,45	2151,95
	44,72	45,65	323,68	125,67	62,59	82,08	2259,39
	47,54	45,82	321,09	124,81	60,73	81,69	2264,30
	51,10	44,29	318,50	123,87	60,63	81,61	2285,99

Tabelle 7-14: Trainingsdatensatz für 7 Eingabeneuronen

Zustand	Parameter						
	6	7	9	13	18	19	20
gut	33,47	38,24	308,93	134,21	53,92	78,84	2344,60
	31,28	36,16	306,53	134,46	54,63	78,70	2340,20
	32,52	36,54	303,98	133,82	55,06	78,56	2322,88
	29,57	36,66	301,42	132,99	53,80	78,43	2322,58
	31,69	36,59	298,87	132,16	54,36	78,29	2323,02
	29,63	34,10	296,32	131,33	53,96	74,30	2358,59
	32,53	36,91	326,82	125,84	53,06	78,78	2280,38
	32,33	37,02	325,26	128,55	53,45	78,75	2279,44
	31,53	36,93	323,70	130,53	53,79	78,72	2265,19
	30,28	35,20	322,13	131,14	53,39	78,69	2296,68
	33,75	35,48	320,57	131,76	53,70	78,66	2311,35
	30,63	35,59	319,00	132,37	52,88	78,62	2295,86
	31,04	35,61	317,44	132,98	53,55	78,59	2307,99
schlecht	37,67	48,78	319,61	128,62	59,75	82,54	2261,00
	35,84	46,34	320,54	128,62	59,70	82,49	2232,72
	40,82	45,95	321,48	128,61	59,36	82,45	2179,12
	36,44	45,90	322,41	128,60	61,02	82,45	2138,51
	34,57	47,73	323,35	128,60	59,30	82,49	2097,54
	40,32	46,90	324,29	128,30	58,21	82,22	2090,61
	48,10	45,17	317,12	122,94	59,75	81,76	2200,73
	45,52	47,05	316,99	122,21	60,03	82,17	2156,72
	36,85	46,35	316,86	121,62	60,49	81,90	2114,57
	41,65	45,77	316,73	121,03	59,20	81,87	2177,33
	42,81	45,89	316,60	120,45	57,78	82,00	2179,87
	42,36	48,57	315,83	119,86	58,10	82,02	2189,09
	39,21	45,66	313,26	119,46	58,75	82,51	2145,14

Tabelle 7-15: Validierungsdatensatz für 7 Eingangsneuronen

Der Validierungssatz ergibt sich entsprechend Tabelle 7-15.

Analog zur vorangehend beschriebenen Bearbeitung wurde die Anzahl der Neuronen in der verdeckten Schicht variiert. Vergleicht man Abbildung 7-21 und Abbildung 7-22, so stellt man fest, dass in diesem Beispiel das Netz mit drei Neuronen der verdeckten Schicht beim Gutzustand im Zeitraum Ende August 2008 eine unbeständigere Ausgabe zeigt. Dies scheint auf den ersten Blick ein Widerspruch zu der vorher getroffenen Aussage zu sein, dass kleinere Netze ein produktiveres Verhalten zeigen. Es muss hier beachtet werden, dass aufgrund der größeren Anzahl an Eingangsneuronen eine Erhöhung der Anzahl der Neuronen der verdeckten Schicht notwendig sein kann. Es ist zu erkennen, dass die zusätzliche Nutzung der beiden Parameter der Druckdifferenzen keine entscheidende Verbesserung der Zustandszuordnung bringt.

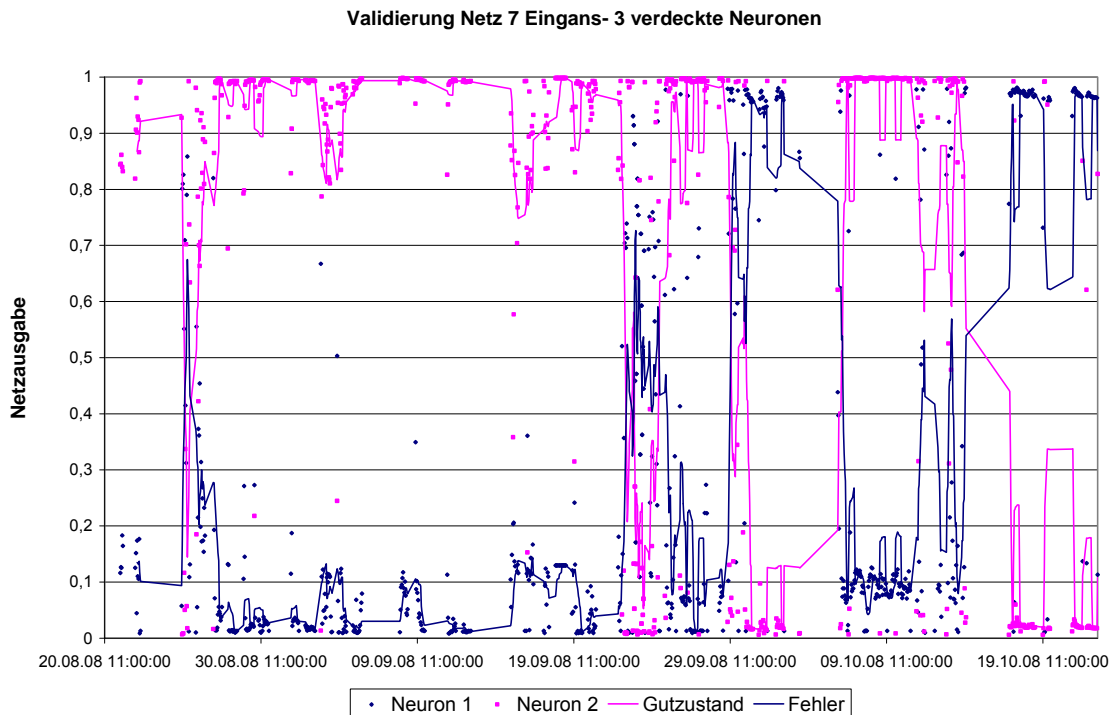


Abbildung 7-21 Netzausgabe für Validierungssatz Netzkonfiguration (7/3)

Klar zu erkennen ist auch, dass das Netz mit drei Neuronen in der verdeckten Schicht noch einmal deutlich den Gutzustand indiziert, ein Verhalten, dass bei den Netzen mit fünf Eingangsneuronen so nicht beobachtet wurde.

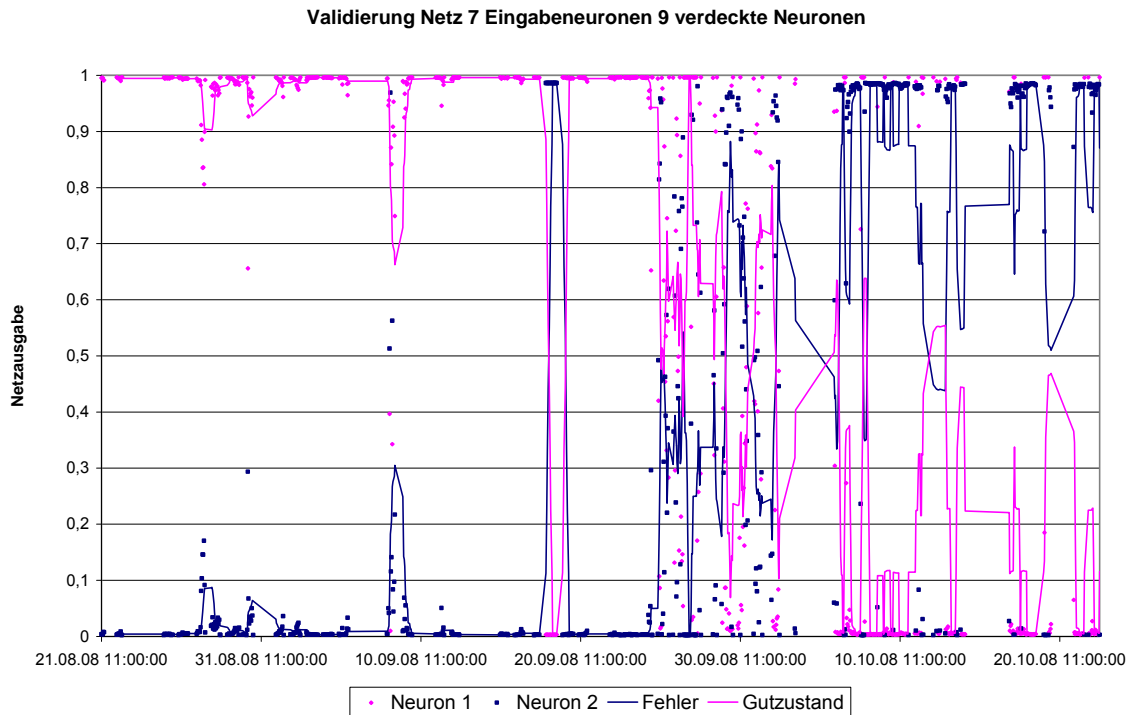


Abbildung 7-22 Netzausgabe für Validierungssatz Netzkonfiguration (7/9)

Das Netz mit neun Neuronen in der verdeckten Schicht zeigt die klare Identifizierung des Gutzustands nicht; hier wird ausschließlich die Zustandsdiagnose kurzfristig uneindeutiger.

8 Programmtechnische Umsetzung

Bevor eine erfolgreiche programmtechnische Umsetzung erfolgen konnte, mussten die grundlegenden Einflussgrößen und Parameter für den Einsatz der neuronalen Netze untersucht werden. Da dies an einem Referenzbeispiel mit unbekanntem Systemverhalten schlecht möglich war, wurde ein Beispiel gewählt, bei dem die Randbedingungen bekannt waren.

8.1 Untersuchungen mit dem Programm „Chem“

Das verwendete Programm basiert auf den Untersuchungen von Himmelblau et al. [Wat89]. Die gezeigte Vorgehensweise lehnt sich an die der Arbeit von Himmelblau et al. an. Es werden jedoch an einigen Stellen Modifizierungen vorgenommen, um die Auswirkung der Veränderung der Netzparameter zu zeigen. Der dynamische chemische Prozess wird durch das Programm „SYSSIM“ realisiert. Es wird die katalytische Gewinnung von Toluol aus Heptan modelliert.

Mit dem Programm „SYSSIM“ werden die Auswirkungen der simulierten Fehler auf die Messgrößen berechnet und so die Trainingsdatensätze erstellt. Nach erfolgtem Training wird durch eine Monte-Carlo Simulation ein dem Netz unbekannter Fehlerdatensatz generiert und so seine Generalisierungsfähigkeit getestet.

Das Programm „Chem“ (vgl. Abbildung 8-1) setzt das neuronale Netz programmtechnisch um. Es handelt sich um ein vorwärtsgerichtetes Netz mit einer verdeckten Schicht Neuronen ohne „short cut“ Verbindungen. Es wird hierbei ein variables Netz generiert, das über eine problembezogene Anzahl an Eingangsgrößen und Ausgangsgrößen verfügt und mit einer variablen Anzahl Neuronen der verdeckten Schicht ausgestattet werden kann. Im Programm implementiert ist der Trainingsdatensatz, so dass der rechentechnische Aufwand reduziert wird, da der Trainingsatz nicht jedes Mal durch das Unterprogramm „SYSSIM“ erstellt werden muss. Für die Verwendung des Trainingsdatensatzes

mit modifizierten Eingabedaten, wurde ein Programmvariante eingesetzt, die über einen modifizierten Trainingsalgorithmus verfügt und bei der die Ausgaben des Unterprogramms „SYSSIM“ entsprechend transformiert werden, bevor sie durch das Netz propagiert werden. Bereits vorprogrammiert sind zwei Aktivierungsfunktionen der Neuronen.

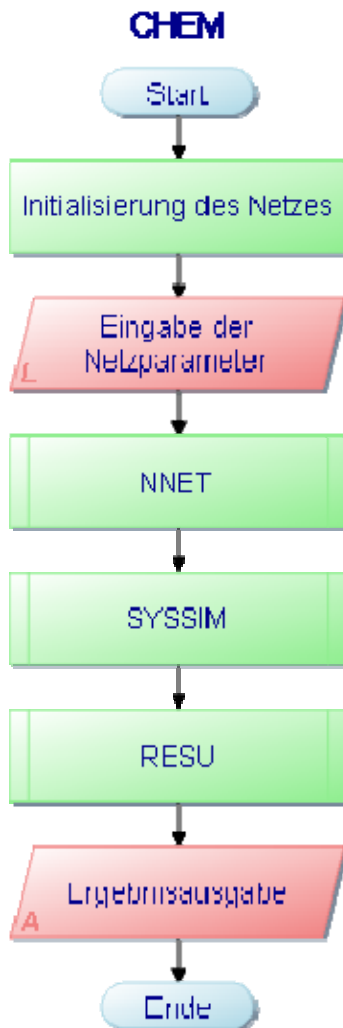


Abbildung 8-1 Programmablaufplan Programm „Chem“

Es kann jedoch durch Hinzufügen modulartiger Routinen oder auch externer Programme jede beliebige Aktivierungsfunktion realisiert werden. Als Lernverfahren ist das Backpropagation Verfahren programmiert, jedoch kann auch hier eine Variation durch Austauschen der entsprechenden Routine erfolgen. Das Training erfolgt iterativ und wird durch zwei Abbruchkriterien beschränkt. Zum

einen wird eine Schranke für die minimale Änderung der Gewichte bestimmt, die das Training abbricht, sobald eine bestimmte Änderung der einzelnen Gewichte unterschritten wird. Dies stellt eine Abweichung von der sonst üblichen Fehlerschranke dar, bei der der Trainingsabbruch beim Unterschreiten der tolerierten Fehlerabweichung erfolgt. Das zweite Abbruchkriterium ist die vorher festlegbare Anzahl der Trainingsspiele. Die Anfangsgewichte der Verbindungen werden über einen Zufallsgenerator bestimmt. Zur besseren Vergleichbarkeit der verschiedenen Rechenläufe werden stets dieselben Anfangsgewichte genutzt.

Mit Hilfe der Programme konnten die unter Kapitel 4 vorgestellten Parameter und Einflussgrößen des neuronalen Netzes experimentell untersucht werden. Der Programmablaufplan ist in Abbildung 8-1 dargestellt. An dieser Stelle werden exemplarisch die Ergebnisse für den Einfluss der Lernrate auf die notwendige Anzahl der Trainingsdurchläufe in Abhängigkeit der Anzahl verdeckter Neuronen sowie ein Beispiel für die Validierung unterschiedlicher Netzgrößen gegeben.

Die Wahl der Lernrate η beeinflusst dabei stark die notwendige Anzahl der Trainingsdurchläufe. Zu kleine Lernraten erhöhen die Anzahl deutlich, wie Abbildung 8-2 zeigt.

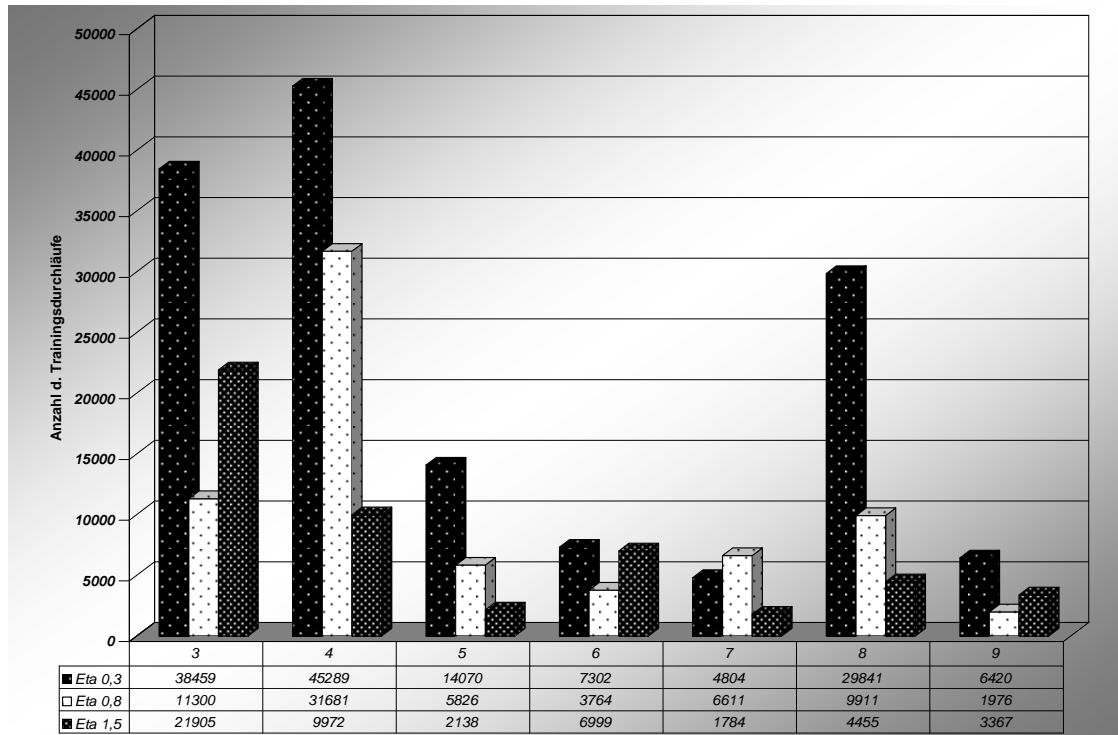


Abbildung 8-2 Anzahl der Trainingsdurchläufe in Abhängigkeit von der Lernrate

Vergleicht man Abbildung 8-2 und Abbildung 8-3 so zeigt sich, dass die Optimierung am Parameter Anzahl der Spiele nicht durchgeführt werden sollte, da mit Erhöhung der Anzahl von Neuronen in der verdeckten Schicht zwar die notwendige Anzahl der Trainingsdurchläufe entscheidend sinkt, aber die Generalisierungsfähigkeit des Netzes stark abnimmt. Vor dem Hintergrund der Leistungsfähigkeit aktueller Prozessoren kann die Anzahl der notwendigen Durchläufe sogar vernachlässigt werden, solange die Problematik des Übertrainings nicht aus den Augen verloren wird.

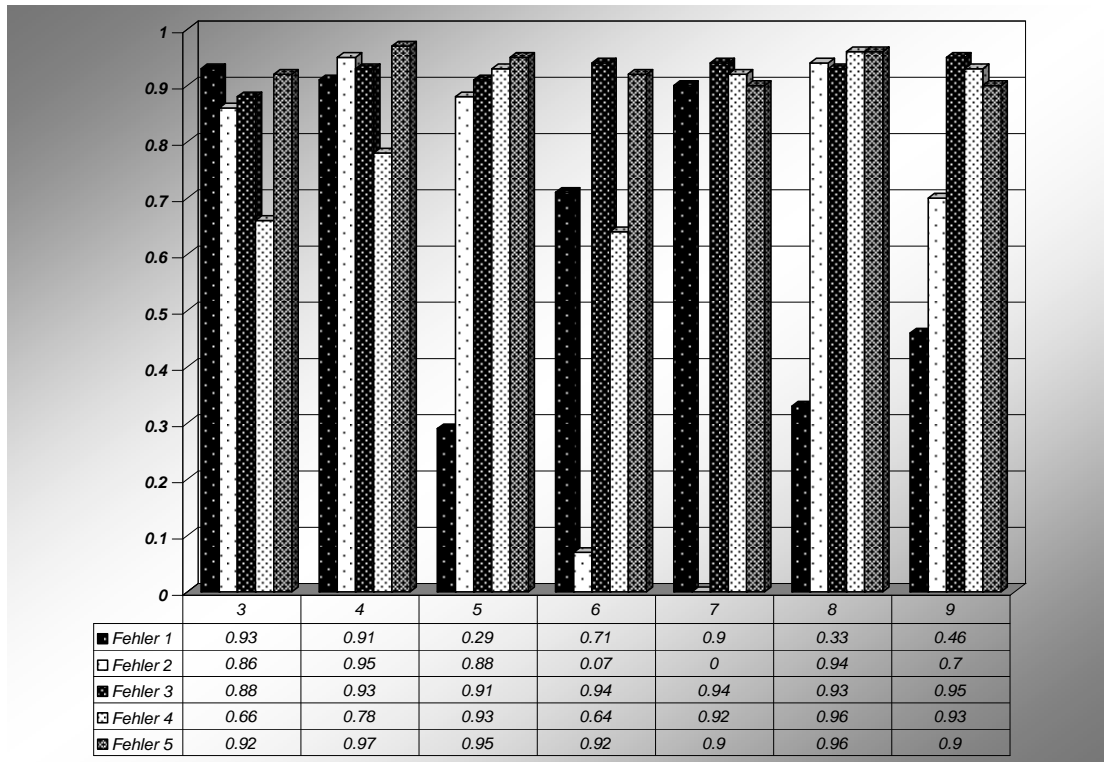


Abbildung 8-3 Generalisierungsfähigkeit in Abhängigkeit der Netzgröße

Aus den Untersuchungen konnte als Erkenntnis gewonnen werden, dass bei der Suche nach einer geeigneten Netzwerkarchitektur mit kleinen Netzen begonnen wird, die dann schrittweise vergrößert werden.

Zum Einsatz kamen ausschließlich vorwärtsgerichtete Netze ohne Rückkopplung. Unter den verwendeten Trainingsalgorithmen stellte sich der Levenberg-Marquardt Algorithmus als oftmals sehr effizient heraus. Dieser und weitere verwendete Trainingsalgorithmen werden im Abschnitt 8.4 vorgestellt. Die notwendige Anzahl von Neuronen in den verdeckten Schichten lässt sich an Hand der Generalisierungsfähigkeit mit Hilfe von dem Netz unbekanntem Daten bewerten. Hierzu wurden mehrere Versuchsreihen durchgeführt. Abbildung 8-4 zeigt eine Auswahl unterschiedlicher Netzkonfigurationen, die mit den Ursprungswerten und skalierten Daten durchgerechnet wurden.

Weiterhin wurde die Anzahl der verdeckten Schichten variiert. Die Ergebnisse sind ebenfalls in Abbildung 8-4 aufgezeigt.

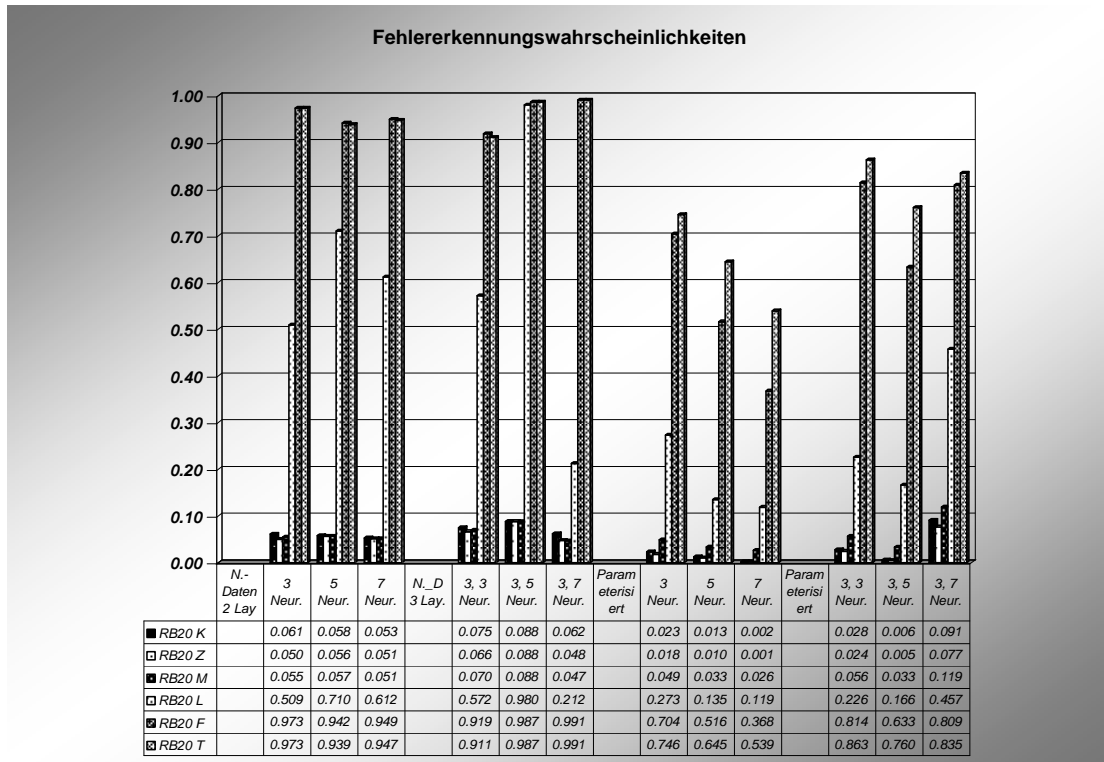


Abbildung 8-4: Trefferquoten unterschiedlicher Netzkonfigurationen

Die Ergebnisse (vgl. Abbildung 8-4 die beiden letzten Rechnungen) zeigen, dass die Normierung der Daten in diesem Fall die Wahrscheinlichkeit, einen Fehler zu entdecken, nicht erhöht. Dies liegt daran, dass in Matlab eine automatische Skalierung der Eingangsdaten und somit eine Normierung in jedem Fall stattfindet.

Abbildung 8-4 zeigt deutlich, dass kleinere Netze in dieser Anwendung, höhere Generalisierungsfähigkeiten besitzen als größere Netze. Auffällig ist, dass meist einer der Fehler deutlich schlechter gelernt und somit schlechter identifiziert wird. Dies entspricht auch den Ergebnissen anderer Arbeiten (vgl. /And97/). Mit zunehmender Netzgröße werden einzelne Fehler nicht mehr hinreichend gut erlernt, so dass das Netz ungenügende Identifikationswahrscheinlichkeiten bei Rechnungen mit nicht trainierten Daten aufweist. Dies wurde schon durch eigene Arbeiten, wie oben beschrieben, bestätigt.

Basierend auf den Ergebnissen wurden die nachfolgend vorgestellten Programme zur rechentechnischen Umsetzung des Verfahrens entwickelt.

8.2 Programmbeschreibungen

Für die programmtechnische Umsetzung der Referenzfälle wurden die Programme in Matlab erstellt. In Matlab sind verschiedene Netzwerkarchitekturen und alle gängigen Trainingsalgorithmen in Form von Funktionen hinterlegt und müssen somit nicht mehr programmiert und verifiziert werden.

Die programmtechnische Umsetzung für die Versuchsreihen für den Referenzfall „Zwischenüberhitzer“ untergliedert sich in zwei Programme. Im Trainingsprogramm „NeuroTrain“ können unterschiedliche Netzarten, Trainingsalgorithmen und Netzwerkarchitekturen eingesetzt werden. Der Programmablaufplan ist in Abbildung 8-5 dargestellt.

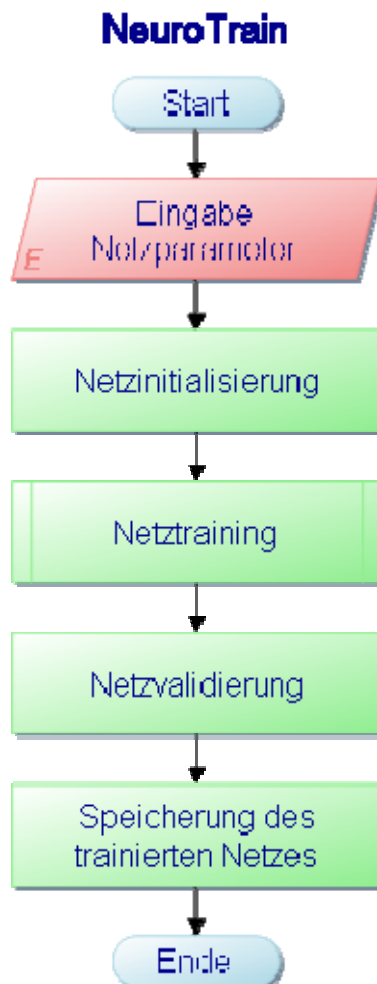


Abbildung 8-5 Programmablaufplan für NeuroTrain

Entsprechend dem Untersuchungsschwerpunkt werden Trainingsdatensätze verwendet (z.B. nur Nennlast, vollständiger Fehlersatz bzw. Teilfehlersatz etc).

Im Programm „NeuroApp“ (vgl. Abbildung 8-6) werden Netze, die erfolgreich trainiert wurden, also eine gute Differenzierung zwischen Fehler- und Normzustand zeigten, aus „NeuroTrain“ mit den vollständigen, nicht zum Training und Verifizierung verwendeten Datensätzen validiert. Dafür wurden die entsprechenden Datensätze eingelesen. Die Ausgabe des Netzes lässt sich, wie an den Ergebnissen in Kapitel 7.1.4 gezeigt, über der Referenzzeit abtragen. Man erhält damit den vom Netz prognostizierten Zustand des Systems und kann diesen mit dem tatsächlich aufgetretenen vergleichen.

Das Programm „NeuroApp“ kann somit auch für die Überwachung im realen Betriebsgeschehen eingesetzt werden. Aus dem Prozessleitsystem werden die entsprechenden Messparameter herausgelesen und vom Netz analysiert.

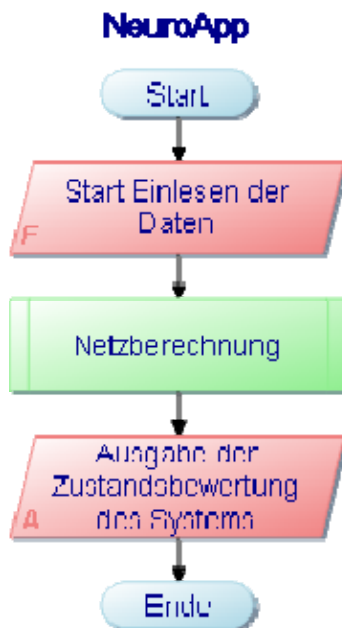


Abbildung 8-6 Programmablaufplan für NeuroApp

An Hand der gewählten Netzkonfiguration wurden die auf den Nennlastbetrieb beschränkten Daten sowohl für den gesamten Fehlerbereich als auch für den eingeschränkten selektierten Fehlerbereich mit dem Programm „NeuroApp“ verifiziert. Da nur eine geringe Anzahl an Datensätzen aus dem Nennlastbereich

für das Training eingesetzt wurde, ist der Einsatz der übrigen Daten desselben Zwischenüberhitzerkreislaufes zur Validierung zulässig.

8.3 Adaption des Programms

Eine Anpassung des neuronalen Netzes an die Ergebnisse der technischen Systemanalyse muss, wie bereits beschrieben, für jede technische Anlage individuell erfolgen. Außerdem muss gegebenenfalls auch der Trainingsalgorithmus variiert werden. Dies ergibt sich aus den unterschiedlichen Ausprägungen der resultierenden Fehlerfunktionen, so dass ein angepasster Trainingsalgorithmus zum Berechnen der Minima der Fehlerfunktion gefunden werden muss. Hier zeigte sich der entscheidende Vorteil der Arbeit mit Matlab. Die Programmroutinen der einzelnen Algorithmen sind bereits hinterlegt und können entsprechend je nach Bedarf aufgerufen werden. Um die zum Teil sehr langwierigen Trainings- und Validierungsmethoden zu optimieren, wurde das Programm „NeuroTrain“ im Verlauf der Entwicklung so geändert, dass sich die einzelnen Parameter, während das Programm abläuft, ändern lassen. Die Änderung des Quellcodes ist damit nicht mehr erforderlich.

Erfolgreiche Netzarchitekturen werden zusammen mit den zugehörigen Gewichten an die Anwendungsroutine „NeuroApp“ übergeben. In der Anwendungsroutine wird das trainierte neuronale Netz als unveränderliche Struktur übernommen und steht zur Analyse von Daten zur Verfügung. Durch Einlesen der Daten erfolgt eine permanente Zustandsbewertung durch das Netz.

8.4 Trainingsalgorithmen

Neben den bereits eingesetzten Trainingsalgorithmen nach Levenberg-Marquardt sowie der klassischen Backpropagation, wurde der „Resilient Backpropagation“ (RPROP) Algorithmus verwendet. Der Algorithmus nach Levenberg-Marquardt stellt dabei das bislang am schnellsten konvergierenden Verfahren dar. Es nutzt, wie auch Backpropagation, das Newton Verfahren, es werden jedoch auch die zweiten Ableitungen verwendet. Damit stellt es ein Verfahren zweiter Ordnung dar. Somit benötigt man Ableitungen zweiter Ordnung der Fehlerfunktion, die die Rechnung aufwendiger machen, aber die Zahl der

Rechenschritte reduzieren. Da es sich bei den vorliegenden Beispielen um kleine Netze handelt, ist die Zahl der Rechenschritte nicht entscheidend und das Finden des globalen Minimums der Fehlerfunktion steht im Vordergrund.

Da das Levenberg – Marquardt Verfahren im Referenzbeispiel „Lagerschaden“ nicht zu zufrieden stellenden Ergebnissen führte, wurde zusätzlich das Resilient Backpropagation Verfahren eingesetzt. Durch diesen Algorithmus kann das Problem, dass auf flachen Plateaus der Fehlerfläche die Gewichtsänderung zwischen den Neuronen zu klein und in steilen Abschnitten die Änderung zu groß wird, behoben werden. Dieses in den meisten Algorithmen auftretende Problem liegt darin begründet, dass für Backpropagation gilt:

$$\Delta w \propto \frac{\delta E}{\delta w} \quad 8-1$$

mit

E – Fehlerfunktion

w – Gewichtsvektor

Die Idee hierbei ist, dass sich die Gewichtsänderung nicht ausschließlich aus der partiellen Ableitung der Fehlerfunktion nach den Gewichten ergibt, sondern sich als Funktion des Vorzeichens der partiellen Ableitung der Fehlerfunktion nach den Gewichten ergibt. Das heißt, nur die Richtung der Gewichtsänderung wird durch die lokalen Gegebenheiten der Fehlerfläche bestimmt, nicht aber deren Ausmaß /Rie97/:

$$\text{Also nicht } \Delta w \neq f\left(-\frac{\delta E}{\delta w}\right) \text{ sondern } \Delta w = f\left(\text{sgn}\left(\frac{\delta E}{\delta w}\right)\right);$$

somit:

$$\Delta w_j^{(i)} = \begin{cases} -\Delta_j^{(i)} & \text{falls } \frac{\delta}{\delta w_j} E(w^{(i)}) < 0 \\ +\Delta_j^{(i)} & \text{falls } \frac{\delta}{\delta w_j} E(w^{(i)}) > 0 \\ 0 & \text{falls } \frac{\delta}{\delta w_j} E(w^{(i)}) = 0 \end{cases} \quad 8-2$$

wobei:

$$\Delta_j^{(i)} = \begin{cases} \eta^+ \Delta_j^{(i-1)} & \text{falls } \frac{\delta}{\delta w_j} E(w^{(i-1)}) \frac{\delta}{\delta w_j} E(w^{(i)}) > 0 \\ \eta^- \Delta_j^{(i-1)} & \text{falls } \frac{\delta}{\delta w_j} E(w^{(i-1)}) \frac{\delta}{\delta w_j} E(w^{(i)}) < 0 \\ \Delta_j^{(i-1)} & \text{falls } \frac{\delta}{\delta w_j} E(w^{(i-1)}) \frac{\delta}{\delta w_j} E(w^{(i)}) = 0 \end{cases} \quad 8-3$$

Dabei muss gelten; $0 < \eta^- < 1 < \eta^+$.

Nachfolgend werden die Ergebnisse ausgewählter Trainingsversuche vorgestellt.

8.5 Ergebnisse

Neben den bereits unter 8.1 vorgestellten Ergebnissen, werden hier die Resultate aus den Anwendungen der unterschiedlichen Trainingsalgorithmen dargestellt. Das Resilient Backpropagation Verfahren zeigte bessere Trainingsergebnisse für das Beispiel „Lagerschaden“. Wie im Vergleich von Abbildung 8-7 und Abbildung 8-8 zu erkennen, erreicht der RPROP Algorithmus deutlich schneller ein Minimum, da er augenscheinlich ein Plateau in der Fehlerfunktion schneller überwinden kann. Ein Plateau ist stets dann zu vermuten, wenn sich über viele Trainingsschritte keine Minimierung erreichen lässt, wie in Abbildung 8-8 zu sehen.

8 Programmtechnische Umsetzung

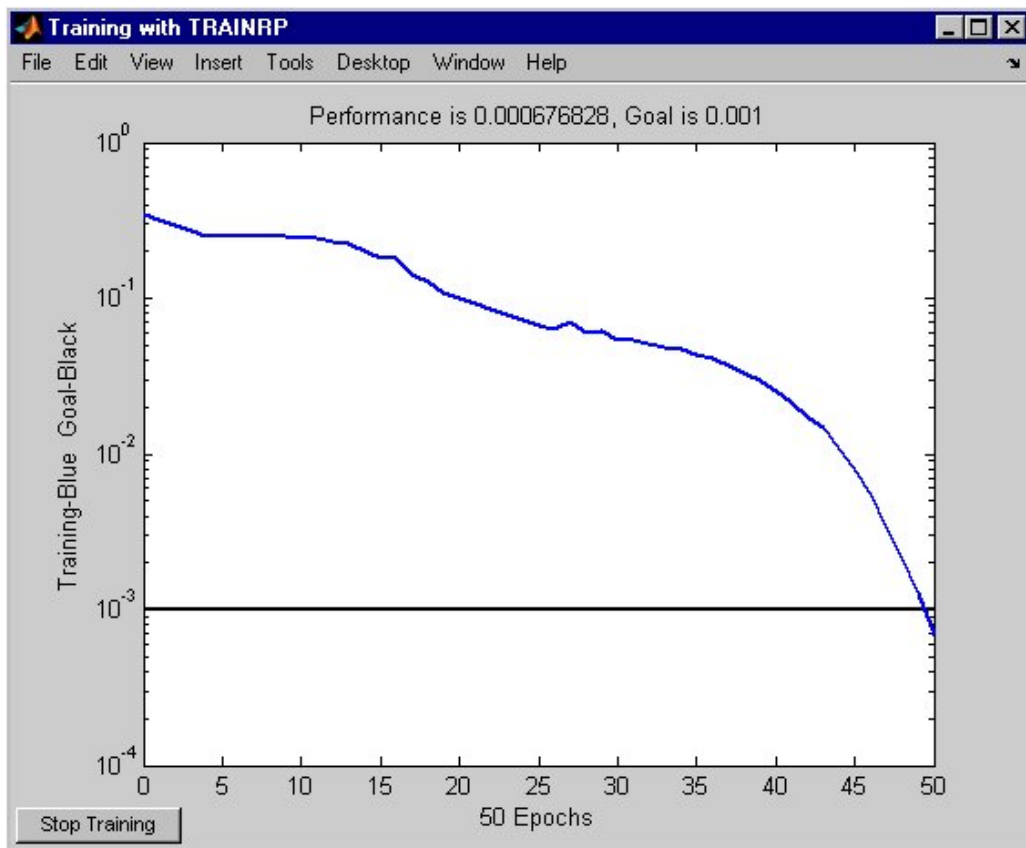


Abbildung 8-7: Verlauf des Trainings mit RPROP

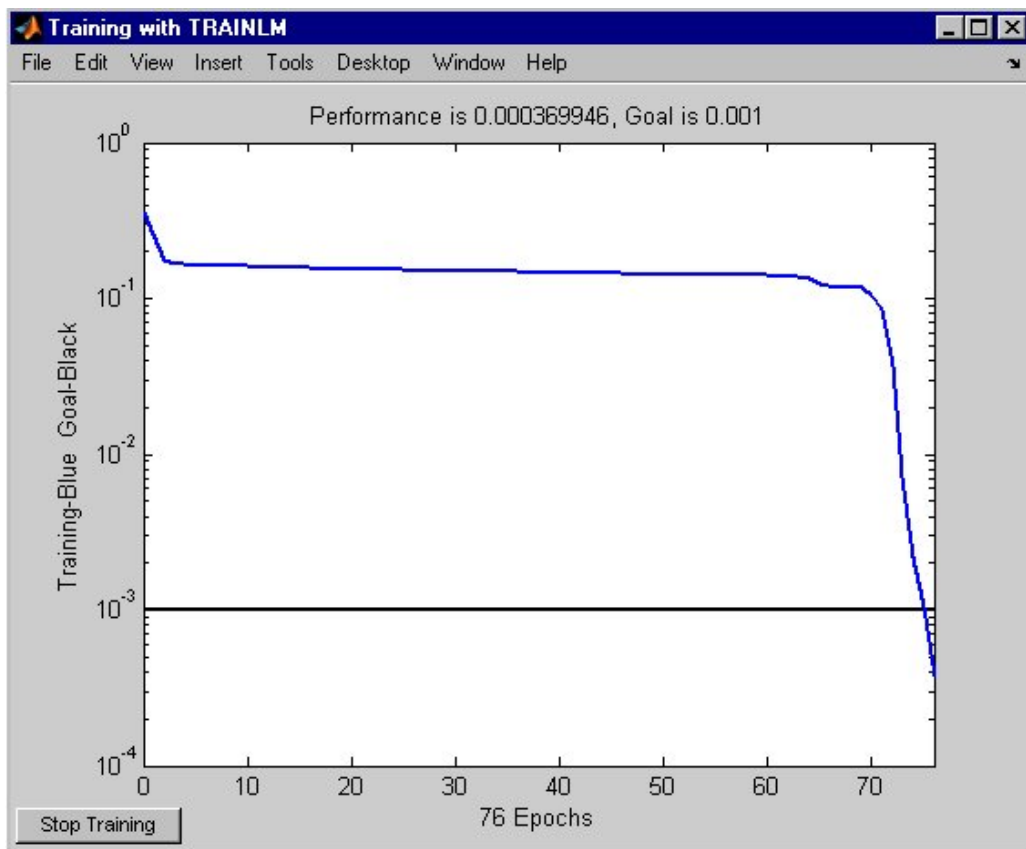


Abbildung 8-8: Verlauf des Trainings mit Levenberg-Marquardt

Obwohl mit beiden Algorithmen die Unterschreitung des vorgegebenen Lernziels gelingt, zeigt Tabelle 8-1, dass nur mittels PROP ein gutes Ergebnis in der Verifizierung erreicht wird. Mit dem Levenberg – Marquardt Algorithmus konnte bei ansonsten gleichen Radbedingungen kein zufrieden stellendes Ergebnis erzielt werden.

Zustand Netz	Fehler	Fehler	Fehler	Übergang	Heile
RPROP	0.99	0.98	0.99	0.47	0.02
Lev- Marqu	0.01	0.01	0.04	0.02	0.01

Tabelle 8-1: Netzausgabe im Verifizierungsschritt

Die Ergebnisse zeigen, dass eine erfolgreiche Früherkennung sich anbahnender Fehler nicht nur von geeigneten Daten abhängt, sondern auch ein geeigneter Algorithmus gefunden und damit das neuronale Netz an das zu behandelnde Problem adaptiert werden muss.

9 Vergleichsmethode

Durch die Anwendung neuronaler Netze zur Klassifizierung wurden die Korrelationen unterschiedlicher Parameter untereinander berücksichtigt. Um Vor- und Nachteile dieses Ansatzes bewerten zu können, musste eine alternative Vorgehensweise gewählt werden, die unter denselben Randbedingungen eingesetzt werden kann. Es wurde hierfür die Fuzzy Methode gewählt. Die Wahl lässt sich wie folgt begründen. Da die Randbedingungen der Untersuchung gleich bleiben, also keine Modellbildung erfolgt und keine zusätzlichen Messgrößen installiert werden, musste eine Vorgehensweise gewählt werden, die es ermöglicht, das Wissen des Anlagenpersonals zu berücksichtigen und ohne eine Modellabbildung des zu Grunde liegenden technischen Systems auszukommen.

Als Vergleichsmethode wurde die Fuzzy Regelung nach dem Ansatz von Mamdani gewählt. Es handelt sich hierbei um einen Ansatz, bei dem das Wissen durch linguistische Regeln implementiert wird. Damit kann ganz speziell der Einsatz der neuronalen Netze bewertet werden, da mit denselben Randbedingungen und auf derselben Wissensbasis die Eingangsgrößen verarbeitet werden.

Nachfolgend wird der Ansatz der Fuzzy Methode und die Einbindung in die Methodik der Fehlerfrüherkennung vorgestellt; anschließend werden die Ergebnisse dargestellt.

9.1 Fuzzy Regelung – Ansatz von Mamdani

Ohne an dieser Stelle eine vollständige Einführung in die Fuzzy Thematik zu geben, werden einige Aspekte, die zum Verständnis der Anwendung notwendig sind, vorgestellt. Im Gegensatz zur klassischen Mengenlehre, wird eine Fuzzy-

menge durch ihre Zugehörigkeitsfunktion charakterisiert. Es wird also, anders als bei den klassischen Mengen, wo die Zugehörigkeit zur Menge stets Null oder Eins ist, eine Zugehörigkeit mit einem Wert aus dem Intervall $[0,1]$ definiert. Somit stellen die klassischen Mengen einen Spezialfall der Fuzzy Mengen dar /Tiz98/.

Der Ansatz von Mamdani /Mam74/ sieht vor, dass ein Experte sein Wissen in Form von linguistischen Regeln beschreibt. Hierfür werden die folgenden Komponenten verwendet:

- Fuzzifizierungs-Interface nimmt die Messwerte auf und wandelt sie in eine Fuzzy Menge bzw. linguistischen Term um
- Die Wissensbasis beinhaltet die Wertebereiche der Mess- und Ausgangsgrößen, die zu den linguistischen Termen assoziierten Fuzzy-Mengen und die Regelbasis
- Die Entscheidungslogik, die aus den Messgrößen mit Hilfe der Wissensbasis Informationen über die „Stellgröße“ gewinnt
- Defuzzifizierung

Zu Beginn werden linguistischen Terme aus den Messparametern gebildet. Es erfolgt eine Partitionierung der Messwertbereiche z. B. in „niedrig“ und „hoch“.

$$\mu_1^{(1)}, \dots, \mu_{p_1}^{(1)} \in X_1 \quad (9-1)$$

Die Partitionierung wird über die Zugehörigkeitsfunktion abgebildet. Die Zugehörigkeitsfunktion kann dabei durch jede, das gesamte Intervall $[0,1]$, überschneidende Funktion realisiert werden. Die in der Arbeit für die Beschreibung der Eingangsgrößen verwendeten Zugehörigkeitsfunktionen sind monoton steigend stückweise lineare (Abbildung 9-1) und die monoton fallend stückweise lineare Funktionen (Abbildung 9-2).

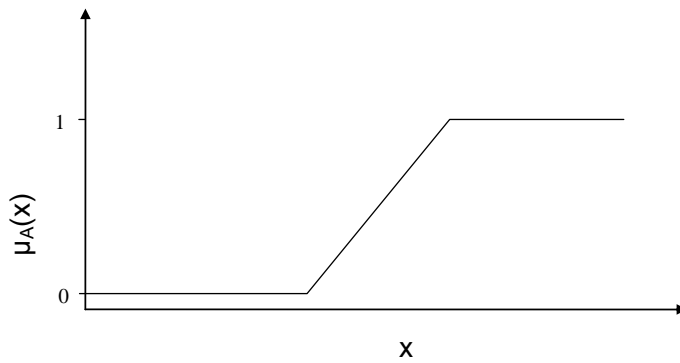


Abbildung 9-1: monoton steigende lineare Zugehörigkeitsfunktion

Hierbei gilt:

$$\mu_A(x) = \begin{cases} 0 & \text{für } x \leq a \\ \frac{x-a}{b-a} & \text{für } a \leq x \leq b \\ 1 & \text{für } x > b \end{cases} \quad (9-2)$$

wobei $a < b$ und a der Wert für Beginn der Steigung und b der Wert für das Ende der Steigung.

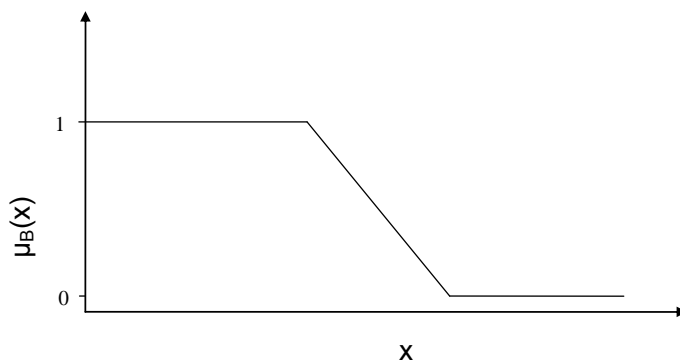


Abbildung 9-2: monoton fallend lineare Zugehörigkeitsfunktion

Entsprechend:

$$\mu_B(x) = \begin{cases} 1 & \text{für } x \leq a \\ \frac{b-x}{b-a} & \text{für } a \leq x \leq b \\ 0 & \text{für } x \geq b \end{cases} \quad (9-3)$$

Es gilt hierbei die Disjunktheitsforderung $\mu_a + \mu_B = 1$.

Die gesuchte Ausgabegröße, in unserem Fall die Zustandsdiagnose des Systems, wird ebenfalls partitioniert. Hierfür wurden die linguistischen Terme „kein Fehler“, „Übergang“, „Fehler“ gebildet. Es ergibt sich:

$$\mu_1, \dots, \mu_p \in Y \quad (9-4)$$

Die entsprechenden mathematischen Grundlagen sowie alternative Zugehörigkeitsfunktionen werden u. a. in /Tiz98/, /Böh93/ und /Kru95/ behandelt.

Die partitionierten Eingangsgrößen werden über die so genannten Kontrollregeln miteinander verbunden. Diese sind nicht als Implikationen sondern als eine stückweise Definition der Ausgabefunktion zu verstehen. Es wird damit jeder denkbaren Kombination eines Eingangstupels, bestehend aus den partitionierten Eingangsgrößen, die entsprechende Ausgabe zugewiesen. Es wird dafür jede Regel einzeln ausgewertet:

$$\text{Wenn } \xi_1 \text{ ist } A_{i_1,r} \text{ und } \dots \text{ und } \xi_n \text{ ist } A_{i_n,r} \text{ dann } \theta \text{ ist } B \quad (r = 1, \dots, k) \quad (9-5)$$

dabei sind $A_{i_1,r} \dots A_{i_n,r}$ und B die linguistischen Terme, die den Fuzzy-Mengen μ_A , μ_B und μ_Y entsprechen.

$$\theta \in \begin{cases} B_{i_1} & \text{falls } \xi_1 \in A_{i_1,1} \text{ und } \dots \text{ und } \xi_n \in A_{i_n,1} \\ \vdots \\ B_{i_k} & \text{falls } \xi_1 \in A_{i_1,k} \text{ und } \dots \text{ und } \xi_n \in A_{i_n,k} \end{cases} \quad (9-6)$$

Die Auswertung erfolgt, in dem der Erfüllungsgrad bestimmt wird, zu dem die Prämisse bei den vorliegenden Messwerten erfüllt ist. Die Prämissen berechnen sich wie folgt:

$$\alpha_r = \min\{\mu_{i,r}^{(1)}(x_1), \dots, \mu_{i,r}^{(n)}(x_n)\} \quad (9-7)$$

Dabei gibt α_r den Erfüllungsgrad der entsprechenden Regel r an.

Als Ausgabe ergibt sich die Fuzzy-Menge an „Stellwerten“, die sich durch „Abschneiden“ der Fuzzymenge μ_i der Regel R_r beim Grad α_r ergibt. Jeder mögliche „Stellwert“ y wird berechnet nach:

$$\mu_{x_1, \dots, x_n}^{\text{output}(R_r)}(y) = \min\{\mu_{i,r}^{(1)}(x_1), \dots, \mu_{i,r}^{(n)}(x_n), \mu_{i,r}(y)\} \quad (9-8)$$

Nachdem die Entscheidungslogik jede einzelne Regel ausgewertet hat, müssen die Ergebnis-Fuzzy-Mengen mittels Maximumbildung zu einer Fuzzymenge zusammengesetzt werden.

$$\mu_{x_1, \dots, x_n}(y) = \max\{\min\{\alpha_1, \mu(y)\}, \dots, \min\{\alpha_n, \mu(y)\}\} \quad (9-9)$$

Im letzten Schritt der Defuzzyfizierung muss die unscharfe Größe in eine scharfe Größe rücktransformiert werden. Diese Größe wird als Wahrscheinlichkeit für den Fehler interpretiert.

Im vorliegenden Zusammenhang wurde die Schwerpunktmethode zur Defuzzyfizierung gewählt. Als Ergebnis E der Defuzzifizierung wird der Wert verwendet, der unter dem Schwerpunkt der von Zugehörigkeitsfunktion und der Ordinate begrenzten Fläche liegt.

$$E = \frac{\int_{y \in Y} y \cdot \mu_{x_1, \dots, x_n}^{\text{output}}(y) dy}{\int_{y \in Y} \mu_{x_1, \dots, x_n}^{\text{output}}(y) dy} \quad (9-10)$$

Die Methode entspringt der Idee, dass die Regeln entsprechend dem Erfüllungsgrad ihrer Prämissen berücksichtigt werden. Nachteil der Methode ist, dass dies formal aus der Theorie der Fuzzy-Mengen kaum zu rechtfertigen ist /Tiz98/.

Das Ergebnis wird dann wiederum als Wahrscheinlichkeit für den Fehler interpretiert. Somit kann für jedes Messtupel eines Zeitschrittes eine Fehlerwahrscheinlichkeit bestimmt werden. Die Ergebnisse können dann direkt mit denen aus der Anwendung der neuronalen Netze verglichen werden.

Die Umsetzung der Anwendung erfolgt mit Hilfe von Matlab. Abbildung 9-3 zeigt den Programmablaufplan.

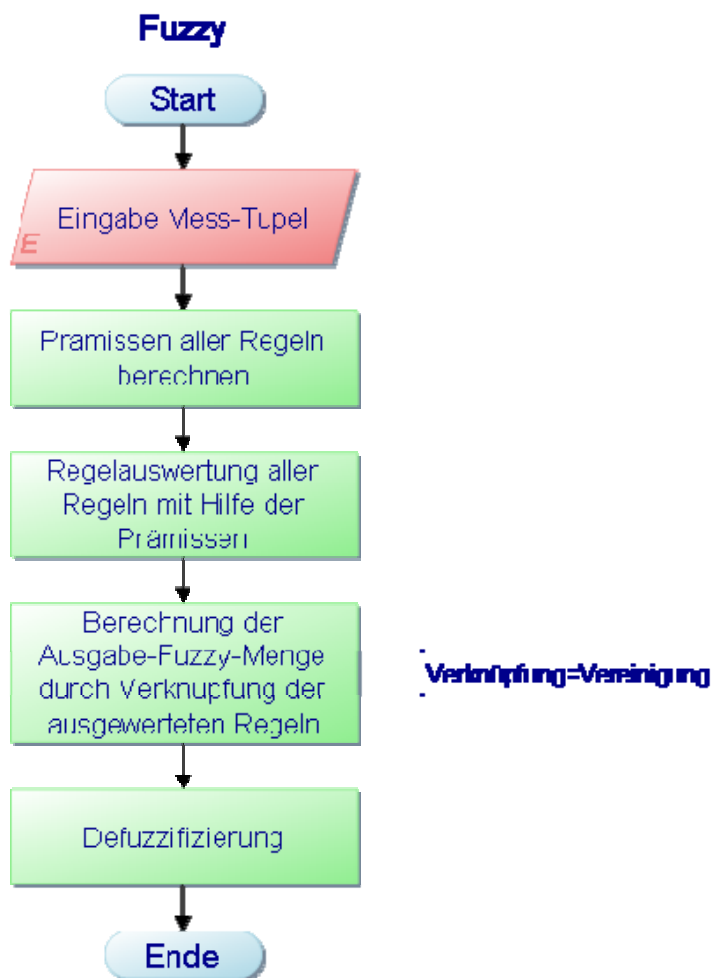


Abbildung 9-3 Programmablaufplan Fuzzy Anwendung

9.2 Referenzfall „Lagerschaden“

Als erster Vergleich wurde der Referenzfall „Lagerschaden“ gewählt. Hier liegen wenige Eingangsgrößen und ein überschaubares Parameterverhalten vor. Im ersten Schritt werden die Eingangsgrößen:

- „low rate“
- „high rate“
- „Schwingung horizontal“
- „Schwingung vertikal“

fuzzyfiziert. Hier stellte sich heraus, dass für die Schwingungsgrößen „horizontal“ und „vertikal“ keine geeigneten Fuzzifizierungsregeln gefunden werden konnten. Daher wurden die Größen „low rate“ und „high rate“ fuzzyfiziert. Dies erfolgte, wie voranstehend beschrieben, durch stückweise lineare Zugehörigkeitsfunktionen. Die Festlegungen der Grenzen für die einzelnen Zugehörigkeitsfunktionen wurden aus den Daten der Trainingssätze gewonnen, indem die Grenzen für „Gut“- und „Fehlerbereiche“ verwendet wurden.

Die Regeln wurden entsprechend den Erkenntnissen der technischen Systemanalyse erstellt. Abbildung 9-4 zeigt das defuzzifizierte Ergebnis.



Abbildung 9-4: Ergebnis der Fuzzy Anwendung für Lager SAC02

Im Vergleich zu Abbildung 7-11 wird deutlich, dass die kleineren Schwankungen eliminiert werden. Dies liegt daran, dass für das Ergebnis der Erfüllungsgrad der Prämissen Ausschlag gebend sind. Solange eine Regel dominiert, ändert sich die Ausgabe nicht. Es wird jedoch eindeutig der Fehler angezeigt.

Abbildung 9-5 zeigt das Ergebnis nach Defuzzifizierung für das Lager SAC03. Entsprechend der Vorgehensweise bei der Anwendung mit neuronalen Netzen wurde dies als Quasi-Realtimeanwendung eingesetzt.

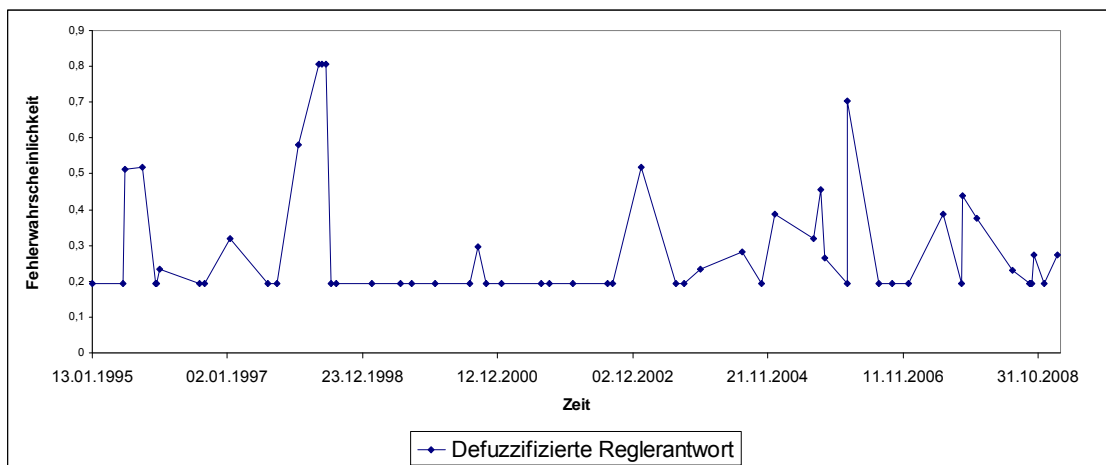


Abbildung 9-5: Ergebnis der Fuzzy Anwendung für Lager SAC03

Von den Daten war vorher nichts bekannt bzw. floss in die Gestaltung der Regeln und Zugehörigkeitsfunktionen ein. So kann geprüft werden, wie sich das System bei unbekanntem Daten verhält. Im Gegensatz zu der Anwendung der neuronalen Netze ist hier keine eindeutige Zustandsdiagnose möglich. Der aufgetretene Fehler im Zeitraum Ende 1997 wird zwar gut erkannt, jedoch suggerieren deutlich ansteigende Fehlerwahrscheinlichkeiten im späteren Verlauf nicht vorhandene Fehler hin.

Hier ist festzuhalten, dass die Anwendung mit neuronalen Netzen vorzuziehen ist. Die Übertragbarkeit zur Analyse häufiger Schadensfälle ist das wesentliche Ziel der vorliegenden Arbeit.

9.3 Referenzfall „Zwischenüberhitzer“

Der Referenzfall Zwischenüberhitzer enthält eine höhere Anzahl an Messparametern als der vorangegangene. Die Eingangsgrößen

- Massenstrom vor Zwischenüberhitzer
- Massenstrom hinter Zwischenüberhitzer
- Dampfdruck hinter ZÜ
- Kondensatablauf
- Differenzmassenstrom
- Notablauf Regelgefäß

wurden fuzzyfiziert. Entsprechend erhöht sich die Anzahl der erforderlichen Regeln, da jede Kombination der Eingangsgrößen mit der Ausgabe verknüpft werden muss. Hier zeigt sich ein weiterer Nachteil der Fuzzy-Methode, da mit zunehmender Anzahl an Eingangsparametern und in Abhängigkeit der Partitionierung, die Anzahl der Regeln stark zunimmt.

Analog der vorangehend beschriebenen Vorgehensweise wurden die einzelnen Schritte durchgeführt. Abbildung 9-6 zeigt das Ergebnis der Auswertung für den Zeitraum von 1997 bis 2006. Mitte 2001 ist eine Änderung der Fehlerwahrscheinlichkeit zu erkennen.

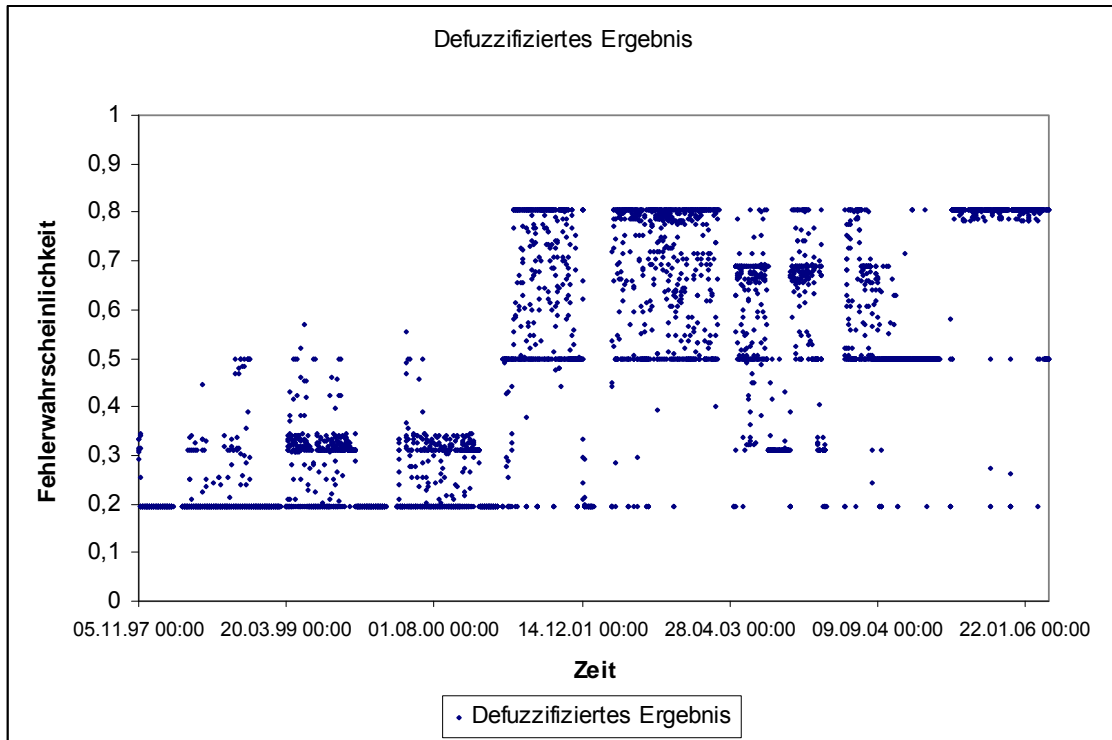


Abbildung 9-6: Ergebnis der Fuzzy Anwendung für RB20

Der Vergleich mit Abbildung 7-4, der entsprechenden Ausgabe des neuronalen Netzes, zeigt, dass zum Fehlerzeitpunkt eine Veränderung der Fehlerwahrscheinlichkeit erfolgt. Die starke Streuung erlaubt jedoch im Vergleich nur eine deutlich schlechtere Aussage bezüglich des Systemzustandes.

Die aufgeführten Beispiele zeigen, dass die Vergleichsmethode für die Anwendungen, bei denen die Auslegungsparameter vorhanden sind, ähnliche Ergebnisse zeigt. Die Übertragbarkeit ist aber deutlich schlechter.

10 Zusammenfassung und Ausblick

Die Ziele der Arbeit waren die Entwicklung einer Methodik zur Erkennung sich anbahnender Störungen und der erfolgreiche Einsatz neuronaler Netze zur Mustererkennung. Dabei entscheidend war die Verwendung realer Betriebsdaten, die auch den Einsatz im laufenden Betriebsgeschehen ermöglicht. Damit kann dem Kraftwerkspersonal eine Bewertungshilfe zur Verfügung gestellt werden, die es erleichtert, im komplexen Arbeitsumfeld spezielle Anlagenzustände besser zu erkennen bzw. bewerten zu können. Es wurde der Nachweis erbracht, dass die Berücksichtigung korrelierter Parameter einen erkennbaren Vorteil, und, im Falle der Früherkennung, eine notwendige Voraussetzung gegenüber der Verwendung der üblichen Grenzwertfestlegung einzelner Parameter darstellt.

Die Erarbeitung der Methodik zur Erkennung sich anbahnender Störungen erlaubt es, unabhängig vom vorliegendem Problemumfeld eine strukturierte Behandlung der Fragestellung vorzunehmen. Durch den modularen Aufbau können einzelne Schritte unabhängig voneinander modifiziert und erweitert werden. Dies vereinfacht beispielsweise die vergleichende Behandlung mittels Fuzzy Sets.

Die Implementierung realer Daten aus Anlagen war eine wichtige Vorgabe der Arbeit. Vorangehende Arbeiten mit neuronalen Netze für die Fehlerfrüherkennung beruhten ausschließlich auf simulierten Fehlerdaten. Somit stellt der erfolgreiche Einsatz realer betrieblicher Daten zur Früherkennung eine Neuerung dar. Dieser bislang in der Literatur noch nicht gezeigte Fortschritt erlaubt den Übergang von der theoretischen Bearbeitung hin zur betrieblichen Anwendung im Anlagenbetrieb.

Die Korrelation der Parameter beruht auf den physikalischen Zusammenhängen. Ein möglicher Weg, diese zu berücksichtigen, besteht darin, den technischen Prozess mathematisch zu modellieren. Dies ist, wie beschrieben, ein aufwendiger Prozess, der durch die Komplexität der technischen Prozesse oftmals nur unter Vereinfachungen umsetzbar ist. Die gemessenen Prozesssigna-

le andererseits spiegeln dabei die Prozessabläufe wider und ergeben so Muster, mit denen man den Prozesszustand beschreiben kann. Es wurde mit neuronalen Netzen eine Möglichkeit der Mustererkennung gewählt, die durch Auswertung vorhandener Betriebsdokumentation den Anlagenzustand bewertet. Da unterstellt werden kann, dass die gemessenen Signale sich bei Soll- und Fehlerzustand der Anlage voneinander unterscheiden und unterschiedliche Fehlerzustände unterschiedliche Signalmuster hervorrufen, kann so nicht nur eine Abweichung vom Sollzustand sondern auch die Art des aufgetretenen Fehlers diagnostiziert werden.

Da die Bewertung der Prozessvorgänge aus den Betriebsdokumentationen zwangsläufig nur in enger Zusammenarbeit mit dem Anlagenpersonal erfolgen kann, wurde deren Erfahrung praktisch konserviert und in abstrakter Weise im neuronalen Netz eingebunden.

Die entwickelte Methode wurde an vier technischen Teilsystemen von unterschiedlichen Kraftwerkstypen erprobt. Bei der ersten untersuchten Teilanlage handelte es sich um den Zwischenüberhitzer eines Siedewasserreaktors. Durch eine über längeren Zeitraum stärker werdende Leckage wurde die Leistung des Zwischenüberhitzers beeinflusst. Durch Berücksichtigung der identifizierten relevanten Parameter konnte die Leckage eindeutig identifiziert werden. Durch Anwendung des trainierten Netzes auf den redundanten Zwischenüberhitzer konnte die Übertragbarkeit nachgewiesen werden. Dies konnte ohne Nachkalibrierung, wie es etwa bei anderen Diagnosesystemen notwendig ist, durchgeführt werden.

In einem weiteren Anwendungsbeispiel wurde der Riss in einem Rohrkompensator zwischen Mittel- und Niederdruckturbine untersucht. Hier stellte sich heraus, dass es sich um einen Ausfall handelte, der nicht durch die installierte Messtechnik erfasst werden kann. Der zeitliche Verlauf der Störung erfolgte dabei so schnell, dass im Gegensatz zum Fall „Zwischenüberhitzer“ keine Änderung an den gemessenen Parametern detektiert werden konnte, die es erlaubte, eine rechtzeitige Erkennung durchzuführen.

Bei der Untersuchung des Lageschadens konnte sowohl die Übertragbarkeit auf baugleiche Anlagen nachgewiesen werden als auch ein Schaden detektiert werden, der nicht in der Betriebsdokumentation erfasst war. Bei diesem Beispiel war es mit Hilfe der zur Auswertung der Daten im Kraftwerk vorhandenen Diagnosesoftware möglich, die neue Methodik im Vergleich zum am Markt vorhandenen Technologie zu erproben. Es konnte gezeigt werden, dass trotz geringer Datenbasis bessere Ergebnisse bei der Diagnose erzielt werden konnten.

Der vierte Referenzfall betraf einen Luftvorwärmer. Hier war eine eindeutige Zustandsbestimmung durch die Differenzdruckauswertung wegen der verschiedenen Betriebszustände und Fahrweisen nicht möglich. Durch die Berücksichtigung der im Umfeld gemessenen Parameter konnten der Fehler nachgewiesen und die betrachteten Zustände separiert werden.

Es kann zusammengefasst werden, dass trotz einer geringen Datenmenge zu den aufgetretenen Fehlern eine erfolgreiche Erprobung an realen Anlagen möglich war. Die Beschränkung auf Nennlastdaten stellt, wie bereits angemerkt, keinen Abstrich bei den Ergebnissen dar, denn Nennlast stellt den Großteil der Betriebsfahrweise der entsprechenden Kraftwerke dar. Weiterhin werden durch die entwickelte Methodik Driftausfälle behandelt, die kein zeitkritisches Verhalten aufweisen und somit die Zustandsdiagnose im Nennlastfall ausreichend ist.

Der Vergleich mit der Fuzzy Methode verdeutlichte die Stärken des Einsatzes neuronaler Netze. Die Mustererkennung und der Umstand, dass keine speziellen Werte als Grenzen festgelegt werden, stellen insbesondere bei der Übertragbarkeit auf baugleiche Anlagen einen entscheidenden Vorteil dar.

Durch die entwickelte Vorgehensweise wird dem Anlagenfahrer eine Bewertungshilfe an die Hand gegeben, die es ihm ermöglicht durch eine Größe den Anlagenzustand zu bestimmen. Besonders in kritischen Situationen, bei denen das Personal unter Stress steht, können so Fehler vermieden werden.

Es können in Abhängigkeit von den Randbedingungen Gegenmaßnahmen getroffen oder geplant werden. Da das Erreichen von Grenzwerten und damit die

Häufigkeit des Ansprechens von Sicherheitseinrichtungen verringert werden, ist ein wirkungsvoller Beitrag zur Erhöhung der Sicherheit gegeben.

Mittels der trainierten Netze können die entsprechenden Anlagen auf den speziellen Fehler hin im laufenden Betrieb überwacht werden. Treten neue Ereignisse derselben Fehlerart auf und werden diese erkannt, so brauchen sie nicht nachtrainiert werden. Ausschließlich neue Fehlerarten müssen nachtrainiert werden. Da die Vorgehensweise jedoch so konzipiert ist, dass für jeden Fehler ein separates Netz erstellt wird, kann dies unabhängig von den bereits trainierten Fehlern erfolgen. Auch hier liegt ein Vorteil gegenüber den bislang in der Literatur verfolgten Strategien. Diese sahen meist die Implementierung aller Fehler in einem Netz vor, was zwangsläufig dazu führt, dass durch jeden neu auftretenden Fehler die komplette Netzwerkarchitektur neu erstellt werden muss und die bereits erfolgreichen Netze nicht weiter verwendet werden können.

Durch eine Erprobung der Anwendung im laufenden Kraftwerksbetrieb kann die erfolgreiche Umsetzung nachgewiesen werden.

Literaturverzeichnis

- /And97/ U. Anders; Statistische neuronale Netze; Verlag Vahlen München, 1997
- /Att96/ I.K. Attieh, A.V. Gribok, J.W. Hines, R.E. Uhrig; Pattern Recognition for Transient Detection to Enhance Nuclear Reactors' Operational Safety; Nuclear Engineering department; University of Tennessee Knoxville; 2000
- /Bau89/ E.B. Baum, D. Haussler; What size net gives valid generalization?; Neural Computation; Band 1;1989
- /Böh93/ G. Böhme, Fuzzy-Logik Einführung in die algebraischen und logischen Grundlagen, Springer Verlag Berlin Heidelberg; 1993
- /Bot97/ H.-H. Bothe; Neuro-Fuzzy-Methoden; Springer Verlag Berlin Heidelberg; 1997
- /Cha89/ Y. Chauvin; A back-propagation algorithm with optimal use of hidden units; Advances in Neural Information Processing Systems; Morgan Kaufmann; 1989
- /Cho90/ M. Chow, S.O. Yee; Real Time Application Of Artificial Neural Networks For Incipient Fault Detection Of Induction Machines; ACM New York; 1990
- /Fah89/ S.E. Fahlmann; An empirical study of learning speed in back-propagation networks; Proc. of the 1988 Connectionist Models Summer School; Carnegie Mellon University; 1988
- /Fah90/ S.E. Fahlmann, C. Lebiere; The cascade correlation learning architecture; Advances in Neural Information Processing Systems; Morgan-Kaufmann; 1990

- /FME06/ Analysetechniken für die Funktionsfähigkeit von Systemen – Verfahren für die Fehlerzustandsart- und –auswirkungsanalyse, DIN EN 60812, November 2006
- /Hau00/ U. Hauptmanns; Incipient fault detection using a neural network: Modelling and Simulation in Plant Safety II; Vorlesungsmaterial O-v-G-Universität Magdeburg; Magdeburg; 2000
- /Hof93/ N. Hoffmann; Neuronale Netze; Vieweg Verlag Braunschweig/Wiesbaden, 1993
- /Ise94/ R. Isermann; Überwachung und Fehlerdiagnose; VDI- Verlag Düsseldorf; 1994
- /Jur91/ M. Jurik; Backpercolation; wahrscheinlich unveröffentlichter Artikel von Jurik resarch and Consulting, PO 2379; Aptos, California USA; 1991
- /Kann04/ B. Knaup; Früherkennung betrieblicher Fehlerzustände in verfahrenstechnischen Anlagen; Dissertation OvG Universität Magdeburg, 2004
- /Kra00/ S. Krause, S. Baginski, H.-D. Kochs, M. Koch, G. Esser; Fehlerdetektion durch neuronale Netze bei der Schweißnahtüberwachung laserstrahlgeschweißter Tailored Blanks; atp, Nr. 42, Heft 7; 2000
- /Kru95/ R. Kruse, J. Gebhardt, F. Klawonn; Fuzzy Systeme; Teubner Verlag Stuttgart, 1995
- /Kru52/ W.H. Kruskal, W.A. Wallis, (1952); Use of ranks in one-criterion variance analysis. JASA, Vol. 47, S. 583-621
- /LeC90/ Y. LeCun, J.S. Denker; S.A. Solla; Optimal Brain Damage; D.S. Touretzky (Ed.): Advances in Neural Processing Systems 2 (NIPS-2), Morgan-Kaufmann, S. 598-605; 1990

- /Lia06/ M. Liangyu, M. Yongguang, M. Jin; Fault Diagnosis for the Feedwater Heater System of a 300MW Coal-Fired Power Generating Unit based on RBF Neural Network"; ICMLC 2005, LNAI 3930, S. 832-841; Springer-Verlag 2006
- /Mam74/ E.H. Mamdani; Applications of Fuzzy Algorithmus for Simple Dynamic Plant; Proc. IEEE 121:1585-1588, 1974
- /Moz89/ M.C. Mozer, P. Smolensky; Skeletonization; A technique for trimming the fat from a network via relevance assessment; D.S. Touretzky (Ed.): Advances in Neural Processing Systems 1; Morgan-Kaufmann; S.107-115; 1989
- /Nau96/ D. Nauck, F. Klawonn, R. Kruse; Neuronale Netze und Fuzzy-Systeme; Vieweg Verlag Braunschweig/Wiesbaden, 1996
- /Neu02/ J. Neumann, K. Schwarze-Benning, T. Wack, G. Deerberg, S. Schlüter; Umweltgerechte Prozessführung und Zustandserkennung in Chemieanlagen mit neuronalen Netzen, Teilvorhaben 1: Entwicklung und Validierung von Zustandserkennungswerkzeugen; Schlussbericht BMBF- Vorhaben 01 RV 9801/1;Fraunhofer Institut Oberhausen; 2002
- /PAA90/ Risikobegrenzung in der Chemie PAAG – Verfahren (HAZOP), Internationale Sektion der IVSS für die Verhütung von Berufsunfällen und Berufskrankheiten in der chemischen Industrie, Heidelberg 1990
- /Pat05/ K. Patan, T. Parisini; Identification of neural dynamic models for fault detection and isolation: the case of a real sugar evaporation process; Journal of Process Control No. 15; 2005
- /Pat97/ D. Patterson; Künstliche neuronale Netze; Prentice Hall Verlag München, 1997
- /Pow04/ Y. Power, P. A. Bahri; A two-step supervisory fault diagnosis framework; Computer & Chemical Engineering; 2004

- /Ren00/ R. Rengaswamy, V. Venkatasubramanian; A fast training neural network and its updation for incipient fault detection and diagnosis; Computers & Chemical Engineering 24/2000 S.434-437; 2000
- /Sch94/ T. Schreiner; Ausdünnungsverfahren für neuronale Netze; Diplomarbeit; Stuttgart 1994
- /Sri05/ R. Srinivasan, C. Wang, W.K. Ho, K.W. Lim; Context-based recognition of process states using neural networks; Chemical Engineering Science No. 60; 2005
- /Tiz98/ H. R. Tizhoosh, Fuzzy-Bildverarbeitung Einführung in Theorie und Praxis, Springer Verlag Berlin Heidelberg New York; 1998
- /Wat89/ K. Watanabe, I. Matsuura, M. Abe, M. Kubota, D.M. Himmelblau; Incipient Fault Diagnosis of Chemical Processes via Artificial Neural Networks; AIChE Journal, November 1989
- /Wolk90/ M.W. Wolkenstein; Entropie und Information; Akademie-Verlag Berlin; 1990
- /Zel97/ A. Zell; Simulation neuronaler Netze; R. Oldenburgerverlag München Wien, 1997

Anhang A: Routinen zur Aufbereitung Datensatz „Kompensator-schaden“

```
Public Sub löschen()  
  
Dim b As Date  
Dim a  
Dim start As Integer, i As Long, j As Long  
  
i = Workbooks.Count  
  
Workbooks(i).Activate  
  
start = ActiveCell.Row  
  
For i = start To 65536  
  
zeile3:  
    a = Cells(i, 1)  
    If IsNull(a) = True Then GoTo zeile1  
  
    On Error Resume Next  
    b = a  
    If Err.Number = 13 Then GoTo zeile1  
    On Error GoTo 0  
  
    If a = "" Then GoTo zeile1  
  
    GoTo zeile2  
  
zeile1:
```

	<pre> Rows(i).Delete Shift:=xlUp On Error GoTo 0 j = j + 1 If j > 200 Then GoTo zeile4 GoTo zeile3 zeile2: On Error GoTo 0 Cells(i, 1).Select j = 0 Next i zeile4: End Sub </pre>
	<pre> Sub suchundkopieren() , , anzahlDursch = 5 'Anzahl der Durchschnitt Reihen in Tabelle1 For i = 1 To anzahlDursch - 2 Sheets.Add Next Sheets("Tabelle1").Select For i = 2 To anzahlDursch + 1 Cells.Find(What:="Durchschn.", After:=ActiveCell, LookIn:=xlFormulas, - LookAt:=xlPart, SearchOrder:=xlByRows, SearchDirection:=xlNext, _ </pre>

```
MatchCase:=False, SearchFormat:=False).Activate
tmpRow = ActiveCell.Row
Range(Cells(1, 1), Cells(tmpRow, 13)).Select
Selection.Copy
Sheets("Tabelle" & i).Select
ActiveSheet.Paste
Sheets("Tabelle1").Select
'Range(Cells(1, 1), Cells(tmpRow, 13)).Select
'Range(Cells(1, 1), Cells(tmpRow, 13)).Select
'Selection.Delete
Rows("1:" & tmpRow).Select
Selection.Delete Shift:=xlUp
'Selection.Delete
Next
End Sub
```

Anhang B: Anlagenbilder Luftvorwärmer

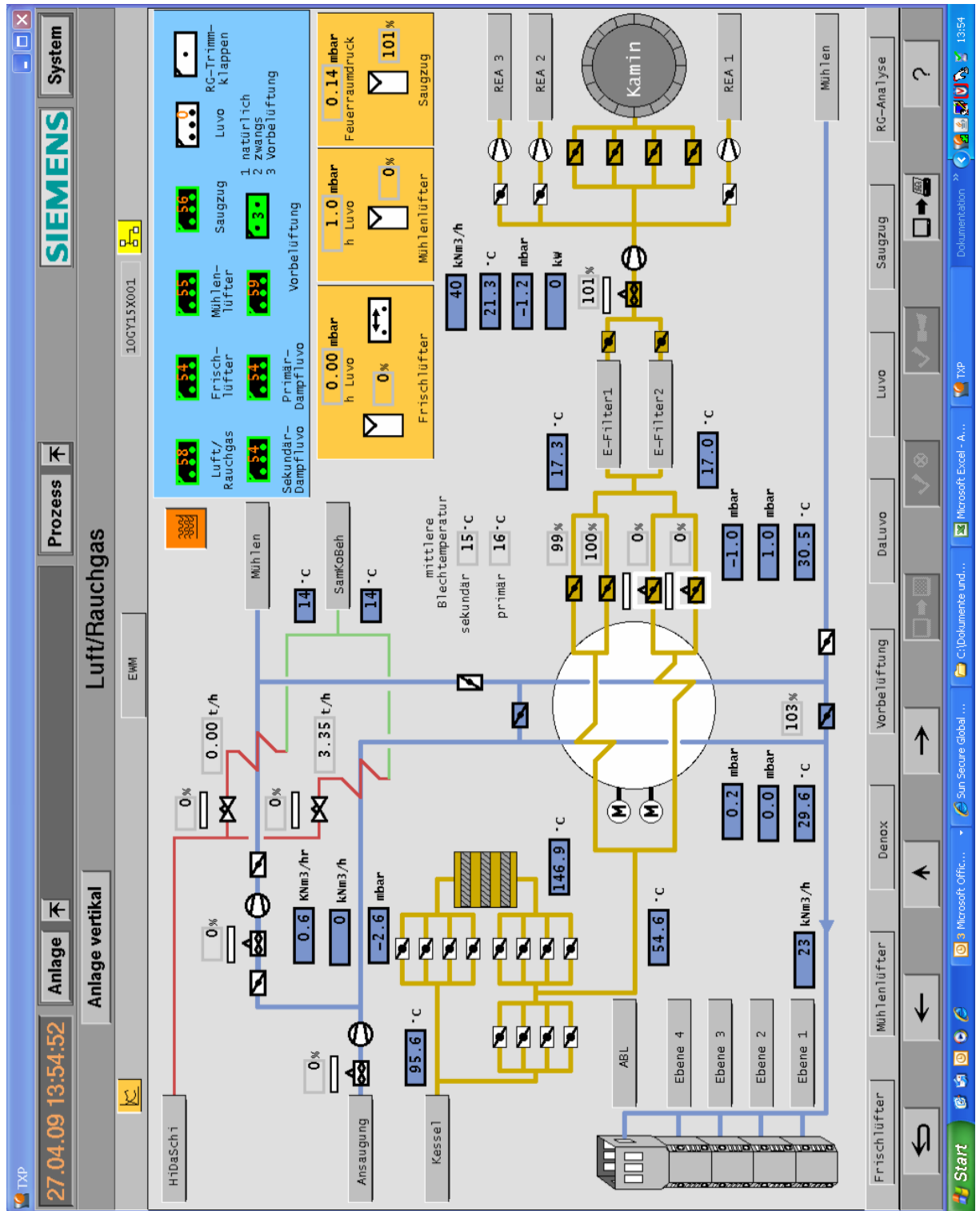


Bild B.1: Luftwege

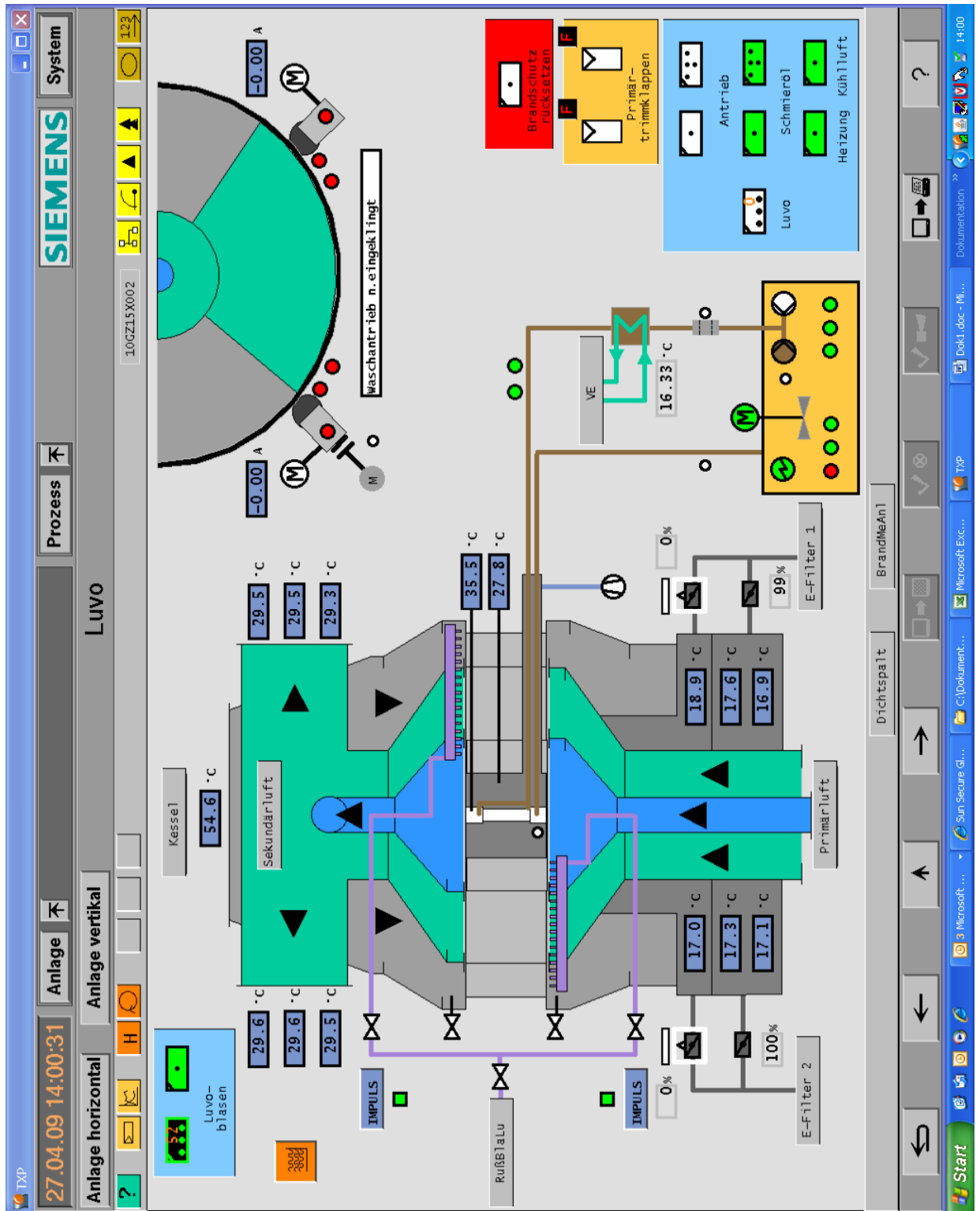


Bild B.2: Temperaturmessstellen

Lebenslauf

Name: Sascha Grünbeck
Geburtstag, -ort: 07.06.1977, Magdeburg
Familienstand: Ledig

Schulische Ausbildung

- Jun. 1995 • Erhalt der allgemeinen Hochschulreife
- Sep. 1990 • Gymnasium „Hegelgymnasium“
- Sep. 1983 • September 1983 Einschulung in die Oberschule „Hans Beimler“ in Magdeburg

Wehrdienst

- 1995 - 1996 • Grundwehrdienst bei der Luftlandepionierkompanie 270 in Wildeshausen

Studium

- 2004 – 2006 • OvG-Universität Magdeburg Masterstudiengang „Sicherheit und Gefahrenabwehr“
- 1996 - 2004 • Studium Energietechnik an OvG-Universität Magdeburg
Diplom als Ingenieur für Energietechnik

Berufstätigkeiterfahrung

- 2004 - 2010 • Wissenschaftlicher Mitarbeiter am Lehrstuhl für Anlagentechnik und Anlagensicherheit der Otto-von-Guericke-Universität Magdeburg