

Fakultät für Informatik
der Technischen Universität München

Routenauswahl in Transportnetzen

Iris Gilsdorf

Vollständiger Abdruck der von der Fakultät für Informatik
der Technischen Universität München zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigten Dissertation.

Vorsitzende: Univ.-Prof. G. J. Klinker, Ph.D.

Prüfer der Dissertation:

1. Univ.-Prof. Dr. Dr. h.c. W. Brauer
2. Univ.-Prof. Dr. E. Jessen, em.

Die Dissertation wurde am 20. Januar 2003 bei der Technischen Universität München
eingereicht und durch die Fakultät für Informatik
am 14. April 2003 angenommen.

Kurzfassung

Die automatische Routenauswahl in Telekommunikationsnetzen mit externen Managementsystemen, sogenannten Transportnetzen, stellt ein unzureichend gelöstes Problem dar.

Die Aufgabe kann reduziert werden auf die Bewertung und Auswahl von Routen in einem gerichteten Graphen - ausgehend von Bewertungen der Kanten und Knoten mit jeweils einer Reihe von sehr verschiedenartigen Parametern, die von den technischen und wirtschaftlichen Gegebenheiten abhängen. Ein wesentliches Problem der globalen Bewertung der Routen liegt darin, daß die Berücksichtigung der lokalen Kanten- und Knotenparameter nicht durch eine einfache Funktion beschrieben werden kann. So können die besten Routen nicht einfach an Hand eines aus den Parametern berechneten Distanzmaßes bestimmt werden.

Sowohl die Bewertung unter Berücksichtigung der lokalen Kanten- und Knotenparameter, als auch die Routenauswahl werden von Experten ausgeführt. Die Auslastung des Netzes und die kostengünstige Bereitstellung der Routen ist dabei von entscheidender Bedeutung. Spezielle strategische Gründe machen es notwendig, das im Regelfall angewandte Vorgehen bei der Routenauswahl temporär und lokal zu umgehen. Die Experten beherrschen diese strategische Feinsteuerung der Routenauswahl.

Automatische Routing-Verfahren, die bisher entwickelt wurden, berücksichtigen nur solche Parameter, aus denen sich ein Distanzmaß berechnen läßt. Als Konsequenz ergibt sich eine Routenauswahl, die wesentliche Aspekte nicht berücksichtigt.

In dieser Arbeit wird ein automatisches Routing-Verfahren vorgestellt, das das Verhalten der Experten nachbildet. Aus dem Expertenwissen wird ein regelbasiertes Fuzzy-System erstellt, das zur Bewertung des Graphen herangezogen wird. Kombiniert mit einem Kürzeste-Wege-Verfahren entsteht aus diesem Fuzzy-System ein flexibel steuerbares Routing-Verfahren. Auf Grundlage dieses Verfahrens entstehen weitere Varianten durch die zusätzliche Optimierung einzelner, additiver Parameter der Kanten. Die strategische Feinsteuerung der Routenauswahl wird durch die Einführung eines speziellen Kantenparameters und durch zusätzliche Fuzzy-Regeln ermöglicht.

Um empirisches Wissen für die Konstruktion des Fuzzy-Systems zu nutzen, wird das Fuzzy-System reversibel in ein neuronales Netz transformiert. Zur Adaption des Fuzzy-Systems an empirische Daten, wird speziell für diese Zwecke ein Trainingsalgorithmus entwickelt.

Der Erfolg der entwickelten Verfahren wird an Hand der Simulation der Routenauswahl für hochprioritäre Übertragungswege gezeigt.

Abstract

Automatic route choice in telecommunication networks using external management systems, also known as transport networks, is an unsatisfactorily solved problem.

The task can be reduced to weighting and choosing routes in a directed graph - starting with weighting edges and vertices using sets of very different parameters, parameters depending on technical and economic conditions. An essential problem of global route weighting is that the consideration of the local edges and vertices parameters cannot be described by a simple function. Therefore, the best route cannot be determined simply by means of a distance measure calculated from the parameters.

Both route weighting, which takes the local parameters of edges and vertices into account, and route choice are done by experts. For this task, consideration of the network's utilisation rate and of economical route provision is essential. Special strategic considerations require temporarily and locally bypassing the usual route choice process. Experts master this detailed fine-tuning of route choice.

Existing automatic routing algorithms take into account only parameters from which a distance measure can be calculated. Therefore, the choice of routes does not take into account essential aspects of route choice.

This thesis presents an automatic routing process emulating the experts' method. Using expert knowledge, a fuzzy rule-based system for weighting the graph is constructed. By combining the fuzzy-rule based system with a shortest path algorithm, a flexible and controllable routing process is created. Based on this process, more variants are provided by additionally optimising single, additive edge parameters. The introduction of a special edge parameter and additional fuzzy rules make strategic fine-tuning of route choice possible.

In order to use empirical knowledge for the construction of the fuzzy system, the fuzzy rule-based system is reversibly transformed into a neural network. A training algorithm has been especially developed for adapting the fuzzy system to empirical data.

The success of all these routing methods is demonstrated by simulating route choice for high priority link connections.

Danksagung

An dieser Stelle danke ich besonders meinem Doktorvater Herrn Professor Dr. Dr. h.c. Wilfried Brauer ganz herzlich für die kontinuierliche und engagierte Unterstützung und Förderung meiner Arbeit. Mit seinen wertvollen Fragen und Hinweisen verhalf er mir zu einem klareren und tieferen Verständnis der zu bearbeitenden Aufgabe.

Herrn Professor Dr. Eike Jessen möchte ich herzlich für die engagierten Diskussionen und die Erstellung des Zweitgutachtens danken.

Mein besonders herzlicher Dank gilt Herrn Dr. Hans Geiger. Es wird wohl immer unbeantwortet bleiben, ob ohne seine „Starthilfe“ in Form von begeisterten Diskussionen und Herstellung vieler Kontakte diese Arbeit überhaupt begonnen worden wäre. Trotz seines übervollen Terminkalenders hat er immer Zeit gefunden für wertvolle Diskussionen.

Herrn Professor Dr. Jochen Pfalzgraf, Institut für Computerwissenschaften der Universität Salzburg, danke ich für viele anregende Diskussionen und für die Möglichkeit, meine Arbeit im Rahmen eines Kolloquiums an der Universität Salzburg vorzustellen.

Die Mitglieder der Forschungsgruppe KI/Kognition haben mich als „Externe“ freundlich und ohne Vorbehalte in ihre Runde aufgenommen. Aus den Gesprächen habe ich viele fachliche Anregungen mitgenommen.

Allen KorrekturleserInnen danke ich dafür, daß sie diese mühevollen Arbeit übernommen haben und für ihre wertvollen Hinweise.

Auch meinen Arbeitskollegen gilt mein ganz besonderer Dank. In vielen Gesprächen und Diskussionen sind Lösungsideen für diese Arbeit entstanden.

Diese Arbeit wurde von meinem Arbeitgeber, der Firma Siemens Business Services GmbH & Co OHG gefördert. Daher bedanke ich mich insbesondere bei meiner Abteilung, die diese Förderung durchgeführt hat. Mein besonderer Dank gilt Herrn Bernd Rückmann und Herrn Georg Voss, die diese Förderung erst möglich gemacht haben.

Eine Promotion erfordert neben der fachlichen auch menschliche Unterstützung. Bei meiner Familie, allen Freunden und Bekannten, die mir während dieser Zeit Rückhalt gaben und mich motivierten, bedanke ich mich deshalb ganz herzlich.

Inhaltsverzeichnis

Kurzfassung	i
Abstract	iii
Danksagung	v
Abbildungsverzeichnis	xi
Tabellenverzeichnis	xv
KAPITEL 1 Einleitung	1
1.1 Problemstellung	1
1.2 Zielsetzung	2
1.3 Motivation der Lösung	3
1.4 Aufbau	4
KAPITEL 2 Anwendungsgebiet	5
2.1 Telekommunikationsnetze	5
2.2 PDH- und SDH-Technik	7
2.3 Netzmanagement	10
2.4 Transportnetze	11
2.5 Routenbestimmung in Transportnetzen	16
2.5.1 Aufgabe	16
2.5.2 Zielsetzung	19

KAPITEL 3	Routenbewertung im Layer-Netzwerk	23
3.1	Objekte und Voraussetzungen	23
3.1.1	Layer-Netzwerk als abstrakter Graph	23
3.1.2	Eigenschaften der Parameter	25
3.2	Vorhandene Lösungskonzepte	28
3.2.1	Kürzeste-Wege-Probleme mit einem Parameter	28
3.2.2	Kürzeste-Wege-Probleme mit mehreren Parametern	29
3.3	Analyse des Expertenverhaltens	31
3.4	Validierung	35
3.5	Fuzzy-Logik-basierte Lösungskonzepte	35
3.5.1	Fuzzy-Routing	37
3.5.2	FuzzyPlus-Routing	38
3.5.3	Fuzzy-Neuro-Verfahren	39
3.5.4	Einbettung existierender Verfahren	40
3.5.5	Fuzzy-Bewertung im Algorithmus von Dijkstra	40
KAPITEL 4	Regelbasiertes Fuzzy-System	43
4.1	Grundlagen der Fuzzy-Logik	43
4.2	Eingesetztes Fuzzy-System	50
KAPITEL 5	Fuzzy-Neuro-Verfahren	57
5.1	Ursprung künstlicher neuronaler Netze	57
5.2	Regelbasierte Fuzzy-Systeme als neuronale Netze	58
5.2.1	RBF-Netze	59
5.2.2	SRCM-Netze	63
5.3	Adaption eines regelbasierten Fuzzy-Systems	68
5.4	Fuzzy-Neuro-Verfahren zur Routenauswahl	70
5.4.1	Entscheidung für SRCM-Netze	70
5.4.2	Training	71
5.4.3	Lernalgorithmen für das neuronale Netz	73
5.5	Überblick über die realisierten Routing Verfahren	77
KAPITEL 6	Validierung	79
6.1	Simulationsumgebung	79
6.1.1	Software	80
6.1.2	Daten	84
6.2	Fuzzy-Routing	90

6.2.1	Beurteilung der gewählten Routen	91
6.2.2	Steuerung der Kantenwahl mittels strategischer Bedeutung	96
6.2.3	Auswirkung auf das Layer-Netzwerk bei längerem Einsatz	99
6.2.4	Zeitliche Veränderung der Kantenwahl	108
6.3	FuzzyPlus-Routing	114
6.3.1	Auswirkungen auf das Layer-Netzwerk bei längerem Einsatz	114
6.4	Fuzzy-Neuro-Verfahren	116
6.4.1	Adaption der Regelgewichte	118
6.5	Bewertung	126
6.5.1	Fuzzy-Routing	126
6.5.2	FuzzyPlus-Routing	127
6.5.3	Fuzzy-Neuro-Verfahren	127
KAPITEL 7	Diskussion und Ausblick	129
7.1	Rückblickende Beurteilung	129
7.2	Ansatzpunkte für zukünftige Arbeiten	131
7.2.1	Erweiterungen der Anwendung	131
7.2.2	Erweiterungen der Technologie	132
7.2.3	Schlußbemerkung	133
ANHANG A	Regelsystem	135
ANHANG B	Daten des Layer-Netzwerks	145
ANHANG C	Ergebnisse Fuzzy-Routing	151
ANHANG D	Ergebnisse FuzzyPlus-Routing	167
ANHANG E	Ergebnisse Fuzzy-Neuro-Verfahren	171
	Literaturverzeichnis	177
	Index	183

Abbildungsverzeichnis

2.1	Einheitliche Benutzung des STM-1 Carriers in allen Wirtschaftszonen	8
2.2	SDH-Multiplexschema nach CCITT G.709, [21]	9
2.3	Allgemeine Beziehung zwischen TMN und Telekommunikationsnetz nach M.3010, [7]	10
2.4	Beispiel für eine Partitionierung	12
2.5	Beispiel für das Erzeugen von „Link Connection“ mittels eines Trails	13
2.6	Übertragungsweg von Netzknoten vq nach Netzknoten vs	15
3.1	Auslastungsveränderung der Kanten bei Verwendung	25
3.2	Beispiel für die ungenaue Beurteilung von Routen durch die Verwendung des maximalen Auslastungswertes	27
3.3	Beispiel zur Demonstration der Routenauswahl der Experten	33
3.4	Parameterverwendung der Fuzzy-Bewertung	36
3.5	Schematische Darstellung der Arbeitsweise von Fuzzy-Routing	37
4.1	Regelbasiertes Fuzzy-System	43
4.2	Trapezfunktion	47
4.3	Beispiel einer linguistischen Variablen	47
4.4	Graphische Darstellung der Regelauswertung nach Mamdani	49
4.5	Beispiel einer Zugehörigkeitsfunktion eines linguistischen Wertes	52
4.6	Gekappte Zugehörigkeitsfunktionen	54
5.1	Generisches Neuron	58
5.2	eindimensionale Gaußfunktion	59

5.3	RBF-Netz für zwei-dimensionale Eingabe und eindimensionale Ausgabe	60
5.4	Netzarchitektur eines RBF-Netz zur Darstellung einer Fuzzy-Regelbasis	63
5.5	Aktivierungsfunktion eines SRCM-Neuron	64
5.6	Neuronenverschaltung für x_{kj}	65
5.7	Neuronenverschaltung für Prämissen	66
5.8	Netzarchitektur für das regelbasierte Fuzzy-System	67
5.9	Neuronenverschaltung für Defuzzifizierung nach Takagi-Sugeno	68
5.10	Beispiel zur Erläuterung des Ist- und Soll-Routen Vergleichs	71
5.11	Schematische Darstellung des Trainings	73
5.12	Vereinfachte Netzarchitektur für die Defuzzifizierung	75
6.1	Architektur der Simulationsumgebung	81
6.2	Anforderungen an die Verfügbarkeit	86
6.3	Ausschnitt aus dem Layer-Netzwerk	91
6.4	Zugehörigkeitsfunktion des linguistischen Wertes Δ_{BitErr_ecoBad}	96
6.5	Kanten zwischen F und Avi	96
6.6	Zugehörigkeitsfunktion von $beGr_m$	99
6.7	Auslastung der Kanten nach dem Einsatz von Fuzzy-Routing	100
6.8	Alternative Route für Ro - Pa	101
6.9	Auslastung der Kanten nach einem Einsatz mit Fuzzy-Routing; Auslastungsregeln doppelt gewichtet	102
6.10	Kosten der Knoten Knoten, die mit zwei Säulen gekennzeichnet sind, sind Knoten, die in einer Route verwendet werden.	103
6.11	Kanten um Athen und Thes2	104
6.12	Verfügbarkeit der verwendeten Kanten mit Fuzzy-Routing	105
6.13	Bitfehlerhäufigkeit der verwendeten Kanten mit Fuzzy-Routing	105
6.14	Verteilung der Bitfehlerhäufigkeit mit Fuzzy-Routing	107
6.15	Verteilung der Verfügbarkeit mit Fuzzy-Routing	108
6.16	Anzahl der Kanten mit 100% Auslastung	109
6.17	Anzahl Kanten mit 80 - 100% Auslastung	110

6.18	Anzahl Kanten mit 60 - 80% Auslastung	111
6.19	Anzahl der verwendeten Kanten, deren Verfügbarkeitswerte im gewünschten Bereich liegen	112
6.20	Anzahl der Kanten, deren Werte für Bitfehlerhäufigkeit im gewünschten Bereich liegen	113
6.21	Auslastung der Kanten mit FuzzyPlus-Routing	114
6.22	Auslastung der Knoten mit FuzzyPlus-Routing	115
6.23	Auslastung der Kanten mit FuzzyPlus-Routing mit doppelt gewichteten Auslastungsregeln	115
6.24	Kosten der Knoten Knoten mit zwei Säulen sind Knoten, die in einer Route verwendet werden	116
6.25	Datenfluß während des Trainings	117
6.26	Minigraph	118
6.27	Zum Training verwendeter Graph	119
6.28	Trainingserfolg mit Gradientenabstiegsverfahren	121
6.29	Trainingserfolg mit Delta-Lernen	121
6.30	Vergleich der Kantenauslastung des Testgraphen	122
6.31	Auslastung der Kanten nach Einsatz des trainierten Systems	124
6.32	Auslastung der Kanten nach dem Einsatz des untrainierten Systems	125
C.1	Auslastung der Kanten nach Einsatz von Length-Routing	151
C.2	Auslastung der Kanten nach Einsatz von Hops-Routing	152
C.3	Auslastung der Knoten nach Einsatz von Fuzzy-Routing mit gleicher Gewichtung aller Regeln	152
C.4	Auslastung der Knoten nach Einsatz von Fuzzy-Routing mit doppelter Gewichtung der Regeln für die Auslastung	153
C.5	Auslastung der Kanten nach 200 Schaltungen mit Fuzzy-Routing	153
C.6	Auslastung der Kanten nach 300 Schaltungen mit Fuzzy-Routing	154
C.7	Kosten der verwendeten Knoten nach Einsatz von Length-Routing	154
C.8	Kosten der verwendeten Knoten nach Einsatz von Hops-Routing	155
C.9	Verfügbarkeit der verwendeten Kanten mit Length-Routing	155
C.10	Bitfehlerhäufigkeit der verwendeten Kanten mit Length-Routing	156

C.11	Verfügbarkeit der verwendeten Kanten mit Hop-Routing	156
C.12	Bitfehlerhäufigkeit der verwendeten Kanten mit Hop-Routing	157
C.13	Verteilung Verfügbarkeit mit Length-Routing	157
C.14	Verteilung Verfügbarkeit mit Hop-Routing	158
C.15	Verteilung Bitfehlerhäufigkeit mit Length-Routing	158
C.16	Verteilung Bitfehlerhäufigkeit mit Hop-Routing	159
C.17	Anzahl der verwendeten Kanten, deren Verfügbarkeitswerte niedriger sind als die Mindestverfügbarkeit	164
C.18	Anzahl der Kanten, deren Verfügbarkeitswerte im wirtschaftlich schlechten Bereich liegen	164
C.19	Anzahl der verwendeten Kanten, deren Werte für die Bitfehlerhäufigkeit die maximal erlaubte überschreiten	165
C.20	Anzahl der verwendeten Kanten, deren Werte für die Bitfehlerhäufigkeit im wirtschaftlich schlechten Bereich liegen	165
D.1	Verfügbarkeit verwendeter Kanten mit FuzzyPlus-Routing	167
D.2	Bitfehlerhäufigkeit verwendeter Kanten mit FuzzyPlus-Routing	168
D.3	Verteilung der Verfügbarkeit mit FuzzyPlus-Routing	168
D.4	Verteilung der Bitfehlerhäufigkeit mit FuzzyPlus-Routing	169

Tabellenverzeichnis

2.1	PDH Hierarchie-Ebenen in verschiedenen Weltwirtschaftszonen	7
3.1	Wichtige Parameter der Kanten für Beispiel aus Abbildung 3.3	33
5.1	Soll-Kantenwerte für Beispiel	72
6.1	Parameter der Kanten des Layer-Netzwerks	84
6.2	Zusätzliche linguistische Variablen	87
6.3	Ausgabe-Parameter	88
6.4	Parameter der betroffenen Kanten und ihre Fuzzy-Bewertung für Beispiel 1	92
6.5	Parameter der betroffenen Kanten und ihre Fuzzy-Bewertung für Beispiel 1	93
6.6	Parameter der betroffenen Kanten und ihre Fuzzy-Bewertung für Beispiel 2	94
6.7	Kantenparameter und Fuzzy-Bewertung	97
6.8	Knotenparameter	97
6.9	Kantenparameter und Fuzzy-Bewertung der Kanten nach Einbau von Lyon2 als neuen Knoten	98
6.10	Anzahl der verwendeten Kanten mit Werten für die Bitfehlerhäufigkeit im angegebenen Bereich	106
6.11	Anzahl der verwendeten Kanten mit Werten für die Verfügbarkeit im angegebenen Bereich	107
6.12	Kanten und ihre Werte für Routen zwischen DA und Mar	120
6.13	Gewichte der Auslastungsregeln nach dem Training	122
6.14	Vergleich des Auslastungsverhaltens	123
6.15	Vergleich der Systeme bezüglich stark ausgelasteter Kanten	124
B.1	Daten der Kanten des Layer-Netzwerks	145

B.2	Daten der Knoten des Layer-Netzwerks	148
C.1	Auslastung der Kanten bei Einsatz von Fuzzy-Routing in Prozent	161
E.1	Daten der Kanten des Trainingsgraphen	171
E.2	Daten der Knoten des Trainingsgraphen	173
E.3	Auslastungswerte der Kanten des Trainingsgraphen	175

1.1 Problemstellung

Die vorliegende interdisziplinäre Arbeit stellt ein automatisches Verfahren zur Routenbestimmung in Transportnetzen. Transportnetze sind Telekommunikationsnetze, die ein externes Managementsystem haben. Die Problemstellung der umfassenden Routenbewertung, die aus der Telekommunikation stammt, wird mit Mitteln der Informatik gelöst. Diese Arbeit richtet sich also an Wissenschaftler aus beiden Disziplinen. Sie ist vor dem Hintergrund einer langjährigen Tätigkeit im Bereich der Telekommunikation Management Systeme der Firma Siemens Business Services entstanden.

Telekommunikationsnetze dienen der weltweiten Übertragung von Daten. Sie haben in den letzten Jahren stark an Bedeutung gewonnen. Die wachsende öffentliche und privatwirtschaftliche Abhängigkeit von Telekommunikationsnetzen verlangt eine große Zuverlässigkeit und hohe Verfügbarkeit der Netze. Durch die Marktöffnung und den Wegfall der Monopole sind andererseits die Betreiber dieser Netze einem hohen Wettbewerbsdruck ausgesetzt. Nur durch Automatisierung und den Einsatz modernster Technologie können die immer umfangreicher werdenden Aufgaben, die im Betrieb von Telekommunikationsnetzen anfallen, gewinnbringend gelöst werden.

Betreiber von Transportnetzen verwenden während der Planungsphase viel Zeit und Geld für die Konfiguration ihrer Netze mit dem Ziel einer verkehrsgerechten Netzdimensionierung und einer Kostenoptimierung.

Während der Einsatzphase wird ein solches Transportnetz durch das Schalten von Übertragungswegen verändert. Diese Schaltvorgänge sind notwendig, um neue Übertragungswege zu schalten und so die Kapazität des Transportnetzes den Marktanforderungen anzupassen, oder um Umschaltungen bestehender Übertragungswege vorzunehmen.

Transportnetze sind hierarchisch strukturiert, das heißt insbesondere, für jede Übertragungsrate existiert ein eigenes Netzwerk, genannt Layer-Netzwerk. Um einen Übertragungsweg zu schalten, muß für diesen Übertragungsweg im entsprechenden Layer-Netzwerk des Transportnetzes eine Route bestimmt werden. Jedes Layer-Netzwerk kann als Graph repräsentiert werden, dessen Knoten die Netzelemente sind und dessen Kanten die Menge der gleichartigen Kanäle zwischen je zwei Knoten sind. Kanten und Knoten sind jeweils mit sehr verschiedenartigen Parametern bewertet, die von den technischen und wirtschaftlichen Gegebenheiten abhängen. Ein wesentliches Problem der globalen Bewertung der Routen liegt darin, daß die Berücksichtigung der lokalen Kanten- und Knotenparameter nicht durch eine einfache Funktion beschrieben werden kann. So können die besten Routen nicht einfach

an Hand eines aus den Parametern berechneten Distanzmaßes als „kürzeste Wege“ im Graphen bestimmt werden.

Diese Routensuche wird in den Schaltzentren von erfahrenen Mitarbeitern der Betreiberfirma des Transportnetzes ausgeführt. Diesen Experten stehen langjährige Erfahrungen und ein gewachsenes Regelwerk zur Beurteilung von Kanten und Knoten zur Verfügung. Sie berücksichtigen auch die Regeln, die aus der Planungsphase in der Betriebsphase eine Rolle spielen.

Diesen Experten gelingt es, die Kanten und Knoten an Hand ihren vielfältigen Parametern zu bewerten und trotz des Fehlens eines Distanzmaßes von einer Kanten- und Knotenbewertung zu einer Routenbewertung zu gelangen.

Die Verwendung aller Parameter der Knoten und Kanten und das Hinzuziehen aller Beurteilungsaspekte gewährleistet, daß die Routen für die Übertragungswege der Netzphilosophie entsprechend gewählt werden. Unter anderem wird auf eine gleichmäßige Auslastung der Kanten und Knoten geachtet.

Um die vielfältigen Aufgaben zu lösen, die während des Betriebes eines Transportnetzes anfallen, werden „Telecommunications Management Network“-Systeme eingesetzt. Um die immer größer werdenden Performanceanforderungen an die Schaltvorgänge zu erfüllen, wird auch die Routensuche von diesen Systemen übernommen. So kommen Kürzeste-Wege-Algorithmen zum Einsatz.

Solche Algorithmen setzen jedoch die Additivität der Parameter entlang jeder Route voraus. Daher wählen viele der heute eingesetzten automatischen Routing-Verfahren einen passenden Parameter aus und verwenden diesen zur Kantenbewertung. So kann zum Beispiel die Routenauswahl die jeweils physikalisch kürzeste Route, oder die Route mit der minimalen Anzahl der Schaltpunkte bestimmen.

Es können auch Algorithmen zur Bestimmung kürzester Wege mit Nebenbedingungen eingesetzt werden, um gleichzeitig mehrere Parameter zu beachten oder Kürzeste-Wege-Algorithmen, die alle Pareto-optimalen Wege bestimmen. Auch bei diesem Vorgehen werden nur die Parameter herangezogen, die additiv entlang jeder Route sind. So kann nur ein Teil der Parameter bei der Routenauswahl berücksichtigt werden.

Wird zum Beispiel bei der Routenauswahl die Auslastung der Knoten nicht berücksichtigt, so kann das dazu führen, daß Knoten zu 100% vollgeschaltet werden, obwohl es passende alternative Routen geben würde. Zu diesen Knoten können nur dann weitere Übertragungswege geschaltet werden, wenn bestehende Übertragungswege umgeschaltet werden. Oder es werden neue Geräte aufgestellt, um diesen Engpaß auszugleichen. Beide Lösungen wirken sich negativ auf die Bilanz des Betreibers aus.

1.2 Zielsetzung

Ziel dieser Arbeit ist es, ein automatisches Routing-Verfahren für Layer-Netzwerke von Transportnetzen bereitzustellen, das die Vorteile des Expertenverhaltens nachbildet. Das bedeutet insbesondere die Verwendung aller wichtigen Parameter, auch der nicht-additiven Parameter, der Kanten und Knoten bei der Routenauswahl.

Der Experte, also ein „menschlicher Bewerter“, hat außer der Beachtung aller Parameter noch einen weiteren Vorteil. Er kann sehr flexibel arbeiten. Er kann temporär, aus strategischen Gründen, einzelne Kanten und Knoten nicht für Schaltungen verwenden, obwohl er sie nach seinen allgemeingültigen Regeln wählen würde, um sie später für eine spezielle Ver-

wendung zur Verfügung zu haben. Auch diese strategische Feinsteuerung der Routenauswahl soll mit dem neuen Verfahren möglich sein.

Da die betrachteten Transportnetze schon lange im Einsatz sind, kann davon ausgegangen werden, daß es Listen von Routen gibt, die als „gute“ Routen bekannt sind und die Aspekte bei der Auswahl verwenden, die noch nicht vollständig in den Regeln abgebildet sind. Das Wissen, das in Form solcher empirischer Daten vorliegt, soll dazu verwendet werden, das Routing-Verfahren zu verbessern.

Das technische und wirtschaftliche Umfeld, in dem die Betreiber von Transportnetzen operieren, ist starken Veränderungen unterworfen. Aus diesem Grund ist es möglich, daß sich die Anforderungen an die Routenauswahl ändern. Das Verfahren, das zur Routenauswahl eingesetzt wird, sollte daher schnell und flexibel an solche Veränderungen anzupassen sein.

1.3 Motivation der Lösung

Eine Analyse des Expertenverhaltens macht zwei Punkte deutlich:

- Die Bewertung der lokalen Objekte, der Kanten und Knoten, an Hand ihrer Parameter kann in verbale Regeln gefaßt werden.
- Von dieser lokalen Bewertung gelangt der Experte zu einer Bewertung der globalen Objekte, der Routen.

Daher liegt es nahe die automatische Routenauswahl in zwei Schritten durchzuführen.

Der erste Schritt besteht darin, die Bewertung der Experten nachzubilden. Da der Bewertungsvorgang an Hand von Regeln beschrieben werden kann, bietet sich der Einsatz von Fuzzy Logik an. Die verbalen Regeln werden dazu verwendet ein Fuzzy-Regelwerk zu formulieren. Das regelbasierte Fuzzy-System wird so konstruiert, daß es die Parameter jeder Kante und ihrer angrenzenden Knoten dazu verwendet, diese Kante zu bewerten. So entsteht ein kantenbewerteter Graph.

Im zweiten Schritt wird auf diesem fuzzy-bewerteten Graphen ein klassischer Kürzester-Wege-Algorithmus eingesetzt. Das so entstandene Verfahren zur automatischen Routenauswahl in Layer-Netzwerken, wird Fuzzy-Routing genannt.

Dieses Verfahren läßt sich auf verschiedene Weise erweitern. Jeder weitere additive Kantenparameter kann nun dazu verwendet werden, unter den Routen, die die Regeln am besten einhalten, die mit der kleinsten Kantensumme dieses zusätzlichen Parameters zu bestimmen. Beispielhaft für alle solche Verfahren wird in dieser Arbeit eine zusätzliche Kostenoptimierung vorgenommen. Das Verfahren wird mit FuzzyPlus-Routing bezeichnet.

Um empirische Daten zu nutzen, wird das automatische Routing-Verfahren zusätzlich lernfähig gemacht. Die Lernfähigkeit ist der Hauptvorteil neuronaler Netze. Sie lernen das Generalisieren an Hand von Beispieldaten.

Um diese Fähigkeit in das Verfahren zu integrieren, wird das regelbasierte Fuzzy-System reversibel in ein neuronales Netz transformiert. Die Reversibilität dient dazu, das veränderte System für einen Experten lesbar und eventuell auch korrigierbar zu machen.

Das durch die Transformation entstehende neuronale Netz erwartet als Trainingsmenge Soll-Werte für Kanten. Daher wird ein Algorithmus entwickelt, der aus der Liste von guten Routen, sogenannten globale Daten, lokale Daten, also Soll-Werte für Kanten erzeugt.

Dieser gesamte Komplex wird abkürzend mit Fuzzy-Neuro-Verfahren bezeichnet.

Die auf diese Weise entstehenden Verfahren erfüllen alle Anforderungen aus 1.2.

1.4 Aufbau

KAPITEL 2 stellt zunächst Begriffe und Fakten aus dem Bereich der Telekommunikation bereit, die für diese Arbeit von Bedeutung sind. Anschließend werden die Aufgabenstellung und die Zielsetzung aus der Sicht der Anwendung beschrieben.

KAPITEL 3 dient der Formulierung der praktischen Aufgabe als abstrakte, informatisch-mathematische Aufgabe. Die Aufgabe wird reduziert auf die Bewertung und Auswahl von Routen in einem gerichteten Graphen, dessen Kanten und Knoten mit jeweils einer Reihe von verschiedenartigen Parametern bewertet sind. Die Zielsetzung dieser Arbeit wird an Hand dieses Graphen und seiner Parameter definiert.

Vorhandene Lösungskonzepte werden untersucht. Nach der Analyse des Expertenverhaltens wird eine Lösungsarchitektur entwickelt. Neue Verfahren zur automatischen Routenauswahl, die auf dieser Architektur basieren, werden vorgestellt.

In KAPITEL 4 werden die Grundlagen für regelbasierte Fuzzy-Systeme vorgestellt. Das regelbasierte Fuzzy-System, das den Kern aller in dieser Arbeit entwickelten Verfahren bildet, wird beschrieben. Insbesondere wird erläutert, wie aus dem vorhandenen Expertenwissen ein Fuzzy-Regelwerk formuliert wird.

KAPITEL 5 stellt zwei Methoden zur reversiblen Transformation von regelbasierten Fuzzy-Systemen in neuronale Netze gegenüber und begründet die Auswahl der in dieser Arbeit verwendeten Methode. Anschließend wird beschrieben, wie für das Training des neuronalen Netzes aus globalen Daten auf lokale Daten geschlossen wird. Verschiedene Adaptionmöglichkeiten des neuronalen Netzes werden vorgestellt.

KAPITEL 6 stellt die Simulationsumgebung, das verwendete Layer-Netzwerk und die Fuzzy-Regelbasis vor. Anschließend wird die Überprüfung von Fuzzy-Routing und FuzzyPlus-Routing vorgestellt und die Ergebnisse erläutert. Die Validierung der Fuzzy-Neuro-Verfahren schließt das Kapitel ab.

KAPITEL 7 rundet die Arbeit mit einer rückblickenden Zusammenfassung und einer Diskussion möglicher Erweiterungen ab.

Die Anhänge enthalten Daten und Ergebnisse, auf die in KAPITEL 6 verwiesen wird.

Um die Zielsetzung dieser Arbeit zu definieren und den Anwendungsfall zu beschreiben, sind Begriffe und Fakten aus dem Bereich der Telekommunikation notwendig. Dieses Kapitel dient dazu, entsprechendes Wissen bereitzustellen.

In 2.1 wird eine Klassifizierung der Telekommunikationsnetze vorgenommen. In 2.2 werden die Charakteristika der Übertragungstechniken PDH und SDH erläutert und die besonderen Merkmale der SDH-Technik dargestellt. Die genaue Kenntnis dieser Technik ist nicht unbedingt für das weitere Verständnis notwendig. Der Vollständigkeit halber wird aber ein Überblick über die wichtigsten Fakten gegeben. Die Funktion des Netzmanagement und dessen Rolle für die Aufgabenstellung wird im darauffolgenden Absatz 2.3 erklärt. Nun stehen alle Begriffe für die Definition eines Transportnetzes zur Verfügung, das in 2.4 beschrieben und von anderen Netzen abgegrenzt wird. In 2.5 wird die Zielsetzung dieser Arbeit erläutert.

Es würde den Rahmen dieser Arbeit sprengen, alle oben erwähnten Themen erschöpfend darzustellen. Dem an weiteren Informationen interessierten Leser sei daher die Lektüre von [24] und [29] empfohlen.

2.1 Telekommunikationsnetze

Telekommunikationsnetze stellen Verbindungswege bereit, um Informationen in Form von Signalen von einem Ort A zu einem anderen Ort B zu transportieren.

Telekommunikationsnetze werden in öffentliche und private Netze eingeteilt. Private Netze dienen der Kommunikation eines abgegrenzten Teilnehmerkreises, wenn sich zum Beispiel alle beteiligten Geräte auf demselben Firmenstandort befinden. Öffentliche Netze stehen unter festgelegten, allgemein geltenden Geschäftsbedingungen zur allgemeinen Benutzung zur Verfügung.

Um eine weltweite Datenkommunikation zu ermöglichen, muß der Übergang, die sogenannte Interconnection der Kommunikationswege, zwischen den verschiedenen öffentlichen Netzen gewährleistet sein. Der reibungslose Datentransfer wird durch Interconnection-Verbindungen zwischen den Betreibern der verschiedenen öffentlichen Netze geregelt.

Diese Arbeit betrachtet ausschließlich öffentliche Telekommunikationsnetze. Im weiteren Text wird daher auf den Zusatz öffentlich verzichtet.

Ein **Telekommunikationsnetz** besteht aus Netzelementen und physikalischen Verbindungen zwischen den Netzelementen. Aus mathematischer Sicht kann es als Graph betrachtet

werden, dessen Kanten die Leitungen zur Signalübertragung und dessen Knoten die Netzelemente sind. **Netzelemente** können Vermittlungseinrichtungen, Router, Multiplexeinrichtungen, Leitungsendgeräte und vieles mehr sein, je nach Art und Ausbau des Netzes.

Telekommunikationsnetze können durch folgende Merkmale klassifiziert werden:

- **Netzstruktur**
Sie bezeichnet die Topologie und die Anordnung des Netzes. Es wird zum Beispiel zwischen unregelmäßigen Netzen, Ring- und Stern-Netzen unterschieden.
- **Versorgungsstruktur**
Das zu bedienende geographische Gebiet und die Lage zum Kunden wird als Unterscheidungsmerkmal herangezogen. Große Betreiber wie zum Beispiel die Deutsche Telekom unterscheiden zwischen Zugangsnetz und Verbindungsnetz. Bei den Computernetzen wird zwischen LAN (Local Area Network) und WAN (Wide Area Network) unterschieden.
- **Übertragungstechnik**
Als Unterscheidungsmerkmal dient die Art der Signalübertragung. Es kann sich um digitale oder analoge Signalübertragung handeln.
- **Multiplextechnik**
Unterscheidungsmerkmal ist der verwendete Multiplexprozeß. Multiplexen bezeichnet das strukturierte Zusammenfassen mehrerer Signale zu einem neuen Signal einer höheren Übertragungsrate¹⁾; der umgekehrte Vorgang, also das Zerlegen, wird mit Demultiplexen bezeichnet. Die Multiplextechniken unterscheiden sich darin, wie die niedrigeren Übertragungsraten in die höheren abgebildet werden und wie auf diese niedrigeren Übertragungsraten wieder zugegriffen werden kann. Sie lassen unterschiedlich standardisierte Übertragungsraten zu und verwenden verschiedene, signalbegleitende Zusatzinformation.
- **Übertragungsmedien**
Als Unterscheidungsmerkmal dient das Leitungsmedium wie zum Beispiel Kupfer, Glasfaser, Richtfunk und so weiter. Die Übertragungsmedien werden in zwei Gruppen unterteilt:
Leitergebundene Übertragung, wie bei Kupfer- oder Glasfaserkabeln,
und nichtleitergebundene Übertragung, wie Richtfunk.
- **Kommunikationsdienste**
Die technischen Eigenschaften des Netzes legen fest, welche Dienste übertragen werden können. Nach diesem Dienst wird das Netz bezeichnet.
Ein Beispiel ist ISDN (Integrated Service Digital Network). Hier wird die Sprachübertragung mit Nicht-Sprach-Diensten wie Videotex, Teletex und FAX, zusammengefaßt.
- **Managementtechnik**
Es gibt Netze mit externem, zentralisiertem Managementsystem und solche mit internem Managementsystem. Hier sind vor allem die Netze, die Internet-Technik verwenden, zu nennen. Auf Netze mit externen Managementsystemen wird in 2.3 und 2.4 genauer eingegangen.

Beispiele für Telekommunikationsnetze sind Mobilfunknetze, Zugangsnetze und Transport-

¹⁾Synonyme für Übertragungsrate sind auch Bitrate und Übertragungsgeschwindigkeit.

netze. Hier werden im wesentlichen Transportnetze betrachtet, daher wird die zugrundeliegende Technik in 2.2 und 2.3 genauer vorgestellt.

2.2 PDH- und SDH-Technik

Seit Anfang der siebziger Jahre verwenden digitale Übertragungssysteme die „Pulse Code Modulation Methode“ (PCM). Mehrere analoge Signale werden pulscodemuliert zu einem Zeitmultiplexsignal zusammengefaßt. 8 Bit lange Codewörter werden zyklisch zeitlich verschaltet. Auf der Basis von 8 kHz Wiederholfrequenz entsteht so eine Übertragungsrate von 64 kbit/s. Analoge Signale, wie zum Beispiel die menschliche Stimme, können so in binärer Form repräsentiert werden. Mehrere PCM-Kanäle lassen sich kombinieren und auf der selben Kupferader übertragen.

Aus mehreren PCM-Kanälen wird ein PDH-Signal aufgebaut. PDH bedeutet „Plesiochrone Digitale Hierarchie“. Sie wurde 1972 von CCITT¹⁾ standardisiert. Das Wort „plesiochron“ stammt aus dem Griechischen und bedeutet „fast synchron“. Verschiedene Geräte generieren nur fast gleiche Übertragungsraten, da deren Taktgeber innerhalb geringer Toleranzen variieren. Geräte für die plesiochrone digitale Übertragungstechnik wurden ab ca. 1982 entwickelt. Hierbei handelt es sich um Multiplexer und Geräte zur digitalen Signalübertragung, also Leitungsendgeräte und Zwischenregeneratoren.

In Europa beruht PDH-Technik auf dem PCM30 Grundsystem, das 30 digitale Datenkanäle mit je 64 kBit/s zusammen mit zwei Kanälen, die Kontrollinformation enthalten, zu einem Primärmultiplexsignal zusammenfaßt. Dieser sogenannte E1-Carrier hat eine Übertragungsrate von 2,048 MBit/s. Da dieses Standard-2 MBit/s-Signal nicht ausreicht, um den immer größer werdenden Datenverkehr zu bewältigen, werden jeweils mehrere dieser 2,048 MBit/s Signale zu höheren Übertragungsraten zusammengefaßt. In den USA und Japan setzt PDH auf dem DS1 mit 24 PCM-Kanälen auf. Das hier entstandene Primärmultiplexsignal hat eine Übertragungsrate von 1,544 MBit/s. So existieren drei differierende, plesiochrone, hierarchische Strukturen, siehe Tabelle 2.1. Dadurch entstanden unterschiedliche Hierarchie-Ebenen, die jeweils für die Belange in den entsprechenden Ländern optimal waren. Durchschaltungen zwischen zwei Ländern sind in dieser Technik nur auf der 64 kBit/s - Ebene möglich.

Europa	Nordamerika	Japan
2,048 MBit/s	1,544 MBit/s	1,544 MBit/s
8,448 MBit/s	6,312 MBit/s	6,312 MBit/s
34,368 MBit/s	44,736 MBit/s	32,064 MBit/s
139,264 MBit/s	139,264 MBit/s	97,728 MBit/s

Tab. 2.1 PDH Hierarchie-Ebenen in verschiedenen Weltwirtschaftszonen

¹⁾CCITT ist die Abkürzung für Comité Consultatif International de Télégraphique et Téléphonique. CCITT ist das Organ der ITU (International Telecommunication Union) für Telefon- und Datenkommunikationssysteme [49].

Die PDH-Technik weist weitere Nachteile auf. Um zum Beispiel einem Kunden eine 2 MBit/s Leitung zur Verfügung zu stellen, die in einer Leitung von höherer Übertragungsrate geführt wird, muß aus dieser Leitung von höherer Übertragungsrate dieses 2 MBit/s Signal abgezweigt werden. Der direkte Zugriff auf diese 2 MBit/s Leitung ist nicht möglich. Um auf einzelne Signale zugreifen zu können, muß das Signal zunächst über alle Hierarchiestufen aufgebrochen und anschließend für den weiteren Transport wieder in eine höhere Übertragungsrate eingebettet werden. Das Durchlaufen dieser gesamten Mux-Demux-Kette macht das Verfahren teuer und unflexibel.

Außerdem handelt es sich bei den PDH-Geräten überwiegend um Geräte mit manuell umkonfigurierbaren Steckverbindungen. Zur Herstellung von Übertragungswegen müssen diese Geräte manuell mit Verteilern verbunden werden. Das Herstellen von Verbindungen ist entsprechend aufwendig und unflexibel. Ersatzschaltungen können oft Stunden, wenn nicht Tage dauern. Schnelles Fehlermanagement und Performance-Kontrolle ist oft nicht möglich.

Um eine Technik ohne die Nachteile der PDH-Technik bereitzustellen, wurde vom CCITT die SDH (Synchrone Digitale Hierarchie) entwickelt. Sie kann als nächster Schritt in der Evolution der Übertragungshierarchie angesehen werden. Parallel zu dieser Entwicklung wurden starke Standardisierungsanstrengungen unternommen. Als Hauptergebnisse sind die CCITT-Empfehlungen G.707, G.708 und G.709 zu nennen, die 1989 veröffentlicht wurden. In den USA verabschiedete ANSI den SONET (Synchronous Optical Network) -Standard, der heute als eine Teilmenge des weltweiten SDH-Standards angesehen werden kann.

Seit der Einführung der SDH-Technik wurde die Weiterentwicklung von PDH-Geräten praktisch gestoppt.

SDH definiert eine Struktur, die es möglich macht, PDH-Signale zusammenzufassen und in ein Standard-SDH-Signal zu verpacken. Dadurch können die PDH-Geräte weiterhin in einem SDH-Netzwerk verwendet werden und ein bestehendes PDH-Netz kann den Anforderungen entsprechend nach und nach um die neuen SDH-Geräte ergänzt werden. In das synchrone Transportmodul der Ebene 1, STM-1, das SDH-Signal unterster Stufe, können alle weltweit gängigen PDH-Signale integriert werden. Außerdem ist es damit möglich, Signale der europäischen Hierarchie mit Signalen der amerikanischen PDH-Hierarchie zu kombinieren. Eine solche Vereinheitlichung hat es vorher nicht gegeben.

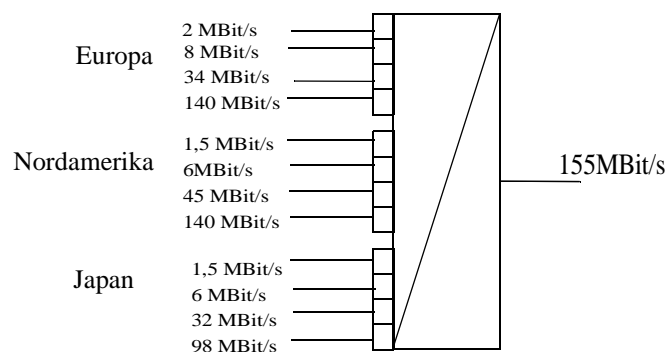


Abb. 2.1 Einheitliche Benutzung des STM-1 Carriers in allen Wirtschaftszonen

Das Transportmodul STM-1 ist als Container konzipiert, der selbst weitere kleinere Container unterschiedlicher Größe aufnehmen kann. Die einzelnen Container sind genau gekennzeichnet, so daß ein einfacher Zugriff auf diese Container jederzeit möglich ist. Die

Container korrespondieren mit je einem der existierenden PDH-Übertragungsraten. Die Nutzinformation des PDH-Signals wird in den entsprechenden Container abgebildet. Dort kann direkt über Zeigerstrukturen auf das Signal zugegriffen werden. Es ist nicht mehr erforderlich, das gesamte Signal zu demultiplexen.

STM-1 ist skalierbar, so daß praktisch beliebig hohe Übertragungsraten erzielt werden können. Die Abbildung 2.2 zeigt eine schematische Darstellung der synchronen Multiplexstruktur.

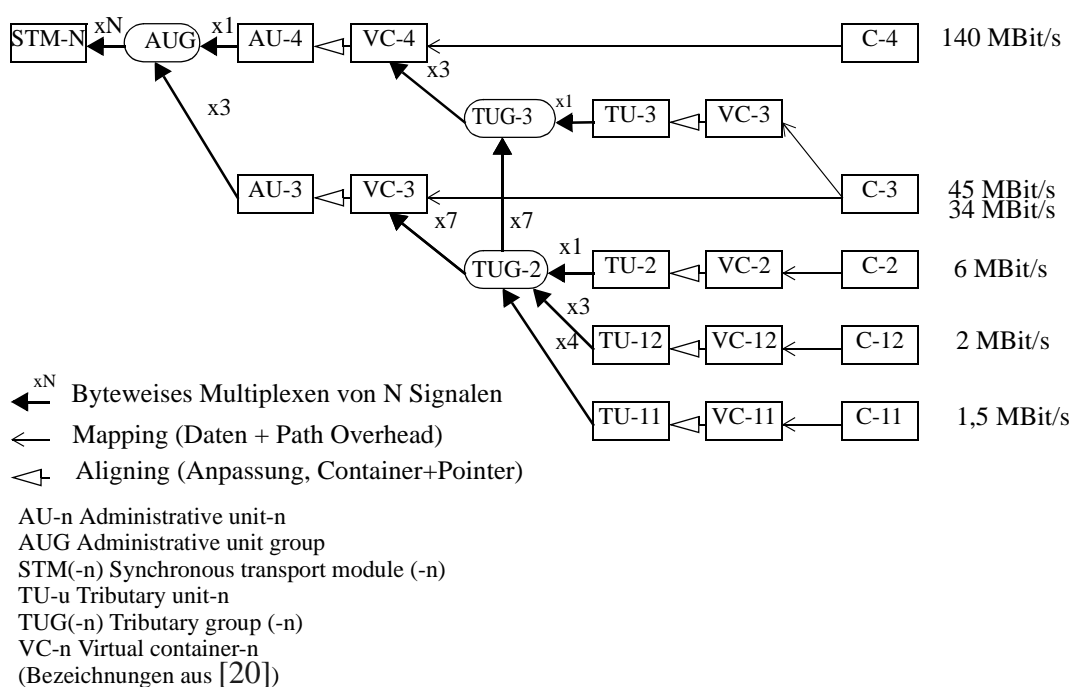


Abb. 2.2 SDH-Multiplexschema nach CCITT G.709, [21]

SDH-Geräte bestehen wie auch PDH-Geräte grundsätzlich aus einem Multiplex-, einem Leitungs- und einem Überwachungsteil. Es sind Multiplexer, Add-Drop-Multiplexer und Cross-Connects. Sie benötigen eine externe Taktzuführung. Im Gegensatz zu den PDH-Geräten besitzen sie eine externe Datenleitung zur digitalen Steuerung. Diese Datenleitung ist logisch getrennt von den Kanälen der Übertragungstechnik und kann von einem Managementsystem genutzt werden.

Die Vorteile der SDH-Technik sind also:

- Standardisierte Hierarchieebenen
- Einfache Einfüge- und Abzweigmöglichkeiten von Signalen; Zugriff durch Zeigertechnik ohne das Durchlaufen der gesamten Mux-Demux-Kette
- Möglichkeit eines zentralen Netzmanagements.

Die SDH-Übertragungstechnik deckt mehr und mehr Bereiche des Transportnetzes ab, siehe 2.4. In PDH-Technik werden keine neuen Investitionen getätigt. So ist es sinnvoll, sich bei der Routenbewertung auf Transportnetze der SDH-Technik zu konzentrieren.

Weitere Details zur PDH- und SDH-Technik können in [9] nachgelesen werden.

2.3 Netzmanagement

Telekommunikationsnetze müssen sowohl Anforderungen der Netzbetreiber, als auch die von deren Kunden erfüllen. Das Netzmanagement ist verantwortlich für einen ausfallsicheren Netzbetrieb und für die Gewährleistung der Dienste, die das Netz bereitstellt. Es umfaßt die Steuerung und Überwachung des Netzes, dient der betriebswirtschaftlichen Beobachtung.

Es gibt zwei verschiedene Arten, ein Netz zu managen. Bei der ersten Art, dem sogenannten Selbstmanagement, beinhaltet das Netz selber alle Fähigkeiten zur eigenen Steuerung. Die Steuerinformation wird in den Netzelementen ausgewertet, verarbeitet und weitergesendet. Alle Managementintelligenz steckt somit im Netz.

Bei der zweiten Art ist das Managementsystem nicht Bestandteil des Netzes. Das Netz wird von außen verwaltet. Das Managementsystem hat so eine globale Sicht auf das Netz. Für die vorliegende Arbeit spielen nur die zuletzt genannten Managementsysteme eine Rolle.

„Telecommunications Management Networks“, kurz TMN genannt, gehören zu den Managementsystemen, die eine globale Sicht auf das Netz haben.

Für die komplexen Aufgaben, die in Managementsystemen von Transportnetzen anfallen, werden computergestützte TMN-Systeme eingesetzt. Um beim Einsatz der angebotenen TMN-Systeme Heterogenität zu erlauben, bedarf es anerkannter Standards. In der CCITT-Empfehlung M.3010, [7], werden die Prinzipien für TMN festgelegt. Hier wird das Zusammenspiel der Netzelemente des Telekommunikationsnetzes mit dem Managementsystem beschrieben. Ein TMN ist ein vollkommen separates Netz mit diversen Schnittstellen zu dem zu managenden Telekommunikationsnetz. Ein TMN umfaßt unter anderem Operationssysteme und ein eigenes Datennetz zur Kommunikation mit den Netzelementen des Telekommunikationsnetzes. Die Abbildung 2.3 zeigt den schematischen Aufbau eines solchen TMN.

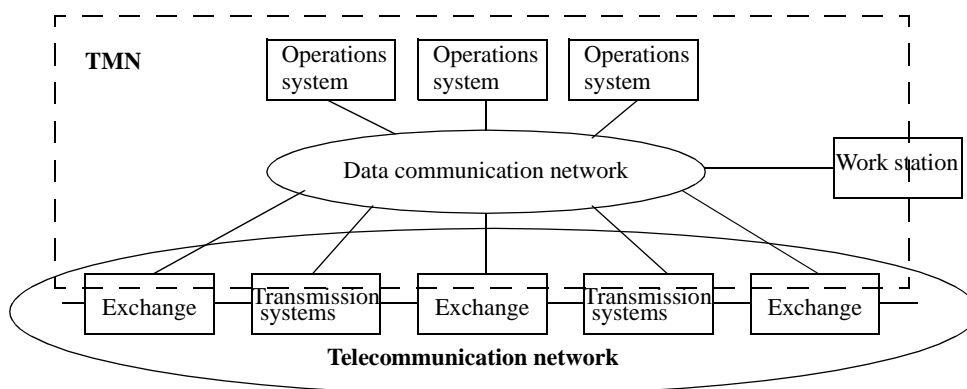


Abb. 2.3 Allgemeine Beziehung zwischen TMN und Telekommunikationsnetz nach M.3010, [7]

TMN umfaßt fünf Management-Funktionsbereiche:

- Fehlermanagement („Fault Management“, FM)
Es dient der Überwachung der einzelnen Netzkomponenten. Durch eine schnelle Fehlererkennung und -behebung wird die Verfügbarkeit des überwachten Netzes gewährleistet.
- Konfigurationsmanagement („Configuration Management“, CM)
Die Aufgabe des CM umfaßt die Funktionen zur Ressourcenverwaltung. Dazu gehört die Einstellung von Systemparametern, die Vorbereitung von Telekommunikationsverbindungen, deren Schaltung, deren Inbetriebnahme und die Beendigung von Telekommunikationsverbindungen.
Die Vorbereitung einer Telekommunikationsverbindung umfaßt unter anderem die Bestimmung einer Route für diese Verbindung.
- Abrechnungsmanagement („Accounting Management“, AM)
Unter AM wird die Gebührenerfassung und -berechnung verstanden. Die Kosten, die durch die Bereitstellung einer Telekommunikationsverbindung entstehen, müssen dem Kostenverursacher zugeordnet werden können. Da sich die Kosten nach der Abrechnungspolitik des jeweiligen Betreibers richten, ist eine Anforderung an das AM, es gemäß der Abrechnungspolitik konfigurieren zu können.
- Leistungsmanagement („Performance Management“, PM)
Wichtige Aufgaben des PM sind die Bestimmung von Dienstgüteparametern, das Überwachen des Telekommunikationsnetzes im Hinblick auf Leistungsengpässe und das Erfassen von Daten zur Leistungs- und Kapazitätsplanung.
- Sicherheitsmanagement („Security Management“, SM)
Das SM ist verantwortlich für die Sicherstellung der Netzintegrität. Es beinhaltet die Zugangs- und Zugriffssicherung und die Integritätsicherung der Datenübertragung.

2.4 Transportnetze

Die Fragestellung, die im weiteren untersucht wird, beschäftigt sich mit Telekommunikationsnetzen, die ein externes Managementsystem haben und zu den öffentlichen Netzen gehören. Ihre Übertragungs- und Multiplextechnik basiert auf PDH und SDH. Von der Versorgungsstruktur her sind es Verbindungsnetze. Solche Netze werden im weiteren Text als **Transportnetze**¹⁾ bezeichnet.

Ein Beispiel für ein Transportnetz ist das PDH/SDH-Transportnetz der Deutschen Telekom. Dieses Netz bestand bis 1995 aus einem Netz von Analog-Netzelementen und PDH-Netzelementen. Die Übertragungsmedien dieses Netzes waren zunächst symmetrische Kupferkabel, später koaxiale Kupferkabel und Richtfunkstrecken. Sie werden mehr und mehr durch Glasfaserkabel ersetzt, da mit diesen eine Übertragung auf hoher Übertragungsrates über große Entfernungen ohne Regeneratoren möglich ist. Um den steigenden Qualitätsanforderungen der Datendienste gerecht zu werden, wird die bestehende PDH-Übertragungstechnik durch die SDH-Technik ersetzt.

¹⁾Transportnetze haben nichts mit dem Transportdienst der Schicht 4 des ISO-OSI-Referenzmodells zu tun; sie können der Schicht 1 zugeordnet werden.

Ein Transportnetz überträgt die Daten aller bereitgestellten Dienste über den, durch die Netzelemente und Übertragungsmedien gebildeten Graphen. In den Netzelementen findet unter anderem das Multiplexen und Demultiplexen der Datenströme statt. Bei den Übertragungsmedien handelt es sich um Kabel oder Funkstrecken. Für die Übertragung der Daten stehen Kanäle unterschiedlicher Übertragungsraten zur Verfügung. Aufgabe des Netzmanagements ist es unter anderem, auf eine gleichmäßige Auslastung und auf genügend freie, fehlerfreie Ressourcen zu achten.

Ein Transportnetz ist ein sehr komplexes Netz. Um ein solches Netz zu managen bzw. Managementsysteme zu realisieren, bedarf es einer funktionalen und logischen Beschreibung. Um eine einheitliche Beschreibung zu gewährleisten, wurde die ITU-T-G.805 „Generic Functional Architecture of Transport Networks“, [23], erstellt. Sie arbeitet mit dem Konzept der Partitionierung („partitioning“) und der Schichtung („layering“) der Netze in ihre logischen Komponenten, um das komplexe Transportnetz mit seinen vielen verschiedenen Komponenten besser handhaben zu können. Mit den hier und in der ITU-T-G.803 „Architecture of Transport Networks based on the Synchronous Digital Hierarchy (SDH)“, [22], bereitgestellten Mitteln ist es möglich, ein Transportnetz vollständig und übersichtlich zu beschreiben. Mit Hilfe der dort definierten topologischen Komponenten, wie zum Beispiel Layer-Netzwerk („layer network“), Subnetz („subnetwork“) und „Link“, ist es möglich, die logische Topologie eines Transportnetzes komplett zu beschreiben.

Die Partitionierung dient dazu, das Netz in übersichtliche Subnetze zu unterteilen. Subnetze können wieder als Netze angesehen werden und somit in weitere Subnetze partitioniert werden. „Links“ stellen die topologische Beziehung und die verfügbare Transportkapazität zwischen zwei Subnetzen dar. Die Details können in den oben angegebenen Standards nachgelesen werden.

Der Betreiber legt bei der Modellierung seines Transportnetzes fest, welche Subnetze es gibt und was in den Subnetzen enthalten ist. Netzelemente werden oft als Subnetze modelliert.

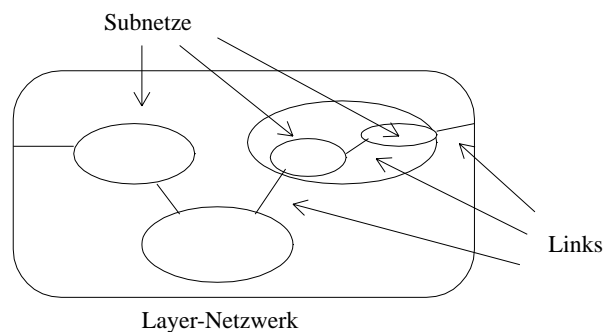


Abb. 2.4 Beispiel für eine Partitionierung

Die Schichtung des Transportnetzes trägt seiner hierarchischen Struktur Rechnung. Signale niedrigerer Übertragungsrate werden durch Multiplexen in Signale höherer Übertragungsrate eingebettet. Nach dem Layering-Konzept bildet jede Übertragungsrate ihr eigenes spezielles Layer-Netzwerk.

Die Partitionierung überträgt sich auf jedes Layer-Netzwerk. Abbildung 2.4 zeigt beispielhaft ein Layer-Netzwerk und seine Partitionierung, die durch die Partitionierung des Transportnetzes entsteht. Angrenzende Layer-Netzwerke haben jeweils eine Client-Server-Beziehung. Am Beispiel der Layer-Netzwerke für die Übertragungsrate 34 MBit/s und

8 MBit/s ist diese Client-Server-Beziehung in Abbildung 2.5 dargestellt und wird im Nachfolgenden noch näher erläutert.

„Transport Entities“ sind Verbindungen („Connections“), die Daten transparent in einem Layer-Netzwerk transportieren, ohne irgendeine Veränderung an den Daten, die die reine Information enthalten, vorzunehmen. Jedoch können Transportinformationen hinzugefügt werden, um den Transport im angrenzenden Layer-Netzwerk zu ermöglichen. „Transport Entities“ enden jeweils an speziellen Punkten, die zu einem Subnetz gehören. Auf diese Punkte wird hier nicht weiter eingegangen, da sie für diese Aufgabenstellung keine Rolle spielen. Es wird vereinfachend, aber nicht einschränkend damit gearbeitet, daß alle diese genannten „Transport Entities“ an Subnetzen enden.

Der Begriff „Transport Entities“ umfaßt unter anderem „Link Connections“, „Network Connections“, „Subnetwork Connections“ und „Trails“.

„Link Connections“ können Informationen transparent über einen „Link“ transportieren.

Eine „Network Connection“ steht zum transparenten Transport von Information über ein Layer-Netzwerk zur Verfügung. Diese „Transport Entity“ besteht aus einer Reihe von „Link Connections“ und/oder „Subnetwork Connections“.

Für eine „Subnetwork Connection“ gilt im allgemeinen das Gleiche wie für eine „Network Connection“ mit dem Unterschied, daß die Information über ein Subnetz transportiert werden kann.

Ein „Trail“ wird aus einer „Network Connection“ plus der entsprechenden Trailertermination gebildet. Mit Hilfe der Trailertermination werden die Daten für den angrenzenden Clientlayer aufbereitet.

Durch das Erzeugen eines „Trails“ im Server Layer-Netzwerk werden in diesem „Link Connections“ belegt und im angrenzenden Client Layer-Netzwerk freie „Link Connections“ erzeugt - siehe Abbildung 2.5. Diese stehen für das Schalten anderer „Transport Entities“ bereit. „Link Connections“ stellen Netzressourcen dar.

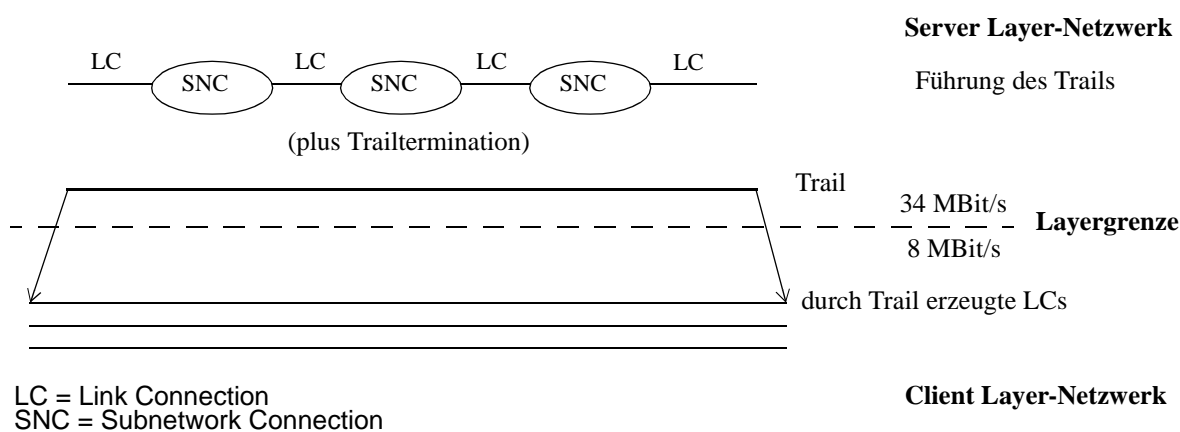


Abb. 2.5 Beispiel für das Erzeugen von „Link Connection“ mittels eines Trails

Auf eine Problematik soll hier besonders hingewiesen werden. Ein „Trail“ kann über mehrere Subnetze geschaltet werden. Er verwendet in diesem Fall als Führung eine Folge von „Link Connections“ und „Subnetwork Connections“. Er kann auf diese Weise in seiner Führung mehrere „Subnetwork Connections“ haben, die bei den entstehenden „Link Connections“ im Client Layer-Netzwerk nicht mehr zu sehen sind. Insbesondere sind also in der „Link Connection“ des Client Layer-Netzwerk nicht mehr die in der Route des „Trails“ vorkommenden Subnetze zu sehen. Abbildung 2.5 veranschaulicht dieses Verhalten. Um sämtli-

che Subnetze zu ermitteln, die in einer „Link Connection“ vorkommen, muß nicht nur das Layer-Netzwerk betrachtet werden, zu dem diese „Link Connection“ durch ihrer Übertragungsrate gehört, sondern auch alle Layer-Netzwerke, die mittels der Führungen betroffen sind.

Genauer soll auf die Definition der Begriffe aus den Standards hier nicht eingegangen werden. Der interessierte Leser findet die exakten Definitionen und entsprechende Beispiele in [22] und [23].

Um englische Begriffe aus den Standards durch deutsche zu ersetzen und um einige Begriffe, die für diese Arbeit sinnvoll sind, einzuführen, werden für den weiteren Text die nachfolgenden Bezeichnungen festgelegt:

Link Connections werden als **Kanäle** bezeichnet. Signale, zum Beispiel für Kundenservices, werden also in Kanälen geführt.

Das Layer-Netzwerk wird als Graph dargestellt, dessen Knoten die Subnetze sind und dessen **Kanten** aus Kanälen zwischen den jeweiligen Subnetzen bestehen. Dieser Graph kann sich verändern; durch das Erzeugen neuer „Trails“ im übergeordneten Layer-Netzwerk entstehen neue Kanäle und somit auch Kanten. Ist das übergeordnete Layer-Netzwerk das physikalische, so entstehen neue Knoten und Kanten durch Aufstellen neuer Netzelemente und das Einrichten neuer Übertragungsmedien.

Da Subnetze die Knoten des Graphen darstellen, der das Layer-Netzwerk für eine bestimmte Übertragungsrate repräsentiert, werden die Subnetze mit **Netzknoten** bezeichnet. Diese Bezeichnung soll verdeutlichen, daß die Netz-Eigenschaft der Netzknoten bei der Sicht auf den Graphen keine Rolle spielt. Die Netzknoten können auch Netze, genauer Layer-Netzwerke sein. Diese Netzsicht spielt erst bei der nächsten Rekursionstufe eine Rolle. Die Aufgaben, die zu lösen sind, können mit Hilfe der Partitionierung solange rekursiv ausgeführt werden, bis die Subnetze nicht mehr unterteilt werden können. Das ist der Fall, wenn sie die Sicht auf ein einzelnes Netzelement repräsentieren, das durch ein nicht weiter aufzuteilendes Gerät dargestellt wird. Genauer und in der Terminologie des Standards gesagt, wenn das Subnetz eine Matrix ist.

Verbindungen im Layer-Netzwerk, die ein Signal durchgängig übertragen können, werden allgemein als **Übertragungswege** bezeichnet.

Zum Beispiel wird ein Übertragungsweg für einen Kundenservice durch „Link Connections“ und „Subnetwork Connections“ bereitgestellt, also durch Kanäle und „Subnetwork Connections“.

Für die Übertragungswege müssen die Kanäle in den Netzknoten miteinander verbunden werden. Das geschieht durch das Einrichten von „Subnetwork Connections“ in den Netzknoten. Das Einrichten einer „Subnetwork Connection“ und das Belegen der, für den Übertragungsweg verwendeten Kanäle, wird mit **Schalten** bezeichnet.

Mit der oben dargelegten Vereinfachung bezüglich der Endpunkte von „Transport Entities“, die also auch die Kanäle betrifft, ist ein Übertragungsweg eine alternierende Folge von Netzknoten und Kanälen.

Die **Route** für einen Übertragungsweg legt die genaue Folge der Netzknoten und Kanäle fest.

Beispiel:

Ein Übertragungsweg vom Netzknoten v_q zum Netzknoten v_s , der über über den Netzknoten v_1 geführt wird, kann wie in Abbildung 2.6 gezeichnet werden.

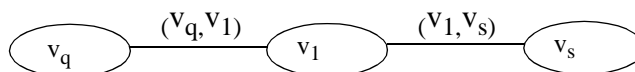


Abb. 2.6 Übertragungsweg von Netznoten v_q nach Netznoten v_s

Die Route dieses Übertragungsweges ist:

$v_q(v_q, v_1), v_1, (v_1, v_s), v_s$ oder auch kurz

„ v_q, v_1, v_s “

Sind zwischen zwei Knoten, zum Beispiel v_1 und v_s , mehrere Kanten vorhanden, so reicht (v_1, v_s) zur eindeutigen Bezeichnung der Kante nicht aus; siehe hierzu auch 3.1.1.

Die Kanten und Knoten des Graphen zeichnen sich durch vielfältige Eigenschaften aus. Ein Teil dieser Eigenschaften leitet sich aus den technischen Eigenschaften der Geräte ab, durch die sie bereitgestellt werden. Andere Eigenschaften sind betriebswirtschaftlich begründet. Im TMN-System werden die Eigenschaften der Kanten und Knoten auf die Parameter der entsprechenden Objekte des TMN-Systems abgebildet. Den Kanten und Knoten ist also jeweils ein Tupel von Parametern zugeordnet. Die Bezeichnungen können von TMN-System zu TMN-System verschieden sein. Auch muß nicht jede Eigenschaft in allen TMN-Systemen abgebildet werden, denn Transportnetze sind nicht alle von gleicher Komplexität.

Um zu klären, ob es Eigenschaften gibt, die für alle Betreiber wichtig sind, wurden eingehende Diskussionen mit TMN-Experten von Siemens Business Services geführt. Die so erarbeiteten Vorstellungen wurden mit Experten der Deutschen Telekom diskutiert, die für verschiedene Bereiche des Transportnetzes zuständig sind. Das Ergebnis zeigte, daß von folgenden Eigenschaften ausgegangen werden kann:

- die Betriebskosten, die für den Netznoten anfallen
- die Art des Netznoten;
ein Netznoten kann zum Beispiel fernschaltbar sein
- die Auslastung des Netznoten
- die Auslastung der Kanten, die angibt, wieviel Prozent der existierenden Kanäle der Kante belegt sind
- die Dienstgüte (quality of service), die ein Kanal einer Kante zur Verfügung stellt. Nach der ITU-T Rec. M.60, [20], ist „quality of service“ frei übersetzt definiert als: „Der kollektive Effekt der Service Performanz, die den Grad der Zufriedenstellung des Benutzers (Kunden) des Service bestimmt“. „Quality of service“ kann als breites Konzept bezeichnet werden, daß viele Aspekte der Performanz eines Service abdeckt und durch zahlreiche Parameter gemessen wird. In [50] zum Beispiel werden Begriffe und Definitionen für Parameter von „quality of service“ aus Sicht der Betreiber und deren Kunden festgelegt. Niemand ist jedoch gezwungen, sich daran zu halten.

Die „quality of service“, die ein Kanal bereitstellt, wird im folgenden kurz mit Qualität bezeichnet. Beispielhaft für die Parameter der Qualität werden in dieser Arbeit Verfügbarkeit und Bitfehlerhäufigkeit betrachtet. Die folgenden Defini-

tionen sind ITU-T Rec. M.60, [20], entnommen und lauten frei übersetzt:

Die **Verfügbarkeit** einer Kante ist ihre Fähigkeit, in einem Zustand zu sein, die geforderte Funktion auszuführen in der vorgegebenen Zeit und unter der Annahme, daß eventuell notwendige externe Ressourcen zur Verfügung stehen. Eine besonders "gute" Kante bezüglich der Verfügbarkeit ist eine Kante, die eine Verfügbarkeit von 99,999% hat [44]. In dieser Arbeit wird mit Verfügbarkeitswerten zwischen 0% und 100% gerechnet.

Die **Bitfehlerhäufigkeit** ist definiert als das Verhältnis der Anzahl der binären Signalelemente, die bei der Übertragung verfälscht wurden, zur Gesamtzahl der ausgesendeten binären Signalelemente.

- die physikalische Länge einer Kante

In dieser Arbeit werden daher diese Eigenschaften zugrunde gelegt und in 3.1.1 als Parameter für die Kanten und Knoten definiert.

2.5 Routenbestimmung in Transportnetzen

2.5.1 Aufgabe

Für ein Transportnetz gibt es im wesentlichen zwei Phasen; die Planungsphase und die Einsatzphase, die sich zyklisch wiederholen. Die Betreiber solcher Transportnetze investieren in die Planungsphase viel Zeit und Geld [13], [52]. In dieser Phase wird zum einen die einsatzbezogene Ressourcenplanung und Kapazitätsoptimierung durchgeführt, zum anderen wird strategisch geplant. Die Aufgabe der strategischen Planung beinhaltet unter anderem die Entscheidung über die zu verwendenden Medien und die Langzeitplanung. Die Planung des Transportnetzes zeichnet sich dadurch aus, daß widersprüchliche Kriterien wirtschaftlicher und technischer Art gegeneinander abgewogen werden müssen, beispielsweise

- verkehrsgerechte Netzdimensionierung entsprechend einer vorangegangenen Bedarfsprognose
- Minimierung der Gesamtkosten für das Netz
- Anforderungen an die Qualität der Übertragungswege

Ein solches Transportnetz kann nach Abschluß der Planung als ein System angesehen werden, das die aufgeführten Kriterien so gut wie möglich beachtet. Das bedeutet im Hinblick auf die hier zu lösende Aufgabe:

- Die schon gebuchten Übertragungswege sind entsprechend geschaltet.
- Für zu erwartende Schaltungen von Übertragungswegen stehen genügend freie Kanäle an den richtigen Stellen im Netz und in verschiedenen Qualitäten bereit.
- Der Überschuß an freien Kanälen ist nicht zu groß, damit die Wirtschaftlichkeit gewährleistet ist.

Nach der Planung kann das Transportnetz in Betrieb gehen und damit beginnt die Einsatzphase.

Der Betrieb hat die Aufgabe, die geplanten Übertragungswege zum gewünschten Termin zu schalten, kurzfristig neue Übertragungswege bereitzustellen und das Funktionieren der Übertragungswege zu gewährleisten. Während des Betriebes wird das Transportnetz also durch

Schaltvorgänge verändert. Schaltvorgänge dienen dazu

- Trails einzurichten, um neue Kanäle einer bestimmten Übertragungsrate zu erzeugen und so die Kapazität der Übertragungsleistung den Marktanforderungen anzupassen, soweit die vorhandenen physikalischen Einrichtungen das zulassen,
- neue Übertragungswege bereitzustellen,
- bestehende Übertragungswege geplant umzuschalten, um zum Beispiel nötige Wartungsarbeiten vorzunehmen,
- bestehende Übertragungswege sofort umzuschalten, wenn eine Störung auftritt. Dieser Vorgang wird Ersatzschaltung genannt.

Um einen solchen Schaltvorgang auszuführen, muß im Layer-Netzwerk, dessen Übertragungsrate der des zu schaltenden Übertragungsweges entspricht, eine Route bestimmt werden, das heißt eine Folge von aneinander grenzenden Kanälen und Netzknoten vom Start-Netzknoten bis zum Ziel-Netzknoten.

Jeder Schaltvorgang bewirkt unter anderem, daß in dem der Übertragungsrate entsprechenden Layer-Netzwerk die Folge von Kanälen belegt wird, die durch die Route vorgegeben ist. Die Belegung der betroffenen Kanäle wird für ein fest definiertes Zeitintervall vorgenommen. In diesem Zeitintervall sind die Kanäle permanent belegt. Der erzeugte Übertragungsweg existiert unabhängig davon, ob er zur Datenübertragung genutzt wird oder nicht. Die von ihm genutzten Kanäle sind und bleiben bis zur Aufhebung des Übertragungsweges belegt¹⁾. Die Größenordnung dieser Zeitintervalle hängt von der Nutzung ab. Leitungen, die für Rundfunk oder Fernsehübertragungen bereitgestellt werden, werden für die Dauer von Stunden geschaltet. Leitungen, die dauerhaft genutzt werden, wie zum Beispiel Leitungen für Telefonkunden, werden ohne Angabe eines Endetermins bereitgestellt.

Die Routensuche für die Schaltvorgänge wurde früher von erfahrenen Mitarbeitern der Betreiberfirma - im weiteren Text Experten genannt - in den Schaltzentren durchgeführt. Langjährige Erfahrungen und ein gewachsenes Regelwerk zur Auswahl dieser Routen standen hier zur Verfügung. So war gewährleistet, daß die Routen für die Schaltungen entsprechend der Anforderungen aus Planung und Betrieb gewählt wurden.

Diese Experten haben die Fähigkeit, die verschiedenen Eigenschaften der Kanten und Knoten bei der Routenbewertung zu berücksichtigen und gegeneinander abzuwägen. Ein Experte wird hier auch Regeln aus der Planungsphase, die die Routenauswahl im Betrieb betreffen, berücksichtigen.

Außerdem kann er, wenn das aus strategischen Gründen nötig ist, bestimmte Kanten nicht für Schaltungen verwenden, um sie später für eine spezielle Verwendung zur Verfügung zu haben.

Die zusätzliche Kapazität, die durch neue Knoten und Kanten zur Verfügung gestellt wird, wird er den Regeln entsprechend verwenden. Kanten, die kaum genutzt werden, da für sie kein Bedarf besteht, wird er möglichst nicht mehr bei der Routenwahl berücksichtigen, um sie freischalten zu können und so die verwendeten Geräte an anderer Stelle nutzen zu können.

Es gibt also solche Kanten, die nicht ausgelastet sind, da sie noch nicht lange in Betrieb sind,

¹⁾Diese Art der Verbindung ist nicht mit der zu verwechseln, die in ATM-Netzen existiert. Hier werden Verbindungen nur kurzfristig für die Zeit der Datenübertragung erzeugt bzw. sie repräsentieren Routen einer potentiellen Verbindung für Pakete.

und solche Kanten, die nicht ausgelastet sind, da für sie kein Bedarf besteht. Für den menschlichen Betrachter ist es kein Problem zwischen beiden Typen nicht ausgelasteter Kanten zu unterscheiden. Er kann also die Routenbewertung sehr flexibel durchführen.

Experten können auch berücksichtigen, daß in einem Transportnetz Routen für verschiedene Arten von Übertragungswegen zu bestimmen sind. Die Übertragungswege unterscheiden sich in der Zeit, die für die Ausführung der Schaltung zur Verfügung steht, und in der Qualität, die für die Übertragungswege erreicht werden soll. Es gibt zum Beispiel:

- langfristig geplante Übertragungswege
- kurzfristig zu schaltende Übertragungswege (zum Beispiel schnelle Ersatzschaltungen)
- preiswert angebotene Übertragungswege, die keine hohe Übertragungsqualität bieten, die in den Parametern Verfügbarkeit und Bitfehlerhäufigkeit gemessen wird.
- hochprioritäre Übertragungswege, deren garantierte minimale Verfügbarkeit (Mindestverfügbarkeit) nicht unterschritten und deren garantierte maximale Bitfehlerhäufigkeit nicht überschritten werden dürfen.

Die Liste ist zwar nicht vollständig, soll aber verdeutlichen, daß jeder der aufgezählten Übertragungswege eine eigene Gruppe bildet. Und jeder dieser Gruppen von Übertragungswegen hat ihre eigenen Regeln, nach denen die Routen zu wählen sind.

In der heutigen Zeit nimmt die Anzahl der täglich durchzuführenden Schaltungen zu und die Reaktionszeit, um eine Schaltung durchzuführen, wird immer kürzer. Um diesen Performanceanforderungen zu genügen, wird auch die Routenauswahl von einem TMN-System durchgeführt. So kommen automatische Routing-Verfahren zum Einsatz.

Die meisten der heute eingesetzten automatischen Routing-Verfahren wählen aus den verfügbaren Parametern einen aus und verwenden diesen als Kantenwert. Durch die Beschriftung der Kanten mit ihrer Länge wird zum Beispiel die physikalisch kürzeste Route ausgewählt. Werden die Kanten des Graphen mit irgend einer konstanten Zahl, zum Beispiel mit 1 beschriftet, so wird auf diese Weise die Anzahl der Knoten in der Route minimiert.

Durch die Wahl eines Kriteriums werden die anderen gleichzeitig nicht beachtet. Die oben genannten Beispiele der Parameterwahl haben zur Folge, daß eine solche Routenwahl nicht die Qualität des Übertragungsweges, nicht die Kosten, die bei der Bereitstellung anfallen, nicht die Auslastung der Netzknoten oder der Kanten berücksichtigt.

Bei diesen Verfahren ist nicht die Möglichkeit geboten, die Regeln der Experten nachzubilden. Der Vorteil, den ein menschlicher Entscheider bei der Routenbewertung hat, geht vollständig verloren. So wird eine Route gewählt, die den Aspekten der Netzplanung und Konfiguration in den meisten Eigenschaften widerspricht. Dadurch wird die wohlüberlegte Netzkonfiguration des Transportnetzes während des Einsatzes zerstört.

Zwei Beispiele sollen zeigen, welche Auswirkungen die Mißachtung eines einzelnen Parameters bei der Routenauswahl haben kann:

1. Wird zum Beispiel die Auslastung nicht berücksichtigt, so kann das dazu führen, daß einerseits bestimmte Kanten und Knoten unterproportional belegt werden und andererseits andere Kanten und Knoten zu 100% beschaltet werden. An den Orten, an denen sich nur noch voll beschaltete Knoten und Kanten befinden, können nur dann neue Übertragungswege für Kunden angeboten werden, wenn neue Kapazität geschaffen

wird. Das kann durch die Anschaffung neuer Netzelemente und ihre Anbindung an das Netz erfolgen oder durch Umschalten bestehender Leitungen. Im ersten Fall kann es so zu Fehlinvestitionen kommen, denn neue Netzelemente werden angeschafft, obwohl es gar nicht notwendig wäre. Im Falle des Umschaltens entsteht Mehraufwand und eine Verkehrsunterbrechung des Übertragungsweges.

Heute werden zur Verbesserung der Netzkonfiguration in bestimmten Zeitabständen sogenannte Grooming-Verfahren¹⁾ eingesetzt. Diese Verfahren bewirken Umschaltungen von bereits bestehenden Übertragungswegen, um das Transportnetz wieder in einen besseren und leistungsfähigeren Zustand zu bringen. Hierzu stehen Regeln für das Belegen von Ressourcen zur Verfügung. Als Folge entstehen Verkehrsunterbrechungen von Übertragungswegen. Diese Verfahren verursachen Kosten und beanspruchen Zeit.

Jede der genannten Maßnahmen wirkt sich negativ auf die Bilanz des Betreibers aus.

2. Berücksichtigt man bei der Auswahl die Qualität der Kanten nicht, so erhält der Kunde zum Beispiel einen Übertragungsweg, dessen Verfügbarkeit unter dem gezahlten Niveau liegt. Die dadurch entstehende Kundenunzufriedenheit kann zu Verlust des Kunden und somit zu Einnahmeausfall und negativer Propaganda führen. Beides ist für den Betreiber von Nachteil.

Es können auch Algorithmen zur Bestimmung Kürzester-Wege mit Nebenbedingungen eingesetzt werden, um gleichzeitig mehrere Parameter zu beachten oder Kürzeste-Wege-Algorithmen, die alle Pareto-optimalen Wege bestimmen. Diese Verfahren eignen sich jedoch nicht für alle Parameter, wie in 3.2 genauer erläutert wird.

2.5.2 Zielsetzung

Ziel dieser Arbeit ist es, die Routenauswahl in den Layer-Netzwerken von Transportnetzen so zu gestalten, daß sie das Verhalten eines Experten in den Schaltzentren nachbildet und automatisch durchführt. Von diesem Experten wird vorausgesetzt, daß er optimal arbeitet, das heißt, er setzt alles ihm zur Verfügung stehende Wissen ein, um die Kanten und Knoten auszuwählen bzw. gegeneinander abzuwägen.

Dieses Wissen besteht aus seiner Erfahrung, den Regeln, die aus der Planungsphase auch für den Betrieb gelten und den Groomingregeln. Diese Arbeit geht davon aus, daß die Erfahrung des Experten sich auch in Form von Regeln erfassen läßt. Das gesamte Wissen kann also in Form von Regeln formuliert werden.

Bei der Bewertung der Routen beachtet der Experte zwei Sichtweisen:

- Sicht auf das gesamte Layer-Netzwerk und das Transportnetz
- Sicht auf die einzelne Kante und den einzelnen Knoten.

Der Zustand des gesamten Transportnetzes ist der Zustand aller seiner einzelnen Layer-Netzwerke. Der Zustandsraum eines Layer-Netzwerkes wird aufgespannt durch alle Parameter jeder seiner Kanten und Knoten. Der Zustand eines Layer-Netzwerkes ist durch die Parameterwerte aller seiner Kanten und Knoten gegeben. Auf diesem Zustandsraum gibt es keine Metrik. Um den Zustand eines Layer-Netzwerkes zu beurteilen, müssen alle Parameterwerte betrachtet werden. Durch die Planung des Transportnetzes ist für jedes seiner Layer-Netz-

¹⁾Grooming-Verfahren werden zur Restrukturierung des Netzes eingesetzt. Das englische Verb *to groom* bedeutet *striegeln, putzen*.

werke ein wirtschaftlich und technisch ausgewogener Zustand hergestellt worden. Während des Einsatzes des Transportnetzes ändern sich die Parameterwerte. Zum Beispiel ändern sich durch die Neuschaltungen die Auslastungswerte der Kanten.

Der Experte beachtet daher bei der Routenauswahl alle Komponenten des Zustandsraumes, das heißt also alle Parameter der Kanten und Knoten. In den Regeln der Experten ist also festgelegt, wie diese Parameter zu berücksichtigen sind. Eine exakte Beschreibung wird im nachfolgenden Kapitel gegeben, abgestimmt auf das dort verwendete Layer-Netzwerk.

Um zu erläutern, wie die Bewertung der Routen durch den Experten aussehen kann, sei hier als Beispiel die Auslastung der einzelnen Kanten betrachtet. Die Auslastung sollte gleichmäßig erhöht werden. Eine gleichmäßige Auslastung der Kanten wird wichtig, wenn das Netz schon sehr voll ist, die zur Verfügung stehenden Ressourcen also sehr knapp werden. So können möglichst viele Schaltungen durchgeführt werden, ohne daß einzelne Knoten nicht mehr erreichbar sind. Insbesondere kann so gewährleistet werden, daß die durch die Planung bereitgestellten Ressourcen ausreichen.

Eine Verbesserung bei der Routenauswahl ist von großer wirtschaftlicher Bedeutung. Da die Betreiber ihre wirtschaftlichen Zahlen hierzu nicht veröffentlicht wissen wollen, kann nur eine beschreibende Erklärung gegeben werden:

In der Planungsphase stehen Algorithmen zur optimierten Ressourcenplanung zur Verfügung. Die geplanten Ressourcen werden in der Betriebsphase zu den gewünschten Terminen geschaltet. Wenn während des Betriebes diese Ressourcen so ungünstig ausgelastet werden, daß die Ressourcen nicht ausreichen, dann muß der Betrieb neue Ressourcen durch Schaltmaßnahmen bereitstellen. Das ist eine Maßnahme, die nicht vorgesehen ist und daher nicht durch entsprechende Software unterstützt wird. Insbesondere stehen die Planungsalgorithmen im Betrieb nicht zur Verfügung. Es handelt sich also um einen sehr unwirtschaftlichen Vorgang, der unbedingt zu vermeiden ist. Die Beträge, die dafür bei Transportnetzen anfallen sind groß.

Um einen Eindruck für die Größenordnung zu geben, sei hier eine typische Konstellation eines Transportnetzes aufgezeigt [43]:

- 350 000 Netzknoten
- 1 500 000 Übertragungswege
- 500 Experten in den Schaltzentren, die gleichzeitig aktiv sind.

Es also wichtig, die durch die Planung zur Verfügung stehenden Ressourcen sehr gut auszulasten. Wenn eine Kunde eine Leitung benötigt, sollte eine vorhanden sein. In [27] wird dargestellt, daß die Kosten, die dem Betreiber für die Bereitstellung einer Leitung entstehen, sehr viel geringer sind als die Einnahmen, die ihm entgehen, wenn keine Ressourcen mehr vorhanden sind, obwohl ein Kunde da ist.

Transportnetze sind schon lange im Einsatz. Es kann also davon ausgegangen werden, daß Listen von Routen existieren, die schon für Schaltungen genutzt wurden und die sich als gut erwiesen haben. Ein weiteres Ziel dieser Arbeit ist es, das Wissen, das implizit in diesen Daten vorhanden ist, für zukünftige Routenbewertung zu nutzen.

Der Telekommunikationsmarkt ist starken Veränderungen unterworfen. Die Anforderungen an die Auswahl der Routen können sich daher ändern. Das Verfahren, das zu diesem Zweck eingesetzt wird, sollte daher schnell und flexibel auf veränderte Anforderungen reagieren können. Hart kodierte Auswahlverfahren, die bei Veränderung der Anforderungen, den Betreiber zum Einsatz einer neuen Softwareversion zwingen, kommen nicht in Betracht.

In den nachfolgenden Kapiteln wird ein Routing-Verfahren vorgestellt, das die vorangegangenen Anforderungen erfüllt.

Das Routing-Verfahren wird für ein einzelnes Layer-Netzwerk entwickelt. Diese Arbeit setzt voraus, daß alle Informationen, die zur Auswahl der Kanten und Knoten wichtig sind, in diesem Layer-Netzwerk zur Verfügung stehen. In der Praxis müssen diese Informationen immer aktuell aus den angrenzenden Layer-Netzwerken ermittelt werden, da sie sich während des Betriebes laufend ändern. Um die exakte Führung eines Kanals zu ermitteln, müssen, wie in 2.4 erläutert, alle Layer-Netzwerke bis hin zur physikalischen Führung betrachtet werden. Wie diese Daten ermittelt werden, ist nicht Bestandteil dieser Arbeit. In den vorhandenen TMN-Systemen sind die Probleme, die sich aus der hierarchischen Struktur der Transportnetze ergeben, durch entsprechende Buchungsverfahren und geschickte Datenhaltung gelöst.

In KAPITEL 2 wurde die Aufgabe aus Sicht der Telekommunikation dargestellt. Das Transportnetz, seine Bestandteile und Eigenschaften wurden dargelegt. In dem nun folgenden Kapitel wird die informatisch-mathematische Aufgabe formuliert. In 3.1 werden die dazu notwendigen Objekte und ihre Eigenschaften definiert. 3.2 untersucht vorhandene Lösungsansätze. In 3.3 werden die Auswahlkriterien der Experten analysiert und mit den nun zur Verfügung stehenden Objekten und ihren Eigenschaften beschrieben. 3.4 beschreibt das gewünschte Ergebnis.

In 3.5 wird der in dieser Arbeit entwickelte Fuzzy-Logik-basierter Lösungsansatz vorgestellt und die daraus resultierenden Verfahren diskutiert.

3.1 Objekte und Voraussetzungen

3.1.1 Layer-Netzwerk als abstrakter Graph

Die Routenauswahl in Layer-Netzwerken von Transportnetzen wird reduziert auf die Bewertung und Auswahl von Routen in einem gerichteten Graphen - ausgehend von Bewertungen der Kanten und Knoten mit jeweils einer Reihe von sehr verschiedenartigen Parametern, die von den technischen und wirtschaftlichen Gegebenheiten abhängen. Dazu wird das Layer-Netzwerk als Graph definiert, der alle Eigenschaften des realen Layer-Netzwerkes hat, die für die Routenauswahl eine Rolle spielen.

Wie schon in 2.4 dargelegt, sind für die Netzknoten und Kanäle des Layer-Netzwerkes die folgenden Parameter wichtig:

- Betriebskosten der Netzknoten,
- Knotenart,
- Auslastung der Netzknoten,
- Qualität der Kanäle, gemessen durch Bitfehlerhäufigkeit und Verfügbarkeit,
- physikalische Länge der Kanäle.

So ergibt sich die nachfolgende Definition für ein Layer-Netzwerk:

Ein **Layer-Netzwerk** der Übertragungsrate $n \in \mathbb{N}$ des Transportnetzes ist ein knoten- und kantenbewerteter Graph $G = (V, E)$, wobei V die endliche, nicht leere Menge der Netzknoten ist und E eine Teilmenge von $V \times V$.

Jedem Element aus V sind die Parameter des Netzknotens zugeordnet. Das sind

- die Betriebskosten des Netzknotens,
- die Knotenart,
Es werden fernschaltbare und nicht fernschaltbare Knoten unterschieden.
- die Auslastung des Netzknotens.
Der Wert der Auslastung gibt an wieviel Prozent der ein- und ausgehenden Kanäle belegt sind.

Eine Kante $(v_1, v_2) \in E$ repräsentiert die Menge der Kanäle von v_1 nach v_2 , die alle dieselben Parameterwerte haben, und deren Übertragungsrate n ist. Diese Parameter und die Auslastung der Kante, die angibt wieviel Prozent der existierenden Kanäle belegt sind, bilden die Parameter der Kante. Das sind also

- die Auslastung der Kante
Der Wert der Auslastung gibt an wieviel Prozent der Kanäle der Kante belegt sind.
- die Verfügbarkeit der Kante
Der Wert der Verfügbarkeit ist der Wert der Verfügbarkeit der in der Kante enthaltenen Kanäle.
- die Bitfehlerhäufigkeit der Kante
Der Wert der Bitfehlerhäufigkeit ist der Wert der Bitfehlerhäufigkeit der in der Kante enthaltenen Kanäle.
- die Länge der Kante.

Zwischen zwei Knoten können mehrere Kanten existieren, wenn Kanäle mit verschiedenen Parameterwerten diese beiden Knoten verbinden. Da in der Praxis die Kanäle durch Multiplexen entstehen bzw. aus physikalischer Sicht in einem Kabel geführt werden, sind häufig die Parameterwerte für mehrere Kanäle dieselben.

Wenn die Übertragungsrichtung der Link Connections berücksichtigt wird, so handelt es sich um einen gerichteten Graphen.

Sind zwischen zwei Knoten v_i und v_j mehrere Kanten vorhanden, so reicht (v_i, v_j) nicht zur eindeutigen Bezeichnung der Kante aus. Die Kanten werden bei mehreren Kanten zwischen zwei Knoten zusätzlich durchnummeriert, zum Beispiel $(v_i, v_j)_1, (v_i, v_j)_2, \dots, (v_i, v_j)_n$ bei n Kanten, $n \in \mathbb{N}$, zwischen den Knoten v_i und v_j . Bei der nachfolgenden Definition der Route ist diese Bezeichnungsweise der besseren Lesbarkeit halber nicht berücksichtigt.

Eine **Route** im Layer-Netzwerk vom Knoten v_q zum Knoten v_s ist eine alternierende Folge von Knoten und Kanten, die jeweils in genau einem Knoten übereinstimmen. Die Knoten sind verschieden:

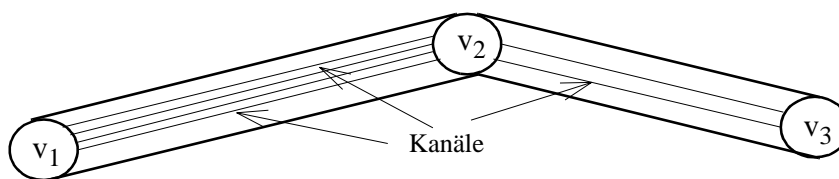
$$v_q, (v_q, v_1), v_1, (v_1, v_2), \dots, (v_{n-1}, v_n), v_n, (v_n, v_s), v_s \text{ mit}$$

$$v_q, v_s, v_1, v_2, \dots, v_n \in V \text{ und } v_i \neq v_j \text{ für } i \neq j$$

Das Schalten eines Übertragungsweges entlang einer bestimmten Route beinhaltet im realen

Layer-Netzwerk das Belegen der betroffenen Kanäle und das Einrichten der Subnetwork Connections für die betroffenen Netzknoten - siehe 2.4. In dem oben definierten abstrakten Layer-Netzwerk wird das Schalten ausschließlich durch Belegung der betroffenen Kanäle angezeigt. Die Auslastung der betroffenen Knoten und Kanten wird dabei entsprechend angepaßt. Das in Abbildung 3.1 dargestellte Beispiel verdeutlicht die Anpassung der Auslastung der Kante.

Das Schalten einer einzigen Route ändert also die Bewertung des Graphen. Das Layer-Netzwerk ändert sich mit jedem geschalteten Übertragungsweg hinsichtlich des Parameters Auslastung seiner Kanten und Knoten. Es handelt sich um eine zeitliche Veränderung des Layer-Netzwerkes.



Ausgangszustand: Kante (v_1, v_2) hat 4 freie Kanäle
Kante (v_2, v_3) hat 2 freie Kanäle

Schalten der 1. Route „ v_1, v_2, v_3 “ bewirkt folgende Veränderung:

ein Kanal aus Kante (v_1, v_2) und ein Kanal aus Kante (v_2, v_3) wird als belegt gekennzeichnet:
Auslastung der Kante (v_1, v_2) ändert sich von 0% nach 25%
Auslastung der Kante (v_2, v_3) ändert sich von 0% nach 50%

Schalten der 2. Route „ v_1, v_2, v_3 “ bewirkt:

Auslastung der Kante (v_1, v_2) ändert sich von 25% nach 50%
Auslastung der Kante (v_2, v_3) ändert sich von 50% nach 100%

Abb. 3.1 Auslastungsveränderung der Kanten bei Verwendung

3.1.2 Eigenschaften der Parameter

Um die Routenauswahl im Sinne der Experten zu gestalten, stehen als Meßgrößen die Parameter der Knoten und Kanten zur Verfügung. Um diese Parameter zur Bewertung von Routen einsetzen zu können, ist es mathematisch interessant für jeden Parameter folgendes zu untersuchen:

- Definiert der Parameter eine Ordnung auf den Kanten oder Knoten und läßt sich diese Ordnung auf die Routen übertragen ?
- Kann aus dem Kanten- oder Knotenparameter an Hand einer einfachen Funktion ein Wert für die Route bestimmt werden und eignet sich diese Funktion als Distanzmaß, so daß die beste Route damit bestimmt werden kann ?
Ist ein Parameter additiv entlang jeder Route, so ist das Distanzmaß durch die Summe der Parameterwerte der Kanten oder Knoten gegeben. Die Bewertung der lokalen Objekte, der Kanten und Knoten, kann so auf die globalen Objekte, die Routen, übertragen werden.

Das wird für die einzelnen Parameter im folgenden untersucht:

Betriebskosten der Knoten

Die Parameterwerte sind reelle Zahlen. Die Knoten können nach den Werten ihrer jeweiligen Betriebskosten geordnet werden. Die Betriebskosten jeder Route können als Summe der Betriebskosten der darin enthaltenen Knoten definiert werden. Die Betriebskosten sind also additiv entlang jeder Route.

Knotenart

Der Parameterwert der Knotenart charakterisiert den Knoten. Als Parameterwerte stehen zwei Werte, fernschaltbar oder nicht fernschaltbar, zur Verfügung. Eine Ordnung im Sinne von schlechter oder besser kann definiert werden. Fernschaltbare Knoten sind besser als nicht fernschaltbare.

Zwei Möglichkeiten mit Hilfe des Knotenparameters die Bewertung der Route vorzunehmen sind offensichtlich:

1. Die Route wird als fernschaltbar bewertet, wenn alle ihre Knoten fernschaltbar sind. Die Route wird als nicht fernschaltbar bewertet, sobald ein einziger ihrer Knoten nicht fernschaltbar ist. Mit dieser Definition ist es nicht möglich Routen zu unterscheiden, die unterschiedlich viele nicht fernschaltbare Knoten enthalten.
2. Die „kürzeste Route“ im Sinne der besten Route hinsichtlich der Knotenart kann definiert werden als die Route mit der kleinsten Anzahl der nicht fernschaltbaren Knoten.

Länge der Kanten

Die Parameterwerte sind reelle Zahlen. Die Kanten können nach ihrer Länge geordnet werden. Die Länge ist additiv entlang jeder Route. Die Länge jeder Route ist die Summe der Länge ihrer Kanten.

Qualität der Kanten

Die Werte der Parameter, die die Qualität der Kante messen, sind reelle Zahlen. Eine Ordnung kann angegeben werden, das heißt es kann festgestellt werden, welche von zwei Kanten die qualitativ bessere bezüglich jeweils eines Parameters ist. Bezogen auf die Verfügbarkeit bedeutet ein höherer Wert, daß die Kante besser ist. Bezogen auf die Bitfehlerhäufigkeit bedeutet ein höherer Wert, daß die Kante schlechter ist.

Um in der Praxis zu einem Wert der Verfügbarkeit der Route zu gelangen, wird oft mit dem Produkt der Verfügbarkeitswerte der einzelnen Kanten gerechnet. Dabei werden die Unabhängigkeit der Verfügbarkeiten der Kanten und sehr hohe Verfügbarkeiten vorausgesetzt. Bei der Routenauswahl sollen nicht ausschließlich hohe Verfügbarkeitswerte vorausgesetzt werden. Die Werte der Parameter sollen keinen Einschränkungen unterliegen.

Betrachtet sei eine Route mit zwei aneinander grenzenden Kanten k_1 und k_2 . Der Wert der Verfügbarkeit von k_1 sei 80% , der von k_2 60%.

Wenn einer Route das Maximum der Werte der Verfügbarkeit ihrer Kanten zugeordnet wird, so ist die Verfügbarkeit der Route zu hoch angesetzt. Für die Beispielroute im vorangegangenen Absatz ist das Maximum 80%. Die Kante k_2 liefert diese Verfügbarkeit nicht.

Wird der Route das Minimum der Verfügbarkeitswerte ihrer Kanten zugeordnet, so werden

qualitativ sehr verschiedene Routen mit dem gleichen Wert versehen. Für die Beispielroute ergibt sich so für die Verfügbarkeit 60%. Aber der Beispielroute, die aus fünf Kanten mit jeweils Verfügbarkeit 60% besteht, wird auch der Wert 60% zugeordnet.

Wenn also solche Routen unterschieden werden sollen, dann kann aus der Verfügbarkeit der Kanten einer Route nicht die Verfügbarkeit der gesamten Route berechnet werden.

In der Praxis wird oft die Bitfehlerhäufigkeit der Route aus den Werten der Bitfehlerhäufigkeit ihrer Kanten durch die Addition dieser Werte berechnet. So entsteht ein Bitfehlerhäufigkeitswert der Route, der den „wirklichen Wert“ der Bitfehlerhäufigkeit der Route nach oben begrenzt. Eine Addition der Werte der Bitfehlerhäufigkeit bestimmt nur dann den exakten Wert, wenn pro Kante verschiedene und unabhängige Bits verfälscht werden.

Die eingehenden Diskussionen, die mit den TMN-Experten geführt wurden, um die in dieser Arbeit zu Grunde gelegten Parameter festzulegen - siehe 2.4, hat ergeben, daß die Werte der Verfügbarkeit und der Bitfehlerhäufigkeit der Kanten einzeln zu berücksichtigen sind.

Verfügbarkeit und Bitfehlerhäufigkeit sind als stellvertretende Parameter für die Qualität zu sehen. Welche Parameter zusätzlich oder in Zukunft für die Qualität verwendet werden, ist noch offen.

Aus allen diesen Gründen wird in dieser Arbeit davon ausgegangen, daß sich aus den Werten der Verfügbarkeit und der Bitfehlerhäufigkeit der einzelnen Kanten einer Route keine Werte für die entsprechenden Parameter der Route berechnen lassen.

Auslastung der Kanten

Die Parameterwerte sind reelle Zahlen aus dem Intervall [0,100]. Die Kanten können also nach ihrer Auslastung geordnet werden.

Um aus der Auslastung der Kanten eine Auslastung der Route zu berechnen, bieten sich Funktionen wie Minimum oder Mittelwert aller ihrer Kantenauslastungen an. Da diese Werte meist kleiner sind als die Auslastungswerte der maximal ausgelasteten Kante der Route, liefern die Funktionen ein ungenaues Bild der Auslastung. Die einzige sinnvolle Möglichkeit, einer Route einen Wert für die Auslastung an Hand der Auslastungswerte ihrer Kanten zuzuordnen, ist ihr das Maximum aller Auslastungswerte ihrer Kanten zuzuordnen. Damit werden aber große Unterschiede zwischen den Routen hinsichtlich der Auslastung der enthaltenen Kanten nicht beachtet. Das wird durch das in Abbildung 3.2 dargestellte Beispiel verdeutlicht.

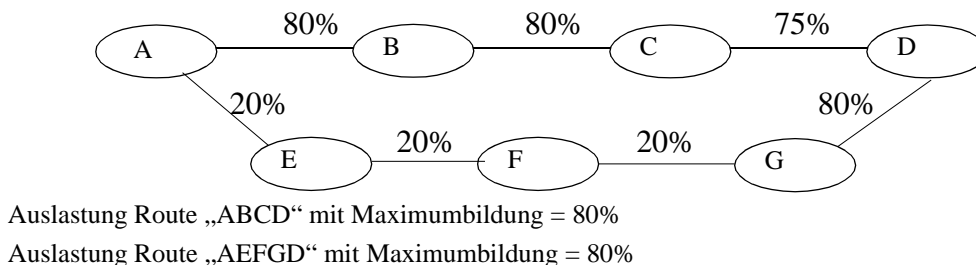


Abb. 3.2 Beispiel für die ungenaue Beurteilung von Routen durch die Verwendung des maximalen Auslastungswertes

Wenn solche Routen unterschieden werden sollen, kann aus der Auslastung der Kanten nicht

die Auslastung der Route berechnet werden.

Auslastung der Knoten

Für diesen Parameter gilt genau dasselbe wie für den Parameter Auslastung der Kanten.

3.2 Vorhandene Lösungskonzepte

An Hand ihrer Parameterwerte können einzelne Kanten und Knoten beurteilt und untereinander geordnet werden. Ein wesentliches Problem liegt darin, daß die Berücksichtigung der lokalen Parameter nicht durch eine einfache Funktion beschrieben werden kann und sich deshalb nicht einfach an Hand eines aus den Parametern berechenbares Distanzmaß die besten Routen als „kürzeste Wege“ im Graphen bestimmen lassen.

Es liegt nahe, das automatische Routing-Verfahren mit Hilfe von Kürzeste-Wege-Algorithmen lösen zu wollen. Der Vollständigkeit halber wird diese Möglichkeit im folgenden Text genauer betrachtet, obwohl sie nicht zum Ziel führt.

Kürzeste-Wege-Algorithmen können in solche unterteilt werden, die auf Graphen arbeiten, deren Kanten mit einem Parameter bewertet sind, und in solche, die auf Graphen arbeiten, deren Kanten mit mehreren Parameter bewertet sind. Im folgenden werden einige der zur Verfügung stehenden Optimierungsverfahren hinsichtlich ihrer Einsatzmöglichkeit für ein automatisches Routing-Verfahren im Layer-Netzwerk betrachtet.

3.2.1 Kürzeste-Wege-Probleme mit einem Parameter

Bei einem Kürzeste-Wege-Problem mit einem Parameter wird auf einem Graphen $G=(V,E)$, dessen Kanten mit je einer positiven reellen Zahl bewertet sind, eine Route von einem Knoten v_q zu einem Knoten v_s gesucht, so daß die Summe der Werte der Kanten entlang der Route minimal ist. Für diese Fragestellung existieren schon lange effiziente, klassische Algorithmen; einer der bekanntesten ist der Algorithmus von Dijkstra [10] aus dem Jahre 1959. Es führt nicht zum Ziel alle diese Algorithmen aufzuführen. In [1], [15], [28] und [31] finden sich einige Möglichkeiten.

Ein neueres Verfahren basiert auf der Darstellung des Kürzeste-Wege-Problems als Semi-ring-based Constraint Logic Programing (SCLP) Problem. Voraussetzung ist eine Beschriftung der Kanten mit Elementen des vollständigen Semiringes $S = \langle R, \text{MIN}, +, +\infty, 0 \rangle$, wobei R die nicht negativen reellen Zahlen inklusive $+\infty$ sind. Insbesondere müssen die Parameter der Kanten additiv entlang der Pfade sein. Bistarelli zeigt auf, wie das SCLP formuliert werden muß, um sowohl die Länge der kürzesten Route als auch die Route selbst zu erhalten [3].

Sind die Kanten des Graphen mit Elementen eines geordneten Semiringes bewertet, so kann das Kürzeste-Wege-Problem genauso gut mit jedem klassischen Kürzesten-Wege-Algorithmus gelöst werden. Der Vorteil von Bistarellis Methode ist, daß sie auch verwendet werden kann, wenn die Kantenwerte nur in einem partiell-geordneten Semiring enthalten sind - siehe 3.2.2.

Bei Transportnetzen handelt es sich um Graphen mit hoher Knotenanzahl. Schon 1995 waren in einem von der Deutschen Telekom verwendeten Telekommunikations-Management-System 20000 Knoten erfaßt [2]. Wenn auf einer Leitung eine Störung auftritt, so muß diese

heute in einer Zeit $t \ll 1$ Minute behoben sein. In dieser Zeitspanne muß sowohl eine neue Route für eine Ersatzschaltung bestimmt sein, als auch die Schaltung durchgeführt sein. Klassische Routingalgorithmen arbeiten zentral und seriell. Neuronale Lösungen für Kürzeste-Wege-Probleme umgehen diesen Nachteil, indem sie parallel und dezentral arbeiten. Die Routensuche kann zum Beispiel von einem Hopfield Netz [19] mit entsprechender Energiefunktion ausgeführt werden. F. Kamoun und M.K. Mehmet Ali definieren eine Energiefunktion und konstruieren das neuronale Netz so, daß es sogar möglich ist neue Kanten in den Graphen aufzunehmen, ohne neue Neuronen erzeugen zu müssen oder die Verbindungen des neuronalen Netzes zu ändern [26]. Die Veränderungen werden ausschließlich durch Bias-Neuronen in das neuronale Netz integriert. Diese Routenbestimmung mit Hopfield Netzen wurde für paketvermittelnde Kommunikationsnetze erarbeitet, für die ebenfalls hohe Performanzanforderungen bestehen. Für diesen Netztyp sind in der Literatur noch andere neuronale Ansätze beschrieben [6], [8],[18].

Sowohl die klassischen Algorithmen wie auch die neuronalen Verfahren setzen voraus, daß die Bewertung der Kanten aus einem positiven, additiven Parameter besteht. Die Kanten und Knoten des Graphen, der das Layer-Netzwerk repräsentiert, sind mit mehreren Parametern bewertet, und nicht alle sind additiv entlang jeder Route. Daher kann kein Verfahren zur Lösung des Kürzeste-Wege-Problems alleine als Lösung für das automatische Routing-Verfahren in Frage kommen.

Wird ein additiver Parameter der Kanten des Layer-Netzwerkes ausgewählt, um alleine als additive Bewertung der Kante zu fungieren, so kann zwar einer der oben beschriebenen Algorithmen zu Routenauswahl angewandt werden, aber diese Routenwahl erfüllt nicht die Aufgabenstellung, wie schon in 2.5.1 dargelegt wurde.

3.2.2 Kürzeste-Wege-Probleme mit mehreren Parametern

Bei einem Kürzeste-Wege-Problem mit mehreren Parametern gibt es die Möglichkeit eine Pareto-optimale Lösung zu suchen. Vereinfacht charakterisiert heißt eine Lösung Pareto-optimal, wenn keine andere Lösung gefunden werden kann, bei der ein Parameter einen besseren Wert hat, ohne daß ein anderer Parameter dieser Lösung einen schlechteren Wert hat.

Um eine genaue Definition zu geben, sei eine Ordnung der Kanten bezüglich der einzelnen Parameter vorausgesetzt. Mit dieser Ordnung kann eine partielle Ordnung für die Kanten bezüglich aller Parameter definiert werden:

Seien $k_1, k_2 \in E$ mit Parametern k_1^i, k_2^i für alle $i = 1, \dots, m$

$$k_1 \leq k_2 \Leftrightarrow k_1^i \leq k_2^i \text{ für alle } i = 1, \dots, m$$

Eine **Pareto-optimale Kante** k^* ist eine Kante aus dem Pareto-optimalen Bereich P mit

$$k^* \in P \Leftrightarrow \{k \in \text{Kanten von } v_q \text{ nach } v_s \mid k \leq k^*\} = \emptyset,$$

wobei $v_q, v_s \in V$ zwie Knoten sind.

Sind alle Parameter der Kanten additiv entlang jeder Route, so kann eine **Pareto-optimale Route** r^* wie folgt definiert werden:

Sie ist ein Element aus dem Pareto-optimalen Bereich P_R aus der Menge der Routen von v_q

nach v_s mit

$$r^* \in P_R \Leftrightarrow \{r \in \text{Route von } v_q \text{ nach } v_s \mid r \leq r^*\} = \emptyset$$

wobei $r \leq r^*$ wie folgt definiert ist:

$$\text{sei } r = (k_1, \dots, k_l) \text{ und } r^* = (k^*_1, \dots, k^*_{l^*}),$$

jede Kante k hat Parameter $k^j \in \mathbb{R}$ mit $j = 1, \dots, m$

$$r \leq r^* : \Leftrightarrow \sum_{i=1}^l k_i^j \leq \sum_{i=1}^{l^*} k^{*j}_i \quad \text{für alle } j = 1, \dots, m.$$

Mit der Additivität der Kantenparameter entsteht so aus der Ordnung für die einzelnen Kantenparameter eine partielle Ordnung für die Routen.

Für Kürzeste-Wege-Probleme mit multiplen, additiven Parametern schlägt Bistarelli vor, das oben erwähnte SCLP für Graphen, deren Kanten mit einem einzigen additiven Parameter bewertet sind, zu erweitern; statt mit S wird mit dem kartesischen Produkt von Instanzen des vollständigen Semirings S gearbeitet. Die Anzahl der Faktoren des kartesischen Produktes ist gleich der Anzahl der Kriterien. In [3] wird das Verfahren für zwei Kriterien vorgestellt: Zeit und Kosten. Jedoch liefert das Verfahren so nicht die Pareto-optimalen Lösungen, sondern die kleinste obere Grenze des jeweiligen Parameters; zum Beispiel statt der „Routenlängen“ $(7,8)$ und $(9,7)$ wird $(7,7)$ ausgegeben. Ist S ein vollständiger Semiring, so ist auch die Hoare Power Domain von S , $P^H(S)$ ein vollständiger Semiring. Wenn $P^H(S)$ statt des kartesischen Produktes von S eingesetzt wird, so kann ein SCLP formuliert werden, das als Lösung die Pareto-optimalen Lösungen bestimmt. Das Ergebnis ist eine Menge nicht vergleichbarer Lösungen. Weitere Details möge der interessierte Leser in [3] nachlesen.

Sind die Kantenparameter nicht-additiv, so kann zwar ein Pareto-optimaler Bereich P_R für die Routen wie folgt definiert werden:

$$r^* \in P_R \Leftrightarrow \{r \in \text{Route von } v_q \text{ nach } v_s \mid r \leq r^*\} = \emptyset$$

wobei $r \leq r^*$ bedeutet: alle Kanten von r sind besser oder gleich gut im Sinne der partiellen Ordnung der Kanten, als die Kanten von r^* . Aber Bistarellis Methode kann in diesem Fall nicht angewendet werden.

Eine andere Möglichkeit ein Kürzeste-Wege-Problem mit multiplen Parametern zu lösen, ist eine Formulierung des Problems als Kürzeste-Wege-Problem mit Nebenbedingungen.

Um nicht alle Parameter gleichzeitig zu optimieren, wird einer ausgesucht und zur optimierenden Kostenfunktion erklärt. Die übrigen Parameter werden als Ressourcenfunktionen betrachtet und durch eine obere Schranke begrenzt:

gegeben sei ein Graph $G(V,E)$ und für jede Kante $k \in E$ ein m -Tupel von Parameter

$$e_1^{(k)}, e_2^{(k)}, \dots, e_m^{(k)}.$$

$$e_1^{(k)} \text{ wird zur Kostenfunktion } c^{(k)} \text{ und } e_2^{(k)}, \dots, e_m^{(k)} \text{ werden zu Ressourcen } r_1^{(k)}, \dots, r_R^{(k)}.$$

Es wird ein Pfad von $v_q \in V$ nach $v_s \in V$ gesucht mit

$$\min \sum_k c^{(k)}$$

$$\sum_k r_i^{(k)} \leq \lambda_i \quad i=1, \dots, R, \lambda_i \in \mathbb{R}.$$

Kosten und Ressourcen werden als nicht negativ und additiv entlang jeder Route vorausgesetzt.

Diese Kürzeste-Wege-Probleme mit Nebenbedingungen gehören zur Klasse der NP-vollständigen Probleme wie Ziegelmann in [54] zeigt. Sie können als diskretes lineares Optimierungsproblem formuliert werden, wenn die Parameter additiv sind. Eine ausführliche Diskussion existierender Lösungen wird in [54] durchgeführt, bevor eine neue effektive Methode dargestellt wird. Diese Methode arbeitet in zwei Schritten:

1. Durch die Lösung einer Relaxierung des Problems wird eine obere und untere Schranke für das Optimum gefunden.
2. Mittels geometrischer Interpretation der Lösung wird ein effektives Verfahren zur Schließung der Lücke begründet und damit das Optimum gefunden.

Je mehr Nebenbedingungen das Kürzeste-Wege-Probleme hat, um so komplizierter und rechenzeitaufwändiger wird das Verfahren, vor allem der 2. Schritt. Wie zu erwarten, arbeitet das Verfahren am besten mit genau einer Nebenbedingung.

Bei allen hier genannten Verfahren wird vorausgesetzt, daß die Parameter additiv sein müssen. Da diese Kanteneigenschaft im Layer-Netzwerk nicht von allen Parametern erfüllt wird, wie in 3.1.2 ausgeführt wird, können die oben genannten Verfahren nicht für die Routenbewertung in Transportnetzen mit den in dieser Arbeit beschriebenen Anforderungen eingesetzt werden.

3.3 Analyse des Expertenverhaltens

Ziel dieser Arbeit ist es, den Experten, der wie in 2.5.2 beschrieben arbeitet, so gut wie möglich nachzubilden. Diesem Experten gelingt es, trotz aller aufgezeigten Probleme, die Routen zu bewerten und für die Übertragungswege Routen auszuwählen. Er kommt also von lokalen Aussagen über Kanten und Knoten zu globalen Aussagen über Routen. Dieses Verhalten der Experten wird im folgenden genauer analysiert.

Für die Routenauswahl steht dem Experten Erfahrungswissen zur Verfügung. Er weiß außerdem, welche Regeln aus der Planungsphase auch in der Betriebsphase zu beachten sind, und er kennt die Belegungsregeln aus den Groomingverfahren.

Bei der Beurteilung der Routen betrachtet der Experte zum einen die einzelnen Kanten und Knoten, und zum anderen das gesamte Layer-Netzwerk.

Aus technischer Sicht soll die Kante möglichst gut und störungsfrei arbeiten, also gilt:

- Je kleiner die Länge einer Kante ist, um so eher ist sie zu bevorzugen.
- Je besser die Qualität der Kante, um so eher ist sie zu bevorzugen.

Es ist wichtig, möglichst viele neue Routen schalten zu können, ohne neue Knoten und Kanten erzeugen zu müssen, denn dadurch entstehen Kosten. Alle Knoten sollten daher möglichst lange über freie Kanäle erreichbar sein.

Knoten und Kanten, die fast nicht verwendet werden, sollten, wenn möglich abgebaut wer-

den.

Daher ist aus betriebswirtschaftlicher Sicht folgendes einzuhalten:

- Kanten und Knoten sollen möglichst gleichmäßig ausgelastet werden.
- Kanten und Knoten, die kaum ausgelastet sind, sind eher abzulehnen.
- Sehr stark ausgelastete Kanten und Knoten sind eher abzulehnen.

Ein weiteres Auswahlkriterium aus betriebswirtschaftlicher Sicht ist:

- Knoten mit niedrigen Betriebskosten sind zu bevorzugen.

Wie schon in 2.5.1 erwähnt, hängt die Beurteilung der Knoten und Kanten auch davon ab, für welche Gruppe von Übertragungswegen eine Route zu bestimmen ist. Für jede Gruppe von Übertragungswegen sind spezielle Auswahlkriterien zu beachten.

Langfristig geplante Übertragungswege können zum Beispiel über nicht-fernschaltbare Knoten geführt werden, da die Zeit zur manuellen Schaltung vorhanden ist. Wieviele und welche Gruppen von Übertragungswegen zu definieren sind, hängt vom jeweiligen Betreiber, seinem Transportnetz und dem Kundenkreis ab. Beispielhaft wird in dieser Arbeit die Gruppe der hochprioritären Übertragungswege betrachtet.

Hochprioritäre Übertragungswege sind Übertragungswege für die eine bestimmte Qualität garantiert wird. Genauer heißt das:

Die Verfügbarkeit darf eine bestimmte Mindestverfügbarkeit nicht unterschreiten, und die Bitfehlerhäufigkeit darf eine maximale Bitfehlerhäufigkeit nicht überschreiten. Zusätzlich wird für solche Übertragungswege im Fehlerfall eine Ersatzschaltung in einer bestimmten Zeitspanne gefordert.

Da Kanten mit hoher Qualität teurer sind als solche mit niedriger Qualität, gilt aus betriebswirtschaftlicher Sicht auch:

- Kanten, deren Bitfehlerhäufigkeit fast gleich der garantierten maximalen Bitfehlerhäufigkeit ist, sind zu bevorzugen.
- Kanten, deren Verfügbarkeit nahe der Mindestverfügbarkeit ist, sind zu bevorzugen.

Für diese Gruppe von Übertragungswegen muß im Fehlerfall schnell Ersatz geschaltet werden. Daher gilt auch:

- Knoten, die fernschaltbar sind, sollen bevorzugt werden.

Zu den allgemein gültigen Regeln gibt es sogenannte strategische Gesichtspunkte, die nur zu bestimmten Zeiten und auch nur für spezielle Kanten gelten. Diese strategischen Gesichtspunkte machen es erforderlich, die Routen im Widerspruch zu den im Regelfall angewandten Vorgehen auszuwählen. Es gibt zum Beispiel Kanten, die für wichtige zu erwartenden Schaltungen frei gehalten werden müssen. Ab einem festen Zeitpunkt sollen sie wieder in den normalen Betrieb integriert werden.

Neu entstandene Kanten, die zu Beginn ihres Einsatzes eine sehr kleine Auslastung haben, sollen, trotz der Regel über die mittlere Auslastung, für Schaltungen verwendet werden. Eine ausführliche Darstellung dieser Anforderung an die Feinsteuerung der Routenbestimmung wird in 6.2.2 gegeben.

Mit Hilfe der Kanten- und Knotenparameter wird also die Bewertung der Kanten verbal beschrieben und ihr qualitativer Zusammenhang erfaßt. Obwohl die besten Routen nicht an Hand eines aus den Parametern berechneten Distanzmaßes bestimmt werden können, gelingt

es dem Experten Kanten und Knoten zu beurteilen und darüber hinaus, diese Beurteilung auf die Routen auszudehnen.

Ein menschlicher Bewerter sieht den Graphen als Bild vor sich und kann die verschiedenen Möglichkeiten zur Wahl einer Route von Knoten v_q nach Knoten v_s gegeneinander abwägen.

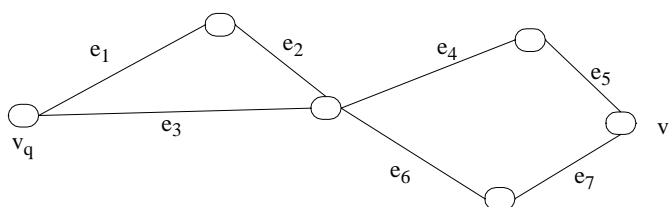


Abb. 3.3 Beispiel zur Demonstration der Routenauswahl der Experten

	e_1	e_2	e_3	e_4	e_5	e_6	e_7
Verfügbarkeit	60%	60%	60%	80%	20%	60%	65%
Bitfehlerhäufigkeit	0,3	0,3	0,3	0,3	0,3	0,3	0,3
Auslastung	50%	90%	65%	60%	60%	60%	60%

Tab. 3.1 Wichtige Parameter der Kanten für Beispiel aus Abbildung 3.3

Um diesen Vorgang prinzipiell zu verdeutlichen, wird das Beispiel, das durch Abbildung 3.3 und Tabelle 3.1 beschrieben ist, betrachtet. Auf die Knoten wird bei der Diskussion des Beispiels wegen der besseren Übersichtlichkeit nicht eingegangen.

Die Kante e_1 sei zu 50% und e_2 sei zu 90% ausgelastet, die Kante e_3 zu 65%. Die Qualitätsparameter aller dieser drei Kanten sind gleich. Der Experte betrachtet alle drei Kanten und entscheidet sich dann für die Kante e_3 . Würde er nur e_1 und e_3 vergleichen, so würde er sich für e_1 entscheiden.

Für die nächste Teilroute würde er zunächst e_4 bevorzugen, da er hohe Qualität bieten will. Da er aber in seine Entscheidung auch die nächsten Kanten einbezieht, so entscheidet er sich für e_6 und e_7 .

Er wägt also an allen Entscheidungsstellen die einzelnen Parameterwerte der jeweils zur Auswahl stehenden Kanten und Knoten gegeneinander ab und blickt auch voraus. Um die Routenauswahl perfekt auszuführen, müßte er alle Möglichkeiten durchdenken und gegeneinander abwägen. Je mehr Möglichkeiten für die Route er betrachtet, um so besser kann seine Auswahl sein. Das dargestellte Beispiel gibt einen Eindruck, wie schwierig es ist alle Parameterwerte zu betrachten. Es ist anzunehmen, daß der Experte sich nicht alle einzelnen Parameter der betrachteten Kanten und Knoten merkt, sondern eine Ordnung der schon betrachteten Objekte.

Es kann zu bestimmende Routen geben, bei denen sich der Experte zwischen Möglichkeiten entscheiden muß, die seinen Regeln widersprechen. Wenn zum Beispiel alle der letzten vier Kanten des Beispiels, e_4 bis e_7 , zu 90% ausgelastet sind, muß er trotzdem eine Wahl treffen, da v_s nicht anders zu erreichen ist.

Der Experte wählt die Routen so aus, daß der Zustand des Layer-Netzwerkes, der nach der

Planung als wirtschaftlich und technisch ausgewogen angesehen werden kann, soweit wie möglich erhalten bleibt. Auf dem Zustandsraum existiert keine Metrik.

Der Zustand des Layer-Netzwerkes wird durch alle Parameterwerte aller seiner Kanten und aller seiner Knoten bestimmt. Die Wahl einer einzelnen Route ändert die Bewertung des Graphen und damit dessen Zustand. Diese Veränderung muß so geschehen, wie es die oben genannten Auswahlkriterien vorschreiben.

Der Experte beachtet auch, wie sich der Zustand während einer Betriebsphase, bei der eine größere Menge von Übertragungswegen geschaltet wird, verändert. So kann er feststellen, ob die Routenauswahl während der Betriebsphase richtig war. Dazu wird das Layer-Netzwerk sowohl aus technischer als auch aus wirtschaftlicher Sicht beurteilt.

Aus wirtschaftlicher Sicht sollte ein Layer-Netzwerk möglichst wenig Kosten verursachen und so viele Einnahmen wie möglich erbringen.

Das wird durch die bevorzugte Wahl von Knoten mit niedrigen Betriebskosten und Kanten mit garantierter, aber nicht zu hoher Qualität gewährleistet.

Aus technischer Sicht müssen die geschalteten Übertragungswege alle durch die Auswahlkriterien geforderten Ansprüche erfüllen. Die Auslastung der einzelnen Kanten und Knoten muß im gewünschten Bereich liegen. Dies ist sowohl von wirtschaftlicher Seite wichtig - es darf nicht zu viele brach liegende Ressourcen geben - wie auch von technischer Seite - für Schaltungen müssen noch genügend freie Ressourcen zur Verfügung stehen.

Die Auslastung der Knoten und Kanten nach dem Einsatz wird mit der Auslastung zu Beginn des Einsatzes verglichen:

Solange noch genügend Ausweichrouten entsprechender Qualität im Layer-Netzwerk vorhanden sind, darf die Auslastung der einzelnen Knoten und Kanten die vorgegebenen Grenzwerte nicht überschreiten. Kanten und Knoten sollen möglichst „gleichmäßig“ ausgelastet werden. Konkrete Werte können nicht gefordert werden, da die Auslastung sowohl von der Auslastung vor dem Einsatz abhängt als auch davon, wieviele Übertragungswege geschaltet werden und welche Anfangs- und Endpunkte diese haben.

3.4 Validierung

Bevor in den nachfolgenden Kapiteln das neue automatische Routing-Verfahren entwickelt wird, wird in diesem Kapitel das gewünschte Ergebnis mit den im vorangegangenen Text bereitgestellten Objekten und ihren Eigenschaften festgelegt.

Während des Einsatzes des noch zu konzipierenden automatischen Routing-Verfahrens ändert sich der Zustand des Layer-Netzwerkes. Der Zustandsraum des Layer-Netzwerkes, der durch alle Parameter jedes seiner Kanten und Knoten aufgespannt wird, hat keine Metrik - siehe 2.5.2. Um festzulegen wie ein Routing-Verfahren den Zustand des Layer-Netzwerkes ändern soll, muß festgelegt werden, wie sich die Parameterwerte seiner Kanten und Knoten verändern sollen. Daher wird im nachfolgenden Text nur noch von Kanten und Knoten und deren Parameter die Rede sein.

Die Analyse des Experten-Verhalten zeigt, wie das noch zu konzipierende automatische Routing-Verfahren zu validieren ist.

Folgenden Zielsetzungen müssen erfüllt werden:

- Auswahl der einzelner Routen:
Die einzelne Routen, die das automatische Routing-Verfahren bestimmt hat, müssen die von den Experten aufgestellten Regeln im Rahmen der in dem Layer-Netzwerk vorhandenen Möglichkeiten erfüllen.
- Möglichkeit zur Feinsteuerung der Auswahl:
Die Feinsteuerung zur Auswahl einzelner Routen, um die vorne beschriebenen strategische Gesichtspunkte einfließen zu lassen, muß gegeben sein.
- Auswirkungen auf das Layer-Netzwerk bei einem längeren Einsatz:
Bei einem Vergleich der Auslastung der Kanten und Knoten vor und nach dem Einsatz sollten die Werte für die Auslastung nur begründet außerhalb der vorgegebenen Bereiche liegen.
Es sollten hauptsächlich Knoten mit niedrigen Betriebskosten verwendet worden sein.
Die Qualität der verwendeten Kanten sollte im geforderten Bereich liegen.

Diese Zielsetzungen können nicht konkreter formuliert werden, da folgendes gilt:

Die Regeln zur Routenauswahl können nur eingehalten werden, wenn im Layer-Netzwerk auch genügend Routen vorhanden sind, die diesen Anforderungen genügen. Das Layer-Netzwerk ändert sich während des Einsatzes. Je mehr Schaltungen durchgeführt worden sind, um so weniger Ausweichrouten stehen zur Verfügung.

Daher ist es sinnvoll, die zeitliche Veränderung der Kanten- und Knotenauswahl während der Betriebsphase aufzuzeichnen. Das Ergebnis dieser Beobachtung muß zeigen, daß die Regeln am Anfang eingehalten werden und zwar solange, wie es in diesem Layer-Netzwerk möglich ist.

Eine weitere Zielsetzung aus der Aufgabenstellung ist:

Das automatische Routing-Verfahren muß sich schnell und flexibel an veränderte Anforderungen anpassen lassen.

3.5 Fuzzy-Logik-basierte Lösungskonzepte

Das oben beschriebene Expertenverhalten weist zwei Merkmale auf, die nachgebildet werden müssen.

Das erste Merkmal ist das Regelwerk, das der Experte zur Beurteilung der Kanten und Knoten verwendet. Mit Hilfe der Kanten- und Knotenparameter wird die Beurteilung der Kanten und Knoten verbal beschrieben.

Das zweite Merkmal ist der Schritt von der Kanten- und Knotenbewertung zur Routenbewertung. Ein menschlicher Bewerter ist durchaus in der Lage, die Werte der Kanten und Knoten einzeln zu betrachten und trotzdem zu einer Auswahl zu kommen. Das ist offensichtlich zeitaufwendig. Um diesen komplizierten Auswahlvorgang automatisch zu gestalten, müssen die Bewertungen der Kanten und Knoten in irgendeiner Weise zu einer Routenbewertung zusammengefaßt werden. Es muß eine Art „Addition“ definiert werden.

Die Idee ist, die automatische Routenauswahl in zwei Schritten durchzuführen:

Der erste Schritt besteht darin, die Regeln der Experten zur Bewertung der Kanten zu verwenden. Dazu bietet sich der Einsatz eines regelbasierten Fuzzy-Systems an, ähnlich wie für „Call Admission Control“ und „Usage Parameter Control“ in [17]. Alle erfaßbaren Regeln können in dieses Verfahren einfließen. Das vorhandene Wissen wird genutzt, um eine Fuzzy-Regelbasis zu erstellen, wie in 4.2 beschrieben wird. Bei der Umsetzung des Wissens in eine logisch gleichwertige Fuzzy-Regelbasis werden die dort beschriebenen Besonderheiten beachtet. Das vollständige Regelsystem wird in 6.2 beschrieben. Mit diesem Fuzzy-Regelsystem wird das Verhalten des idealen Experten bei der Beurteilung der Kanten und Knoten nachgebildet.

Bei dieser Fuzzy-Bewertung der Kante werden die Parameter der Kante und ihrer angrenzenden Knoten verwendet - siehe Abbildung 3.4.

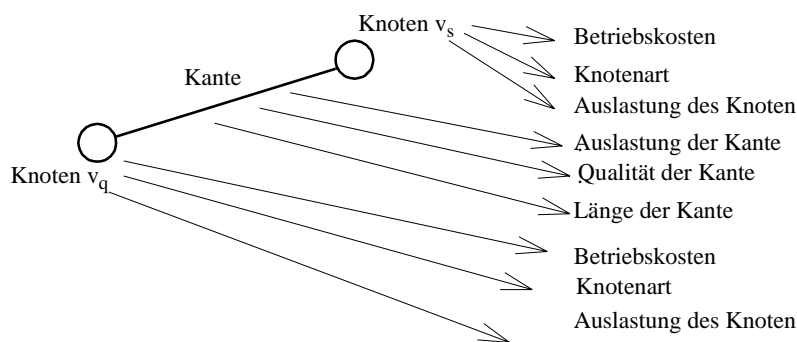


Abb. 3.4 Parameterverwendung der Fuzzy-Bewertung

Mit Hilfe des regelbasierten Fuzzy-Systems werden diese Parameter auf eine reelle Zahl projiziert. Die der Kante so zugeordnete reelle Zahl entspricht der Beurteilung der Kante durch den Experten. So entsteht ein kantenbewerteter Graph.

Durch die Projektion auf die reellen Zahlen entsteht eine additive Bewertung der Kanten, obwohl die projizierten Parameter zum nicht-additiv sind beziehungsweise sich für sie kein zur Routenbewertung geeignetes Distanzmaß angeben läßt. So wird das Verhalten eines menschlichen Entscheiders nachgebildet. Er merkt sich seine Beurteilung der Kanten und Knoten und nutzt diese für die nächste Entscheidung. Die guten Ergebnisse - siehe KAPITEL 6 - rechtfertigen dieses Vorgehen.

Im zweiten Schritt wird auf diesem fuzzy-bewerteten Graphen ein Routingalgorithmus eingesetzt - siehe Abbildung 3.5. Mit dieser Architektur bieten sich nun verschiedene automati-

sche Routing-Verfahren für Layer-Netzwerke an, die in den nachfolgenden Kapiteln im Einzelnen beschrieben werden.

3.5.1 Fuzzy-Routing

Mit den durch das regelbasierte Fuzzy-System ermittelten Werten werden die Kanten des Graphen neu bewertet. Das Ergebnis ist ein kantenbewerteter Graph, auf dem ein klassischer Kürzeste-Wege-Algorithmus eingesetzt werden kann. In dieser Arbeit wird ein klassischer Dijkstra Algorithmus eingesetzt.

Für den Graphen, der das Layer-Netzwerk repräsentiert, existieren also zwei Bewertungen:

- Die ursprüngliche Knoten- und Kantenbewertung, die durch die Parameter der Knoten und Kanten gegeben ist.
- Die Kantenbewertung, die durch das regelbasierte Fuzzy-System aus den Parameterwerten der Kanten und ihrer angrenzenden Knoten bestimmt wird. Diese Bewertung wird in dieser Arbeit mit Fuzzy-Bewertung bezeichnet.

Das gesamte Routing-Verfahren besteht aus zwei zunächst unabhängigen Komponenten, dem Fuzzy-Regelsystem, das die Fuzzy-Bewertung vornimmt, und dem Routensucher, der unter Verwendung des so entstandenen fuzzy-bewerteten Graphen die kürzeste Route bestimmt. Die Architektur ist in Abbildung 3.5 schematisch dargestellt.

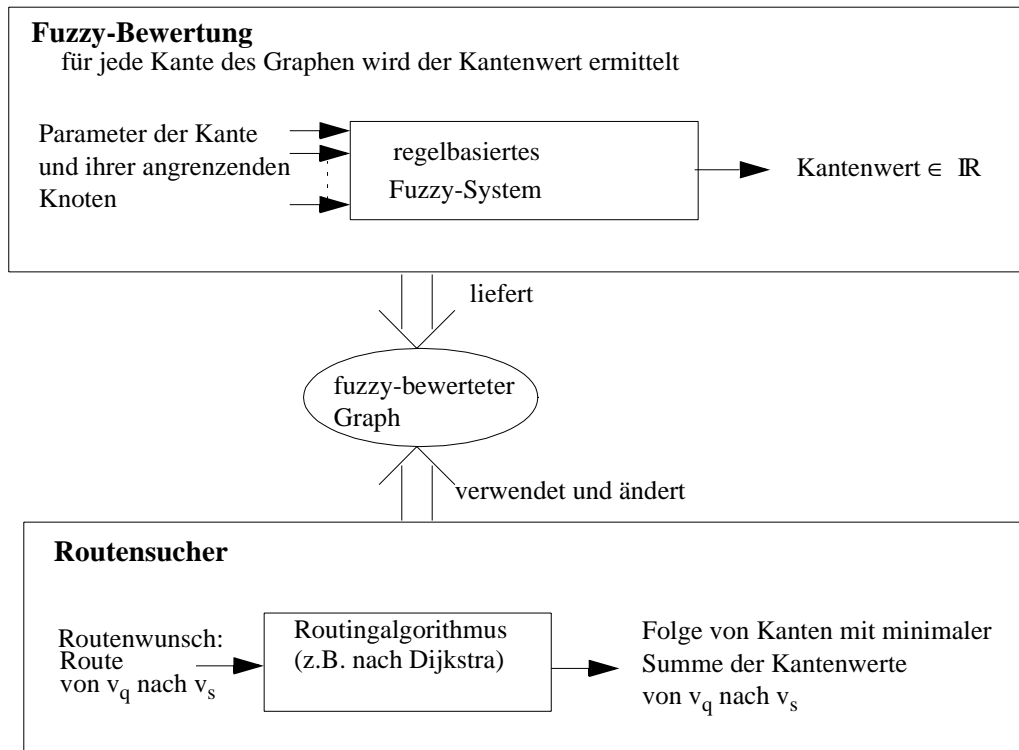


Abb. 3.5 Schematische Darstellung der Arbeitsweise von Fuzzy-Routing

Es wird davon ausgegangen, daß eine vom Routensucher bestimmte Route auch für eine Schaltung verwendet wird. Also ändert sich die Bewertung des Graphen, da sich, wie schon

oben erwähnt, die Auslastungswerte der in der Route enthaltenen Knoten und Kanten ändern. So muß nach Bestimmung einer Route auch die Fuzzy-Bewertung des Graphen mit Hilfe des regelbasierten Fuzzy-Systems neu bestimmt werden, um für die nächste Routensuche eine korrekte Fuzzy-Bewertung zu haben.

Beide Komponenten, Fuzzy-Bewertung des Graphen und Routensucher arbeiten also immer abwechselnd, wobei mit der Fuzzy-Bewertung des Graphen gestartet wird.

Bei sehr großen Transportnetzen bedeutet das Verwenden eines einzelnen Kanals eine prozentual kleine Veränderung der Auslastung der entsprechenden Kante und Knoten, so daß eine erneute Fuzzy-Bewertung des Graphen nicht nach jeder Routensuche durchzuführen ist. Wie oft die Fuzzy-Bewertung des Graphen in der Praxis notwendig und sinnvoll ist, hängt von der Größe des zugrunde liegenden Transportnetzes und dem Schaltaufkommen ab. Um eine gute Performanz eines solchen Systems zu gewährleisten, ist eine Fuzzy-Bewertung des Graphen in Zeiten mit geringem Schaltaufkommen anzuraten.

Das gesamte Verfahren, bestehend aus Fuzzy-Bewertung und Routensuche, wird mit **Fuzzy-Routing** bezeichnet. Das verwendete regelbasierte Fuzzy-System wird in KAPITEL 4 beschrieben. Fuzzy-Routing bestimmt Routen, deren Kanten und Knoten die gegebenen Regeln, soweit das im jeweils vorliegenden Layer-Netzwerk möglich ist, erfüllen.

Mit Fuzzy-Routing ist auch die Feinsteuerung der Routenauswahl möglich. Dazu wird ein zusätzlicher Parameter für jede Kante eingeführt und Regeln zu dessen Beurteilung. Die genaue Arbeitsweise der Feinsteuerung wird in 6.1.2.2 beschrieben.

Fuzzy-Routing ist mittels der in 6.1 beschriebenen Simulation getestet worden. Alle in 3.4 aufgeführten Validierungskriterien sind erfüllt worden. Obwohl die Qualitätskriterien nur als ein Kriterium neben vielen anderen in die Fuzzy-Regeln einfließen, werden vorwiegend Routen ausgewählt, die den Qualitätsansprüchen genügen. Eine ausführliche Darstellung der Ergebnisse ist in 6.2 gegeben.

Das Fuzzy-Regelsystem kann schnell und flexibel geändert werden, wenn sich die Anforderungen an die Bewertung ändern.

3.5.2 FuzzyPlus-Routing

Mit Fuzzy-Routing werden die Routen bestimmt, die die Regeln einhalten. Auf der so eingeschränkten Menge der Routen, können nun weitere Kriterien optimiert werden.

Für die Betreiber eines Transportnetzes sind die Einnahmen, die er mit seinem Transportnetz erzielen kann, ein wichtiger Aspekt sind. Deshalb wird als zusätzliches Optimierungskriterium bei der Routenauswahl die Kostenoptimierung eingesetzt.

Die Einnahmen für die Kundenservices werden von den Marktpreisen bestimmt und hängen nicht unbedingt von den realen Kosten ab, die dem Betreiber zur Bereitstellung der Übertragungswege entstehen. Da die Einnahmen außerdem nicht vom Routing-Verfahren beeinflusst werden können, werden sie hier nicht aufgenommen. Beurteilt werden ausschließlich die Kosten, die dem Betreiber durch die Bereitstellung der einzelnen Übertragungswege entstehen.

Für die Berechnung dieser Kosten existieren Funktionen. Jedoch sind diese Funktionen für die jeweiligen Betreiber der Transportnetze sehr vertrauliche Informationen, die aus Wettbewerbsgründen nicht veröffentlicht werden können. Daher verwendet diese Arbeit eine fiktive Kostenfunktion. Da die Kosten für die Knoten schon in den Regeln beachtet werden, wird

eine Kostenfunktion ausschließlich auf der Menge der Kanten betrachtet.

$$k: E \rightarrow \mathbb{R}$$

Die Kosten einer Route $r = (e_1, e_2, \dots, e_n)$, $e_i \in E$, $i = 1, \dots, n$ berechnen sich durch

$$\text{Kost}: \{\text{Routen im Transportnetz}\} \rightarrow \mathbb{R}$$

$$\text{Kost}(r) = \sum_{i=1}^n k(e_i)$$

Die Kosten sind so also als ein additiver Kantenparameter zu betrachten.

Es wird davon ausgegangen, daß die Kosten der Kanten von ihren Parameterwerten und den Parameterwerten der angrenzenden Knoten abhängen. Die in dieser Arbeit verwendete fiktive Kostenfunktion berechnet die Kosten jeder Kante in Abhängigkeit von den Werten ihrer Parameter Verfügbarkeit, Bitfehlerhäufigkeit und Länge. Die Regeln, die zur Beurteilung der Kanten und Knoten von den Experten verwendet werden, sagen implizit etwas über die Kosten aus, denn die Auswahl der Experten ist auch betriebswirtschaftlich angemessen. Daher werden die Kosten einer Kante nicht in die Fuzzy-Regeln integriert. Die Kostenoptimierung wird zusätzlich auf der Menge der Routen ausgeführt, die die Regeln einhalten.

Um diese harte Kostenoptimierung vorzunehmen, wird aus den k besten mit Fuzzy-Routing bestimmten Routen von einer Quelle v_q zu einer Senke v_s die mit den kleinsten Kosten gewählt. Dieses Verfahren wird mit **FuzzyPlus-Routing** bezeichnet. k kann vom Benutzer gewählt werden. Je größer k gewählt wird, um so eher können Routen gewählt werden, die zwar kostengünstig sind, aber schlechter im Sinne der Regeln.

Auch dieses Verfahren wurde mit der in 6.1 beschriebenen Simulation getestet und erfüllt alle in 3.4 aufgeführten Anforderungen. Eine ausführliche Darstellung der Ergebnisse wird in 6.3 gegeben.

Diese zusätzliche Kostenoptimierung kann auch mit den in 3.2.2 vorgestellten Verfahren durchgeführt werden, denn durch die Projektion der Parameter der Kanten und Knoten auf die reellen Zahlen liegen nun ausschließlich additive Parameter vor, die Kosten der Kante und ihre Fuzzy-Bewertung.

Jeder additive Kantenparameter eignet sich grundsätzlich dazu, eine Variante von FuzzyPlus-Routing zu definieren. Zum Beispiel kann statt einer zusätzlicher Kostenoptimierung auch die Minimierung der Schaltpunkte hinzugefügt werden. Dazu wird aus den ersten k besten mit Fuzzy-Routing bestimmten Routen von einer Quelle v_q zu einer Senke v_s , die mit der kleinsten Anzahl der Kanten ausgewählt. In dieser Arbeit wird beispielhaft die Kostenoptimierung betrachtet.

3.5.3 Fuzzy-Neuro-Verfahren

Eines der in 2.5.2 genannten Ziele ist mit Hilfe von Fuzzy-Routing und FuzzyPlus-Routing noch nicht erreicht. Bis jetzt wurde vorausgesetzt, daß das Regelwerk der Experten vollständig ist. In der Realität ist jedoch davon auszugehen, daß das Wissen der Experten nicht vollständig erfaßt ist. Daher soll es zusätzlich möglich sein, empirisches Wissen in die Regeln aufzunehmen.

Diese Daten bestehen jeweils aus einer Liste von einzelnen Routen, die sich im Sinne der Validierung als gut für das Layer-Netzwerk erwiesen haben.

Diese Anforderung kann mit Hilfe von künstlichen neuronalen Netzen erfüllt werden. Künstliche neuronale Netze können mit Hilfe ihrer Lernfähigkeit durch das Verarbeiten von Beispieldaten eine „Wissensbasis“ aufbauen. Um diese Fähigkeit nutzen zu können, wird das regelbasierte Fuzzy-System reversibel in ein neuronales Netz transformiert. Das der Fuzzy-Regelbasis äquivalente neuronale Netz wird mit Hilfe der vorhandenen Daten trainiert. Das in dem regelbasierten Fuzzy-System bereits enthaltene Wissen dient durch die Transformation in ein neuronales Netz als Initialisierung dieses neuronalen Netzes. Da die Transformation reversibel ist, kann nach Abschluß des Lernvorganges das neuronale Netz in ein regelbasiertes Fuzzy-System rücktransformiert werden. Nach der Rücktransformation in ein Fuzzy-System ist das vorher implizit vorhandene Wissen nun explizit in Form von Fuzzy-Regeln vorhanden. Diese Regeln können von den Anwendern gelesen, interpretiert und gegebenenfalls korrigiert werden. Das durch das Training veränderte Fuzzy-System kann zur Fuzzy-Bewertung des Graphen für Fuzzy-Routing und FuzzyPlus-Routing eingesetzt werden.

Dieser gesamte Prozeß wird mit **Fuzzy-Neuro-Verfahren** bezeichnet und in KAPITEL 5 vorgestellt.

Die Tests haben gezeigt, daß mit Hilfe dieses Verfahrens, das regelbasierte Fuzzy-System den Daten entsprechend adaptiert werden kann. Eine genaue Beschreibung der Ergebnisse wird in 6.4 gegeben.

3.5.4 Einbettung existierender Verfahren

Nach der Projektion der Kanten- und Knotenparameter auf die reellen Zahlen können alle in 3.2 aufgeführten Verfahren zur eigentlichen Routenbestimmung eingesetzt werden.

Zum Beispiel können alle nicht-additiven Parameter der Kanten und Knoten mit Hilfe des regelbasierten Fuzzy-Systems auf die reellen Zahlen projiziert werden. Dann kann mit dem von Bistarelli beschriebenen SCLP die Route bestimmt werden. Das Ergebnis ist zunächst nicht eine Route, sondern eine Menge „gleich guter“ Routen, aus der ein menschlicher Entscheider eine auswählen muß.

Gesucht ist jedoch ein Verfahren, das ohne menschlichen Entscheider auskommt. SCLP müßte also noch um ein automatisches Auswahlverfahren ergänzt werden. Das ist leicht möglich. Die einfachste Variante ist, immer die erste Route der Ergebnisliste zu nehmen. Als aufwendigere Lösung bietet sich der Einsatz eines neuronalen Netzes an, das die Auswahl des menschlichen Entscheiders lernt und nach der Trainingsphase die Entscheidung vornimmt.

Die in FuzzyPlus-Routing zusätzlich vorgenommene Kostenoptimierung kann auch mit dem von Ziegelmann entwickelten Verfahren durchgeführt werden. Die Beschriftung der Kanten besteht in diesem Fall aus Kosten und der Fuzzy-Bewertung.

Diese hier genannten Verfahren werden in dieser Arbeit jedoch nicht weiter verfolgt, da sie bereits in der genannten Literatur ausführlich beschrieben werden.

3.5.5 Fuzzy-Bewertung im Algorithmus von Dijkstra

Bei den bisher beschriebenen Verfahren wird die Fuzzy-Bewertung des Graphen durchgeführt, bevor ein Kürzeste-Wege-Algorithmus seine Arbeit beginnt. Es ist nun zu überlegen, ob es sinnvoll ist, die Bewertung der Kanten, die durch das regelbasierte Fuzzy-System zur

Verfügung steht, in den Kürzeste-Wege-Algorithmus zu integrieren. Diese Überlegungen werden im nachfolgenden Text für den Algorithmus von Dijkstra durchgeführt.

Im klassischen Algorithmus von Dijkstra zur Bestimmung einer kürzesten Route von dem Knoten v_q zu dem Knoten v_s ist für jeden Knoten v des Graphen das folgende Minimum zu bestimmen:

$$\text{Min}(D[v], D[w]+l(v,w))$$

wobei $D[v]$ die Länge der kürzesten Route von v_q nach v ist, $D[w]$ die entsprechende Länge zum Knoten w und $l(v,w)$ die Länge der Kante von v nach w . Statt der Länge wird jeweils der vom regelbasierten Fuzzy-System bestimmte Wert der Kante verwendet.

Wenn die Fuzzy-Bewertung der einzelnen Kanten jeweils aktuell während der Berechnung von $\text{Min}(D[v], D[w]+l(v,w))$ durchgeführt wird, so ist das Ergebnis dasselbe wie das von Fuzzy-Routing. Denn die jeweils aktuell bestimmten Kantenwerte sind dieselben, die bei der Fuzzy-Bewertung vor der Routensuche berechnet werden. Die Belegung der Kante und damit die Veränderung der Auslastung der Kanten und Knoten wird nach der Bestimmung der Route vorgenommen. Ein Vorteil für die Performanz entsteht nicht, da auch bei dem Algorithmus von Dijkstra alle Kanten fuzzy-bewertet werden, genau wie bei der Gewichtung des gesamten Graphen. Im Gegenteil: wenn die Fuzzy-Bewertung in den Dijkstra-Algorithmus integriert ist, werden dieselben Kanten mehrfach einer Fuzzy-Bewertung unterzogen.

Angenommen, der zu betrachtende Graph ist mit mehreren Parametern kantenbewertet und jeder Parameter ist additiv entlang jeder Route. Dann ist es möglich, wie schon in [14] erläutert, die Bewertung der Kanten durch das regelbasierte Fuzzy-System in den Algorithmus von Dijkstra zu integrieren.

Da aber nicht alle Parameter des Layer-Netzwerkes additiv entlang jeder Route sind, wie schon in 3.1.2 dargelegt wurde, ist diese Idee an dieser Stelle nicht weiter verfolgt worden.

In diesem Kapitel wird das regelbasierte Fuzzy-System beschrieben, das den Kern aller in dieser Arbeit vorgestellten Verfahren bildet. In 4.1 werden die Grundlagen der Fuzzy-Logik kurz vorgestellt, um die verwendete Terminologie zu klären. 4.2 beschreibt die von Fuzzy-Routing verwendeten Komponenten des regelbasierten Fuzzy-Systems.

4.1 Grundlagen der Fuzzy-Logik

Die Fuzzy-Logik, die eine Verallgemeinerung der zweiwertigen Logik darstellt, wurde Mitte der 60-iger Jahre von Lotfi Zadeh entwickelt. Auf diesen Grundlagen basiert das 1973 vorgestellte regelbasierte Fuzzy-System, das Mamdani 1977 weiterentwickelte. Es dient der Modellierung eines komplexen Systems mit dem Ziel, Wissen von Experten nachzubilden. Die Modellierung eines solchen Systems besteht im Kern aus einer Regelbasis, die zusammen mit den festgelegten Fuzzy-Funktionen die sogenannte Wissensbasis darstellt, und einer Inferenz-Maschine. Die Daten des Eingaberaumes X müssen zunächst fuzzyfiziert werden, bevor sie von der Inferenz-Maschine verarbeitet werden. Die Inferenz-Maschine liefert als Ausgabe die Werte der einzelnen Regeln oder ihren gemeinsamen Wert, je nach verwendetem Inferenz-Mechanismus. Um die ermittelten Werte verwenden zu können, muß aus den Fuzzy-Regelausgaben mittels Defuzzifizierung ein harter Wert des Ausgaberaumes Y hergestellt werden. Abbildung 4.1 zeigt den schematischen Aufbau eines solchen Systems.

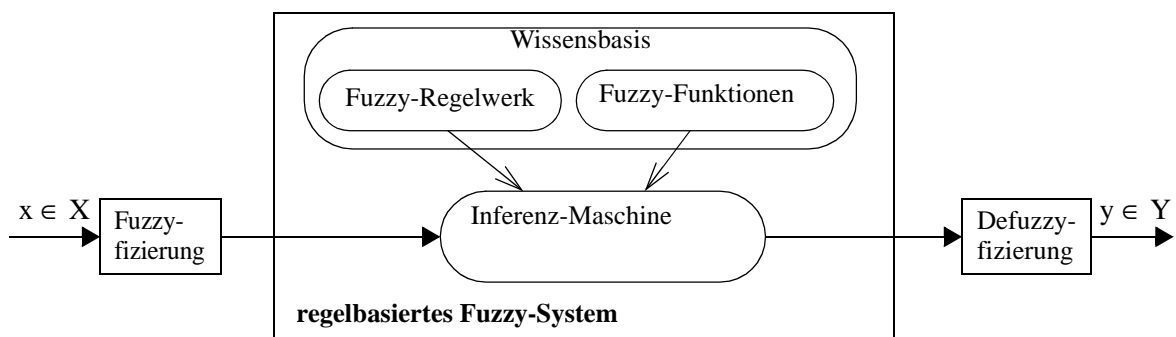


Abb. 4.1 Regelbasiertes Fuzzy-System

Die mathematischen Methoden, auf denen diese Systeme basieren, werden im folgenden dar-

gestellt. Diese Darstellung geht nur auf die Fakten und Definitionen genauer ein, die für diese Arbeit von Bedeutung sind. Der an weiteren Details interessierte Leser sei zum Beispiel auf die Bücher von Bothe [5], Klir, George, Yuan [25] oder Driankov, Hellendoorn, Reinfrank [11] verwiesen. Dem letztgenannten ist die hier verwendete Notation entnommen.

Zwischen Fuzzy-Logik und Fuzzy-Mengenlehre besteht eine analoge Isomorphie wie zwischen der klassischen Logik und der klassischen Mengenlehre, die die Übertragung aller Operationen und Eigenschaften aus der Mengenlehre auf die Logik erlaubt.

Im Gegensatz zu den Zugehörigkeitsfunktionen der klassischen Mengenlehre nehmen die Zugehörigkeitsfunktionen der Fuzzy-Mengen jeden beliebigen Wert aus dem Intervall $[0,1]$ an. Eine Fuzzy-Menge ist durch ihre Definitionsmenge, auch Universum genannt, und durch ihre Zugehörigkeitsfunktion bestimmt.

Def 4.1: Eine **Fuzzy-Menge** auf dem Universum U ist eine Menge

$$F = \{(u, \mu_F(u)) \mid u \in U\}$$

wobei

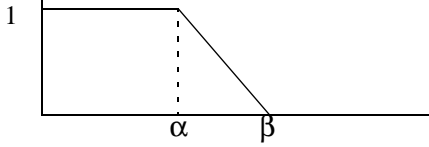
$\mu_F: U \rightarrow [0, 1]$, eine Funktion ist, die **Zugehörigkeitsfunktion** von F genannt wird.

$\mu_F(u)$ bezeichnet den Grad, mit dem u zur Menge F gehört.

Viel verwendete Zugehörigkeitsfunktion für die reellen Zahlen, $U = \mathbb{R}$ sind zum Beispiel:

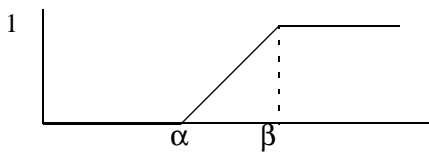
Def 4.2: Seien $\alpha, \beta \in \mathbb{R}$ mit $\alpha < \beta$.

Die **L-Funktion** ist eine Funktion von \mathbb{R} nach $[0,1]$ mit

$$L(u; \alpha; \beta) = \begin{cases} 1 & \text{für } u < \alpha \\ \frac{\beta - u}{\beta - \alpha} & \text{für } \alpha \leq u \leq \beta \\ 0 & \text{für } u > \beta \end{cases}$$


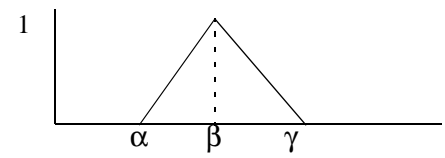
Def 4.3: Seien $\alpha, \beta \in \mathbb{R}$ mit $\alpha < \beta$.

Die **Γ -Funktion** ist eine Funktion von \mathbb{R} nach $[0,1]$ mit

$$\Gamma(u; \alpha; \beta) = \begin{cases} 0 & \text{für } u < \alpha \\ \frac{u - \alpha}{\beta - \alpha} & \text{für } \alpha \leq u \leq \beta \\ 1 & \text{für } u > \beta \end{cases}$$


Def 4.4: Seien $\alpha, \beta, \gamma \in \mathbb{R}$ mit $\alpha < \beta < \gamma$.

Die **Λ -Funktion** ist eine Funktion von \mathbb{R} nach $[0,1]$ mit

$$\Lambda(u; \alpha; \beta; \gamma) = \begin{cases} 0 & \text{für } u < \alpha \\ \frac{u - \alpha}{\beta - \alpha} & \text{für } \alpha \leq u \leq \beta \\ \frac{\gamma - u}{\gamma - \beta} & \text{für } \beta \leq u \leq \gamma \\ 0 & \text{für } u > \gamma \end{cases}$$


Der Definitionsbereich einer Zugehörigkeitsfunktion wird durch die Anwendung vorgegeben. Die verwendete Form der Zugehörigkeitsfunktion hängt von vielen Faktoren ab. Da für jede Variable im System mehrere Zugehörigkeitsfunktionen zu definieren sind, spielen zum Beispiel Speicherverbrauch, Recheneffizienz und Performanz eine große Rolle bei der Wahl der Zugehörigkeitsfunktionen. Da sich die \wedge -Funktionen in dieser Hinsicht als ökonomisch erwiesen haben, werden sie oft eingesetzt.

Alle Operatoren der klassischen Mengenlehre lassen sich auf Fuzzy-Mengen übertragen. Auf den Randwerten 0 und 1 sollten sich diese Operatoren wie die entsprechenden Operatoren der zweiwertigen Logik verhalten.

Seien $A = \{(u, \mu_A(u)) | u \in U\}$ und $B = \{(u, \mu_B(u)) | u \in U\}$ zwei Fuzzy-Mengen auf dem Universum U , so ist

$$A \cup B = \{(u, \mu_{A \cup B}(u)) | u \in U\},$$

$$A \cap B = \{(u, \mu_{A \cap B}(u)) | u \in U\},$$

$$A^c = \{(u, \mu_{A^c}(u)) | u \in U\},$$

mit Zugehörigkeitsfunktionen wie nachfolgend beschrieben:

Die grundlegenden Eigenschaften von Durchschnitt und Vereinigung zweier Mengen sind mit Hilfe von T-Norm und S-Norm erfaßt. Zugehörigkeitsfunktionen, die die Eigenschaften einer S-Norm aufweisen, sind für die Definition des Durchschnitts zweier Mengen geeignet und solche, die die Eigenschaften einer T-Norm aufweisen für die Definition der Vereinigung.

Die Zugehörigkeitsfunktion für das Komplement einer Menge ist wie folgt definiert:

$$\forall u \in U: \mu_{A^c}(u) = 1 - \mu_A(u).$$

Zadeh schlug für die Zugehörigkeitsfunktionen von Durchschnitt und Vereinigung zweier Mengen folgendes vor [11]:

$$\forall u \in U: \mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u)),$$

$$\forall u \in U: \mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u)),$$

Diese Zugehörigkeitsfunktionen haben die oben genannten Eigenschaften. Auch

$$\forall u \in U: \mu_{A \cap B}(u) = \mu_A(u) \cdot \mu_B(u)$$

ist eine T-Norm und

$$\forall u \in U: \mu_{A \cup B}(u) = \min(1, \mu_A(u) + \mu_B(u))$$

ist eine S-Norm.

Die zuletzt genannte Zugehörigkeitsfunktion der Vereinigung wird zum Beispiel bei der Abbildung eines regelbasierten Fuzzy-Systems in ein RBF-Netz verwendet; siehe hierzu 5.2.1.

Die genaue Definition der T- und S-Norm wie auch eine Liste der gebräuchlichsten Zugehörigkeitsfunktionen für die Mengenoperationen kann zum Beispiel in dem Buch von Fullér [12] nachgelesen werden.

Sei U eine klassische Menge und A und B zwei klassische Teilmengen von U . Die Eigenschaft eines Elementes x aus U zur Menge A zu gehören wird durch die Zugehörigkeitsfunktion

$$\mu_A(x) = \begin{cases} 1 & \text{für } x \in A \\ 0 & \text{für } x \notin A \end{cases}$$

beschrieben. Der Wahrheitswert τ , der Aussage $p = „x \text{ ist in } A“$ ist durch diese Zugehörigkeitsfunktion gegeben und die Zugehörigkeitsfunktion μ_A ist durch den Wahrheitswert der Aussage p gegeben. Es gilt:

$$\tau(p) = \mu_A(x)$$

Sei y eine Element von U und q eine weitere Aussage mit $q = „y \text{ ist in } B“$. Der Wahrheitswert der Implikation $p \rightarrow q$ ist durch die Wahrheitswerte der Aussage von p und q gegeben, denn es gilt

$$p \rightarrow q \Leftrightarrow (p \wedge q) \vee \neg p ,$$

Somit ist der Wahrheitswert von $p \rightarrow q$ auch durch die Werte der Zugehörigkeitsfunktionen der Mengen A und B gegeben. Er lässt sich als Funktion der Zugehörigkeitsfunktionen der Mengen A und B ausdrücken.

Wenn A und B zwei Fuzzy-Mengen mit Zugehörigkeitsfunktionen μ_A und μ_B sind, dann wird die Zugehörigkeitsfunktion für die **Fuzzy-Implikation**, daran angelehnt wie folgt punktweise als Funktion I der Zugehörigkeitfunktionen μ_A und μ_B definiert:

$$\mu_I(x, y) = I(\mu_A(x), \mu_B(y))$$

Zadeh berechnete

$$\mu_I(x, y) = \max(\min(\mu_A(x), \mu_B(y)), 1 - \mu_A(x)) .$$

Das entspricht der Übertragung von $p \rightarrow q \Leftrightarrow (p \wedge q) \vee \neg p$ aus der klassischen Logik.

Mamdani verwendet

$$\mu_I(x, y) = \min(\mu_A(x), \mu_B(y)) .$$

und interpretiert damit $p \rightarrow q$ als $p \wedge q$. Obwohl diese Interpretation für die Wahrheitswerte 0 der Aussage p falsch ist, wird sie häufig verwendet, denn in regelbasierten Fuzzy-System spielen Aussagen, deren erster Teil falsch ist, meist keine Rolle [12].

Mit Hilfe von Fuzzy-Mengen können qualitative Eigenschaften von Objekten beschrieben werden. Zum Beispiel kann die Eigenschaft einer Zahl $u \in \mathbb{R}$ verbal durch

„ u ist ungefähr in $[a, b]“$

beschrieben als Fuzzy-Menge beschrieben werden. Als Zugehörigkeitsfunktion wird eine Trapezfunktion, die aus L-Funktion und Λ -Funktion gebildet werden kann, verwendet. Die in Abbildung 4.2 dargestellte Zugehörigkeitsfunktion stellt diese Eigenschaft, „u ist ungefähr in [a,b]“, dar.

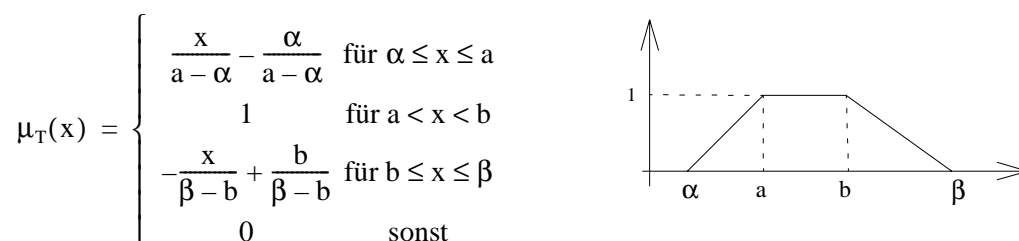


Abb. 4.2 Trapezfunktion

Um die Eigenschaften von Objekten angemessen erfassen zu können, wird der Begriff der linguistischen Variablen eingeführt. Eine linguistische Variable faßt mehrere Fuzzy-Mengen unter einem beschreibenden Namen zusammen. Sie bildet sprachlich beschriebene Eigenschaften auf mathematische Funktionen, besser auf Fuzzy-Mengen ab. Genauer:

Def 4.5: Eine **linguistische Variable** ist ein 4-Tupel

$\langle X, LX, PX, M_X \rangle$, mit:

X bezeichnet den symbolischen Namen.

LX bezeichnet die linguistischen Werte.

PX ist der Wertebereich. Er kann sowohl diskret als auch kontinuierlich sein.

M_X ist die Interpretation des linguistischen Wertes in quantitativer Form. M_X ordnet jedem Element aus LX eine konkrete Zugehörigkeitsfunktion über PX zu.

So kann die Eigenschaft einer Kante, eine kleine, mittlere oder große Verfügbarkeit zu haben, mit Hilfe der linguistischen Variablen mit dem symbolischen Namen Verfügbarkeit und den linguistischen Werten groß, mittel und klein beschrieben werden. Durch die Zuordnung von Zugehörigkeitsfunktionen wird den sprachlichen Werten eine mathematische Bedeutung gegeben. Diese Abbildung der qualitativen Eigenschaften groß, mittel und klein des Parameters Verfügbarkeit einer Kante auf die entsprechende linguistische Variable ist in Abbildung 4.3 dargestellt.

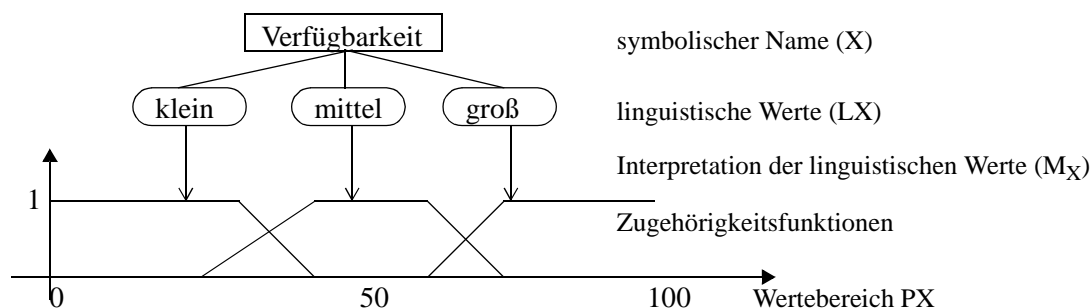


Abb. 4.3 Beispiel einer linguistischen Variablen

Sei X eine linguistische Variable $\langle X, LX, PX, M_X \rangle$ mit $LX = \{Z, S, M, B\}$. Aussagen der Form

$$X \text{ ist } Z$$

werden als **atomare Fuzzy-Aussagen** bezeichnet. Die Zugehörigkeitsfunktion für den linguistischen Wert Z gibt den Wahrheitswert der Aussage „ X ist Z “ an.

Die sprachliche Aussage:

Die Verfügbarkeit hat einen großen Wert.

kann so als die atomare Fuzzy-Aussage

$$\text{Verfügbarkeit ist groß}$$

ausgedrückt werden.

Mit Hilfe der Fuzzy-Mengenoperatoren können fuzzy-logische Verknüpfungen, auch fuzzy-logische Operatoren genannt, definiert werden. Zum Beispiel

$$X \text{ ist } Z \text{ ODER } X \text{ ist } M$$

wird definiert durch

$$X \text{ ist } Z \cup M.$$

Und

$$X \text{ ist } Z \text{ UND } X \text{ ist } M$$

wird definiert durch

$$X \text{ ist } Z \cap M.$$

Mit Hilfe der so definierten fuzzy-logischen Operatoren können mehrere atomare Fuzzy-Aussagen zu einer Fuzzy-Aussage zusammengefaßt werden. Mit diesen Mitteln lassen sich **Fuzzy-Regeln** formulieren. Sie haben die Form:

$$\text{if } \langle \text{Fuzzy-Aussage} \rangle \text{ then } \langle \text{Fuzzy-Aussage} \rangle,$$

wobei $\langle \text{Fuzzy-Aussage} \rangle$ sowohl eine atomare wie auch eine zusammengesetzte Aussage sein kann. Die Bedeutung der Fuzzy-Regel ist durch die entsprechende Fuzzy-Implikation gegeben:

sei X eine linguistische Variable $\langle X, LX, PX, M_X \rangle$ mit $LX = \{Z, S, M, B\}$ und Y eine weitere linguistische Variable $\langle Y, LY, PY, M_Y \rangle$ mit $LY = \{Z, S, M, B\}$. Z sei auf PX durch die Zugehörigkeitsfunktion $\mu_Z: PX \rightarrow [0, 1]$, und B auf PY durch $\mu_B: PY \rightarrow [0, 1]$ gegeben ist. Die Bedeutung der Regel

$$\text{if } X \text{ ist } Z \text{ then } Y \text{ ist } B$$

ist gegeben durch:

$$\begin{aligned} \mu_I: PX \times PY &\rightarrow [0, 1] \\ (x, y) &\rightarrow \mu_I(x, y) = I(\mu_Z(x), \mu_B(y)) \end{aligned}$$

mit I wie weiter oben definiert.

Sind statt der angegebenen atomaren Fuzzy-Aussagen zusammengesetzte Aussagen gegeben, so wird $\mu_Z(x)$ bzw. $\mu_B(y)$ durch die entsprechende Zugehörigkeitsfunktion ersetzt.

Das hier präsentierte Verfahren verwendet die von Mamdani vorgeschlagene Fuzzy-Implikation. Sie ist im Bereich Fuzzy-Control eine der wichtigsten Implikationsmethoden.

Für die Regel

if X ist Z then Y ist B

wird die linguistische Variable X auch als Eingabe-Variable der Regel bezeichnet, Y als Ausgabe-Variable.

Ein- und Ausgabe einer Regel besteht je aus einer Fuzzy-Menge. Die Ausgabe wird wie nachfolgend beschrieben aus der Eingabe berechnet:

Als Voraussetzung sei die oben genannte Regel gegeben und die Tatsache, daß X den Wert x^* aus PX annimmt. Das wird mit Hilfe der Fuzzy-Menge E

$$\mu_E(x) = \begin{cases} 1 & \text{für } x=x^* \\ 0 & \text{sonst} \end{cases}$$

fuzzifiziert als „X ist E“ dargestellt. Es gilt $\mu_Z(x^*) = c$. Dann wird die Konklusion, also die Ausgabe der Regel,

„y ist B“

mit Hilfe der Kompositionsregel der Inferenz berechnet und es gilt:

$$\mu_{B'}(y) = \sup(\min\{\mu_E(x), I(\mu_Z(x), \mu_B(y))\} \mid x \in PX) \quad \forall y \in PY.$$

Wird für die Implikation die von Mamdani verwendet, so ist also die Ausgabe der Regel für die Eingabe „X ist E“:

$$\mu_{B'}(y) = \min(\mu_Z(x^*), \mu_B(y)) = \min(c, \mu_B(y))$$

Dies entspricht der auf der Höhe c gekappten Zugehörigkeitsfunktion der Ausgabe-Variable Y wie in Abbildung 4.4 dargestellt. c wird **Erfüllungsgrad** der Regel genannt. Er bezeichnet den Wert der Zugehörigkeitsfunktion ihrer Prämisse an dem entsprechenden harten Eingabewert.

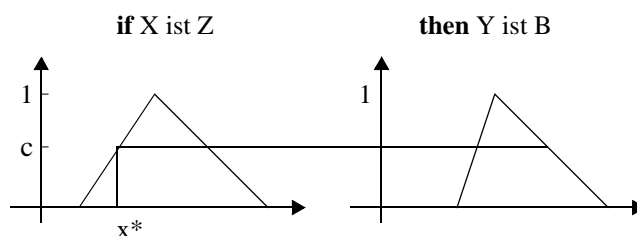


Abb. 4.4 Graphische Darstellung der Regelauswertung nach Mamdani

Ein Fuzzy-Regelsystem besteht aus einer Menge von Regeln $R_i, i=1..n$. Es gibt zwei verschiedene Wege, die Ausgabe des Regelsystems bei einer konkreten Eingabe zu berechnen.

Zum Einen können alle Regeln zu einer kombiniert werden. Aus dieser Kombination wird dann die Konklusion wie oben beschrieben berechnet. Nach Mamdani wird zum Beispiel die

Kombination der Regeln R durch

$$R = \bigcup_{i=1}^n R_i$$

berechnet.

Zum Anderen kann die Konklusion für jede Regel einzeln berechnet werden, und die einzelnen Ergebnisse werden kombiniert.

Die verwendete Fuzzy-Implikation und die Art und Weise wie die einzelnen Regeln in die Berechnung des Gesamtergebnis eingehen, wird im folgenden zusammenfassend mit Inferenzmechanismus bezeichnet.

Ein regelbasiertes Fuzzy-System erwartet als Eingabe Fuzzy-Mengen. Daher wird eine „harte“ Eingabe zunächst fuzzifiziert. Die Ausgabe eines Regelsystems ist ebenfalls eine Fuzzy-Menge. Um einen „harten“ Ausgabewert zu erhalten, wird die Ausgabe des Regelsystems defuzzifiziert.

Eine Darstellung der verschiedenen Möglichkeiten, ein regelbasiertes Fuzzy-System auszuwerten, wie auch eine Klassifizierung der verschiedenen Defuzzifizierungsmethoden, findet sich zum Beispiel in [16] und [12].

Die Menge der Regeln sollte auf Vollständigkeit, Konsistenz und Stetigkeit überprüft werden. Die Vollständigkeit eines Regelsystems ist in der Praxis meist nicht gegeben, da nicht alle Eingabekombinationen in der realen Anwendung vorkommen. Vollständigkeit in der Praxis heißt, alle in der realen Anwendung vorkommenden Kombinationen werden abgedeckt.

Für die Konstruktion eines regelbasierten Fuzzy-Systems sind also folgende Bestandteile festzulegen:

- Mengenoperatoren bzw. fuzzy-logische Operatoren
- linguistische Variablen für Eingabe und Ausgabe
- Regelsystem
- Inferenzmechanismus
- Defuzzifizierungsmethode

Die Veränderung jedes einzelnen Bestandteils beeinflusst die Bedeutung und Mächtigkeit der anderen.

4.2 Eingesetztes Fuzzy-System

Für diese Arbeit wird das regelbasierte Fuzzy-System mit Hilfe eines neuronalen Netzes realisiert. Um eine einfache und übersichtliche Implementierung zu erhalten, sind die im folgenden aufgeführten Mengenoperatoren gewählt worden. Eine genaue Beschreibung der Abbildung eines regelbasierten Fuzzy-Systems mit Hilfe eines neuronalen Netzes findet sich in 5.2. Die Motivation für die Auswahl des hier beschriebenen Verfahrens ist in 5.4 dargestellt.

Im folgenden werden die verwendeten Bestandteile des Fuzzy-Regelsystems beschrieben.

Mengenoperatoren

Die Zugehörigkeitsfunktionen von Durchschnitt und Vereinigung sind wie folgt definiert:

Seien A und B zwei Fuzzy-Mengen.

$$\mu_{A \cap B} = \max(\mu_A(x) + \mu_B(x) - 1, 0) \quad (4.1)$$

$$\mu_{A \cup B} = \max\left(\frac{1}{2}(\mu_A(x) + \mu_B(x)), 0\right) \quad (4.2)$$

Die Definition des Durchschnitts geht auf Lukasiewicz zurück und erfüllt alle Eigenschaften einer T-Norm [12]. Die Vereinigung verhält sich auf den Randwerten 0 und 1 nicht wie in der klassischen Mengenlehre:

$$\mu_{A \cup B}(0, 0) = 0 \quad (4.3)$$

$$\mu_{A \cup B}(1, 0) = \mu_{A \cup B}(0, 1) = 0,5 \neq 1 \quad (4.4)$$

Dieser Operator ist also strenger. Wegen Gleichung (4.4) ist er keine S-Norm. Außerdem ist „tertium non datur“ nicht erfüllt, denn

$$\mu_{A \cup B}(x, 1 - x) = 0,5 \neq 1 \quad (4.5)$$

Dies ist bei Umformungen von Fuzzy-Ausdrücken, die mittels fuzzy-logischen Operatoren gebildet werden, die auf diesen Mengenoperationen basieren, zu beachten. Da jedoch die fuzzy-logischen Ausdrücke aus den Regeln direkt durch eine neuronale Verschaltung von UND- und ODER-Neuronen abgebildet werden, ist diese Unzulässigkeit der Umformung nicht relevant [4].

Linguistische Variablen für Eingabe und Ausgabe

Gegeben sei ein beliebiges Layer-Netzwerk $G(V,E)$ wie in 3.1.1 definiert. Für die Fuzzy-Bewertung jeder Kante $e = (v,w) \in E$ mit $v, w \in V$ werden folgende Parameter verwendet:

- Betriebskosten des Knoten v
- Knotenart des Knoten v
- Auslastung des Knoten v
- Auslastung der Kante e
- Verfügbarkeit der Kante e
- Bitfehlerhäufigkeit der Kante e
- physikalische Länge der Kante e
- Betriebskosten des Knoten w
- Knotenart des Knoten w
- Auslastung des Knoten w

Für jeden dieser Parameter wird eine linguistische Eingabe-Variable definiert. Die Potenzmenge aus den Wertemengen jeder dieser Kantenparameter definiert den Eingaberaum. Die genaue Definition der Parameter und ihrer Wertebereiche kann in 6.1.2.1 nachgelesen werden.

Der Ausgaberaum ist \mathbb{R} . Es gibt eine einzige linguistische Ausgabe-Variable, den Kantenswert. Er beinhaltet das Projektionsergebnis der Kanten- und Knotenparameter auf die reellen Zahlen.

Für jede linguistische Variable werden drei linguistische Werte definiert, die mit Hilfe von Γ , Λ oder L -Funktionen dargestellt werden. Abbildung 4.5 zeigt ein Beispiel.

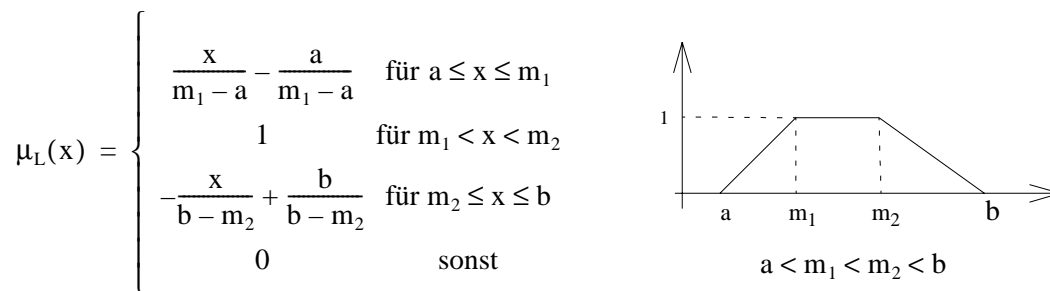


Abb. 4.5 Beispiel einer Zugehörigkeitsfunktion eines linguistischen Wertes

Regelsystem

Um das Regelsystem zu formulieren, steht das Wissen der Experten, wie in 3.3 beschrieben, zur Verfügung. Die Experten in den Schaltzentren verfügen über Erfahrungs- und Expertenwissen. Dieses Wissen wird durch Interviews und Diskussionen erfaßt und anschließend adäquat modelliert. So entsteht ein Überblick über das gesamte Wissen zur Fuzzy-Bewertung der Kanten, das als Basis zur Routenauswahl in Festnetzen dient.

Ziel ist die Bewertung der Kanten zur Routensuche für Neu- und Ersatzschaltungen im laufenden Betrieb des Festnetzes. In dieser Arbeit wird eine Lösung für die hochprioritären Übertragungswege aufgezeigt, wie schon in 3.3 dargelegt.

Im Zuge der Wissensmodellierung entsteht eine Liste von Aussagen der folgenden Art:

- *Teure Knoten sollten in einer Route möglichst nicht verwendet werden.*
- *Die Auslastung der Kanten sollte möglichst gleichmäßig sein.*
- *Kanten mit hoher Verfügbarkeit und kleiner Bitfehlerhäufigkeit sind zu bevorzugen.*
- *Es sollte keine Kante gewählt werden, deren Verfügbarkeit eine bestimmte Grenze unterschreitet, wenn der Kunde einen qualitativ hochwertigen Übertragungswege bestellt hat.*
- *Ist die Bitfehlerhäufigkeit einer Kante sehr hoch, so sollte sie nicht für hochprioritäre Übertragungswege genutzt werden.*

Diese Liste der Aussagen soll einen Eindruck geben von der Art des vorliegenden Materials und ist nur exemplarisch zu verstehen. Sie zeigt auch schon die Entscheidungskonflikte, die bei der Beachtung von mehreren Parametern entstehen können; zum Beispiel können Kanten mit hoher Verfügbarkeit an teuren Knoten enden.

Nachdem die Liste der Aussagen vollständig erfaßt ist, erfolgt die Wissensrepräsentation, d.h. die Abbildung des Wissens in eine passende formale Struktur. Diese formale Struktur ist gegeben durch:

- die Vorschrift, die Aussagen in Form von Fuzzy-Aussagen und Fuzzy-Regeln abzubilden
- die Objekte und ihre Attribute, also die Kanten und Knoten, und ihre Parameter
- die Eingabe- und Ausgabe-Variablen

Die linguistischen Werte der linguistischen Ein- und Ausgabe-Variablen werden bei diesem Vorgang festgelegt.

Als Ergebnis entstehen Regeln, deren Prämissen die logischen Verknüpfungen von den verschiedenen Parameterwerten der Kanten sind. Die Konklusion der Regeln ist jeweils ein linguistischer Wert der linguistischen Variable Kantenwert. Je kleiner der Kantenwert, um so eher wird diese Kante gewählt, da der Kürzeste-Wege-Algorithmus die Summe der Kantenwerte minimiert. So entstehen Regeln der Form:

IF Betriebskosten des angrenzenden Knoten v hoch **UND** Betriebskosten für angrenzenden Knoten w hoch **THEN** Kantenwert = hoch

Die Aussagen

- *Kaum ausgelastete Knoten sollen nicht mehr genutzt werden, um sie eventuell ganz frei schalten zu können und anschließend an einem anderen Ort einsetzen zu können.*
- *Kanten mit sehr kleiner Auslastung sind abzulehnen.*
- *Knoten und Kanten sollen möglichst gleichmäßig ausgelastet werden.*

betreffen den zeitlich veränderbaren Parameter Auslastung. Regeln bewerten die Kanten zu einem festen Zeitpunkt. Mit der Formulierung

IF Auslastung der Kante sehr klein **THEN** Kantenwert = hoch

IF Auslastung der Kante mittel **THEN** Kantenwert = klein

IF Auslastung der Kante groß **THEN** Kantenwert = hoch

und den entsprechenden Regeln für die Auslastung der angrenzenden Knoten wird die gewünschte Bewertung erreicht. Denn solange noch Kanten mit mittlerer Auslastung vorhanden sind, werden diese bevorzugt. Sind nach längerem Einsatz fast alle Kanten sehr hoch ausgelastet, dann werden alle gleich behandelt.

Nicht alle Aussagen lassen sich direkt in eine solche Regel übertragen. Strategische Aussagen der Form

- *Es gibt Kanten, die nur für ganz bestimmte Schaltungen genutzt werden sollten.*
- *Der Kunde sollte nicht mehr Qualität bekommen als ihm zugesagt wurde, da sonst der Aufwand für die Leitung zu hoch ist.*

bedürfen einer weitergehenden Modellierung. Sie werden durch die Einführung zusätzlicher Kantenparameter und durch das Hinzufügen von besonderen Regeln erfaßt - siehe 6.1.2.2.

Alle Regeln werden mit Gewichten versehen, die eine Feinsteuerung des Systems ermöglichen. Eine detaillierte Beschreibung der Konstruktion des Regelsystems wird in 6.1.2 gegeben. Das vollständige Regelsystem ist in ANHANG A aufgelistet.

Inferenzmechanismus

Die Regeln werden einzeln ausgewertet. Als Implikationsmethode wird die von Mamdani

verwendet - siehe 4.1. So liefert jede Regel als Ausgabe eine Zugehörigkeitsfunktion eines linguistischen Wertes der Ausgabe-Variablen, die auf Höhe ihres Erfüllungsgrades abgeschnitten ist. So entsteht pro Regel eine trapezförmige Fläche. Diese Flächen können sich überlappen. Seien out_s , out_m und out_b die linguistischen Werte des Kantenwertes. Abbildung 4.6 verdeutlicht die entstehenden Flächen für vier Regeln mit Erfüllungsgrad μ_1 , μ_2 , μ_3 und μ_4 .

Die Ergebnisse der Regeln werden einzeln an die Defuzzifizierung weitergereicht.

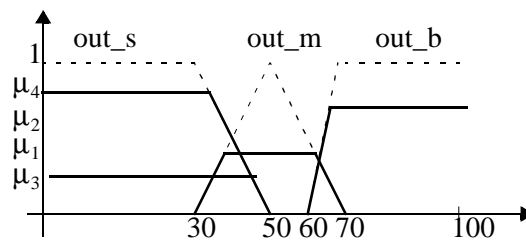


Abb. 4.6 Gekappte Zugehörigkeitsfunktionen

Defuzzifizierungsmethode

Die Gewichtung der einzelnen Regeln soll dazu genutzt werden, das System zu justieren. In einem erweiterten Lösungsansatz werden die Gewichte auch dazu verwendet, das System weitere Eigenschaften lernen zu lassen - siehe 5.4. Aus diesem Grund wird die Defuzzifizierung so gewählt, daß der Einfluß der einzelnen Regeln so genau wie möglich berücksichtigt wird. Daher müssen bei der Berechnung des Endergebnisses auch die Anzahl der sich überlappenden Flächen berücksichtigt werden. In der Literatur wird diese Eigenschaft mit „weight counting“ bezeichnet [16]. Die modifizierte Schwerpunktmethod hat diese Eigenschaft. Alle Flächen, auch die überlappenden Flächen, gehen in die Berechnung ein. Flächen, die mehrfach vorkommen, werden auch mehrfach berücksichtigt. Die Defuzzifizierung mittels modifizierter Schwerpunktmethod berechnet sich wie folgt:

Seien N , n_i , n Elemente der natürlichen Zahlen. Sei $\langle X, LX, PX, M_X \rangle$ eine linguistische Ausgabe-Variable mit N linguistischen Werten $LX = \{L_1, \dots, L_N\}$. Die Anzahl der Regeln mit Konklusion L_i sei n_i . Ein Eingabewert liefert also n_i Erfüllungsgrade $\mu_{i,1}, \mu_{i,2}, \dots, \mu_{i,n_i}$ für die linguistischen Werte L_i .

$f_{i,k}$ sei die Zugehörigkeitsfunktion der linguistischen Werte L_i abgeschnitten auf der Höhe $\mu_{i,k}$. Der harte Ausgabe-Wert out_x ist definiert wie:

$$out_x = \frac{\sum_{i=1}^N \sum_{k=1}^{n_i} \int x f_{i,k}(x) dx}{\sum_{i=1}^N \sum_{k=1}^{n_i} \int f_{i,k}(x) dx} \quad (4.6)$$

Je öfter ein Wert einer linguistischen Ausgabe-Variablen als Konklusion vorkommt, um so größer ist der Einfluß dieses Ausgabewertes auf das Ergebnis. Dieses Verhalten muß bei der Transformation der sprachlichen Regeln in die Fuzzy-Regeln berücksichtigt werden.

Um einen gleichmäßigen Einfluß jedes Beurteilungskriteriums, das einen oder mehrere Para-

meter umfassen kann, auf das Ergebnis zu gewährleisten, muß die Anzahl der Regelkonklusionen, die dieses Kriterium verursacht, für alle Kriterien dieselbe sein. Das kann erreicht werden, indem die Parameter der Kanten in inhaltlich zusammenhängende Gruppen eingeteilt werden, die einem Kriterium entsprechen. Pro Parametergruppe werden die Regeln so formuliert, daß eine Regel pro linguistischem Ausgabewert erzeugt wird. Dazu werden die Aussagen entsprechend logisch verknüpft und in einer Prämisse zusammengefaßt. Auf diese Weise hat jede Gruppe und jeder linguistische Ausgabewert den gleichen Einfluß auf das Ergebnis. Ist bei der Wissenserfassung festgelegt worden, daß eine Regel eine höhere Relevanz hat, so wird das durch eine entsprechende Gewichtung der Regel berücksichtigt. In 6.1.2.2 kann ein Beispiel für dieses Vorgehen nachgelesen werden.

Dieses Kapitel beginnt zur Klärung der verwendeten Terminologie mit einer kurzen Einführung in die künstlichen neuronalen Netze. Im folgenden Kapitel 5.2 werden zwei mögliche Methoden zur Transformation von Fuzzy-Regelbasen in neuronale Netze vorgestellt, die Transformation in ein Radiale-Basisfunktionen-Netz und die Transformation in ein SRCM-Netze, also ein Netz, das semilineare Rate Coded Modell-Neuronen verwendet. Diese beiden Methoden werden in 5.4 gegeneinander abgewogen und die für das Fuzzy-Neuro-Verfahren verwendete Methode, Transformation mit SRCM-Netzen, wird erläutert.

5.1 Ursprung künstlicher neuronaler Netze

Biologische Informationsverarbeitungssysteme bestehen aus einem Netzwerk von Nervenzellen, Neuronen genannt. Das menschliche Gehirn zum Beispiel besteht aus einem Netzwerk von mehr als 10^{10} Nervenzellen, die jeweils mit mindestens 10^4 anderen verbunden sind [34].

Ein Neuron ist eine einfache Verarbeitungseinheit. Es besteht aus einem Zellkörper und zahlreichen Fortsetzungen. Diese Fortsetzungen dienen der Weiterleitung und dem Empfang von Signalen. Jedes Neuron hat genau eine Fortsetzung, die der Übermittlung von Signalen an nachgeschaltete Neuronen dient. Diese Fortsetzung, Axon genannt, verzweigt sich mannigfaltig, um die Signale an zahlreiche andere Neuronen weiterzuleiten. Außerdem hat jedes Neuron zahlreiche Fortsetzungen, die ausschließlich dem Empfang von Signalen dienen. Diese werden Dendriten genannt.

Die Verknüpfung und Arbeitsweise dieser Neuronen diente als Anregung für die Definition und Konstruktion künstlicher neuronaler Netze, im weiteren einfach neuronale Netze genannt. Ein neuronales Netz besteht aus einer Reihe einfacher Verarbeitungseinheiten, die gewichtet miteinander vernetzt sind. Diese Verarbeitungseinheiten werden in Anlehnung an das biologische Vorbild Neuronen genannt. Den ersten Vorschlag für ein solches Neuron machten W. McCulloch und W.Pitts 1943 [30]. Ein generisches Neuron besteht aus n Eingängen, einem Integrationsteil, in dem die Eingaben zusammengefaßt werden, einem Ausgabeteil, der die Werte der Ausgabe berechnet, und einer Ausgabe die diese Werte weiterleitet. Ein solches Neuron ist in Abbildung 5.1 dargestellt. g wird Integrationsfunktion und f Aktivierungsfunktion genannt.

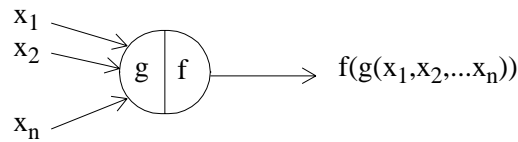


Abb. 5.1 Generisches Neuron

Ein neuronales Netz besteht aus einer Menge vernetzter Neuronen und wird entscheidend durch seine Netzarchitektur bestimmt, die nach Rojas [37] wie folgt definiert ist:

Def 5.1: Eine **Netzarchitektur** ist ein Tupel (E, N, A, T) , wobei E eine Menge von n_e Eingabestellen, N eine Menge von n Neuronen, A eine Menge von n_a Ausgabestellen und T eine Menge gewichteter gerichteter Kanten bezeichnet mit $n_e, n, n_a \in \mathbb{IN}$. Eine gewichtete gerichtete Kante ist ein Tupel (k_i, k_j, w_{ij}) , wobei $k_i \in E \cup N, k_j \in N \cup A$ und das Gewicht w_{ij} eine reelle Zahl ist.

Neuronale Netze unterscheiden sich in den drei folgenden Aspekten:

- Neuronen
- Netzarchitektur
- Lernalgorithmen

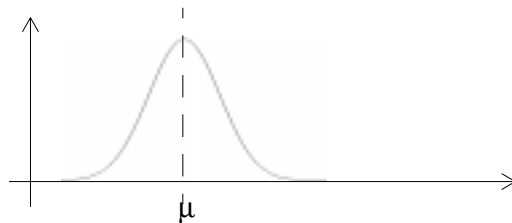
Neuronale Netze bieten eine hochgradig parallele Informationsverarbeitung. Mit Hilfe von neuronalen Netzen können von der Natur in Jahrsmillionen entwickelte Strategien soweit wie möglich und sinnvoll übernommen werden. Die Hauptaufgabe neuronaler Netze ist, auf eine bestimmte Eingabe mit einer adäquaten Ausgabe zu reagieren. Dieses Verhalten wird mit Hilfe einer Trainingsmenge, die aus einer Anzahl von Eingabe-Ausgabe-Paaren besteht, trainiert. Neuronale Netze lernen das Generalisieren an Hand von Beispieldaten. Die Lernfähigkeit ist der Hauptvorteil der neuronalen Netze. Neben der zu trainierenden Aufgabe, haben auch die verwendeten Neuronen und ihre Aktivierungsfunktion einen großen Einfluß auf den Aufwand für ein erfolgreiches Training. Für eine detaillierte Einführung in die Theorie der neuronalen Netze sei auf die Bücher von Rojas [37] und Ritter [36] verwiesen.

5.2 Regelbasierte Fuzzy-Systeme als neuronale Netze

In dieser Arbeit wird die Transformation von Fuzzy-Regelbasen in neuronale Netze verwendet. Es gibt viele Möglichkeiten, diese Transformation durchzuführen. Am häufigsten werden dazu Radiale-Basisfunktionen-Netze (RBF-Netze) eingesetzt. Deshalb wird im folgenden die Transformation von regelbasierten Fuzzy-Systemen in neuronale Netze mit Hilfe von RBF-Netzen, neben der hier bevorzugten Transformation mit Hilfe von SRCM-Netze dargestellt. Die jeweils dazu verwendete Netzarchitektur wird vorgestellt.

5.2.1 RBF-Netze

Eine radialsymmetrische Funktion ist eine Funktion, die in einem Zentrum den höchsten Absolutbetrag annimmt, und deren Betrag des Funktionswertes radialsymmetrisch abnimmt, je weiter die Argumente vom diesem Zentrum entfernt sind. Diese Funktionen werden auch radiale Funktionen genannt. Die Gaußsche Exponentialfunktion ist ein Beispiel für eine radialsymmetrische Funktion.



$$f(x) = \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

$\mu \in \mathbb{R}$ ist das Zentrum und

$\sigma \in \mathbb{R}$ legt die Weite fest

Abb. 5.2 eindimensionale Gaußfunktion

Die Neuronen in RBF-Netzen haben radiale Aktivierungsfunktionen. Ein solches Neuron ist nur aktiv, wenn seine Eingabe relativ nahe am Zentrum der radialen Funktion liegt. So eignen sich RBF-Netze besonders für Klassifikationsaufgaben, bei denen lokal begrenzte Cluster zu identifizieren sind. Ein einzelnes Neuron kann für ein solches Cluster zuständig sein.

Def 5.2: Ein **RBF-Netz** ist ein neuronales Netz mit der Netzarchitektur (E, N, A, T) .

Die Integrationsfunktion und die Aktivierungsfunktion der Neuronen aus E ist die Identität.

Die Integrationsfunktion der Neuronen aus A ist die Summenfunktion, die Aktivierungsfunktion ist die Identität.

Seien $a, e \in \mathbb{N}$. e bezeichne die Dimension des Eingaberaumes und a die Dimension des Ausgaberaumes. Die Neuronen aus N haben e -dimensionale radiale Aktivierungsfunktionen. Das k -te Neuron hat die Aktivierungsfunktion:

$$b_k(x_1, \dots, x_e) = \exp\left(-\sum_{j=1}^e \frac{(x_j - \mu_{jk})^2}{2\sigma_{jk}^2}\right). \quad (5.1)$$

Die Kantenmenge T ist so definiert, daß ein dreischichtiges Netz entsteht. Von jedem Eingabeneuron gibt es eine Kante zu jedem Neuron aus N . Von jedem Neuron aus N gibt es eine gewichtete Kante zu jedem Neuron der Ausgabeschicht. Die Ausgabe des Neurons i aus A ist gegeben durch:

$$y_i(x) = \sum_{k=1}^n w_{ik} b_k(x) \quad (5.2)$$

wobei die $w_{ik} \in \mathbb{R}$ durch die Trainingsaufgabe bestimmt sind.

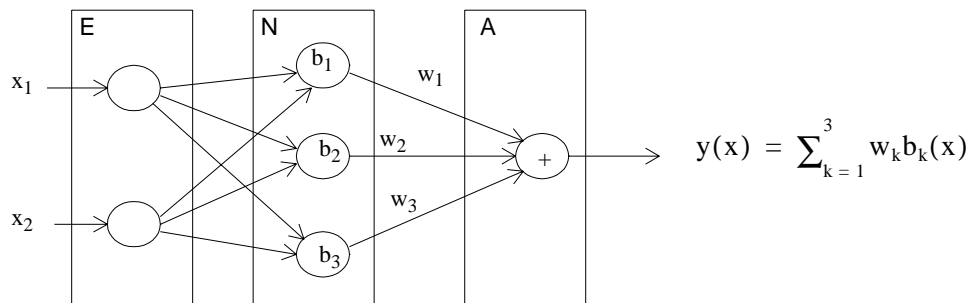


Abb. 5.3 RBF-Netz für zwei-dimensionale Eingabe und eindimensionale Ausgabe

Da für die in dieser Arbeit betrachtete Anwendung nur eindimensionale Ausgaberräume von Interesse sind, werden im folgenden Text ausschließlich RBF-Netze mit eindimensionalem Ausgaberraum dargestellt. So vereinfacht sich die Notation: y_i degeneriert zu y , w_{ik} zu w_k .

Bei sogenannten **einfachen RBF-Netzen** wird pro vorhandenem Trainingsmuster ein inneres Neuron angelegt. Sei $M = \{(m^1, t^1), \dots, (m^n, t^n)\}$, $n \in \mathbb{N}$, die Menge der Trainingsmuster. Die Trainingsmuster seien paarweise verschieden. Dann können die Gewichte durch das Lösen des linearen Gleichungssystems, das durch

$$t^j = \sum_i^n w_i b_i(m^j) \quad j = 1, 2, \dots, n \quad (5.3)$$

gegeben ist, bestimmt werden. Das bedeutet einen Rechenaufwand der Ordnung $O(n^3)$ für die Invertierung der Matrix. Die Anzahl der Neuronen aus N entspricht der Anzahl der Trainingsmuster und ist im allgemeinen sehr hoch. Außerdem kann eine einmal berechnete Approximation bei neuen Daten nicht wiederverwendet werden.

Um die hohe Anzahl von Neuronen aus N zu vermeiden, wird für RBF-Netze eine Auswahl aus den Trainingsmustern verwendet. Das zu lösende lineare Gleichungssystem ist damit überbestimmt. Für die Gewichte werden Näherungslösungen unter Verwendung der Pseudoinversen bestimmt. Genauereres kann in [32] nachgelesen werden.

Die Werte von zwei überlagerten radialen Basisfunktionen werden an Stellen, die entfernt von beiden Zentren liegen, sehr klein, wenn die Zentren weit auseinander liegen. Dieses Absacken kann eventuell nicht gewünscht sein. Um das Interpolationsverhalten zwischen den Basisfunktionen zu verbessern und so weniger Neuronen zur Abdeckung des Eingaberraumes zu benötigen, führten Moody und Darken [33] sogenannte **normalisierte RBF-Netze** ein. Die Ausgabe dieser Netze wird durch die Funktion in Gleichung (5.4) berechnet.

$$y(x) = \frac{\sum_{k=1}^n w_k b_k(x)}{\sum_{k=1}^n b_k(x)} \quad (5.4)$$

Oder aber durch Gleichung (5.5), wenn mit zusätzlichen Gewichten $\kappa_k \in [0, 1]$ gearbeitet wird.

$$y(x) = \frac{\sum_{k=1}^n w_k \kappa_k b_k(x)}{\sum_{k=1}^n \kappa_k b_k(x)} \quad (5.5)$$

Bei RBF-Netzen können verschiedene Parametergruppen unterschieden werden

- Radien und Zentren der radialen Aktivierungsfunktionen der Neuronen
- Gewichte
- Trainingsmuster, die zur Bestimmung des RBF-Netzes verwendet werden

Die Auswahl und die Reihenfolge der Adaption dieser Parametergruppen stellen Freiheitsgrade beim Training von RBF-Netzen dar. Beispielhaft seien die folgenden Strategien genannt:

- Die Radien und Zentren werden fest gewählt und die Gewichte zur Ausgangsschicht berechnet. Arbeitet das RBF-Netz nicht wie gewünscht, so werden andere Muster ausgewählt und der Vorgang wird wiederholt. Das ist die Variante, die am wenigsten aufwendig ist. Es findet kein wirkliches Lernen statt.
- Statt die Zentren und Radien fest zu wählen, werden diese durch selbstorganisierendes Lernen bestimmt. Hier bieten sich Clusterbestimmungsverfahren an. Die Gewichte werden durch überwachtes Lernen bestimmt.
- Die dritte Möglichkeit besteht im überwachten Lernen aller Parameter. Der Trainingsaufwand für dieses Verfahren ist hoch. Jedes RBF-Neuron hat $e^2 + e$ freie Parameter. e bezeichne die Dimension des Eingaberaumes. Zum Vergleich: ein Neuron im Multilayer-Perzeptron hat nur e freie Parameter.

In [32] wird darauf verwiesen, daß es Beispiele für RBF-Netze gibt, bei denen alle Parameter trainiert wurden. Diese RBF-Netze generalisieren dann besser als ein vergleichbares Multilayer-Perzeptron, das mit Backpropagation trainiert wurde. Allgemein läßt sich jedoch nicht entscheiden, ob eine lokale Klassifikationsstrategie, also der Einsatz von RBF-Netzen, oder eine globale Klassifikationsstrategie besser ist. Es hängt von der jeweiligen Lernaufgabe ab.

RBF-Netze eignen sich dazu, Fuzzy-Regelbasen als neuronale Netze abzubilden. Dazu seien Fuzzy-Regelbasen betrachtet, die den folgenden Randbedingungen genügen:

- Die Zugehörigkeitsfunktionen der linguistischen Variablen sind radiale Basisfunktionen.
- Die Konklusion jeder Regel läßt sich als Funktion der Eingabe darstellen; zum Beispiel:
if x_1 is A AND x_2 is B THEN $y=f(x_1, x_2)$
Diese Art des regelbasierten Fuzzy-Systems wird von Takagi und Sugeno in [47] vorgestellt. Eine einfache Variante dieses Systems ist durch die Wahl der konstanten Funktion für f gegeben. f wird als Singleton repräsentiert.
- Der UND-Operator wird durch das Produkt der Zugehörigkeitsfunktionen abgebildet - siehe 4.1.

Den schematischen Aufbau eines regelbasierten Fuzzy-Systems stellt Abbildung 4.1 dar. Die dort gezeigten Bestandteile Fuzzifizierung, Inferenz-Maschine und Defuzzifizierung finden sich in der Netzarchitektur wieder:

Fuzzifizierung

Seien x^1 bis x^e die linguistischen Eingabe-Variablen des zu betrachtenden Systems. Jede Eingabe-Variable x^k habe $I \in \mathbb{N}$ linguistische Werte x^k_1, x^k_2 bis x^k_I . Die Zugehörigkeitsfunktionen für x^k_j ist gegeben durch:

$$b_j^k(x^k) = \exp\left(-\frac{(x^k - \mu_{jk})^2}{2\sigma_{jk}^2}\right) \quad (5.6)$$

Die Fuzzifizierung der Variablen x^k kann also durch ein Neuron mit radialer Aktivierungsfunktion berechnet werden.

Inferenz-Maschine

Sei out die linguistische Ausgabe-Variable mit den linguistischen Werten out_1, out_2 und out_3 , deren Zugehörigkeitsfunktionen als Singletons mit den Werten out_1, out_2 und out_3 dargestellt werden. Beispielphaft sei hier die folgende Regel betrachtet:

IF $x^k = x^k_1$ AND $x^j = x^j_3$ THEN $y = out_2$

Der Erfüllungsgrad r dieser Regel berechnet sich mit diesen Voraussetzungen wie folgt:

$$\begin{aligned} r(x^k, x^j) &= b_1^k(x^k) \cdot b_3^j(x^j) \quad (5.7) \\ &= \exp\left(-\frac{(x^k - \mu_{1k})^2}{2\sigma_{1k}^2}\right) \exp\left(-\frac{(x^j - \mu_{3j})^2}{2\sigma_{3j}^2}\right) \\ &= \exp\left(-\frac{1}{2}\left(\frac{(x^k - \mu_{1k})^2}{\sigma_{1k}^2} + \frac{(x^j - \mu_{3j})^2}{\sigma_{3j}^2}\right)\right) \\ &= b_r(x^k, x^j) \end{aligned}$$

Somit entspricht der Erfüllungsgrad dieser Regel einer radialen Funktion.

Defuzzifizierung

Das Fuzzy-Regelsystem bestehe aus R Regeln, deren Erfüllungsgrad jeweils durch $b_r, r = 1, \dots, R$ gegeben sei. Da die Zugehörigkeitsfunktionen der linguistischen Ausgabewerte Singletons sind, berechnet sich der harte Ausgabewert mit der Defuzzifizierungsmethode von Takagi-Sugeno durch

$$y = \frac{\sum_{r=1}^R y_r b_r}{\sum_{r=1}^R b_r}, \quad (5.8)$$

wobei y_r der Wert der Konklusion der Regel r ist. Er kann out_1, out_2 oder out_3 sein. Das ent-

spricht dem Netzausgang eines normalisierten RBF-Netzes, wie aus der Gleichung (5.4) hervorgeht. Wenn die Regeln jeweils mit $w_r \in [0, 1]$ gewichtet werden, so entspricht der harte Ausgangswert des Systems dem Netzausgang eines gewichteten normalisierten RBF-Netzes. Das dazugehörige RBF-Netz ist in Abbildung 5.4 dargestellt.

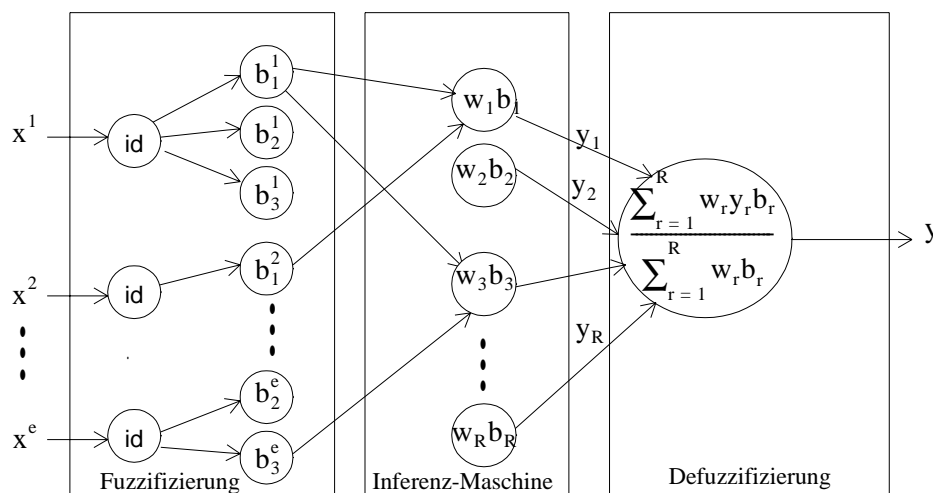


Abb. 5.4 Netzarchitektur eines RBF-Netz zur Darstellung einer Fuzzy-Regelbasis

Handelt es sich um ein Regelsystem mit Regeln, deren Prämissen aus je gleich vielen linguistischen Eingabe-Variablen bestehen, so hat jedes der Neuronen aus der Inferenz-Maschine gleichdimensionale Aktivierungsfunktionen. In der Praxis kommen jedoch auch Regelsysteme vor, bei denen die Prämissen der Regeln unterschiedliche Anzahlen von Eingabe-Variablen logisch kombinieren. So entstehen in der Inferenz-Maschine Neuronen mit radialen Aktivierungsfunktionen unterschiedlicher Dimension.

Sind die Aktivierungsfunktionen der Neuronen der Inferenz-Maschine das Produkt aus den Aktivierungsfunktionen ihrer präsynaptischen Neuronen, so ist eine Rücktransformation des RBF-Netzes in eine regelbasiertes Fuzzy-System möglich.

5.2.2 SRCM-Netze

Die Aktivierungsfunktion der RBF-Neuronen ist im Vergleich zu der Aktivierungsfunktion der ersten Neuronenmodelle, die von W. McCulloch und W.Pitts 1943 vorgeschlagen wurden, kompliziert und die Lernalgorithmen entsprechend aufwendig. Dieses Kapitel stellt ein Verfahren vor, das als Neuronentyp im Prinzip ein einfaches Perzeptron verwendet. Statt der Treppenfunktion sind jedoch semilineare Aktivierungsfunktionen zugelassen. Ein Perzeptron mit semilinearer Aktivierungsfunktion sei als semilineares Rate Coded Modell-Neuron bezeichnet, kurz SRCM-Neuron. Neuronale Netze, die ausschließlich SRCM-Neuronen verwenden, seien mit SRCM-Netze bezeichnet.

Das in diesem Kapitel vorgestellte Verfahren zur reversiblen Transformation von Fuzzy-Regelbasen in neuronale Netze entspricht zum großen Teil dem 1994 von Angela Bücherl vorgestellten Verfahren [4]. Die Details können dort nachgelesen werden. Der Vollständigkeit halber wird hier die grundsätzliche Architektur erläutert. Um das Nachlesen zu vereinfachen, lehnt sich die Notation im wesentlichen an die in [4] an. Abbildungen und Beispiele sind teilweise [4] entnommen. Zur Abbildung des regelbasierten Fuzzy-Systems werden

mehrschichtige, strukturierte und inhomogene SRCM-Netze verwendet.

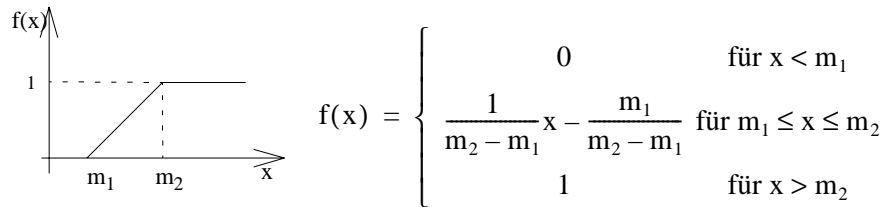


Abb. 5.5 Aktivierungsfunktion eines SRCM-Neuron

Betrachtet seien regelbasierte Fuzzy-Systeme, die den folgenden Voraussetzungen genügen:

- die Zugehörigkeitsfunktionen der linguistischen Werte sind dreiecks- oder trapezförmig
- die verwendeten fuzzy-logischen Operatoren sind assoziativ
- die Regeln enthalten keine negierten logischen Ausdrücke, da die Semantik der Fuzzy-Negation nicht eindeutig geklärt ist

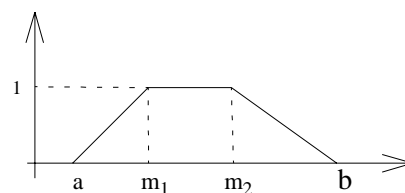
Die Regeln können beliebig mit UND- und ODER-Verknüpfungen formuliert werden. Diese fuzzy-logischen Ausdrücke werden, wie schon vorne erwähnt, direkt durch neuronale Verschaltungen von UND- und ODER-Neuronen abgebildet, dadurch entfällt das Umformen der Regeln in eine bestimmte Form. Fehler durch logische Umformungen, die bei den meisten Fuzzy-Operatoren unvermeidlich sind, wie in Beispiel 3.4.2 in [4] gezeigt wurde, können so nicht vorkommen. Das Verbot der Negation stellt im allgemeinen keine Einschränkung dar, da Fuzzy-Regeln im allgemeinen positiv formuliert werden, das heißt, die verwendeten linguistischen Variablen werden nicht negiert verwendet, wie schon in [4] dargestellt.

Die schematische Gliederung des regelbasierten Fuzzy-Systems in Fuzzifizierung, Inferenz-Maschine und Defuzzifizierung findet sich in der Netzarchitektur wieder:

Fuzzifizierung

Seien x^1 bis x^c die linguistischen Eingabe-Variablen des zu betrachtenden Systems. Jede Eingabe-Variable x^k habe $I \in \mathbb{N}$ linguistische Werte x_1^k, x_2^k bis x_I^k . Im folgenden sei die Zugehörigkeitsfunktion des linguistischen Wertes x_j^k beispielhaft betrachtet. x sei der zu x^k gehörige Wert der physikalischen Variablen

$$x_j^k(x) = \begin{cases} \frac{x}{m_1 - a} - \frac{a}{m_1 - a} & \text{für } a \leq x \leq m_1 \\ 1 & \text{für } m_1 < x < m_2 \\ -\frac{x}{b - m_2} + \frac{b}{b - m_2} & \text{für } m_2 \leq x \leq b \\ 0 & \text{sonst} \end{cases}$$



mit $a, b, m_1, m_2 \in \mathbb{R}$.

Sei $f: \mathbb{R} \rightarrow [0, 1]$ mit

$$f(x) = \begin{cases} 0 & \text{für } x < 0 \\ x & \text{für } 0 \leq x \leq 1 \\ 1 & \text{für } x > 1 \end{cases} \quad (5.9)$$

gegeben, dann gilt

$$x_j^k(x) = f(f(w_3x + w_1) - f(w_4x + w_2)) \quad (5.10)$$

mit $w_1 = -\frac{a}{m_1 - a}$, $w_2 = -\frac{m_2}{b - m_2}$, $w_3 = \frac{1}{m_1 - a}$ und $w_4 = \frac{1}{b - m_2}$.

Damit ergibt sich die in Abbildung 5.6 dargestellte Neuronenverschaltung. Mit pv_in, compute_lv und lv_in sind die Layer des neuronalen Netzes bezeichnet. Diese Layer bezeichnen weitere Unterteilungen der Neuronenschichten E, N, A aus Def. 5.1.

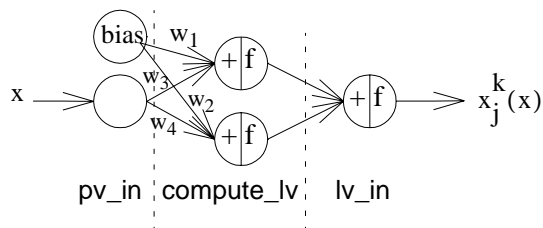


Abb. 5.6 Neuronenverschaltung für x_j^k

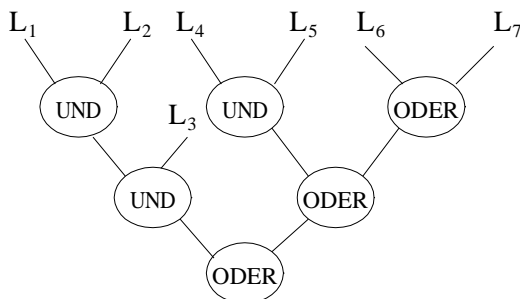
Inferenz-Maschine

Da nur assoziative, logische Operatoren verwendet werden, können logische Ausdrücke als binärer Baum dargestellt werden. Das folgende Beispiel verdeutlicht die Aussage:

$L_i, i=1, \dots, 7$ seien linguistische Werte. Der logische Ausdruck

$$(L_1 \wedge L_2 \wedge L_3) \vee ((L_4 \wedge L_5) \vee L_6 \vee L_7) = ((L_1 \wedge L_2) \wedge L_3) \vee ((L_4 \wedge L_5) \vee (L_6 \vee L_7)) \quad (5.11)$$

entspricht dem folgenden binären Baum



Als logische Operatoren werden die in Gleichung (4.1) und Gleichung (4.2) in 4.2 definierten verwendet. Sie erfüllen an den Randpunkten 0 und 1 diese Voraussetzung. Da diese Funktionen stetig sind, gilt das „fuzzy-mäßig“ auch im Inneren des Intervalls. Sei nun

$$f_{\text{und}}(x) = \begin{cases} 0 & \text{für } x < 1 \\ x - 1 & \text{für } 1 \leq x \leq 2 \\ 1 & \text{sonst} \end{cases} \quad (5.12)$$

und

$$f_{\text{oder}}(x) = \begin{cases} 0 & \text{für } x < 0 \\ \frac{1}{2}x & \text{für } 0 \leq x \leq 2 \\ 1 & \text{sonst} \end{cases} \quad (5.13)$$

Seien μ_A und μ_B Zugehörigkeitsfunktionen von linguistischen Variablen, dann gilt:

$$\mu_{A \cap B}(x, y) = f_{\text{und}}(\mu_A(x) + \mu_B(y)) \quad (5.14)$$

und

$$\mu_{A \cup B}(x, y) = f_{\text{oder}}(\mu_A(x) + \mu_B(y)) \quad (5.15)$$

Somit kann die Regel, die Gleichung (5.11) als Prämisse hat,

$$w * \text{IF } ((L_1 \wedge L_2) \wedge L_3) \vee ((L_4 \wedge L_5) \vee (L_6 \vee L_7)) \text{ THEN output} = \text{out}_k \quad (5.16)$$

mit $w \in [0, 1]$, durch folgende neuronale Verschaltung dargestellt werden:

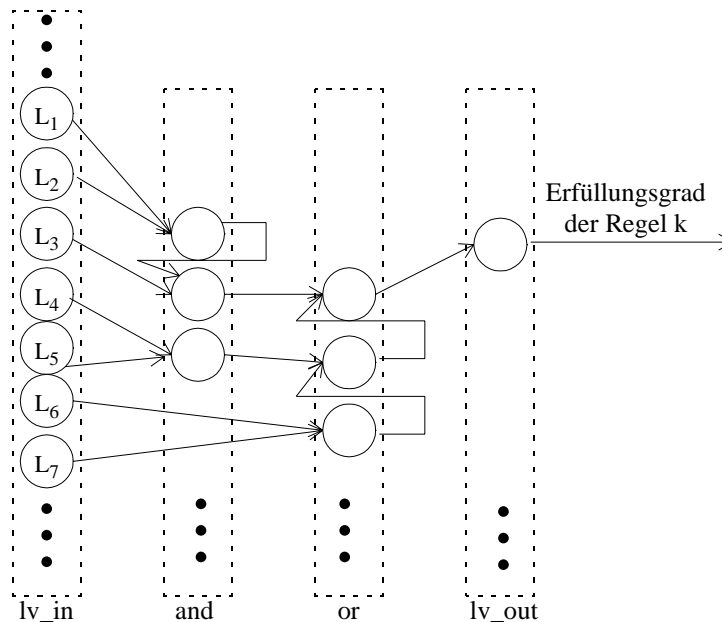


Abb. 5.7 Neuronenverschaltung für Prämissen

Mit den bisherigen Berechnungen ergibt sich die in Abbildung 5.8 dargestellte Netzarchitektur. Der Übersichtlichkeit halber sind nicht alle Verbindungen dargestellt. Insbesondere ist pro Eingabe-Variable nur ein linguistischer Wert als Verschaltung gezeigt. Die Neuronenschaltung für die Defuzzifizierung ist hier nicht dargestellt, da zunächst alle Methoden zugelassen sind. Mit w_1, w_2, \dots, w_R sind die Gewichte der Regeln bezeichnet.

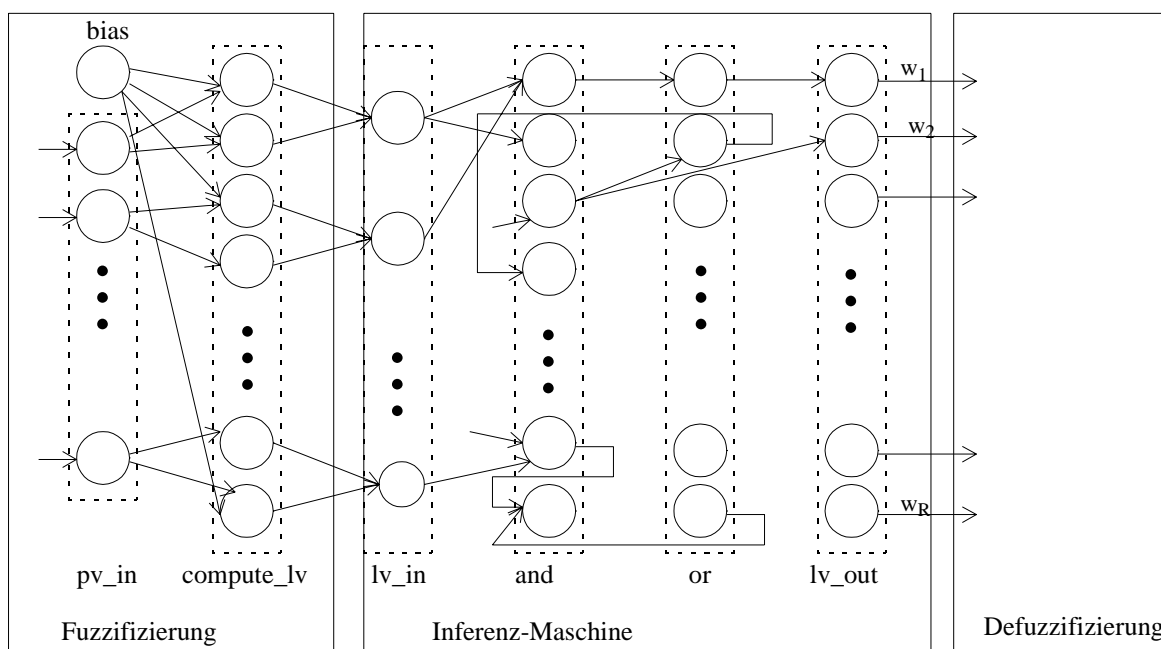


Abb. 5.8 Netzarchitektur für das regelbasierte Fuzzy-System

Defuzzifizierung

Diese Abbildung der Fuzzy-Regelbasis kann als Grundlage für alle Inferenz- und Defuzzifizierungsmethoden genutzt werden. Um die in 4.2 beschriebenen Methoden als neuronales Netz abzubilden gibt es mehrere Möglichkeiten, die in [4] genau beschrieben sind.

Um die beiden Methoden zur reversiblen Transformation einer Fuzzy-Regelbasis in ein neuronales Netz, die hier vorgestellt werden, miteinander vergleichen zu können, wird die Inferenz- und Defuzzifizierungsmethode in dieser Netzstruktur abgebildet, die auch in dem RBF-Netz verwendet wird. Statt die linguistischen Werte der Ausgabe-Variablen mit trapezförmigen Zugehörigkeitsfunktionen darzustellen, werden hier für die linguistischen Werte Singleton-Funktionen gewählt.

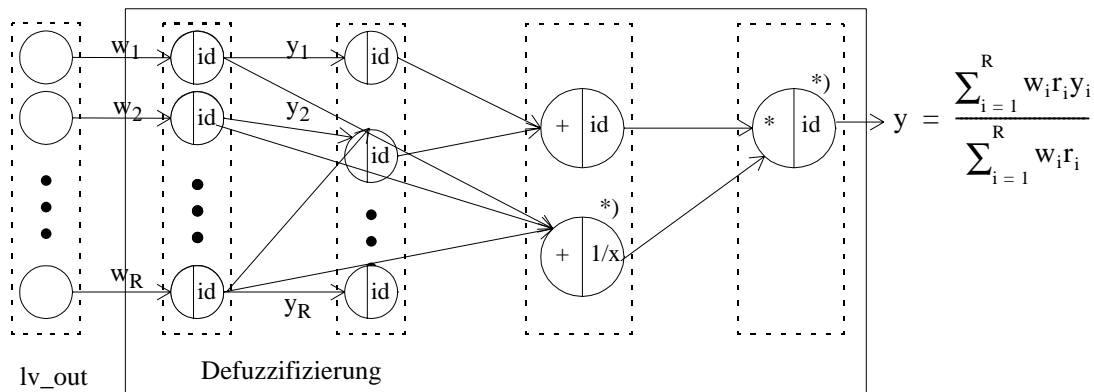
Gegeben sei ein Fuzzy-Regelsystem mit R Regeln. Der Erfüllungsgrad der i -ten Regel sei r_i und der Wert ihrer Konklusion sei y_i . Der harte Ausgabewert des Systems mit der Defuzzifizierungsmethode nach Tagaki-Sugeno berechnet sich durch

$$y = \frac{\sum_{i=1}^R r_i y_i}{\sum_{i=1}^R r_i} \quad (5.17)$$

Sind die Regeln jeweils mit w_i gewichtet, dann gilt:

$$y = \frac{\sum_{i=1}^R w_i r_i y_i}{\sum_{i=1}^R w_i r_i} \quad (5.18)$$

Die Defuzzifizierung kann wie folgt als neuronales Netz dargestellt werden:



Nicht beschriftete Kanten haben als Gewicht eine 1

*) : Diese Neuronen sind keine SRCM-Neuronen.
Die zu adaptierenden Gewichte liegen jedoch im reinen SRCM-Netz

Abb. 5.9 Neuronenverschaltung für Defuzzifizierung nach Takagi-Sugeno

Jedes Regelsystem, das die oben gemachten Einschränkungen erfüllt, läßt sich in der oben beschriebenen Weise als SRCM-Netz abbilden. Eine eindeutige Rücktransformation ist immer möglich.

5.3 Adaption eines regelbasierten Fuzzy-Systems

Für die so entstandenen neuronalen Netze können entsprechende Lernverfahren angewandt werden. So kann das in den Daten enthaltene Wissen in das regelbasierte Fuzzy-System einfließen. Für das Training wird überwachtes Lernen eingesetzt.

Die Adaption eines regelbasierten Fuzzy-Systems an seine Aufgabe kann an folgenden Komponenten vorgenommen werden:

- fuzzy-logische Operatoren
Sie können ausgetauscht werden.
- linguistische Variablen für Eingabe und Ausgabe
Die Zugehörigkeitsfunktionen können verändert werden.
Neue linguistische Variablen können ergänzt werden.

- Regelsystem
Die Wichtigkeit der bestehenden Regeln kann mittels Veränderung der Regelgewichte angepaßt werden.
- Inferenzmechanismus
Er kann ausgetauscht werden.
- Defuzzifizierungsmethode
Sie kann ausgetauscht werden.

Die Veränderung der logischen Operatoren, des Inferenzmechanismus und der Defuzzifizierungsmethode kann ohne die Transformation in ein neuronales Netz geschehen. Zur Adaption der anderen genannten Punkte wird jedoch mit Hilfe der Transformation die Lernfähigkeit der neuronalen Netze genutzt. Das neuronale Netz darf während des Trainings nur so geändert werden, daß es weiterhin einem regelbasierten Fuzzy-System entspricht.

Adaptionspotential der RBF-Netze

Die Darstellung der Fuzzy-Regelbasis als RBF-Netz bietet die Möglichkeit mittels Lernverfahren zu verändern:

- die Form und die Zentren der Zugehörigkeitsfunktionen der linguistischen Eingabe-Variablen,
- die Singleton-Werte der Ausgangsvariablen,
- die Wichtigkeit der Regeln über die Regelgewichte.

Wenn nach der Lernphase das RBF-Netz weiterhin einer Fuzzy-Regelbasis entsprechen soll, so müssen sich nach dem Training weiterhin die Zugehörigkeitsfunktionen überlappen und die Singleton-Werte müssen als getrennte Werte erhalten bleiben. Außerdem müssen die Aktivierungsfunktionen der Neuronen der Inferenz-Maschine das Produkt aus den Aktivierungsfunktionen ihrer präsynaptischen Neuronen sein.

Eine Beschreibung dieser Adaptionstechnik ist in [42] zu finden. Hier wird auch eine interessante Anwendung, eine Prognose des Deutschen Aktienindex, für diese Transformation von Fuzzy-Regelbasen in RBF-Netze vorgestellt. Mit einer leicht abgewandelten Netzarchitektur werden gute Ergebnisse erzielt. Die Anwendung paßt sehr gut zu den Gegebenheiten dieser Transformation. Die Regeln betreffen immer nur je zwei Parameter und als logische Verknüpfung wird ausschließlich der UND-Operator verwendet. Dadurch ist die Struktur des Netzes sehr übersichtlich.

Die Adaption wird prinzipiell mit Gradientenverfahren vorgenommen. Der Trainingsaufwand und der Rechenaufwand ist bei RBF-Netzen wie schon in 5.2.1 erwähnt höher als bei SRCM-Netzen.

Adaptionspotential der SRCM-Netze

Die Abbildung des regelbasierten Fuzzy-Systems als SRCM-Netz stellt alle Parameter als Gewichte der Verbindungen des neuronalen Netzes dar. So können grundsätzlich alle Parameter adaptiert werden, insbesondere die folgenden:

- die Gewichte der Regeln
- die Zugehörigkeitsfunktionen der linguistischen Eingabe- und Ausgabe-Variablen

Eine genaue Darstellung der möglichen Adaptionen wird in 5.4.3 „Lernalgorithmen für das

neuronale Netz" gegeben.

Damit das Netz nach dem Training einem sinnvollen regelbasierten Fuzzy-System entspricht, müssen sich die Zugehörigkeitsfunktionen überlappen und den gesamten Wertebereich abdecken. Die Singleton-Werte der linguistischen Ausgabe-Variablen müssen weiterhin unterschiedliche Werte haben. Änderungen, die diesen Bedingungen widersprechen, sind unzulässig.

Die Architektur des SRCM-Netzes ermöglicht das Trainieren mittels einfachem Delta-Verfahren [51] oder Gradientenverfahren. Der Rechen- und Trainingsaufwand ist wegen der einfachen Aktivierungsfunktion der Neuronen im Vergleich zum RBF-Netz wesentlich geringer. Da die Semantik jeder Schicht und ihre Strukturierung genau bekannt sind, können gezielte Adaptionen durchgeführt werden.

Das Aufnehmen neuer linguistischer Variablen bedeutet eine Änderung der Netzarchitektur. Das ist zum Beispiel sinnvoll, wenn die Analyse der letzten Lernschritte eine zyklische Verschiebung der linguistischen Variable aufzeigt.

5.4 Fuzzy-Neuro-Verfahren zur Routenauswahl

5.4.1 Entscheidung für SRCM-Netze

Das regelbasierte Fuzzy-System, das transformiert werden soll, wird zur Routenbewertung in Transportnetzen eingesetzt. Die Regeln werden wie in 4.2 beschrieben festgelegt. Bei der Formulierung dieser Regeln werden sowohl UND- als auch ODER-Verknüpfungen verwendet. Diese Regeln sollen so übernommen werden, wie sie vorgeschlagen werden. Sie sollen nicht umformuliert werden, da eine Umformulierung von fuzzy-logischen Ausdrücken den Wahrheitswert dieses Ausdrucks verändern kann:

Ist die Fuzzy-Mengenoperation für die Vereinigung gegeben durch

$$\mu_{A \cup B} = \max\left(\frac{1}{2}(\mu_A(x) + \mu_B(x)), 0\right),$$

so erfüllt die entsprechende fuzzy-logische ODER-Verknüpfung die Regel „tertium non datur“ nicht.

Wird ODER durch die Addition und UND durch die Multiplikation der Zugehörigkeitsfunktionen abgebildet, so gilt die Distributivität nicht.

Da diese Gesetze aber unter anderem bei einer Umformung verwendet würden, würden die Wahrheitswerte der Ausdrücke verändert.

Schon diese Bedingung spricht gegen den Einsatz der Transformation in ein RBF-Netz. ODER-Verknüpfungen sind dort nicht vorgesehen. Außerdem spiegelt sich die ungeordnete Struktur der Regelbasis in der Struktur des RBF-Netzes wider, wie in 5.2.1 dargestellt.

Beide Netze bieten Lernmöglichkeiten für

- die Zugehörigkeitsfunktionen der linguistischen Eingabe- und Ausgabe-Variablen
- und für die Gewichte der Regeln.

Der Trainingsaufwand ist für das RBF-Netz höher als der für das SRCM-Netz.

Diese Arbeit verwendet deshalb die Transformation des regelbasierten Fuzzy-Systems in ein SRCM-Netz. Das hat folgende Vorteile gegeben:

- Einfache Aktivierungsfunktionen der Neuronen führen zu einem geringen Rechenaufwand im Vergleich mit den RBF-Neuronen.
- Fuzzy-logische Ausdrücke für die Regeln können sowohl mit UND- als auch mit ODER-Operanden verknüpft werden. So sind die natürlichsprachlichen Regeln der Experten ohne Einschränkungen formulierbar.
- Es stehen zwei verschiedene Defuzzifizierungsverfahren zur Verfügung, die beide den in 4.2 gewünschten Anforderungen entsprechen.
- Es können einfache Lernverfahren angewandt werden.

5.4.2 Training

Die Transformation des regelbasierten Fuzzy-Systems in ein neuronales Netz hat das Ziel, die Fuzzy-Bewertung der Kanten an die durch die Daten gegebenen Kriterien anzupassen.

Gleich welche Komponente des Fuzzy-Regelsystems durch das Training adaptiert werden soll, für ein überwachtes Lernen des SRCM-Netzes werden Soll-Werte für die Kantenwerte benötigt. Soll-Kantenwerte sind nicht vorhanden. Zum Training stehen sogenannte Routenlisten zur Verfügung. Diese Routenlisten enthalten einzelne Routen, die zur Bewertung der Kanten ein oder mehrere Kriterien heranziehen, die in dieser Ausprägung noch gar nicht oder noch nicht exakt genug in den bereits erfaßten Fuzzy-Regeln enthalten sind. Ziel des Trainings ist es, die Fuzzy-Regeln so zu ändern, daß sie diese Bewertung zusätzlich beachten. Die in den Routenlisten enthaltenen Routen werden Soll-Routen genannt und liegen in Form von Kantenfolgen vor.

Um das neuronale Netz in seiner Arbeitsweise zu überprüfen, müssen aus den Soll-Routen Soll-Kantenwerte ermittelt werden. Aus den globalen Aussagen über Routen muß auf die lokalen Aussagen über Kanten geschlossen werden. Der Graph des Layer-Netzwerks wird mit Hilfe des neuronalen Netzes fuzzy-bewertet, wie in 3.5 beschrieben. Jeder Soll-Route liegt ein Routenwunsch, Quelle und Ziel der Route zugrunde. Für diesen Routenwunsch wird auf dem fuzzy-bewerteten Graphen mit Fuzzy-Routing die Route bestimmt, die sogenannte Ist-Route. Ist- und Soll-Route werden miteinander verglichen. Durch diesen Vergleich werden auf die folgende Weise Soll-Kantenwerte für das Training des neuronalen Netzes erzeugt:

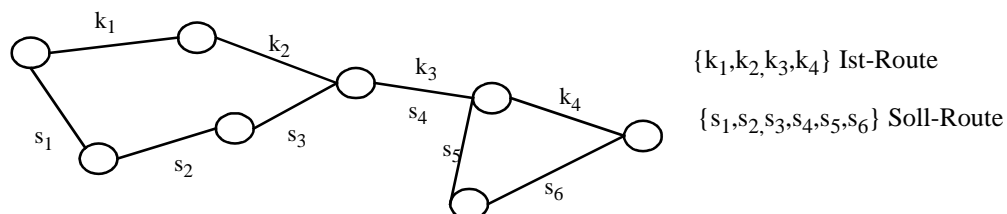


Abb. 5.10 Beispiel zur Erläuterung des Ist- und Soll-Routen Vergleichs

Sei $\{k_1, \dots, k_n\}$ die vom automatischen Routing-Verfahren bestimmte Ist-Route und sei $\{s_1, \dots, s_m\}$ die vorgegebene Soll-Route, wobei $k_i, i=1, \dots, n$ und $s_j, j=1, \dots, m$ Kanten des Layer-Netzwerkes sind. Es gibt drei verschiedene Arten von Kanten bezüglich der Zugehörigkeit

der beiden Routen:

1. Kanten, wie k_1 , k_2 und k_4 aus dem in Abbildung 5.10 zur Veranschaulichung dargestellten Beispiel, die zur Ist-Route gehören, es aber nicht sollten. Deren Kantenwert ist zu niedrig und wird daher um eine feste reelle Zahl δ_1 erhöht.
2. Kanten, wie s_1, s_2, s_3, s_5 und s_6 aus dem Beispiel, die nicht zur Route gehören, aber dazu gehören sollten. Der Kantenwert dieser Route ist zu hoch und wird daher um eine feste reelle Zahl δ_2 vermindert.
3. Kanten, wie k_3 und s_4 aus dem Beispiel, die zu beiden Routen gehören. Deren Kantenwert wird nicht verändert.

Die so korrigierten Kantenwerte werden als Soll-Kantenwerte in Form von Kantenlisten zum Training des neuronalen Netzes bereitgestellt. Sie enthalten je Trainingsmuster die Kantenbezeichnung und den Soll-Kantenwert. Für das in Abbildung 5.10 dargestellte Beispiel entsteht die in Tabelle 5.1 dargestellte Kantenliste.

Kante	Soll-Wert
k_1	Kantenwert von $k_1 + \delta_1$
k_2	Kantenwert von $k_2 + \delta_1$
k_4	Kantenwert von $k_4 + \delta_1$
s_1	Kantenwert von $s_1 - \delta_2$
s_2	Kantenwert von $s_2 - \delta_2$
s_3	Kantenwert von $s_3 - \delta_2$
s_5	Kantenwert von $s_5 - \delta_2$
s_6	Kantenwert von $s_6 - \delta_2$

Tab. 5.1 Soll-Kantenwerte für Beispiel

Das **Training des Systems** wird also in zwei Schritten durchgeführt (Abbildung 5.11):

- Für jede Route aus der Routenliste wird in dem fuzzy-bewerteten Graphen die Ist-Route bestimmt, deren Quelle und Ziel durch die Soll-Routen vorgegeben sind. Durch den Vergleich der Ist- und Soll-Route entstehen veränderte Kantenwerte, die als Soll-Kantenwerte für den Lernalgorithmus des neuronalen Netzes in Form einer Kantenliste bereitgestellt werden.
- Der Lernalgorithmus wird mit diesen Soll-Kantenwerten durchgeführt. Anschließend wird der Graph mit dem angepaßten neuronalen Netzwerk gewichtet. Der Vorgang beginnt von neuem.

Wenn alle Ist-Routen mit den Soll-Routen übereinstimmen, ist das Ziel des Trainings erreicht und es kann abgebrochen werden. Dieses ideale Ergebnis muß jedoch nicht unbedingt eintreffen, daher sind weichere Abbruchkriterien nötig. Ein Abbruch ist zum Beispiel sinnvoll, wenn die Routenliste oft durchlaufen wurde, und sich die neu bestimmten Routen nicht mehr verändert haben. Weitere Details dazu sind in 6.4 beschrieben.

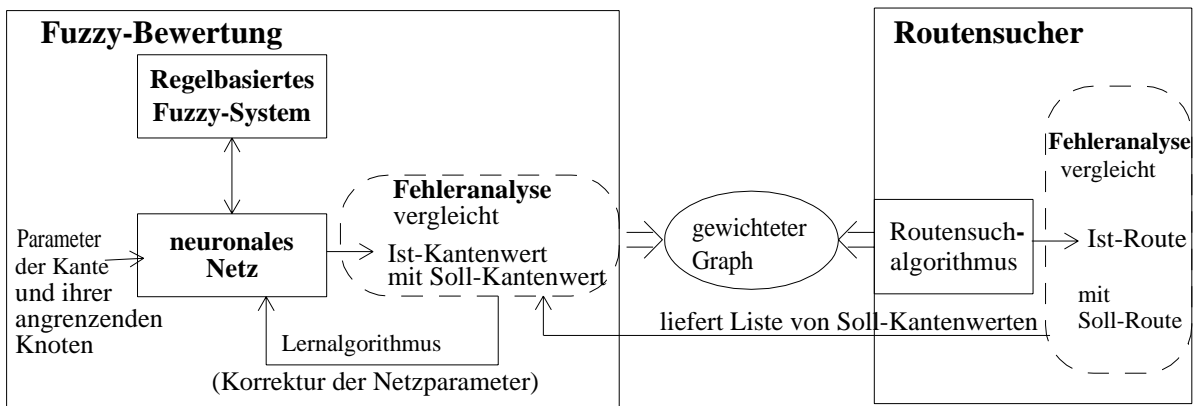


Abb. 5.11 Schematische Darstellung des Trainings

5.4.3 Lernalgorithmen für das neuronale Netz

Wie schon in 5.3 erklärt, sind für das Fuzzy-Neuro-Verfahren verschiedene Lernverfahren möglich. Die verwendete Defuzzifizierungsmethode spielt eine große Rolle, da von ihr die Struktur des neuronalen Netzes abhängt und somit die möglichen Lernverfahren. Im folgenden werden die wichtigsten Möglichkeiten beschrieben.

Adaption der Regelgewichte

Mit diesem Verfahren werden ausschließlich Gewichte von Regeln, die im regelbasierten Fuzzy-System enthalten sind, adaptiert. Die Gewichte der Fuzzy-Regeln sind in der Netzstruktur ebenfalls Gewichte und können daher trainiert werden. Als Grundlage dient die in 5.2.2 beschriebene Methode zur Transformation des regelbasierten Fuzzy-Systems in ein neuronales Netz. Als Defuzzifizierungsmethode wird die in [4] beschriebene modifizierte Schwerpunktmethode verwendet. Das Lernverfahren für die Regelgewichte wird mit Hilfe eines Gradientenabstiegsverfahren bzw. mit Delta-Lernen realisiert [38], [51]. Das heißt, die Regelgewichte werden wie folgt in einem Lernschritt verändert:

$$w_{t+1} = w_t + \alpha \cdot \text{grad}E \text{ für Gradientenverfahren} \tag{5.19}$$

$$w_{t+1} = w_t \pm \alpha \cdot \delta \text{ für Delta-Lernen} \tag{5.20}$$

α bezeichnet die Lernrate, die gewählt werden kann, und E ist die Fehlerfunktion des Systems für diese Trainingsdaten. w_t bezeichnet ein Gewicht im Lernzyklus t . Da diese Gewichte nach der Transformation des neuronalen Netzes in ein regelbasiertes Fuzzy-System den Gewichten der Regeln entsprechen, sind Werte für die Gewichte zwischen 0 und 1 erlaubt. Daher wird, wenn $w_{t+1} < 0$, $w_{t+1} = 0$ gesetzt und wenn $w_{t+1} > 1$, so wird $w_{t+1} = 1$ gesetzt. Das Lernverfahren gilt als erfolgreich beendet, wenn die Fehlerfunktion die vorgegebene Fehlertoleranz unterschreitet.

Tests haben gezeigt, daß dieses System lernt, einen bereits in den Regeln enthaltenden Kantenparameter für die Bewertung so zu beachten, wie das durch die Trainingsmenge vorgegeben ist. So kann diese Methode dazu verwendet werden, ein bereits bestehendes Regelsystem im Sinne der Daten zu justieren.

Um mit Hilfe dieser Methode einen neuen Parameter in das Regelsystem aufzunehmen, müssen zunächst Regeln für diesen Parameter in das bestehende Regelsystem aufgenommen werden. Das ist nur möglich, wenn Grundwissen über den Einfluß des neuen Parameters vorhanden ist und als Regeln formuliert werden kann. Mit den vorhandenen Daten können dann alle Gewichte des Systems entsprechend eingestellt werden.

Die durchgeführten Tests und ihre Ergebnisse sind in 6.4.1 beschrieben.

Adaption der Eingabe-Variablen

Eine weitere Möglichkeit, das regelbasierte Fuzzy-System an die implizit in den Trainingsdaten enthaltenen Regeln anzupassen, besteht darin, die Zugehörigkeitsfunktionen der Eingabe-Variablen zu verschieben. Die Parameter der Zugehörigkeitsfunktionen sind Gewichte des neuronalen Netzes. Diese Methode für die Transformation mit modifizierter Schwerpunktmethod ist in [4] beschrieben. Sie wird hier nur skizziert:

Durch die Erhöhung des Erfüllungsgrades jeder Regel von μ auf $\mu + \delta$ wird der Einfluß jeder Regel auf das Endergebnis y ermittelt. So wird bestimmt, ob der Erfüllungsgrad einer Regel erhöht oder vermindert werden muß, um das Ergebnis in die richtige Richtung zu verschieben. Die linguistischen Variablen und die Werte der Prämissen dieser Regeln werden bestimmt. Die Zugehörigkeitsfunktionen dieser linguistischen Werte werden um einen kleinen Betrag verschoben, der die entsprechende Veränderung der Erfüllungsgrade bewirkt.

Sei zum Beispiel der Ist-Kantenwert größer als der Soll-Kantenwert, so ist y zu vermindern. Angenommen, um dies zu erreichen, muß der Erfüllungsgrad der Regel j erhöht werden. Nun werden die linguistischen Variablen und ihre Werte der Prämisse der Regel j bestimmt. Die Zugehörigkeitsfunktionen dieser linguistischen Werte werden nach links verschoben.

Diese Adaptionmethode wurde zur groben Lenksteuerung für ein einspuriges Fahrzeug erfolgreich eingesetzt [4].

Adaption der Ausgabe-Variablen

Die Werte der linguistischen Ausgabe-Variablen sind für die beiden in dieser Arbeit verwendeten Defuzzifizierungsmethoden als Gewichte der Verbindungen des neuronalen Netzes darzustellen.

In [4] wird eine Netzarchitektur vorgestellt, die unter Verwendung der modifizierten Schwerpunktmethod die Parameter der Zugehörigkeitsfunktionen für die linguistischen Werte der Ausgabe-Variablen als Netzgewichte darstellt. Außerdem wird die zugehörige Adaptionmethode vorgestellt.

In Abbildung 5.9 ist die Netzarchitektur für die Defuzzifizierungsmethode nach Takagi-Sugeno zu sehen, die die Singleton-Werte der linguistischen Ausgabe-Variablen als Gewichte des neuronalen Netzes zeigt. Die dort dargestellte Netzarchitektur kann für die in dieser Arbeit vorgestellten Anwendung noch vereinfacht werden:

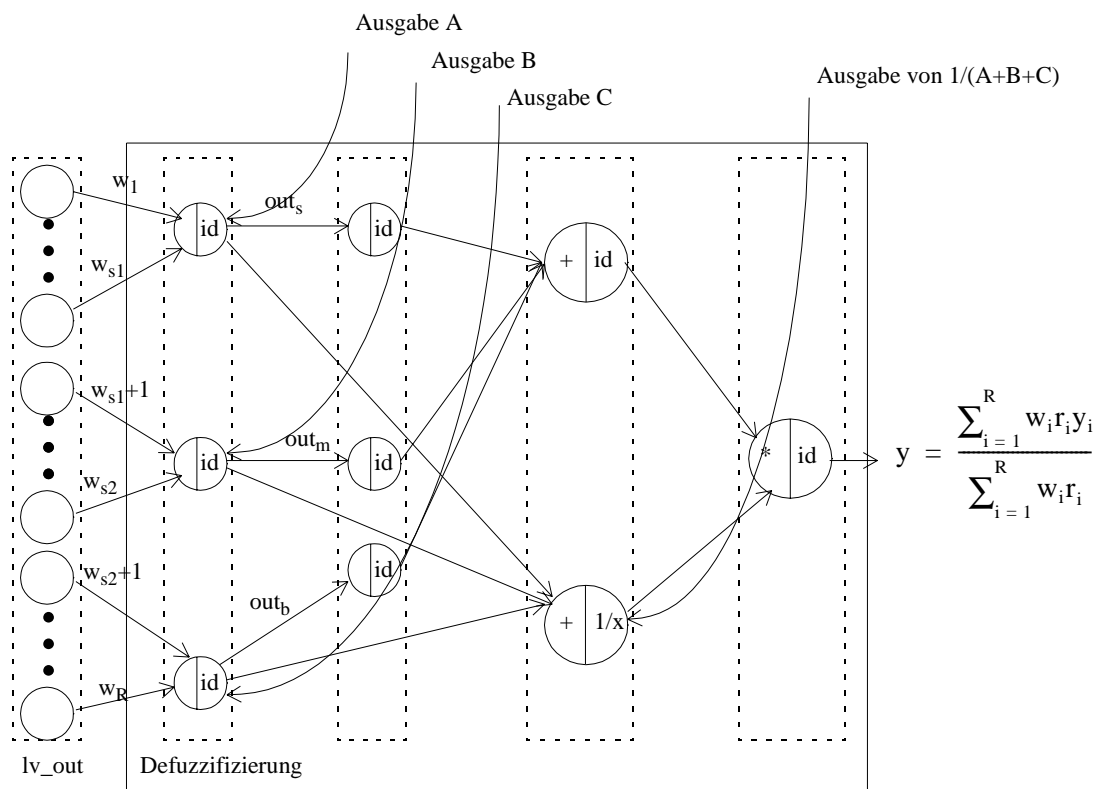
Betrachtet sei ein Regelsystem mit R Regeln, $R \in \mathbb{N}$, wie in 5.2 „Defuzzifizierung“ vorgestellt. Für Fuzzy-Routing wird eine linguistische Ausgabe-Variable mit drei linguistischen Werten verwendet. Die drei linguistischen Werte seien mit out_s , out_m und out_b bezeichnet. Es seien $s_1, s_2, s_3 \in \mathbb{N}$ mit $s_1 + s_2 + s_3 = R$. Je s_1 der R Regeln haben out_s , je s_2 der R Regeln haben out_m und je s_3 der R Regeln haben out_b als Konklusion. Dabei wird ohne Beschränkung der Allgemeinheit davon ausgegangen, daß die ersten s_1 Regeln den linguistischen Wert out_s , die nächsten s_2 Regeln out_b und die letzten s_3 Regeln out_b als Konklusion

haben. Diese Numerierung dient der einfacheren Darstellung. Der harte Ausgabewert errechnet sich durch

$$y = \frac{\sum_{i=1}^R w_i r_i y_i}{\sum_{i=1}^R w_i r_i} = \frac{\left(\sum_{i=1}^{s_1} w_i r_i\right) \text{out}_s + \left(\sum_{i=1}^{s_2} w_i r_i\right) \text{out}_m + \left(\sum_{i=1}^{s_3} w_i r_i\right) \text{out}_b}{\sum_{i=1}^R w_i r_i} \quad (5.21)$$

$$= \frac{A \text{out}_s + B \text{out}_m + C \text{out}_b}{\sum_{i=1}^R w_i r_i}$$

Die Netzarchitektur, die in Abbildung 5.9 gezeigt wird, kann wie in Abbildung 5.12 dargestellt, vereinfacht werden.



Nicht beschriftete Kanten haben als Gewicht eine 1

Abb. 5.12 Vereinfachte Netzarchitektur für die Defuzzifizierung

Die Werte für A, B, C und 1/(A+B+C) können wie in Abbildung 5.12 gezeigt in dem neuronalen Netz abgegriffen werden.

Eine Möglichkeit, die Singletons der linguistischen Ausgabe-Werte zu verschieben, ist die folgende:

Für jeden linguistischen Ausgabe-Wert out_s , out_m und out_b wird festgelegt in welchem Inter-

vall er liegen muß. Diese drei Intervalle überlappen sich nicht, grenzen direkt aneinander und sind so angeordnet, daß die Ordnung $out_s < out_m < out_b$ gilt. Der neue Wert für out_s berechnet sich:

$$out_{s_{new}} = out_{s_{old}} \pm \frac{A_{old}}{A_{old} + B_{old} + C_{old}} \eta \quad (5.22)$$

Für out_m und out_b gilt die entsprechende Formel. Wenn der Ausgabewert y kleiner ist als der Soll-Wert, so werden alle linguistischen Ausgabe-Werte vergrößert, und es gilt das Pluszeichen. Wenn der Ausgabewert y größer als der Soll-Wert ist, so werden alle linguistischen Ausgabe-Werte verkleinert und es gilt das Minuszeichen. η ist eine Lernkonstante, deren Wert experimentell zu bestimmen ist.

Wenn der neue Wert in seinem vorgegebenen Intervall liegt, so wird die Änderung ausgeführt, sonst nicht. Wenn während des Trainings für jeden linguistischen Wert getrennt die Verletzungen der Intervallgrenzen gespeichert werden, so kann eine Analyse dieser Daten zu einer Veränderung des regelbasierten Fuzzy-Systems verwendet werden. Verletzt zum Beispiel ein linguistischer Wert zyklisch seine obere und untere Intervallgrenze, so ist es sinnvoll, einen zusätzlichen linguistischen Wert hinzuzufügen.

Eine weitere Möglichkeit, die Werte der linguistischen Ausgabe-Variablen zu verschieben, ist die folgende:

Die Werte der linguistischen Ausgabe-Variablen sind wie im vorher beschriebenen Verfahren geordnet. Es gilt $0 < out_s < out_m < out_b < \text{maximale Obergrenze}$. Mit $distPlus_s$ sei der Abstand zum größeren Nachbarn von out_s geteilt durch 2 bezeichnet, also

$$distPlus_s = \left| \frac{out_m - out_s}{2} \right|; \quad (5.23)$$

mit $distMinus_s$ sei der Abstand zum unteren Nachbarn von out_s bezeichnet, also

$$distMinus_s = \left| \frac{0 - out_s}{2} \right|. \quad (5.24)$$

Für out_m und out_b gelten die entsprechenden Bezeichnungen.

Der Ausgabewert y sei größer als der Soll-Wert. Also müssen alle Werte verkleinert werden. Die neuen Werte berechnen sich wie in Gleichung (5.25) dargestellt, wobei die Werte für out_m und out_b entsprechend berechnet werden.

$$out_{s_{new}} = out_{s_{old}} - \frac{A_{old}}{A_{old} + B_{old} + C_{old}} \eta distMinus_s \quad (5.25)$$

Ist der Ausgabewert y kleiner als der Soll-Wert, so werden die Werte alle vergrößert. Die neuen Werte berechnen sich wie in Gleichung (5.26) dargestellt, wobei die Werte für out_m und out_b entsprechend berechnet werden.

$$out_{s_{new}} = out_{s_{old}} + \frac{A_{old}}{A_{old} + B_{old} + C_{old}} \eta distPlus_s \quad (5.26)$$

Die Werte für die Distanzen müssen bei jedem Lernschritt neu berechnet werden. Die Überprüfung der Einhaltung der Intervallgrenzen entfällt.

Weitere Adaptionenmöglichkeiten

Die beschriebenen Adaptionenmöglichkeiten können auch kombiniert werden - siehe auch 7.2.2. In einem ersten einfachen Ansatz könnte der für das Training vorhandene Datensatz geteilt werden. Mit dem ersten Teil werden ausschließlich die Regelgewichte adaptiert, während mit dem zweiten Teil die Ausgabe-Variablen adaptiert werden.

Weitere Ideen zur Adaption des regelbasierten Fuzzy-Systems sind im Ausblick von [4] zu finden.

Alle diese Möglichkeiten werden hier jedoch nicht verfolgt.

5.5 Überblick über die realisierten Routing Verfahren

Nachdem in dem vorangegangenen Text die verwendete Technik bezüglich der Fuzzy-System und der neuronalen Netze vorgestellt wurde, gibt dieser Abschnitt abschließend einen Überblick über die realisierten Verfahren.

Das regelbasierte Fuzzy-System wird als neuronales Netz mit zwei Defuzzifizierungsmethoden realisiert, mit modifizierter Schwerpunktmethod und mit der Methode nach Takagi-Sugeno.

So entstehen für Fuzzy-Routing zwei Realisierungen. Da FuzzyPlus-Routing auf Fuzzy-Routing basiert, hat auch FuzzyPlus-Routing diese beiden Ausprägungen.

Für das Fuzzy-Neuro-Verfahren wird als Lernalgorithmus die Adaption der Regelgewichte realisiert, wie in 5.4.3 beschrieben. Es hat als Grundlage die Darstellung des regelbasierten Fuzzy-Systems als neuronales Netz mit der modifizierten Schwerpunktmethod. Sowohl Delta Lernen als auch das Gradientenabstiegsverfahren werden eingesetzt. Das durch das Training entstehende Fuzzy-System wird anschließend für Fuzzy-Routing verwendet.

Die Vorteile und die Wirksamkeit der vorgestellten Routing-Verfahren sind in der praktischen Anwendung zu überprüfen. Zu diesem Zweck ist eine Simulationsumgebung geschaffen worden. Mit ihr können die Aufgaben, die während des Betriebes anfallen und die zur Überprüfung des Routing-Verfahrens notwendig sind, simuliert werden. Mit Hilfe dieser Simulationsumgebung können sowohl kleine Demonstrationsbeispiele untersucht werden, als auch große, einem realen Festnetz entsprechende Graphen bearbeitet werden.

Die beste Methode, die Vorteile der vorgestellten Routing-Verfahren nachzuweisen, wäre der Einsatz der Verfahren im Betrieb eines Festnetzes. Dieses Vorhaben scheitert leider daran, daß kein Betreiber eines Festnetzes seine Daten offen legt oder gar einen Zugang zum Netzmanagementsystem zuläßt.

Die Simulationsumgebung wird im folgenden beschrieben, bevor die Untersuchungen und die Ergebnisse für die einzelnen automatischen Routing-Verfahren dargelegt werden.

6.1 Simulationsumgebung

Wie schon in 3.4 beschrieben, müssen für die einzelnen Routing-Verfahren folgende Prüfungen durchgeführt werden:

- Beurteilung der einzelnen Route
- Feinsteuerung der Auswahl zur Beachtung strategischer Gesichtspunkte
- Auswirkungen auf das Layer-Netzwerk nach längerem Einsatz

Um diese Prüfungen durchzuführen ist folgende Funktionalität notwendig:

- Layer-Netzwerke werden erstellt und verändert:
Knoten und Kanten werden erzeugt, gelöscht, verändert und für die weitere Verwendung gespeichert.
- Ein einzelner Schaltvorgang wird ausgeführt:
dazu wird eine Route mit dem gewählten Routing-Verfahren bestimmt und die gewählten Kanäle werden als belegt gekennzeichnet. Das bedeutet, jeweils ein Kanal der gewählten Kanten wird als belegt gekennzeichnet und die Auslastung der betroffenen Kante und die Auslastung der angrenzenden Knoten wird entsprechend verändert. Die Fuzzy-Bewertung des Graphen wird neu durchgeführt,

- wenn das bei dem gewählten Routing-Verfahren notwendig ist.
- Eines der Routing-Verfahren, zum Beispiel Fuzzy-Routing, wird ausgewählt.
 - Der Einsatz eines Routing-Verfahrens wird simuliert: dazu werden beliebig viele Schaltungen durchgeführt.
 - Die Veränderungen, die durch einen simulierten Einsatz im Festnetz entstanden sind, werden dokumentiert.

Um die Routenbestimmung durchzuführen ist folgende Funktionalität notwendig:

- Fuzzy-Regelbasen werden erstellt und verändert.
- Aus einer Fuzzy-Regelbasis wird ein regelbasiertes Fuzzy-System erzeugt.
- Ein dem Layer-Netzwerk entsprechender Graph wird mit Hilfe des regelbasierten Fuzzy-Systems bewertet.
- Ein Kürzeste-Wege-Algorithmus führt die Routenbestimmung auf dem bewerteten Graphen durch.
- Die kostengünstigste Route aus der Menge der k-kürzesten mit Fuzzy-Routing bestimmten Routen, wird bestimmt. k kann vom Anwender gewählt werden.

6.1.1 Software

Die Software für diese Simulationsumgebung besteht aus zwei Komponenten:

TransMerkur

ist zuständig für die Betriebssimulation des Layer-Netzwerkes und der Routensuche, also die Aufgabe des Routensuchers aus Abbildung 3.5.

TransMerkur ist eine MS Windows Applikation zur Bearbeitung von Graphen, die eigens zu diesem Zweck erstellt wurde. Diese Komponente arbeitet wie in Abbildung 6.1 dargestellt mit NEUROTool nach dem Client-Server-Prinzip.

NEUROTool

ist zuständig für alle Aufgaben, die regelbasierte Fuzzy-Systeme und neuronale Netze betreffen. In diesem Teil wird auch die Aufgabe der Fuzzy-Bewertung aus Abbildung 3.5 durchgeführt.

NEUROTool ist ein Kommandointerpreter zur Erzeugung und Bearbeitung von neuronalen Netzen und zur Erzeugung von regelbasierten Fuzzy-Systemen. Er wird am Lehrstuhl von Prof. Wilfried Brauer eingesetzt und wurde von Dr. Hans Geiger entwickelt. Die Funktionalität von NEUROTool umfaßt unter anderem die Transformation von regelbasierten Fuzzy-Systemen in neuronale Netze nach der in 5.4 dargelegten Methode. NEUROTool wird über Kommandoeingabe gesteuert. Es verfügt über keine graphische Bedienoberfläche. Die graphische Bedienoberfläche wird durch TransMerkur bereitgestellt.

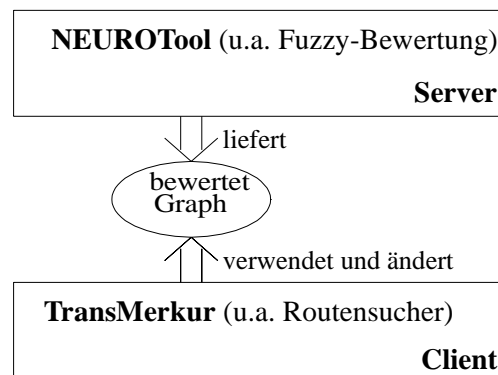


Abb. 6.1 Architektur der Simulationsumgebung

NEUROTool dient als Server und wird für folgende Aufgaben eingesetzt:

- Erstellen, Speichern und Lesen von Fuzzy-Regelbasen
- Erzeugen von regelbasierten Fuzzy-Systemen aus Fuzzy-Regelbasen
- Reversible Transformation von Fuzzy-Regelbasen in neuronale Netze
- Lesen, Bewerten und Speichern von Graphen
- Trainieren neuronaler Netze
- Darstellung der Trainingsergebnisse:
 - Ausgabe der Fehler pro Lernschritt in Datei
(Lernschritt bedeutet hier die Verarbeitung eines Musters)
 - Ausgabe der Fehler pro Lernzyklus in Datei
(Lernzyklus bedeutet hier die Verarbeitung aller Muster).

Mit Hilfe der durch NEUROTool bereitgestellten Kommandosprache können Fuzzy-Regeln als Objekte erfaßt und bearbeitet werden. Eine genaue Definition der Objekte und ihrer Arbeitsweise kann in [4] nachgelesen werden. Zusätzlich zu der dort beschriebenen Defuzzifizierungsmethode ist die in 5.2 beschriebene realisiert worden. Das regelbasierte Fuzzy-System ist als neuronales Netz realisiert. NEUROTOOL wurde um eine Auswahlmöglichkeit der Defuzzifizierungsmethode ergänzt. Beim Start von NEUROTool wird die Defuzzifizierungsmethode festgelegt und somit die Architektur des neuronalen Netzes, das zur Fuzzy-Bewertung des Graphen verwendet wird.

Die Funktionalität zur Bearbeitung von Graphen wurde speziell für den Einsatz mit TransMerkur ergänzt. Hinter der Fuzzy-Bewertung des Graphen verbirgt sich die Bewertung jeder seiner Kanten mit dem durch das regelbasierten Fuzzy-System ermittelten Wert, wie in Abbildung 3.5 verdeutlicht ist.

Die Funktionen zur Adaption der Regelgewichte sind speziell für diese Anwendung realisiert worden.

TransMerkur arbeitet als Client und führt folgende Aufgaben aus:

- Erstellen und Verändern von Graphen, deren Kanten und Knoten die in 6.1.2.1 definierten Parameter haben,

- Bedienung der Kommandoschnittstelle zu NEUROTool um
 - Regelbasierte Fuzzy-Systeme aus vorhandenen Fuzzy-Regelbasen zu erstellen
 - Regelbasierte Fuzzy-Systeme zu trainieren
 - Fuzzy-Bewertungen von Graphen auszuführen
- Routensuche nach verschiedenen Verfahren auf kantenbewerteten Graphen
 - Zur Bestimmung einer einzelnen Route
Es wird davon ausgegangen, daß die bestimmte Route für einen Übertragungsweg verwendet wird, also auch geschaltet wird; siehe auch 2.4. So beinhaltet die Bestimmung der Route auch die Belegung der verwendeten Kanäle und somit die Veränderung der Auslastung der Kanten und ihrer jeweils angrenzenden Knoten. Ist als Routing-Verfahren Fuzzy-Routing oder FuzzyPlus-Routing ausgewählt worden, so werden die Kanten, deren Auslastung sich verändert hat, neu fuzzy-bewertet.
 - Zur Bestimmung einer Liste von Routen, um so einen Einsatz zu simulieren. Für alle Elemente der Liste wird die Bestimmung der einzelnen Route durchgeführt.
- Trainieren des Systems mit Hilfe von Routenwunschlisten
- Ausgabe von Ergebnissen
 - nach einem simulierten Einsatz:
Ausgabe der Parameterwerte aller Knoten und Kanten des Graphen;
Ausgabe der Parameterwerte aller in den gefundenen Routen verwendeten Knoten und Kanten;
die Ausgabe erfolgt in Datei; die graphische Darstellung erfolgt mit MS-EXCEL.
 - Während des Trainings:
Ausgabe der prozentualen Anzahl falscher Kanten in den Ist-Routen nach jedem Trainingsschritt.

Ein Graph kann über die graphische Oberfläche durch Positionieren der Knoten und Kanten auf der Oberfläche und Eingabe der entsprechenden Parameter erzeugt und verändert werden oder über die zur Verfügung stehende Dateischnittstelle. Abbildung 6.3 zeigt die Darstellung eines Layer-Netzwerkes als Graph an der Oberfläche.

Zur Bestimmung einer einzigen kürzesten Route wird ein klassischer Dijkstra Algorithmus verwendet [1]. Zur Bestimmung von k-kürzesten Routen für FuzzyPlus-Routing wird der Double-Sweep-Algorithmus von Shier [39] verwendet.

Das Gewicht jeder Kante ist in ihrem Parameter *edgeVal* abgelegt, wie in 6.1.2.1 beschrieben. Dieses Gewicht wird zur Bestimmung der kürzesten Route verwendet. Die kürzeste Route ist also die, deren Kantenfolge die kleinste Summe der *edgeVal* Werte hat.

Für die Routensuche können drei Minimierungskriterien ausgewählt werden:

- Minimale physikalische Länge
TransMerkur versorgt *edgeVal* jeder Kante mit ihrer physikalischen Länge.
Das Routing-Verfahren, das dieses Minimierungskriterium verwendet, wird im folgenden mit **Length-Routing** bezeichnet.
- Minimale Anzahl der Hops
TransMerkur versorgt *edgeVal* jeder Kante mit 1. So wird durch den Algorithmus sichergestellt, daß die Route mit minimaler Anzahl der Knoten gefunden wird. Das bedeutet in dieser Anwendung eine Minimierung der Schaltpunkte, auch Hops genannt, denn Schaltpunkte liegen bei dieser Modellierung ausschließlich in den Knoten.
Das Routing-Verfahren, das dieses Minimierungskriterium verwendet, wird im folgenden mit **Hop-Routing** bezeichnet.
- Regelbeachtung
TransMerkur beauftragt NEUROTool mit der Fuzzy-Bewertung des Graphen und versorgt so *edgeVal* jeder Kante mit dem Wert, der bei der Bewertung der Kante durch das regelbasierte Fuzzy-System entsteht. So wird bei der Routenbestimmung an jeder Auswahlstelle die Kante bevorzugt, die am besten zu den Regeln paßt.

Die so entstehenden Routing-Verfahren, Length-Routing und Hop-Routing, werden für den Vergleich mit den neuen Routing-Verfahren verwendet.

Die Regelbeachtung bei der Routenbestimmung führt zu Fuzzy-Routing oder FuzzyPlus-Routing je nach weiterer Verarbeitung.

6.1.2 Daten

Als Testobjekte werden Layer-Netzwerke benötigt, das heißt, Kanten und Knoten und ihre Parameter. Zur Bewertung der Knoten und Kanten sind die zugehörigen Fuzzy-Regelsysteme zu formulieren.

Es ist aus Gründen der Geheimhaltung und des Wettbewerbs nicht möglich, die Daten eines realen Festnetzes zu präsentieren. Deshalb wurden „künstliche“ Daten erzeugt. Um Daten für die Tests zu erzeugen, wurden zahlreiche Interviews und Diskussionen mit Experten von Betreibern von Festnetzen und TMN-Experten von Siemens Business Services durchgeführt. An Hand der so erhaltenen Informationen wurden die im folgenden vorgestellten Parameter der Kanten und Knoten und die Regeln festgelegt. Dazu passend wurden Layer-Netzwerke erzeugt. Dieses Datenmaterial wurde den Experten der Betreiber vorgestellt und von ihnen als aussagekräftig und realistisch beurteilt. Daher wird davon ausgegangen, daß das hier vorgestellte Datenmaterial genutzt werden kann, um die Methoden sinnvoll und realistisch zu überprüfen.

Wenn eines der hier vorgestellten Routenverfahren in einem realen Festnetz eingesetzt wird, so müssen die Parameter und Regeln auf die entsprechenden Layer-Netzwerke abgestimmt werden.

6.1.2.1 Parameter des Layer-Netzwerks

Das Layer-Netzwerk besteht aus zwei verschiedenen Objekttypen, Kanten und Knoten. Eine Kante wird durch zwei Knoten, Knoten A und Knoten B, begrenzt. Handelt es sich bei der Kante um eine Schleife, so sind die beiden Knoten identisch, sonst sind sie unterschiedlich. Als Parameter für Knoten und Kanten werden die in 3.1 beschriebenen verwendet. Zur Fuzzy-Bewertung jeder Kante werden die Parameter der Kante selbst und die ihrer angrenzenden Knoten verwendet, wie schon in 3.5 erläutert. Diese Parameter der Kanten sind in Tabelle 6.1 aufgelistet. In einigen der folgenden Beispiele werden jedoch die Parameter der Knoten getrennt aufgelistet, wenn das der Übersichtlichkeit dient. Die Tabelle zeigt sowohl die Eingabe-Parameter und ihre Wertebereiche wie auch die in der Fuzzy-Regelbasis verwendeten linguistischen Eingabe-Variablen und ihre linguistischen Werte. Die Definition der Zugehörigkeitsfunktionen für die linguistischen Werte kann in ANHANG A nachgelesen werden.

Eingabe-Parameter	Wertebereich	linguistische Variable	linguistische Werte
Knotenart des Knotens A	fernschaltbar (1) / nicht fernschaltbar (0)	ndCIA	ndCIA_fs ndCIA_nfs
Knotenart des Knotens B	fernschaltbar (1) / nicht fernschaltbar (0)	ndCIB	ndCIB_fs ndCIB_nfs
Auslastung des Knotens A	[0,100] Angabe in Prozent	ndLsA	ndLsA_s, ndLsA_m, ndLsA_b

*) : zum Test sind Werte aus [0,100] verwendet worden, abstatt der üblichen Werte

Tab. 6.1 Parameter der Kanten des Layer-Netzwerks

Eingabe-Parameter	Wertebereich	linguistische Variable	linguistische Werte
Auslastung des Knotens B	[0,100] Angabe in Prozent	ndLsB	ndLsB_s, ndLsB_m, ndLsB_b
Betriebskosten des Knoten A	[0,100] Angabe in Prozent der maximalen Knotenkosten	ndCoA	ndCoA_s ndCoA_m, ndCoA_b
Betriebskosten des Knoten B	[0,100] Angabe in Prozent der maximalen Knotenkosten	ndCoB	ndCoB_s ndCoB_m, ndCoB_b
Auslastung der Kante	[0,100] Angabe in Prozent, zeitabhängig	beGr	beGr_s, beGr_m, beGr_b
Bitfehlerhäufigkeit der Kante	[0,100] Angabe in Prozent, je kleiner um so besser *	bitEr	bitEr_s bitEr_m, bitEr_b
Verfügbarkeit	[0,100] Angabe in Prozent, je größer um so besser	avai	avai_s, avai_m, avai_b
physikalische Länge	[0, max. Distanz im Netz] Angabe in Pixel, da die Darstellung des Layer-Netzwerkes auf dem Bildschirm erfolgt	length	length_s, length_m, length_b

*) : zum Test sind Werte aus [0,100] verwendet worden, abstatt der üblichen Werte

Tab. 6.1 Parameter der Kanten des Layer-Netzwerks

Eine Kante des Layer-Netzwerks besteht, wie schon in 3.1 dargestellt, aus einer Menge von Kanälen, deren Endpunkte dieselben Knoten sind und die alle die selben Parameterwerte haben.

Die Kanäle werden als bidirektional angenommen und somit auch die Kanten.

6.1.2.2 Regelsystem

Für das oben beschriebene Layer-Netzwerk wird ein Regelwerk erstellt. Beispielhaft wird ein Regelsystem für hochprioritäre Übertragungswege erstellt. Wie schon in 3.3 dargelegt, wird bei hochprioritären Übertragungswegen dem Kunden eine gewisse Mindestqualität garantiert, das bedeutet die Verfügbarkeit darf die garantierte Mindestverfügbarkeit nicht unterschreiten und die Bitfehlerhäufigkeit darf die garantierte maximale Bitfehlerhäufigkeit nicht überschreiten. Auf Grund der eingehenden Diskussionen mit TMN-Experten eines TMN-System-Herstellers, Siemens Business Services, wie auch eines TMN-System-Nutzers, Deutsche Telekom,- siehe auch 2.4 -, und mit der in 3.3 ausgeführten Analyse ergeben

sich folgenden Aspekte zur Beurteilung der Kanten und Knoten:

- Aus betrieblicher Sicht:
 - *Kanten mit hoher Verfügbarkeit sind zu bevorzugen.*
 - *Kanten mit kleiner Bitfehlerhäufigkeit sind zu bevorzugen.*
 - *Die Verfügbarkeit der Kante darf nicht kleiner sein als die dem Kunden garantierte Mindestverfügbarkeit.*
 - *Die Bitfehlerhäufigkeit darf nicht größer sein als die dem Kunden garantierte maximale Bitfehlerhäufigkeit.*
 - *Kanten mit großer physikalischer Länge sind eher abzulehnen und solche mit kleiner Länge zu bevorzugen.*
- Aus wirtschaftlicher Sicht:
 - *Kaum ausgelastete Knoten sollen nicht mehr genutzt werden, um sie eventuell ganz frei schalten zu können und anschließend an einem anderen Ort einsetzen zu können.*
 - *Kanten mit sehr kleiner Auslastung sind abzulehnen.*
 - *Knoten und Kanten sollen möglichst gleichmäßig ausgelastet werden.*
 - *Knoten mit hohen Kosten sind abzulehnen und solche mit niedrigen Kosten sind zu bevorzugen.*
 - *Kanten zwischen fernschaltbaren Knoten, sind gegenüber den Kanten zwischen nicht fernschaltbaren Knoten zu bevorzugen.*
 - *Die Kanten sollen zwar den Anforderung an die Verfügbarkeit und die Bitfehlerhäufigkeit genügen, aber es soll nicht mehr Aufwand als unbedingt nötig entstehen. Denn je höher die Verfügbarkeit eine Kante ist, um so mehr müssen Wartungsarbeiten durchgeführt werden, um diese Verfügbarkeit zu bieten. Das Gleiche gilt für die Bitfehlerhäufigkeit. Die Abbildung 6.2 verdeutlicht diese Anforderung.*

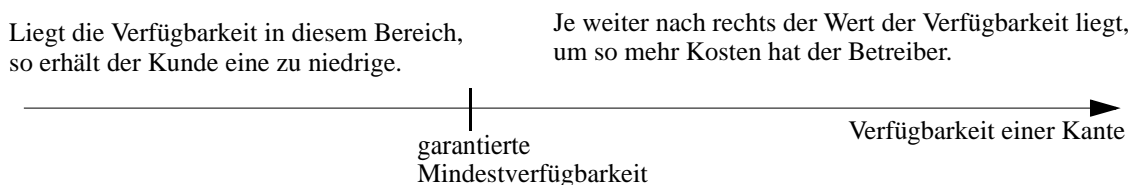


Abb. 6.2 Anforderungen an die Verfügbarkeit

- *Es gibt Kanten, die aus strategischen Gründen gesondert behandelt werden sollen. Sie sind für besondere Verbindungen reserviert. Es soll möglich sein, solche Kanten nicht für die Routensuche zu verwenden oder sie gezielt zu verwenden.*

Um die beiden letzten Aussagen auf Fuzzy-Regeln abbilden zu können, sind den Parametern der Kante, die in Tabelle 6.1 aufgeführt sind, die in der Tabelle 6.2 genannten hinzugefügt

worden.

Eingabe-Parameter	Wertebereich	linguistische Variable	linguistische Werte
Abweichung von der maximal erlaubten Bitfehlerhäufigkeit der Kante	[-100,100]	deltabitErr	deltabitErr_good deltabitErr_bad, deltabitErr_ecoBad
Abweichung von der Mindestverfügbarkeit	[-100,100] Angabe in Prozent	deltaAvali	deltaAvali_good, deltaAvali_bad, deltaAvali_ecoBad
strategische Bedeutung	[0,10] je höher um so wichtiger	stratMean	stratMean_s, stratMean_m, stratMean_b

Tab. 6.2 Zusätzliche linguistische Variablen

Es handelt sich um zusätzliche Eingabe-Parameter und ihre linguistischen Variablen. Sie werden eigens zum Formulieren der Fuzzy-Regeln erzeugt. Ihre Werte errechnen sich aus den Werten der schon vorhandenen Parameter, wie *deltabitErr* und *deltaAvali*, oder aber sie müssen zusätzlich vom Benutzer vergeben werden, wie *stratMean*.

Die Anforderung an die Abweichung von der Mindestverfügbarkeit kann mit Hilfe von *deltaAvali* wie folgt formuliert werden:

- *Ist die Abweichung von der Mindestverfügbarkeit positiv aber gering (deltaAvali_good), so ist die Kante zu bevorzugen.*
- *Ist die Abweichung von der Mindestverfügbarkeit negativ (deltaAvali_bad), so ist die Kante abzulehnen.*
- *Ist die Abweichung von der Mindestverfügbarkeit positiv aber sehr groß (deltaAvali_ecoBad), so ist die Kante abzulehnen.*

Die Anforderung an die Abweichung von der erlaubten Bitfehlerhäufigkeit wird entsprechend formuliert, siehe ANHANG A.

Um Kanten aus strategischen Gesichtspunkten in einem Fall zu bevorzugen und im anderen Fall abzulehnen, wird mit Hilfe des neuen Parameters *stratMean* folgende Regel formuliert:

- *Kanten mit hoher strategischer Bedeutung (stratMean_b) sind zu bevorzugen.*

Alle Kanten des Layer-Netzwerkes haben zu Beginn denselben Wert für diesen Parameter. Dieser Wert wird als Defaultwert bezeichnet. Durch das Verändern der Werte von *stratMean* einer Kante hat der Benutzer nun die Möglichkeit, ihre Verwendung zu bevorzugen oder abzulehnen und trotzdem für die Wahl der Kanten das automatische Routing-Verfahren zu nutzen.

Mit Hilfe dieses Parameters kann auch das Problem gelöst werden, das entsteht, wenn neue Knoten in den Graphen aufgenommen werden:

Es kann in der Praxis vorkommen, daß neue Netzknoten aufgestellt und in das bestehende Transportnetz aufgenommen werden. Dem entspricht in der Modellierung das Aufnehmen eines neuen Knoten in das Layer-Netzwerk. Die Auslastung dieser Knoten und der in ihnen startend und endenden Kanten ist zu diesem Zeitpunkt zunächst Null. Sind diese Knoten

jedoch nicht Quelle oder Senke der zu bestimmenden Route, so würden die in ihnen startend und endenden Kanten normalerweise mit Fuzzy-Routing erst dann gewählt, wenn alle anderen Kanten „schlecht“ im Sinne der Regeln wären, das heißt insbesondere überlastet oder die Qualität sehr viel schlechter ist als die der neu angebotenen Kanten.

Um dieses Verhalten zu vermeiden und die Kanten, die diesen neuen Knoten als Anfangs- oder Endpunkt haben, trotz ihrer schlechten Auslastung eher zu wählen, kann der Wert ihrer strategischen Bedeutung hoch gesetzt werden. Erst wenn die Auslastung dieser Kanten im üblichen Bereich liegt, wird der Wert für die strategische Bedeutung dieser Kante wieder auf den Defaultwert gesetzt. So werden die Regeln speziell für eine ausgewählte Kante umgangen und zwar solange wie es der Benutzer wünscht.

Dieses Verfahren eignet sich auch dafür, neue Leitungen in das Transportnetz aufzunehmen, die als neue Kanten des Graphen auch zunächst eine Auslastung von Null aufweisen.

Die Steuerungsmöglichkeit mittels strategischer Bedeutung wird in 6.2 demonstriert.

Aus allen diesen Regeln, die zum Teil widersprüchlich sind, wird ein konsistentes Fuzzy-System erzeugt, wie in 4.2 unter „Regelsystem“ beschrieben. Als linguistische Ausgabe-Variable wird der Kantenwert eingeführt - siehe Tabelle 6.3. Dieses Fuzzy-System bestimmt für jede Kante des Graphen aus deren Parametern und den Parametern ihrer angrenzenden Knoten ihren Kantenwert: es weist also jeder Kante einen Wert für *edgeVal* zu. So wird der fuzzy-bewertete Graph erzeugt, auf dem eine Routensuche stattfinden kann.

Ausgabe-Parameter	Wertebereich	linguistische Variable	linguistische Werte
Kantenwert	[0,50] Je kleiner der Wert, um so besser die Kante, d.h. um so eher wird sie bei der Routenwahl verwendet	edgeVal	edgeVal_s, edgeVal_m, edgeVal_b

Tab. 6.3 Ausgabe-Parameter

Die einzelnen Regeln werden mit Gewichten versehen, um ihren Einfluß auf die Beurteilung steuern zu können. Bei der Steuerung des Regeleinflusses geht es in dieser Anwendung darum, den Einfluß der einzelnen Parameter unterschiedlich zu gewichten. Es kann zum Beispiel gewünscht sein, den Einfluß der Auslastung der einzelnen Kanten höher zu bewerten als den Einfluß ihrer Bitfehlerhäufigkeit. Das hier verwendete regelbasierte Fuzzy-System hat, wie schon in 4.2 ausgeführt, die folgende Eigenschaft:

Je öfter ein linguistischer Wert der linguistischen Ausgabe-Variablen als Konklusion vorkommt, um so größer ist sein Einfluß auf das Ergebnis.

Daher werden die Regeln wie in 4.2 vorgeschlagen umgruppiert.

Beispiel:

Die Regel

Knoten mit hohen Kosten sind abzulehnen und solche mit niedrigen Kosten sind zu bevorzugen.

wird mit Hilfe der linguistischen Variablen so formuliert:

IF ndCoA = ndCoA_s **AND** ndCoB = ndCoB_s **THEN** edgeVal_s
IF ndCoA = ndCoA_s **AND** ndCoB = ndCoB_m **THEN** edgeVal_s
IF ndCoA = ndCoA_s **AND** ndCoB = ndCoB_b **THEN** edgeVal_b
IF ndCoA = ndCoA_m **AND** ndCoB = ndCoB_s **THEN** edgeVal_m
IF ndCoA = ndCoA_m **AND** ndCoB = ndCoB_m **THEN** edgeVal_m
IF ndCoA = ndCoA_m **AND** ndCoB = ndCoB_b **THEN** edgeVal_b
IF ndCoA = ndCoA_b **AND** ndCoB = ndCoB_s **THEN** edgeVal_b
IF ndCoA = ndCoA_b **AND** ndCoB = ndCoB_m **THEN** edgeVal_b
IF ndCoA = ndCoA_b **AND** ndCoB = ndCoB_b **THEN** edgeVal_b

Daraus wird nach der Umgruppierung:

IF (ndCoA = ndCoA_s **AND** ndCoB = ndCoB_s) **OR** (ndCoA = ndCoA_s **AND** ndCoB = ndCoB_m) **THEN** edgeVal_s
IF (ndCoA = ndCoA_m **AND** ndCoB = ndCoB_s) **OR** (ndCoA = ndCoA_m **AND** ndCoB = ndCoB_m) **THEN** edgeVal_m
IF (ndCoA = ndCoA_s **AND** ndCoB = ndCoB_b) **OR** (ndCoA = ndCoA_m **AND** ndCoB = ndCoB_b) **OR** (ndCoA = ndCoA_b **AND** ndCoB = ndCoB_s) **OR** (ndCoA = ndCoA_b **AND** ndCoB = ndCoB_m) **OR** (ndCoA = ndCoA_b **AND** ndCoB = ndCoB_b) **THEN** edgeVal_b

Zu jedem der drei linguistischen Ausgabewerte gibt es nun genau eine Regeln für die Parametergruppe „Knotenkosten“.

Aus diesem umformulierten Regelsystem wird mit Hilfe von NEUROTool das entsprechende regelbasiertes Fuzzy-System erzeugt.

Das gesamte Regelsystem und seine umformulierte Form ist in ANHANG A aufgeführt.

6.2 Fuzzy-Routing

Die im folgenden beschriebenen Tests für Fuzzy-Routing und für FuzzyPlus-Routing wurden mit beiden Defuzzifizierungsmethoden ausgeführt, mit modifizierter Schwerpunktmethode und mit der Methode nach Takagi-Sugeno. Die Ergebnisse haben gezeigt, daß die unterschiedlichen Defuzzifizierungsmethoden keinen Einfluß auf die Wirksamkeit dieser Routing-Verfahren hat. Die Ergebnisse, die in 6.2 „Fuzzy-Routing“ und 6.3 „FuzzyPlus-Routing“ beschrieben werden, beziehen sich daher nur auf die Darstellung des regelbasierten Fuzzy-Systems als neuronales Netz mit der modifizierten Schwerpunktmethode. Als Kürzeste-Wege-Algorithmus auf dem fuzzy-bewerteten Graphen wird der Dijkstra-Algorithmus verwendet. Die genaue Arbeitsweise von Fuzzy-Routing ist in 3.5.1 beschrieben, das zugrundliegende regelbasierte Fuzzy-System in 4.2.

Für die hier vorgestellten Tests wird ein Layer-Netzwerk aus 38 Knoten und 80 Kanten verwendet. Die Parameterwerte der Knoten und Kanten sind in ANHANG B aufgelistet. Abbildung 6.3 zeigt einen Ausschnitt des Layer-Netzwerks mit Hilfe der Graphdarstellung in TransMerkur. Die Zahl, die an den Kanten gezeigt wird, ist der Wert von *edgeVal*. In der Darstellung ist der Kantenwert mit 100 multipliziert.

Die Auslastung jedes Knoten und jeder Kante ist zu Beginn auf 40% gesetzt.

Die Mindestverfügbarkeit ist auf 60 gesetzt und die maximal erlaubte Bitfehlerhäufigkeit auf 40.

Die im nachfolgenden beschriebenen Tests wurden auch für weitere Layer-Netzwerke durchgeführt. Sie haben die nachfolgend dargestellten Ergebnisse bestätigt.

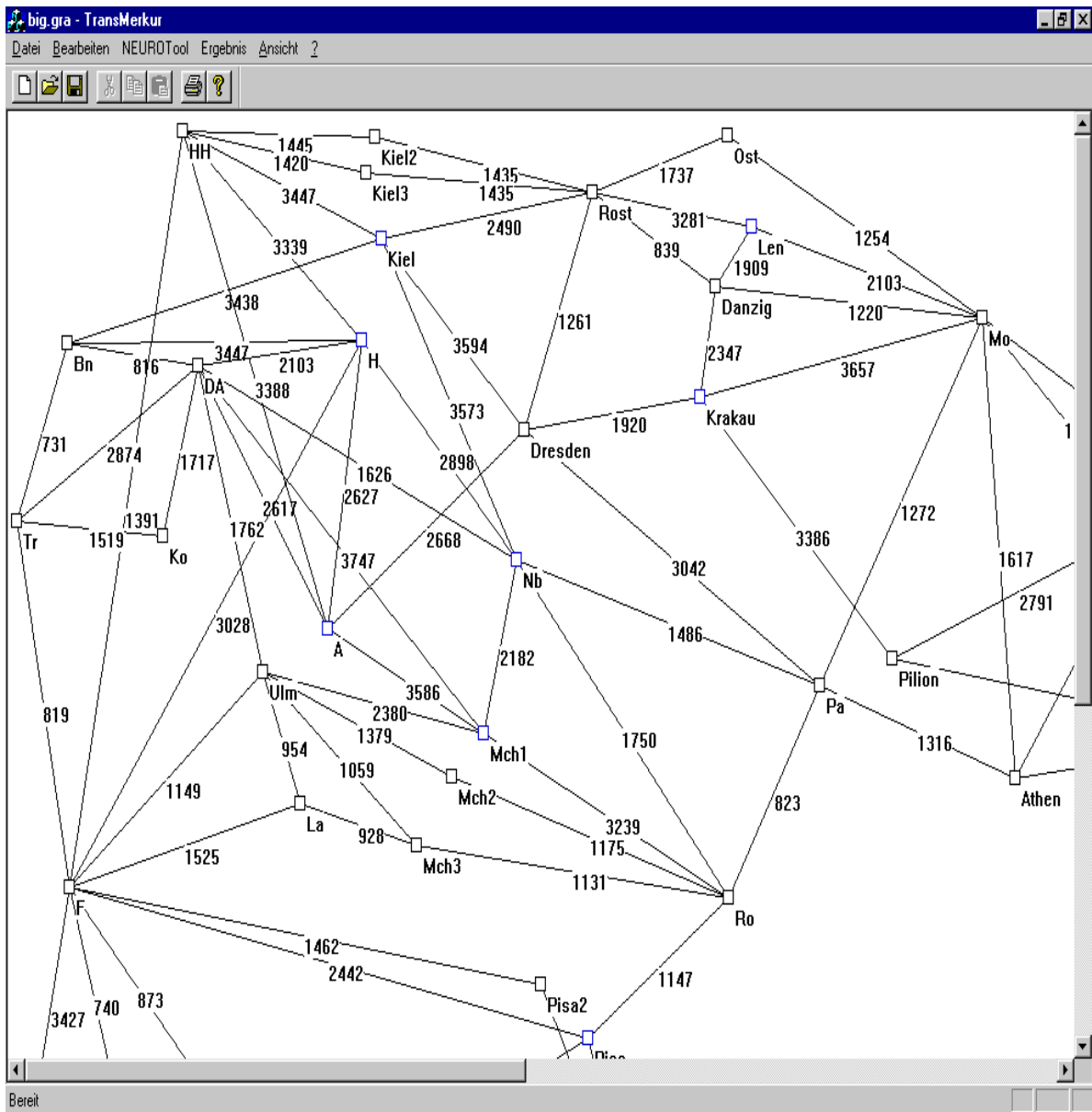


Abb. 6.3 Ausschnitt aus dem Layer-Netzwerk

6.2.1 Beurteilung der gewählten Routen

Im folgenden Text wird zusammenfassend Verfügbarkeit und Bitfehlerhäufigkeit mit Qualität bezeichnet. Eine Unter- beziehungsweise Überschreitung der Grenzwerte dieser Parameter wird mit Verletzung der Qualitätskriterien bezeichnet.

Da das Verfahren die Route mit der kleinsten Summe der Kantenwerte ihrer Kantenfolge bestimmt und der Graph zusammenhängend ist, wird für jeden Routenwunsch eine Route gefunden. Wie gut diese gefundene Route den Regeln genügt, hängt unter anderem davon ab, welche Kanten im Layer-Netzwerk zur Verfügung stehen. Wenn die Werte für Verfügbarkeit und Bitfehlerhäufigkeit aller Kanten des Layer-Netzwerks, die noch freie Kanäle haben, nicht den Anforderungen genügen, so werden auch nur Routen gefunden, deren Kanten die

Qualitätskriterien nicht erfüllen. Um beispielhaft das Verhalten von Fuzzy-Routing bei der Auswahl zu dokumentieren, werden nachfolgend die Ergebnisse für zwei Routenwünsche im Layer-Netzwerk dargestellt und interpretiert. Außer mit Fuzzy-Routing wurde die Route auch jeweils mit Length- und Hop-Routing bestimmt.

• **Beispiel 1:**

Routenwunsch: *HH - Mo*

Der Ausschnitt aus dem Layer-Netzwerk, der für diesen Routenwunsch wichtig ist, ist in Abbildung 6.3 dargestellt.

HH ist oben links und *Mo* oben rechts zu sehen.

Ergebnisse:

Fuzzy-Routing: *HH - Kiel3 - Rost - Danzig - Mo*

Hop-Routing: *HH - Kiel - Dresden - Krakau - Mo*

Length-Routing: dieselbe Route wie Fuzzy-Routing

	HH-Kiel	HH-Kiel2	HH-Kiel3	Kiel3-Rost	Rost-Ost	Rost-Len	Rost-Danzig	Danzig-Mo
strategische Bedeutung	5	5	5	5	5	5	5	5
physikalische Länge	194	175	169	206	128	146	128	243
Auslastung	40	40	40	40	40	40	40	40
Verfügbarkeit	30	80	80	80	90	50	70	60
Abweichung von der Mindestverfügbarkeit	-30	20	20	20	30	-10	10	0
Bitfehlerhäufigkeit	60	10	10	7	12	70	30	40
Abweichung von der maximal erlaubten Bitfehlerhäufigkeit	-20	30	30	33	28	-30	10	0
Kantenwert	34,47	14,45	14,20	14,35	17,37	32,81	8,39	12,20

Tab. 6.4 Parameter der betroffenen Kanten und ihre Fuzzy-Bewertung für Beispiel 1

Interpretation:

Die für die nachfolgende Interpretation wichtigsten Parameterwerte sind in Tabelle 6.4 und Tabelle 6.5 aufgelistet.

In der Fuzzy-Routing Route werden nur Knoten verwendet, die fernschaltbar sind. Das entspricht den Anforderungen an eine Route, denn fernschaltbare Knoten sind zu bevorzugen.

Die Knotenkosten von *Kiel2* und *Kiel3* sind 40, während die Knotenkosten von *Kiel* 94 sind. Außerdem ist *Kiel* nicht fernschaltbar. Aus der Menge der Knoten, die für

eine Route von *HH* in Richtung *Mo* in Frage kommen würden, ist also *Kiel2* oder *Kiel3* zu bevorzugen. Die Kante *HH - Kiel* verletzt außerdem die Qualitätskriterien. Zur Auswahl stehen also noch die Kanten *HH - Kiel2* und *HH - Kiel3*. Beide Kanten haben bis auf die Länge die selben Parameterwerte. Da *HH - Kiel3* die kürzere der beiden Kanten ist, ist diese zu bevorzugen. Das zeigen auch die *edgeVal*-Werte der beiden Kanten. Die nächste Kante *Kiel3 - Rost* erfüllt die Qualitätsanforderungen an hochprioritäre Routen und unterscheidet sich fast nicht in ihren Parameterwerten von der Kante *Kiel2 - Rost*.

Die zu betrachtenden Alternativen, um von *Rost* in Richtung *Mo* zu gelangen, sind *Rost - Len*, *Rost - Ost* und *Rost - Danzig*.

Rost - Len, bietet nicht die gewünschte Qualität,

Rost - Ost erfüllt zwar die Grenzwerte, die an die Qualität gestellt werden, aber überschreitet sie um 30 für die Verfügbarkeit und um 28 für die Bitfehlerhäufigkeit. Die Kante *Rost - Danzig* überschreitet sie nur um 10, sowohl für die Verfügbarkeit als auch für die Bitfehlerhäufigkeit. Da beide sonst gleichwertig sind, ist *Rost - Danzig* zu bevorzugen.

Die beste der Alternativen wurde auch gewählt.

Die Kante *Danzig - Mo* entspricht ebenfalls genau den Qualitätsanforderungen.

	Kiel-Dresden	Dresden-Krakau	Krakau-Mo
strategische Bedeutung	5	5	5
physikalische Länge	182	161	262
Auslastung	40	40	40
Verfügbarkeit	27	60	30
Abweichung von der Mindestverfügbarkeit	-33	0	-30
Bitfehlerhäufigkeit	79	40	90
Abweichung von der maximal erlaubten Bitfehlerhäufigkeit	-39	0	-50
Kantenwert	35,94	19,20	36,57

Tab. 6.5 Parameter der betroffenen Kanten und ihre Fuzzy-Bewertung für Beispiel 1

Da Length-Routing zu demselben Ergebnis kommt wie Fuzzy-Routing, beachtet also Fuzzy-Routing im diesem Fall nicht nur die Qualitätskriterien, sondern hat auch noch die Route mit der kürzesten Länge gewählt. Der Parameter Länge wird bei den Regeln auch beachtet, und zwar nach dem Prinzip „je kürzer je besser“. Die genaue Übereinstimmung der Ergebnisse von Fuzzy-Routing und Length-Routing ist auf die speziellen Daten zurückzuführen. Sie ist im Normalfall eher unwahr-

scheinlich.

Hop-Routing wählt eine Route, in der zwei nicht fernschaltbare Knoten vorkommen, *Kiel* und *Krakau*. Das widerspricht den Regeln, die für eine gute Route gelten. Die Kante *Dresden - Krakau* erfüllt die Qualitätskriterien, aber alle anderen gewählten Kanten verletzen die Qualitätskriterien.

- **Beispiel 2:**

Routenwunsch: *Tr - Ro*

Tr ist in der Abbildung 6.3 links in der Mitte und *Ro* unten rechts zu sehen.

Ergebnisse:

Fuzzy-Routing: *Tr-F-Ulm-Mch3-Ro*

Hop-Routing: *Tr-DA-Mch1-Ro*

Length-Routing: wie Hop-Routing

	Tr-F	F-Ulm	F-La	Ulm-Mch3	Mch3-Ro	Tr-DA	DA-Mch1	Mch1-Ro
strategische Bedeutung	5	5	5	5	5	5	5	5
physikalische Länge	249	227	217	181	286	195	357	248
Auslastung	40	40	40	40	40	40	40	40
Verfügbarkeit	80	80	60	60	65	30	30	30
Abweichung von der Mindestverfügbarkeit	20	20	0	0	5	-30	-30	-30
Bitfehlerhäufigkeit	20	15	12	30	35	80	71	10
Abweichung von der maximal erlaubten Bitfehlerhäufigkeit	20	25	28	10	5	-40	-31	30
Kantenwert	8,19	11,49	15,25	10,59	11,31	28,74	37,47	32,39

Tab. 6.6 Parameter der betroffenen Kanten und ihre Fuzzy-Bewertung für Beispiel 2

Interpretation:

Die für die nachfolgende Interpretation wichtigsten Parameterwerte sind in Tabelle 6.6 aufgelistet.

Um von *Tr* aus in Richtung *Ro* zu gelangen, sind als erste Alternativen die Knoten *DA* und *F* zu betrachten. Beides sind fernschaltbare Knoten. Der Wert der Knotenkosten von *DA* beträgt 20, der von *F* 30.

Die Kante *Tr - DA* erfüllt nicht die für hoch prioritätäre Routen notwendigen Qualitätskriterien, während die Kante *Tr - F* die Qualitätskriterien einhält. Also ist es im Sinne der Regeln richtig, die Kante *Tr - F* zu wählen, trotz der höheren Knotenkosten von *F*.

Eine Routewahl von F nach Ro muß Routen über die nachfolgend aufgelisteten Knoten betrachten:

	Mch1	Mch2	Mch3	Pisa
Knotenart	nicht fernschaltbar	fernschaltbar	fernschaltbar	nicht fernschaltbar
Betriebskosten des Knotens	85	12	10	50

Die Knoten $Mch2$ und $Mch3$ sind den Knoten $Mch1$ und $Pisa$ vorzuziehen, da sie billiger und fernschaltbar sind. $Mch3$ ist der billigere der beiden zu bevorzugenden Knoten.

Um von F nach $Mch3$ zu gelangen, gibt es die Möglichkeit $F - La - Mch3$ oder $F - Ulm - Mch3$.

Die beiden Kanten $F - La$ und $F - Ulm$ unterscheiden sich nur unwesentlich in der Länge. Bezüglich der Abweichung von der Verfügbarkeit ist die Kante $F - La$ genauso gut wie die Kante $F - Ulm$. Die Kante $F - La$ hat eine Abweichung von 0, die Kante $F - Ulm$ eine Abweichung von 20.

Bezüglich der Verfügbarkeit ist die Kante $F - Ulm$ leicht besser als $F - La$, denn es gilt je höher die Verfügbarkeit um so eher wähle die Kante.

Ein weiterer Unterschied für die Bewertung der beiden Kanten liegt an der unterschiedlichen Bitfehlerhäufigkeit. Zwar ist die Bitfehlerhäufigkeit der Kante $F - La$ kleiner als die der Kante $F - Ulm$, daher ist $F - La$ hier besser.

Aber die Abweichung der Kante $F - Ulm$ ist 25 und die der Kante $F - La$ ist 28.

Die Experten würden sich also für $F - Ulm - Mch3$ entscheiden. Denn ihre Anforderung ist den Wert für die Bitfehlerhäufigkeit so nahe wie möglich am Grenzwert zu wählen, aber im positiven Bereich und in eine Toleranz von 20 zu zulassen.

Fuzzy-Routing entscheidet genauso:

Die unterschiedliche Abweichung von der Mindestverfügbarkeit wird gleich beurteilt. Denn nur die Zugehörigkeitsfunktion von deltaAvail_Good liefert für die beiden Kanten einen von Null verschiedenen Wert, und zwar für beide Kanten $F - Ulm$ und $F - La$ den Wert 1.

Für die Abweichung von der maximal erlaubten Bitfehlerhäufigkeit gilt die Regel:

IF $\text{deltaBitErr} = \text{deltaBitErr_ecoBad}$ **THEN** $\text{edgeVal} = \text{edgeVal_b}$

Die Zugehörigkeitsfunktion von $\text{deltaBitErr_ecoBad}$ ist in Abbildung 6.4 dargestellt. Daraus geht hervor, daß der Wert der Zugehörigkeitsfunktion von $\text{deltaBitErr_ecoBad}$ für die Kante $F - Ulm$ niedriger ist als der Wert der Zugehörigkeitsfunktion von $\text{deltaBitErr_ecoBad}$ für die Kante $F - La$. Also ist der Erfüllungsgrad der oben genannten Regel für die Kante $F - La$ größer.

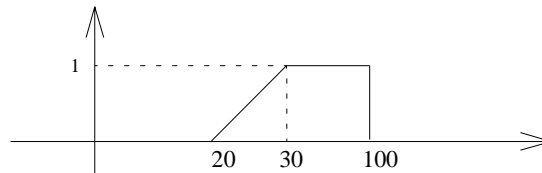


Abb. 6.4 Zugehörigkeitsfunktion des linguistischen Wertes $\text{deltaBitErr_ecoBad}$

Alle diese Bewertungen zusammen führen dazu, daß sich das Systems für $F - \text{Ulm} - \text{Mch3}$ entscheidet.

Für die Kante $\text{Ulm} - \text{Mch3}$ gibt es nach der bisherigen Wahl keine Alternative mehr. Sie erfüllt zudem die geforderten Qualitätskriterien.

Alle Kanten der von Hop- und Length-Routing bestimmten Route halten die Qualitätskriterien nicht ein. Außerdem ist der verwendete Knoten Mch1 nicht fernschaltbar.

Da die Auslastung aller Kanten zu Beginn denselben Wert hat, ist dieser Parameter bei der Interpretation dieser beiden Beispiele nicht interessant.

6.2.2 Steuerung der Kantenwahl mittels strategischer Bedeutung

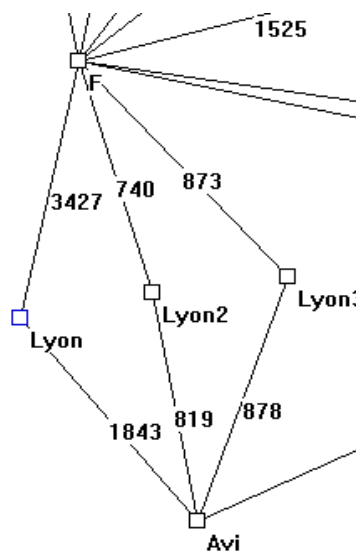


Abb. 6.5 Kanten zwischen F und Avi

Die Möglichkeiten der Feinsteuerung der Routenauswahl werden im nachfolgenden Text am Beispiel eines neu ins Layer-Netzwerk aufgenommenen Knoten demonstriert.

Die Feinsteuerung wird, wie bereits in 6.1.2.2 „Regelsystem“ erläutert, mit Hilfe des Parameters stratMean realisiert.

Zunächst sei der Zustand des Layer-Netzwerks in dem betrachteten Bereich bei gleichmäßi-

ger Auslastung erläutert.

Zwischen den Knoten *F* und *Avi* existieren drei verschiedene Routen, wie in Abbildung 6.5 zu sehen ist. Sie führen über drei verschiedene Knoten, *Lyon*, *Lyon2* und *Lyon3*. *Lyon* ist ein nicht fernschaltbarer Knoten, während *Lyon2* und *Lyon3* fernschaltbar sind. Die Parameter der Kanten sind in Tabelle 6.7, die der Knoten in Tabelle 6.8 zu sehen.

	F-Lyon	F-Lyon2	F-Lyon3
strategische Bedeutung	5	5	5
physikalische Länge	165	152	184
Auslastung	40	40	40
Verfügbarkeit	40	80	80
Abweichung von der Mindestverfügbarkeit	-20	20	20
Bitfehlerhäufigkeit	80	20	20
Abweichung von der maximal erlaubten Bitfehlerhäufigkeit	-40	20	20
Kantenwert	34,27	7,4	8,73

Tab. 6.7 Kantenparameter und Fuzzy-Bewertung

	F	Lyon	Lyon2	Lyon3
Knotenart	1	0	1	1
Betriebskosten	30	80	50	50
Auslastung	40	40	40	40

Tab. 6.8 Knotenparameter

Die Kanten *F - Lyon* und *Lyon - Avi* haben eine niedrige Verfügbarkeit und eine hohe Bitfehlerhäufigkeit. Sie weichen von den geforderten Grenzwerten für diese Parameter ab, so daß bei ihrer Wahl die Kundenanforderung bezüglich der Qualität nicht erfüllt würde. *Lyon* ist außerdem nicht fernschaltbar. Die Route *F-Lyon-Avi* ist also für eine Verbindung eher abzulehnen.

Die Knoten *Lyon2* und *Lyon3* sind gleichwertig, ebenso die Kanten *F-Lyon2*, *F-Lyon3* und *Lyon2-Avi*, *Lyon3-Avi*. Sie unterscheiden sich nur in der Länge. Die Kantenwerte zeigen, daß das automatische Routing-Verfahren, wie erwartet, auch zu dieser Bewertung kommt. Die Kantenwerte der Kanten *F-Lyon2*, *F-Lyon3* und *Lyon2-Avi*, *Lyon3-Avi* weichen nur geringfügig voneinander ab. Der strategischen Bedeutung, also dem Parameter *stratMean* kommt bei dieser Bewertung keine besondere Aufgabe zu, da jede Kante den selben Wert für *stratMean* hat.

Nun sei eine leicht veränderte Situation betrachtet. Bei dem Knoten *Lyon2* handele es sich um einen neu aufgenommen Knoten. Die veränderte Situation zeigt sich in der veränderten

Auslastung der Kante F - $Lyon2$ und $Lyon2$ - Avi - siehe Tabelle 6.9. So entsteht auch eine veränderte Fuzzy-Bewertung, wie in Tabelle 6.9 ebenfalls gezeigt wird.

	F-Lyon2	F-Lyon3	Lyon2-Avi	Lyon3-Avi
strategische Bedeutung	5 / 10 / 10	5 / 5 / 1	5 / 10 / 10	5 / 5 / 1
physikalische Länge	152	184	146	163
Auslastung	1	40	1	40
Verfügbarkeit	80	80	80	80
Abweichung von der Mindestverfügbarkeit	20	20	20	20
Bitfehlerhäufigkeit	20	20	20	20
Abweichung von der maximal erlaubten Bitfehlerhäufigkeit	20	20	20	20
Kantenwert *	15,92 / 10,98 / 10,98	9,17 / 9,17 / 11,53	16,79 / 11,75 / 11,75	9,14 / 9,14 / 11,53

*) : Die drei Werte entsprechen den verschiedenen Werten von *stratMean*

Tab. 6.9 Kantenparameter und Fuzzy-Bewertung der Kanten nach Einbau von $Lyon2$ als neuen Knoten

Das hat zur Folge, daß für Routen nach Avi die Kanten $F-Lyon3$ und $Lyon3-Avi$ gewählt würden, denn deren Kantewerte sind die kleinsten. Gewünscht ist aber, daß der Knoten $Lyon2$ in der Routensuche als gleichwertiger Knoten zu $Lyon3$ verwendet wird, obwohl er nach den Regeln als schlechter eingestuft ist. Mit der Zeit soll eine gleichmäßige Auslastung dieser Knoten erzeugt werden.

Um dies zu erreichen, wird der Wert für *stratMean* der Kanten $F-Lyon2$ und $Lyon2-Avi$ heraufgesetzt und der Wert für *stratMean* der Kanten $F-Lyon3$ und $Lyon3-Avi$ heruntergesetzt. Die Veränderung der Kantewerte, abhängig von der Veränderung von *stratMean*, sind für die Kanten in Tabelle 6.9 dargestellt. Durch die Veränderung der Werte von *stratMean* werden die Kanten für das Fuzzy-System also quasi gleichwertig. Wenn genügend Routen geschaltet wurden und die Auslastung der Kanten $F-Lyon3$ und $F-Lyon2$ wieder ähnlich ist, so werden die Werte von *stratMean* zurückgesetzt.

Der Wert von *stratMean* wird nicht vom System gesetzt, sondern vom Benutzer. Es handelt sich um eine zeitlich begrenzte und auf nur die Kante, für die der Wert von *stratMean* verändert wird, bezogene Maßnahme. Mit Hilfe dieses Parameters wird in dem oben aufgeführten Beispiel die schlechtere Bewertung der Kante, die die Ursache in ihrer geringen Auslastung hat, aufgehoben. Die Regeln des Fuzzy-Systems werden so speziell für diese Kante scheinbar verändert, während für alle anderen Kanten die ursprünglichen gelten. So hat der Benutzer die Möglichkeit, lokal und gezielt die Routenauswahl zu steuern.

6.2.3 Auswirkung auf das Layer-Netzwerk bei längerem Einsatz

Die oben aufgeführten Beispiele machen deutlich, daß die einzelnen Routen von Fuzzy-Routing auf die gewünschte Weise bestimmt werden. Nun sollen die Auswirkungen von Fuzzy-Routing auf das gesamte Layer-Netzwerk bei einem längerem Einsatz des Verfahrens betrachtet werden.

Der Einsatz wurde durch die Bearbeitung von 400 Routenwünschen simuliert. Außer Fuzzy-Routing wurden auch Hop-Routing und Length-Routing eingesetzt, um die Ergebnisse zu vergleichen. Nachdem alle Routen bestimmt waren, wurden die folgenden Kriterien untersucht:

1. Auslastung der Knoten und Kanten des Layer-Netzwerks
2. Kosten der in den Routen verwendeten Knoten im Vergleich zu den Kosten der zur Verfügung stehenden Knoten
3. Bitfehlerhäufigkeit und Verfügbarkeit der in den Routen verwendeten Kanten
4. Verteilung der Bitfehlerhäufigkeit und Verfügbarkeit der verwendeten Kanten

1. Auslastung

Laut Fuzzy-Regelsystem sind Kanten zu bevorzugen, deren Auslastung den Wert $beGr_m$ hat - siehe hierzu auch 4.2 „Regelsystem“. Die Zugehörigkeitsfunktion für $beGr_m$ ist in Abbildung 6.6 dargestellt. Der bevorzugte Wertebereich liegt zwischen 40 und 60. Kanten, deren Belegungsgrad größer als 80 ist, sind abzulehnen.

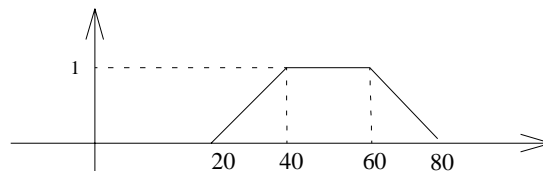


Abb. 6.6 Zugehörigkeitsfunktion von $beGr_m$

Nach dem Einsatz haben die Kanten die in Abbildung 6.7 dargestellten Auslastungswerte.

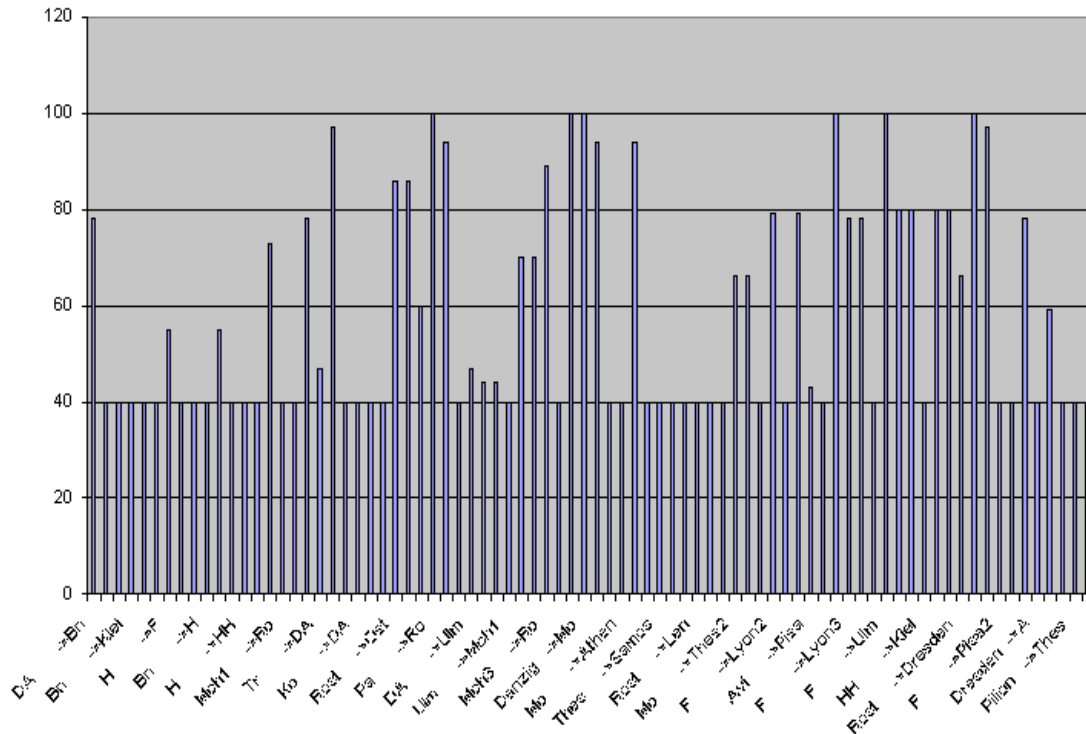


Abb. 6.7 Auslastung der Kanten nach dem Einsatz von Fuzzy-Routing

Die Auslastungswerte von 14 Kanten überschreiten die 80%-Grenze, davon sind 6 zu 100% ausgelastet. Das hat mehrere Ursachen:

Wenn eine Route mehrmals gefunden und so auch für eine Schaltung verwendet wird, so sind nach einer gewissen Verwendungsanzahl die Auslastungswerte ihrer Kanten und Knoten so, daß eine alternative Route verwendet werden sollte. Das kann nur geschehen, wenn eine solche Route existiert. Besteht jedoch diese alternative Route aus deutlich mehr Kanten als die ursprüngliche, so können zwar die einzelnen Kanten kleinere Kantenwerte haben als die Kanten der ursprünglichen Route, aber die Summe ihrer Kantenwerte übersteigt trotzdem die Summe der Kantenwerte der Ursprungsrouten. So wird trotz hoher Auslastung die erste Route weiter verwendet.

Außerdem sind die Regeln so formuliert, daß die Auslastung nur einer von vielen zu beachtenden Parametern ist.

Für eine konkrete Beschreibung der Ursachen, wird im folgenden die Überschreitung der gewünschten Auslastungswerte an zwei Kanten analysiert:

Die Liste der Routenwünsche besteht aus fünf verschiedenen Quelle - Senke Paaren, die je achtmal wiederholt werden. Die fünf einzelnen Routenwünsche sind:

HH - Mo

Tr - Ro

Ulm - Avi

F - Pa

Dresden - Samos

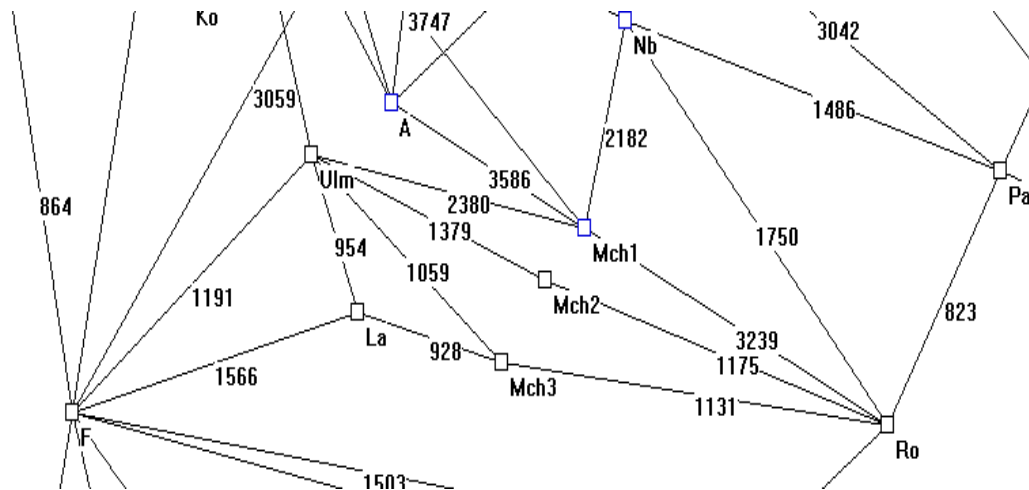


Abb. 6.8 Alternative Route für $Ro - Pa$

Betrachtet sei die Kante $Ro - Pa$, die während dieses Einsatzes zu 100% ausgelastet wird. In Abbildung 6.8 ist der für dieses Beispiel signifikante Ausschnitt des Graphen gezeigt. Um von F nach Pa zu gelangen, wird die Route über $Ulm - Mch2/Mch3 - Ro - Pa$ bevorzugt. Als alternative Route bietet sich $F - Ulm - Mch1 - Nb - Pa$ an: die ist jedoch eher abzulehnen, da sowohl $Mch1$ als auch Nb nicht fernschaltbare Knoten sind. Also kommt die Kante $Ro - Pa$ in der bevorzugten Route vor. Diese Kante kann aber in dieser Konstellation nur durch $Ro - Nb - Pa$ ersetzt werden. Die Summe der Kantenwerte dieser beiden Kanten ist aber sehr viel höher als die der einzelnen Kante $Ro - Pa$, auch wenn der Wert für deren Auslastung schon so hoch ist, daß sie abzulehnen ist.

Die Kante $Rost - Danzig$ kann ebenfalls nur durch eine Kantenfolge aus zwei Kanten ersetzt werden, $Rost - Len - Danzig$ wie in Abbildung 6.3 zu sehen ist. So ergibt sich als Summe der Kantenwerte für die Umgehung ein höherer Wert als der Kantenwert für die Kante $Rost - Danzig$. Diese Kante wird also gewählt. Daher kommt es auch in diesem Fall zu einer Auslastung von 100%.

Vergleicht man die Auslastung der Knoten und Kanten nach dem simulierten Einsatz mit Fuzzy-Routing, mit der Auslastung der Kanten und Knoten, die nach einem Einsatz mit Length-Routing bzw. Hop-Routing entsteht, so wird klar, daß Fuzzy-Routing das weitaus bessere Auslastungsverhalten liefert. Nach dem Einsatz mit Length-Routing sind 17 Kanten zu 100% ausgelastet. Es werden erst dann andere Kanten verwendet, wenn die physikalisch kürzesten nicht mehr zur Verfügung stehen, da sie zu 100% belegt sind. Nach dem Einsatz mit Hop-Routing sind 15 Kanten über 80% ausgelastet, davon sind 14% zu 100% belegt. Siehe hierzu Abbildung C.1, „Auslastung der Kanten nach Einsatz von Length-Routing“ und Abbildung C.2, „Auslastung der Kanten nach Einsatz von Hops-Routing“ in ANHANG C.

Derselbe Einsatz wurde noch einmal simuliert, jedoch wurden die Gewichte der Regeln für die Auslastung der Kanten auf 1 gesetzt, die der übrigen Regeln auf 0.5 belassen. Die Auslastung der Kanten stellt sich wie in Abbildung 6.9 gezeigt dar.

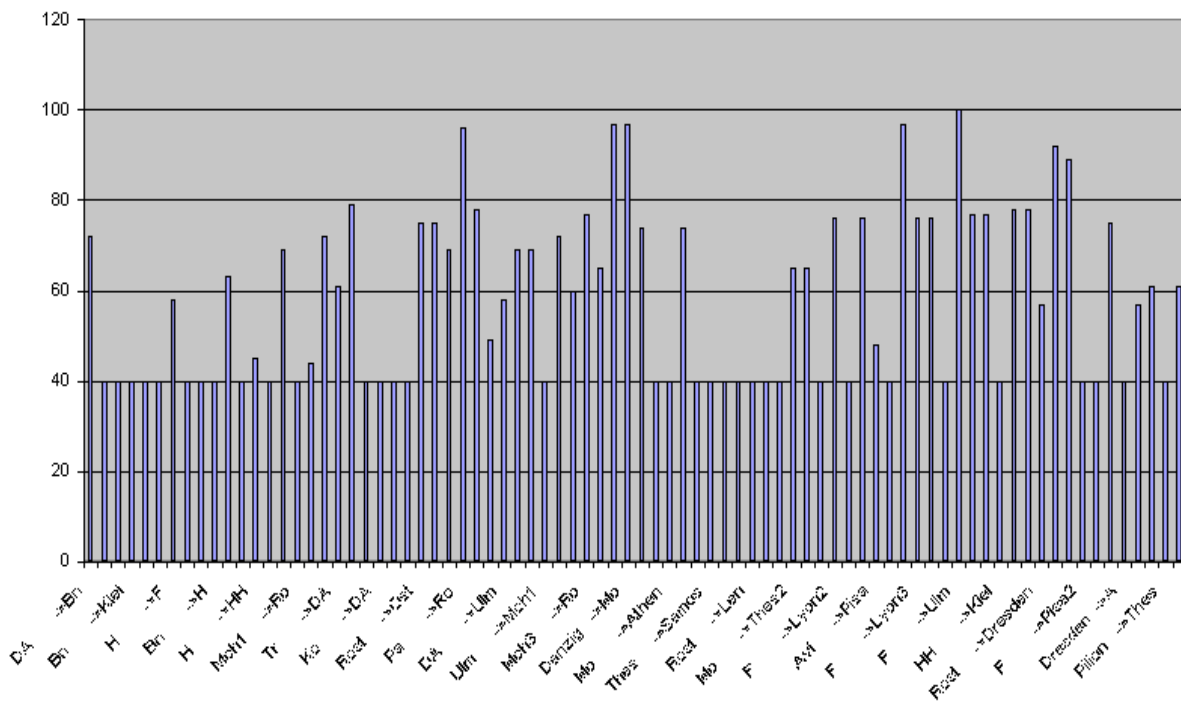


Abb. 6.9 Auslastung der Kanten nach einem Einsatz mit Fuzzy-Routing; Auslastungsregeln doppelt gewichtet

Wird also mehr Gewicht auf die Auslastungsregeln gelegt, so ist nur noch eine einzige Kante zu 100% ausgelastet und nur 7 haben die Grenze von 80% überschritten. Eine solche Verbesserung kann auch bei der Auslastung der Knoten beobachtet werden, wie in Abbildung C.3 und Abbildung C.4 in ANHANG C zu sehen ist.

Die Werte der Auslastung für Kanten und Knoten hängen natürlich davon ab, wieviele Verbindungen im Layer-Netzwerk geschaltet werden. In dem hier vorgestellten Testlauf sind 400 Routen für Verbindungen geschaltet worden. Die Auslastungswerte verändern sich mit jeder geschalteten Verbindung. Es liegt eine zeitliche Veränderung des Layer-Netzwerkes vor, wobei die Zeit durch die Anzahl der geschalteten Verbindungen gemessen wird. Um ein Bild dieser zeitlich veränderten Auslastung zu zeichnen, sind in Abbildung C.5 und Abbildung C.6 in ANHANG C Diagramme für die Auslastung der Kanten nach jeweils 200 und 300 Schaltungen abgebildet. Hier wird deutlich, daß die Regeln für die Auslastung eingehalten werden, solange das im Layer-Netzwerk möglich ist. Die Kanten werden nur bis zu einer gewissen Grenze „vollgeschaltet“. Für die nächste zu schaltende Route wird eine alternative Kante gewählt, wenn eine solche vorhanden ist. Die Grenze hängt davon ab, ob eine alternative Kante existiert und welche Parameter diese hat.

Weitere Ergebnisse zur zeitlichen Veränderung der Kantenwahl sind in 6.2.4 dargestellt.

2. Kosten der Knoten

Nach dem simulierten Einsatz zeigt die Auswertung der verwendeten Knoten, daß bevorzugt keine teuren Knoten verwendet wurden. Dies macht Abbildung 6.10 deutlich. In ihr sind die Knoten des Graphen und die Werte ihrer Betriebskosten dargestellt. Jeder Knoten hat über seinem Namen eine Säule proportional zur Höhe seiner Betriebskosten. Ist ein Knoten in

einer Route verwendet worden, so hat er zur Kennzeichnung noch eine zweite dunklere Säule direkt rechts an die erste anschließend. Knoten, die mit zwei Säulen dargestellt sind, sind also die in mindestens einer Route verwendeten.

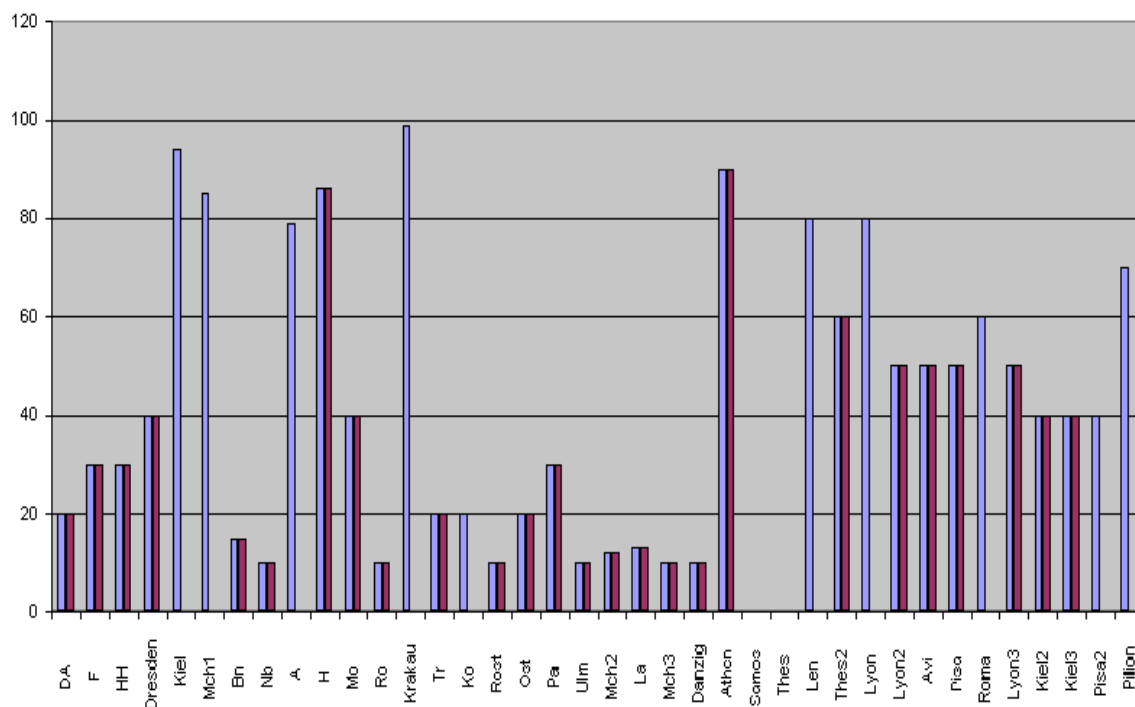


Abb. 6.10 Kosten der Knoten
Knoten, die mit zwei Säulen gekennzeichnet sind, sind Knoten, die in einer Route verwendet werden.

Die Ursachen für die Verwendung der drei teuersten, verwendeten Knoten *H*, *Athen* und *Thes2* werden im folgenden genauer betrachtet. In Abbildung 6.11 ist die Umgebung von *Athen* und *Thes2* dargestellt. Der Graph in der Umgebung von *H* ist in Abbildung 6.3 zu sehen.

Die Analyse ergibt, daß *Athen* in einer Route *Dresden* nach *Samos* verwendet wird. *Athen* ist ein fernschaltbarer Knoten, der aber für einen Knoten von diesem Typ extrem hohe Betriebskosten hat, nämlich 90. Die Kanten *Pa - Athen* und *Athen - Samos* sind von guter Qualität. Alle anderen in Frage kommenden Routen haben mehr Kanten als *Dresden - Pa - Athen - Samos*. Oder ihre Kanten sind schlechter wie zum Beispiel *Dresden - Krakau - Pilion - Samos*. Sowohl *Krakau* als auch *Pilion* zählen mit Knotenkosten von 99 beziehungsweise 70 zu den teureren Knoten.

Wenn die Auslastung der Kanten in *Dresden - Pa - Athen - Samos* keine weiteren Belegungen mehr zuläßt, dann werden Routen über *Thes2* verwendet.

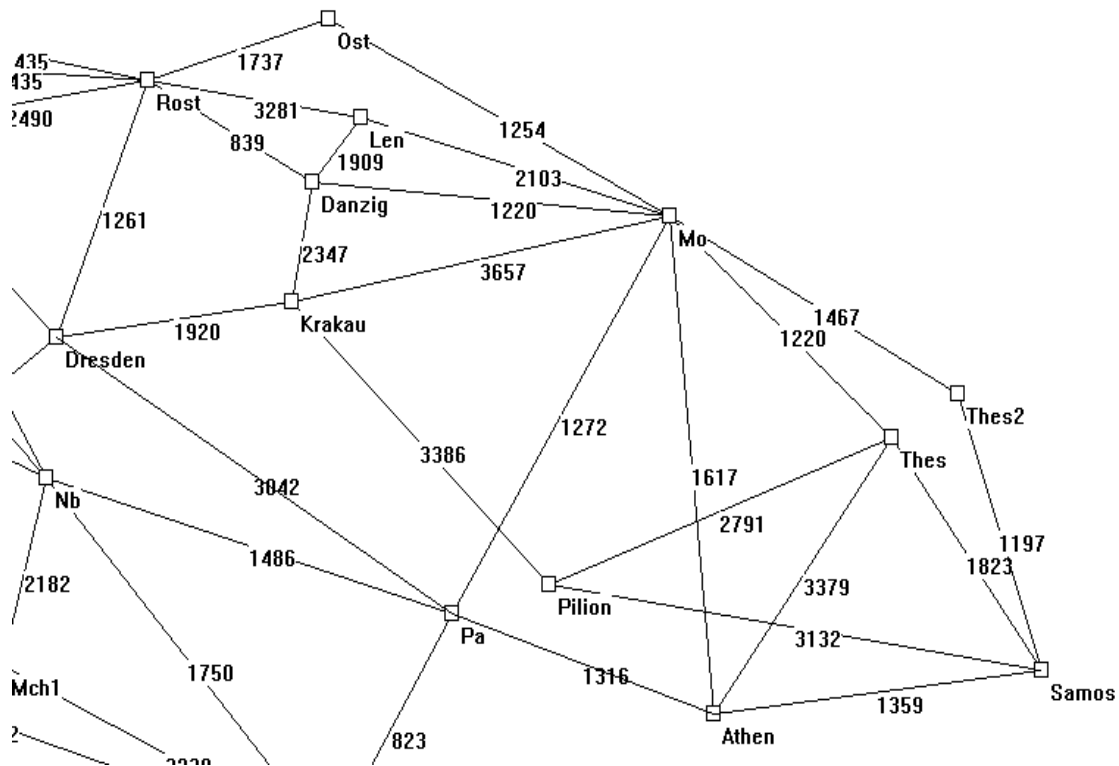


Abb. 6.11 Kanten um Athen und Thes2

Der Knoten H wird bei der Erfüllung des Routenwunsches $F - Pa$ verwendet. Dieses Beispiel zeigt auch nochmals, daß die Anzahl der Kanten in der gewählten Route bei Fuzzy-Routing implizit mitbeachtet wird. Es gibt eine Route von F nach Pa mit nur drei Kanten, das ist $F - H - Nb - Pa$. Zu Beginn des Einsatzes sind die Kantenwerte der Kanten dieser Route sehr viel höher als die anderer möglicher Kanten. Daher reicht die geringere Anzahl der Kanten nicht aus, um die höhere Anzahl der Summanden bei den anderen Routen auszugleichen. Nach längerem Einsatz werden jedoch die Auslastungswerte und damit auch die Kantenwerte der zunächst bevorzugten Kanten höher. Die geringe Anzahl der Summanden in der Route $F - H - Nb - Pa$ führt dann zu ihrer Wahl, obwohl die Kanten andere Routen einzeln betrachtet besser wären. Solange die Kanten der Route, die aus mehr Kanten besteht, sehr viel besser sind als die Kanten der Route, die aus weniger Kanten besteht, wird die Route, die aus mehr Kanten besteht gewählt.

Es liegt also an der Struktur des Layer-Netzwerkes warum diese teuren Knoten doch zum Einsatz kommen.

In Abbildung C.7 und in Abbildung C.8 in ANHANG C sind zum Vergleich die Kosten der Knoten für Length- und Hop-Routing abgebildet. Beide Verfahren verwenden mehr teure Knoten als Fuzzy-Routing.

3. Bitfehlerhäufigkeit und Verfügbarkeit der verwendeten Kanten

Die maximal erlaubte Bitfehlerhäufigkeit ist mit 40, die Mindestverfügbarkeit mit 60 festgelegt. Bei Fuzzy-Routing werden diese Werte ausschließlich durch die Regeln gesteuert. Die Regeln für die Steuerung der Verletzung der Qualitätskriterien sind höher gewichtet als die übrigen, mit 0.9, da diese Kriterien möglichst nicht verletzt werden sollen. Trotzdem handelt

es sich um eine „weiche“ Steuerung. Es ist also nicht sicher, daß Routen gewählt werden, die die Qualitätskriterien einhalten.

Die nachfolgenden Diagramme zeigen die Werte dieser Parameter für die verwendeten Kanten.

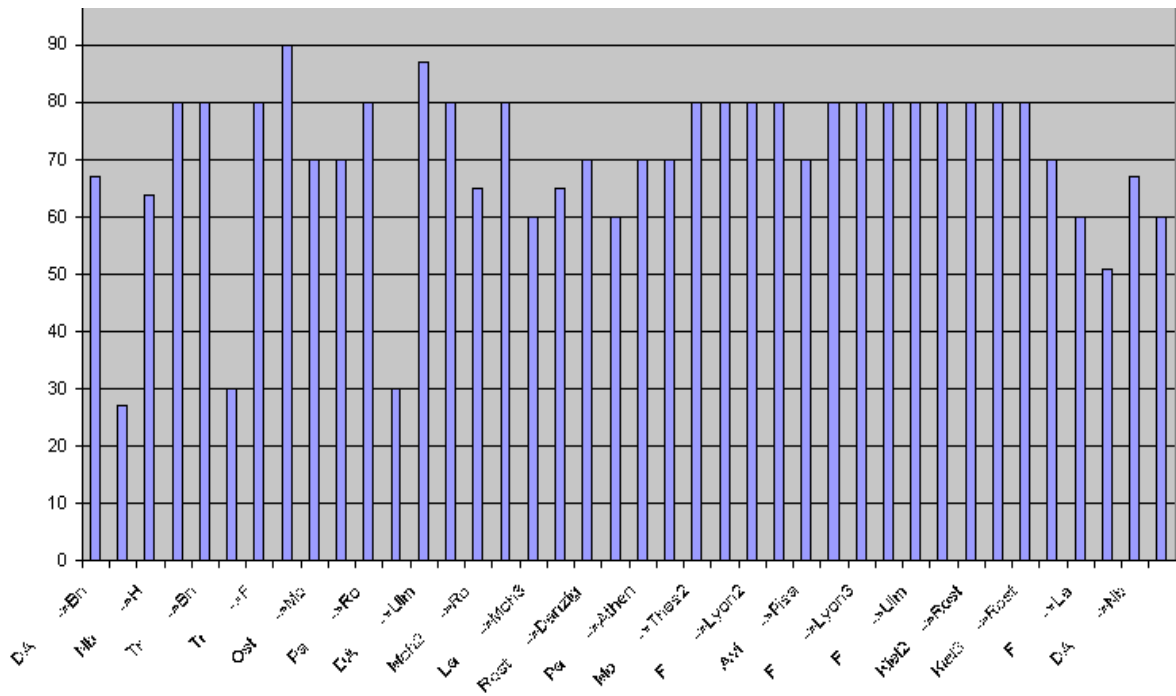


Abb. 6.12 Verfügbarkeit der verwendeten Kanten mit Fuzzy-Routing

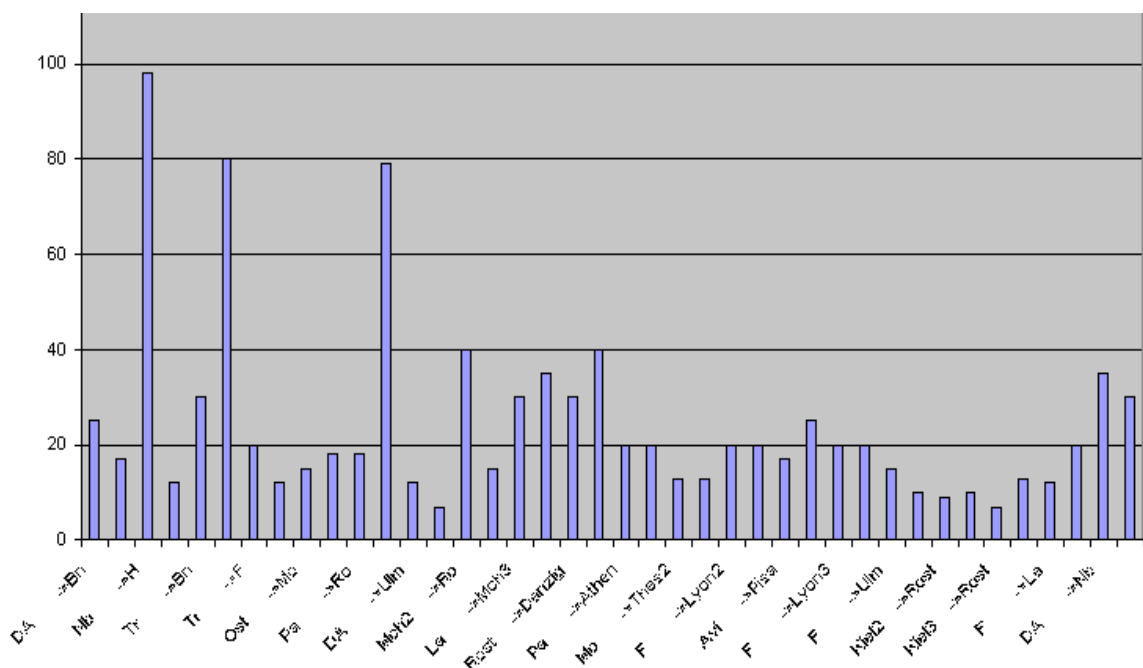


Abb. 6.13 Bitfehlerhäufigkeit der verwendeten Kanten mit Fuzzy-Routing

Nur vier der verwendeten Kanten unterschreiten die Mindestverfügbarkeit und nur drei überschreiten die maximal erlaubte Bitfehlerhäufigkeit. Es werden also hauptsächlich Kanten mit den gewünschten Werten verwendet, obwohl keine harte Prüfung auf diese Kriterien erfolgt.

Bei Length- und Hop-Routing wird auf die Parameter Bitfehlerhäufigkeit und Verfügbarkeit nicht geachtet. Bei Length-Routing unterschreiten neun der verwendeten Kanten die Mindestverfügbarkeit und acht überschreiten die maximal erlaubte Bitfehlerhäufigkeit. Bei Hop-Routing unterschreiten zwölf der verwendeten Kanten die Mindestverfügbarkeit und zehn überschreiten die maximal erlaubte Bitfehlerhäufigkeit. Die Diagramme für die Verfahren Length- und Hop-Routing sind in Abbildung C.9 bis Abbildung C.12 gezeigt.

4. Verteilung der Bitfehlerhäufigkeit und Verfügbarkeit

Wie in 6.1.2.2 dargelegt, sollen aus wirtschaftlichen Gründen keine Kanten gewählt werden, deren Werte für die Qualität zu gut sind. Durch die Regeln ist festgelegt, daß eine Abweichung von 0 bis 20 von den jeweiligen Grenzwerten zu bevorzugen ist. Für die Bitfehlerhäufigkeit wird daher ein Wert zwischen 20 und 40 bevorzugt, für die Verfügbarkeit ein Wert zwischen 60 und 80. Aus den beiden nachfolgenden Diagrammen geht hervor, daß vorwiegend Kanten mit diesen Werten gewählt wurden. Da keine harte Prüfung auf die Einhaltung der Grenzwerte durchgeführt wird, sind auch Kanten mit nicht erlaubten Werten für diese beiden Parameter verwendet worden, jedoch nur wenige. Diese Auswertung wurde auch für Length- und Hop-Routing gemacht, um sicherzustellen, daß die gute Verteilung dieser Werte für Fuzzy-Routing nicht von den Daten abhängt. In Tabelle 6.10 und Tabelle 6.11 ist die genaue Anzahl der Kanten aufgelistet, deren Parameterwerte für die Bitfehlerhäufigkeit und für die Verfügbarkeit jeweils innerhalb und außerhalb des bevorzugten Bereichs liegt.

Die Kanten in den mit *) gekennzeichneten Spalten erfüllen die Anforderung an die Qualität nicht. Die Kanten in den mit **) gekennzeichneten Spalten haben zu gute Werte für die Verfügbarkeit und die Bitfehlerhäufigkeit.

Die Werte in diesen beiden Tabellen zeigen deutlich die Vorteile von Fuzzy-Routing gegenüber Length- und Hop-Routing.

Routing-Verfahren	Abweichung von der maximal erlaubten Bitfehlerhäufigkeit ist negativ *)	Bitfehlerhäufigkeit liegt im bevorzugten Bereich	Abweichung von der maximal erlaubten Bitfehlerhäufigkeit ist positiv **)
Fuzzy-Routing	76	772	622
FuzzyPlus-Routing	103	788	630
Length-Routing	320	560	540
Hop-Routing	510	370	460

Tab. 6.10 Anzahl der verwendeten Kanten mit Werten für die Bitfehlerhäufigkeit im angegebenen Bereich

Routing-Verfahren	Abweichung von der Mindestverfügbarkeit ist negativ *)	Verfügbarkeit liegt im bevorzugten Bereich	Abweichung von der Mindestverfügbarkeit ist positiv **)
Fuzzy-Routing	133	1284	53
FuzzyPlus-Routing	113	1353	53
Length-Routing	380	1020	20
Hop-Routing	610	690	40

Tab. 6.11 Anzahl der verwendeten Kanten mit Werten für die Verfügbarkeit im angegebenen Bereich

In Abbildung 6.14 und Abbildung 6.15 ist die Verteilung der Kanten bezüglich der Parameter Verfügbarkeit und Bitfehlerhäufigkeit graphisch dargestellt. In Abbildung C.13, Abbildung C.14, Abbildung C.15 und Abbildung C.16 in ANHANG C sind die Auswertungen für Length- und Hop-Routing dargestellt.

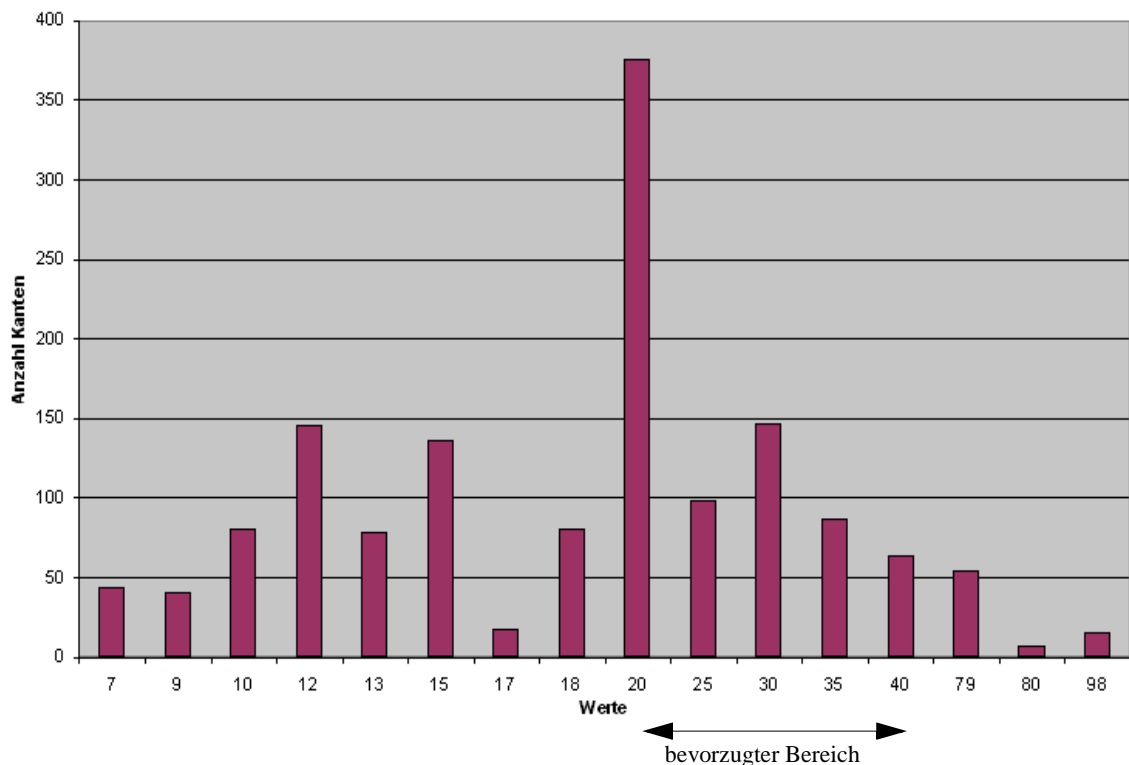


Abb. 6.14 Verteilung der Bitfehlerhäufigkeit mit Fuzzy-Routing

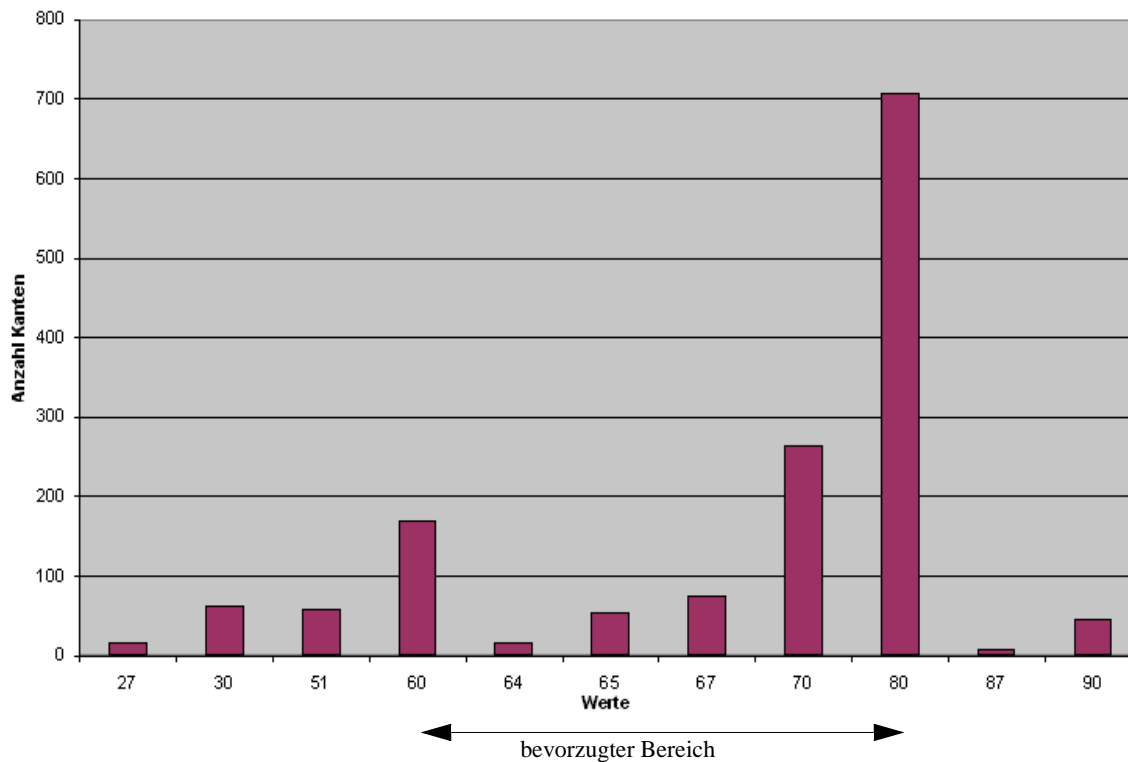


Abb. 6.15 Verteilung der Verfügbarkeit mit Fuzzy-Routing

6.2.4 Zeitliche Veränderung der Kantenwahl

Während des Einsatzes ändert sich das Layer-Netzwerk mit jedem geschalteten Übertragungsweg, da sich die Auslastung der Kanten und Knoten verändert. Es liegt also, wie schon oben erwähnt, eine zeitliche Veränderung des Layer-Netzwerkes vor. Die Kantenwerte ändern sich mit der steigenden Auslastung der Kanten und Knoten. Manche Kanten sind nach einer gewissen Zeit zu 100% ausgelastet und stehen also nicht mehr zur Verfügung.

Um ein Bild dieser zeitlichen Veränderung zu zeichnen, sind solange Routen geschaltet worden, bis eine Schaltung dieser Routen wegen 100% Auslastung aller dafür möglicher Kanten unmöglich war. Es wurden dieselben Routenwünsche verwendet wie bei den vorangegangenen Tests.

Alle fünfzig Schaltungen wurden die folgenden Werte mitgeschrieben:

- Auslastung aller Kanten des Layer-Netzwerkes,
- Anzahl der Kanten, die zu 100% ausgelastet sind,
- Anzahl der Kanten, deren Auslastung kleiner 100% und größer gleich 80% ist,
- Anzahl der Kanten, deren Auslastung kleiner als 80% und größer gleich 60% ist,
- Anzahl der verwendeten Kanten, deren Verfügbarkeit eine negative Abweichung von der Mindestverfügbarkeit hat (0 - 60, schlecht),
- Anzahl der verwendeten Kanten, deren Verfügbarkeit im bevorzugten Bereich liegt (60 - 80, gut),

- Anzahl der Kanten, deren Verfügbarkeit eine positive Abweichung von der Mindestverfügbarkeit hat (> 80 , wirtschaftlich schlecht),
- Anzahl der Kanten, deren Bitfehlerhäufigkeit eine negative Abweichung von der maximal erlaubten hat (> 40 , schlecht),
- Anzahl der Kanten, deren Bitfehlerhäufigkeit im bevorzugten Bereich liegt (20 - 40, gut),
- Anzahl der Kanten, deren Bitfehlerhäufigkeit eine positive Abweichung von der maximal erlaubten hat (0 - 20, wirtschaftlich schlecht).

Dieser Test wurde für Fuzzy-Routing, Fuzzy-Routing mit doppelt gewichteten Auslastungsregeln, hier mit FuzzyAus-Routing bezeichnet, Length- und Hop-Routing durchgeführt. So konnten die verschiedenen Routing-Verfahren verglichen werden.

In den nachfolgenden Graphiken werden jeweils die Ergebnisse der oben genannten Routing-Verfahren im Vergleich dargestellt. Auf der x-Achse sind jeweils die Anzahl der Schaltungen aufgetragen. Die Beschriftung der y-Achse ist jeweils den Abbildungstiteln zu entnehmen.

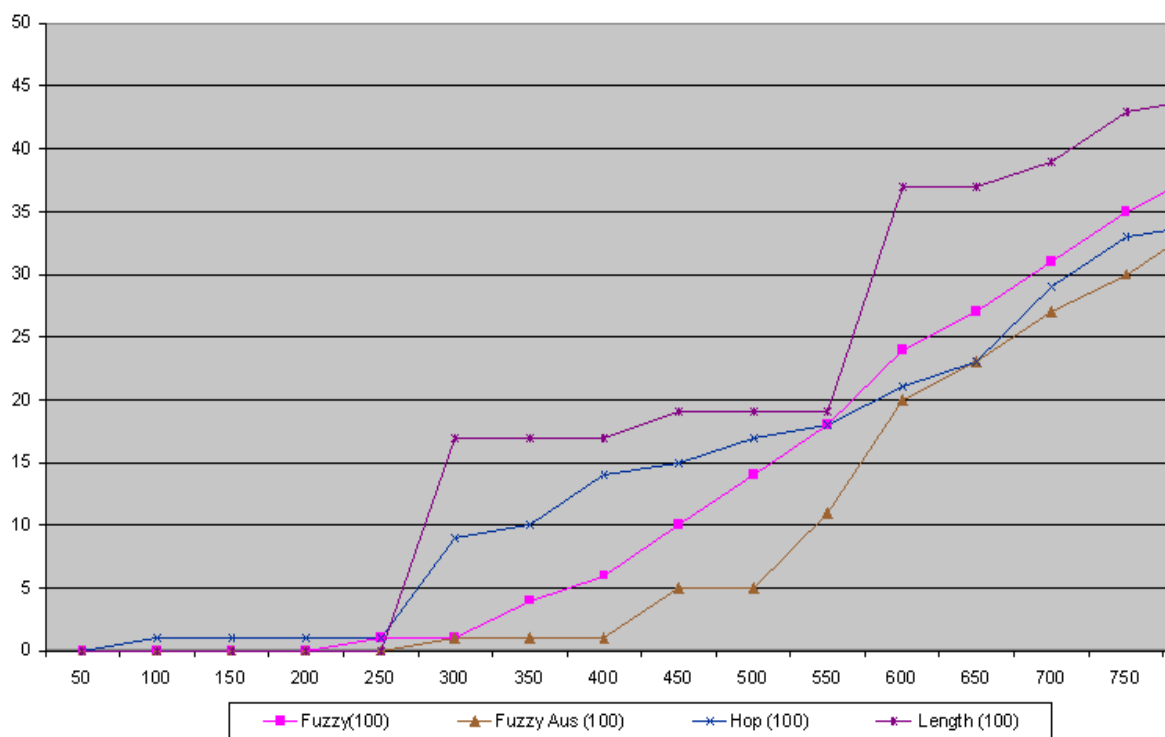


Abb. 6.16 Anzahl der Kanten mit 100% Auslastung

Abbildung 6.16 zeigt deutlich, daß Fuzzy-Routing bis 550 Schaltungen wesentlich weniger Kanten zu 100% auslastet als Hop- und Length-Routing. FuzzyAus-Routing hat bis zum Ende des Tests die wenigsten Kanten zu 100 Prozent gefüllt.

Nach 250 Schaltungen ist erst eine einzige Kante von Fuzzy-Routing ganz zu 100 Prozent ausgelastet worden, während Hop-Routing schon 9 und Length-Routing nach 300 Schaltungen bereits 17 Kanten zu 100 Prozent voll geschaltet hat.

Es ist zu klären, warum Fuzzy-Routing Kanten zu 100 Prozent voll schaltet, obwohl das den Regeln widerspricht. Dazu werden die Daten aus der Tabelle C.1 in ANHANG C herangezogen. Dort ist die prozentuale Auslastung jeder Kante nach jeder fünfzigsten Schaltung dargestellt.

Die Kante, die nach 250 Schaltungen von Fuzzy-Routing zu 100 Prozent voll geschaltet wurde, ist die Kante *F - Ulm*. Für den Routenwunsch von *F* nach *Pa* ist sie Teil der besten Route, wie schon in 6.2.3 unter „1. Auslastung“ erklärt wurde. Eine Ausweichroute für die Kante *F - Ulm* kann nur über zwei Kanten gebildet werden - siehe Abbildung 6.8. Außerdem wird die Kante *F - Ulm* auch noch für die Route von *Ulm* nach *Avi* verwendet.

Nach 350 Schaltungen sind drei weitere Routen bei Einsatz von Fuzzy-Routing zu hundert Prozent voll:

- *Rost - Danzig*
- *Danzig - Mo*
- *Pa - Ro*

Die Ursachen für die Vollschtung der Kanten *Rost - Danzig* und *Pa - Ro* sind schon in 6.2.3 unter „1. Auslastung“ erläutert. Für *Danzig - Mo* können ähnliche Argumente vorgebracht werden.

Wegen der stärker gewichteten Auslastungsregeln ist FuzzyAus-Routing leicht besser als Fuzzy-Routing, was in der Abbildung 6.16 auch deutlich zu sehen ist.

Die beiden nachfolgenden Abbildungen stellen die Anzahl der Kanten mit einer Auslastung im Intervall [80,100[beziehungsweise im Intervall [60,80[dar.

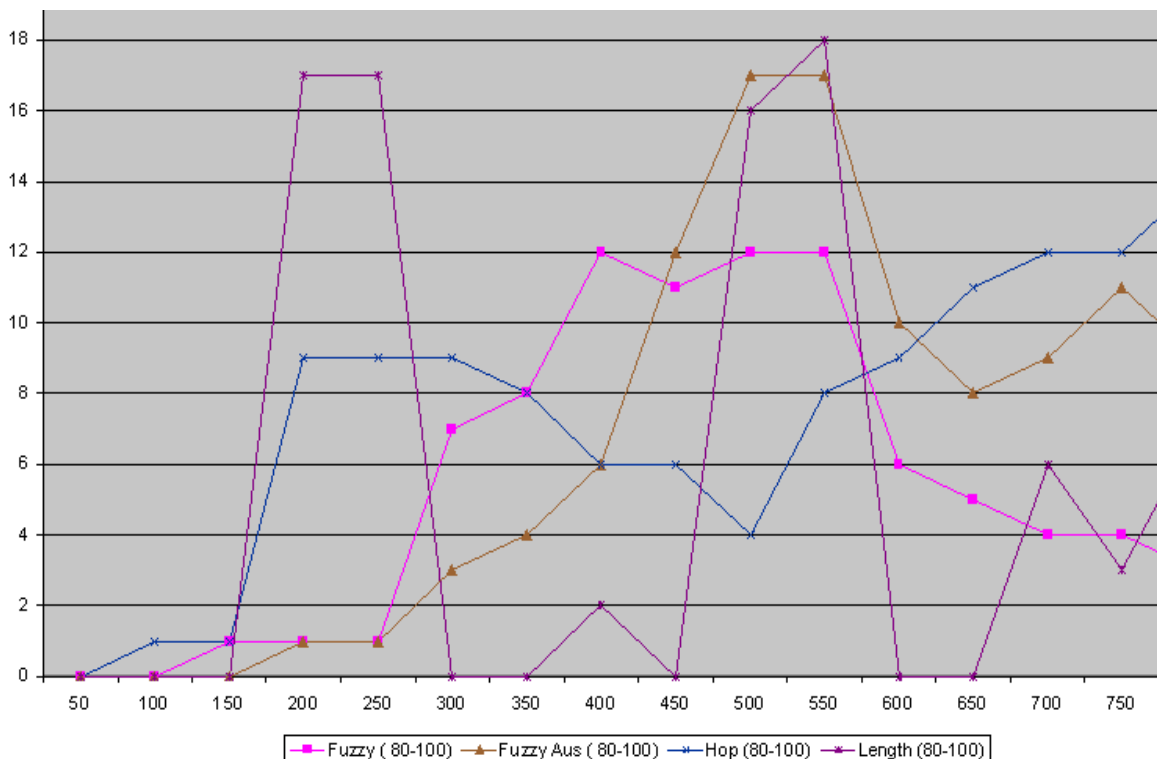


Abb. 6.17 Anzahl Kanten mit 80 - 100% Auslastung

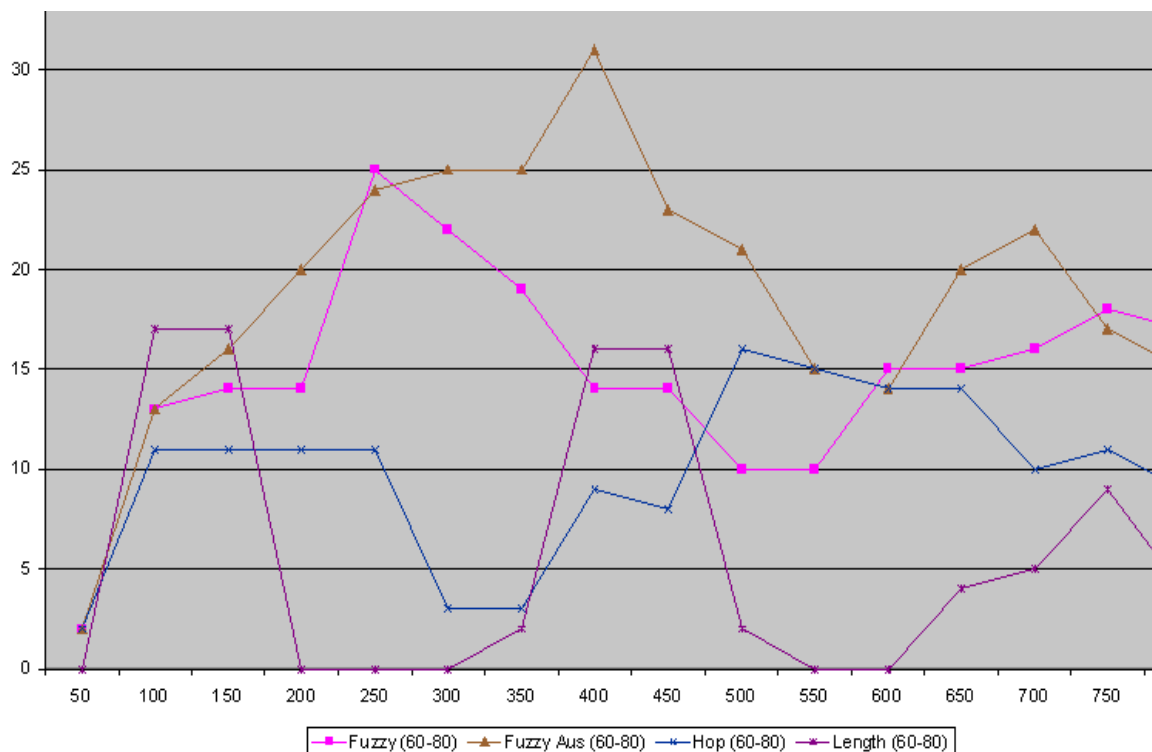


Abb. 6.18 Anzahl Kanten mit 60 - 80% Auslastung

In der Abbildung 6.17 und Abbildung 6.18 bedeutet eine negative Steigung, daß die entsprechende Anzahl von Kanten zu 100 beziehungsweise zu 80 Prozent voll geschaltet wurde und dem jeweils darüberliegenden Bereich zugeordnet wurde. Im Sinne der Regeln, die für die Routenauswahl gelten, ist es also gut, möglichst lange keine negative Steigung zu haben. Denn die Kanten sollen möglichst gleichmäßig ausgelastet werden.

Aus diesen drei Graphiken geht hervor, daß Fuzzy-Routing und FuzzyAus-Routing am längsten die Auslastungswerte und die dazugehörigen Regeln beachten. Length- und Hop-Routing schalten das Layer-Netzwerk sehr schnell „lokal“ voll.

Mit größer werdender Auslastung des Layer-Netzwerkes bezüglich der Kanten, die für die zu bestimmenden Routen in Frage kommen, wird es auch für Fuzzy-Routing und FuzzyAus-Routing schwerer, die Auslastungsregeln zu beachten. Ausweichrouten haben dann sehr viel mehr Kanten als die direkte Route, die aus fast vollen Kanten besteht. Die Summe der Kantenwerte der Ausweichroute ist daher höher, obwohl die Kantenwerte der einzelnen Kanten niedriger sind.

Durch eine hohe Gewichtung der Regeln, die die Auslastung betreffen, ist es möglich, den Zeitpunkt für die hohe Anzahl voller Kanten hinauszuzögern, wie Abbildung 6.16 zeigt. Wenn jedoch der Parameter Auslastung stärker beachtet wird, so heißt dies auch, daß die übrigen Parameter weniger stark beachtet werden. Das wird auch bei den nachfolgenden Auswertungen deutlich.

Fuzzy-Routing und FuzzyAus-Routing wählen wesentlich mehr Kanten, die im gewünschten Bereich der Verfügbarkeit liegen - siehe Abbildung 6.19.

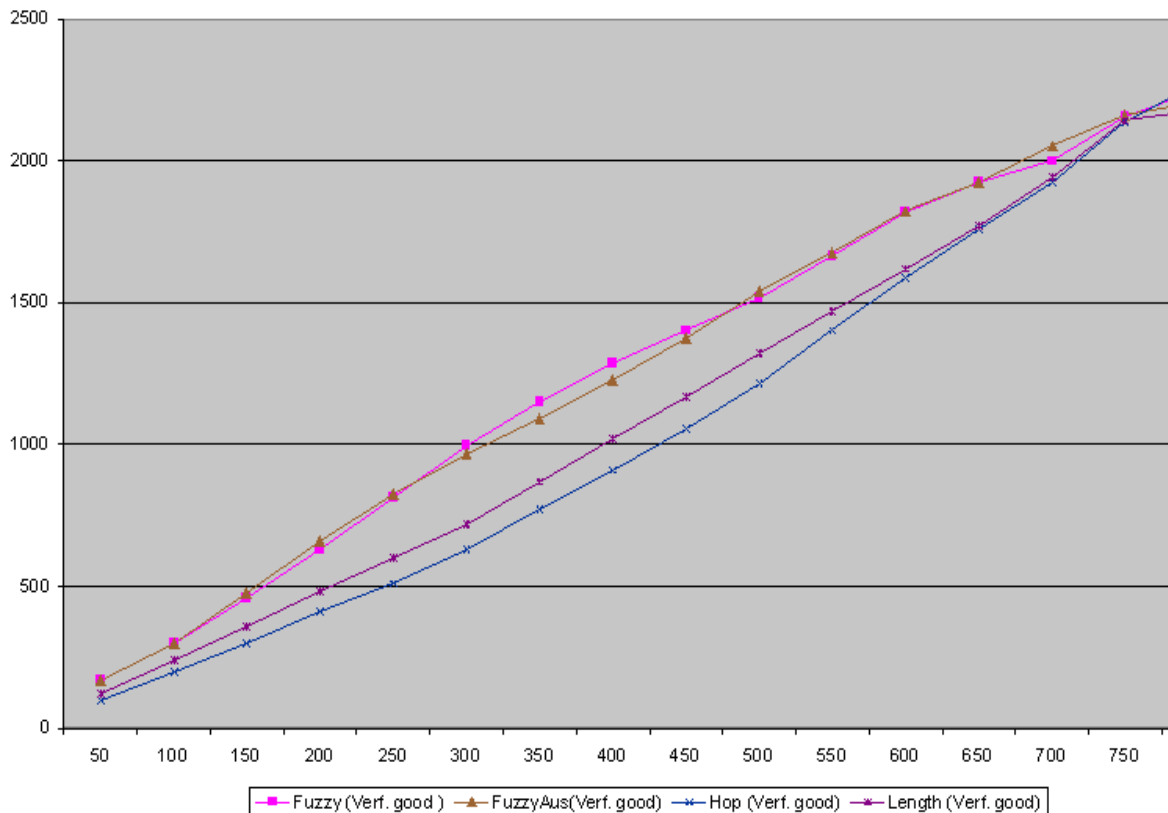


Abb. 6.19 Anzahl der verwendeten Kanten, deren Verfügbarkeitswerte im gewünschten Bereich liegen

Die Abbildung C.17 aus ANHANG C zeigt, daß Fuzzy-Routing und FuzzyAus-Routing wesentlich weniger Kanten wählen, die die Mindestverfügbarkeit unterschreiten. Die Abbildung C.18 stellt die verwendeten Kanten mit wirtschaftlich schlechten Verfügbarkeitswerten dar. Diese Abbildung für sich betrachtet, legt den Schluß nahe, daß Length- und Hop-Routing besser sind. Für die Bewertung der Routing-Verfahren bezüglich der Beachtung der Verfügbarkeitswerte müssen jedoch Abbildung 6.19, Abbildung C.17 und Abbildung C.18 zusammen beachtet werden. Dann wird folgendes deutlich:

Length- und Hop-Routing haben fast alle Kanten aus dem Bereich gewählt, in dem die Mindestverfügbarkeit unterschritten wird, und haben deswegen weniger Kanten als Fuzzy-Routing und FuzzyAus-Routing (Abbildung C.18).

Fuzzy-Routing beachtet die Regeln, die die Verfügbarkeit betreffen, leicht besser als FuzzyAus-Routing. Das ist auch nicht anders zu erwarten, denn die Auslastungsregeln werden bei FuzzyAus-Routing doppelt gewichtet und so sind die Regeln, die die Verfügbarkeit betreffen, hier weniger wichtig.

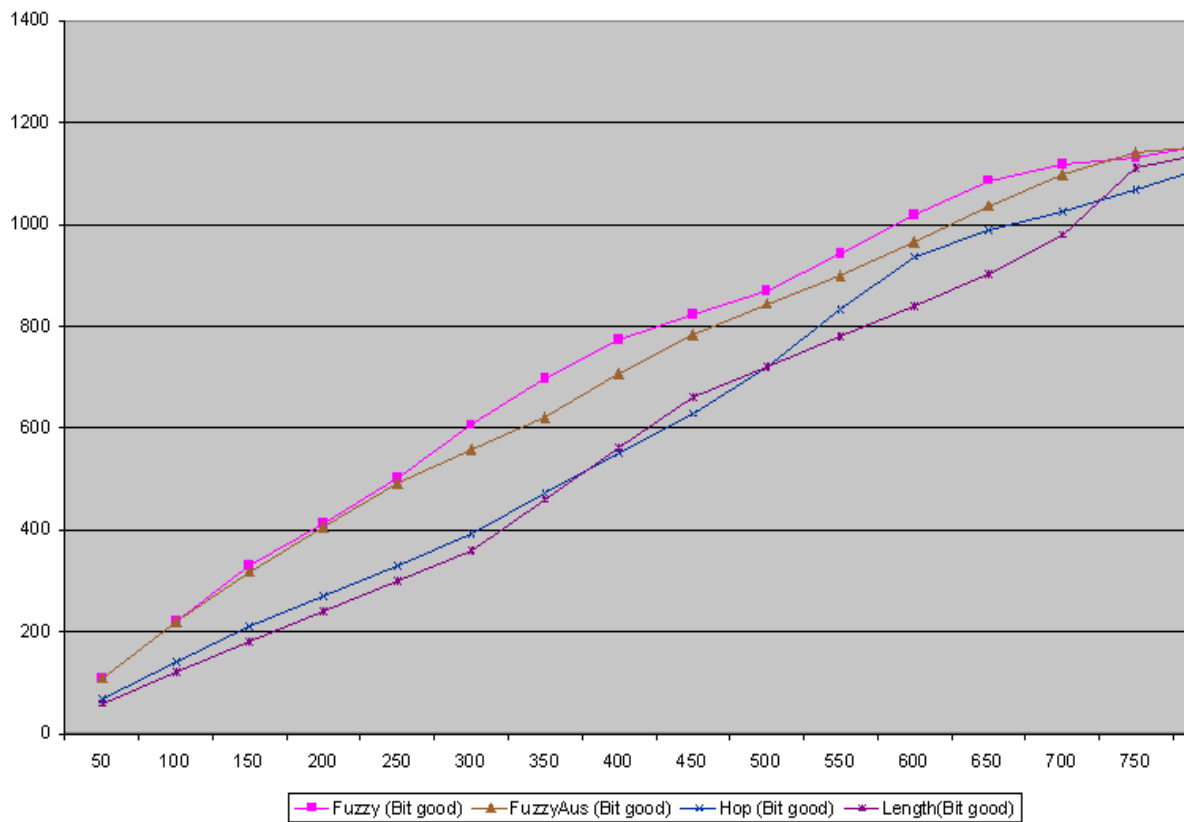


Abb. 6.20 Anzahl der Kanten, deren Werte für Bitfehlerhäufigkeit im gewünschten Bereich liegen

Bei der Auswahl der Kanten im Hinblick auf ihre Werte der Bitfehlerhäufigkeit verhalten sich die Routing-Verfahren genauso wie bei der Auswahl der Kanten im Hinblick auf ihre Verfügbarkeitswerte. Das wird in Abbildung 6.20, und Abbildung C.19 und Abbildung C.20 in ANHANG C deutlich.

6.3 FuzzyPlus-Routing

FuzzyPlus-Routing ist Fuzzy-Routing ergänzt um eine harte Kostenoptimierung, wie in 3.5.2 genau dargestellt ist. Um FuzzyPlus-Routing zu testen, wird für jede Kante mit Hilfe der in 3.5.2 erwähnten fiktiven Kostenfunktion ein zusätzlicher Parameter *edgeCost* erzeugt. *edgeCost* enthält für jede Kante den Werte der Kostenfunktion. Dieser Parameter wird in den Regeln nicht beachtet. Die Werte des Parameters *edgeCost* der Kanten des Layer-Netzwerkes sind in ANHANG B aufgelistet.

In dieser Simulation wurden aus den fünf besten, mit Fuzzy-Routing bestimmten Routen, die mit dem kleinsten Wert für *edgeCost* ausgewählt.

6.3.1 Auswirkungen auf das Layer-Netzwerk bei längerem Einsatz

1. Auslastungsverhalten

Das Auslastungsverhalten ist schlechter als bei Fuzzy-Routing, wie durch den Vergleich von Abbildung 6.21 mit Abbildung 6.7 und Abbildung 6.22 mit Abbildung C.3 festgestellt werden kann. 15 Kanten sind zu mehr als 80% ausgelastet, davon 8 zu 100%. Das war auch zu erwarten, da bei FuzzyPlus-Routing zusätzlich zur Regelbeachtung eine Kostenoptimierung stattfindet und so die Beachtung der Auslastung weniger wichtig wird.

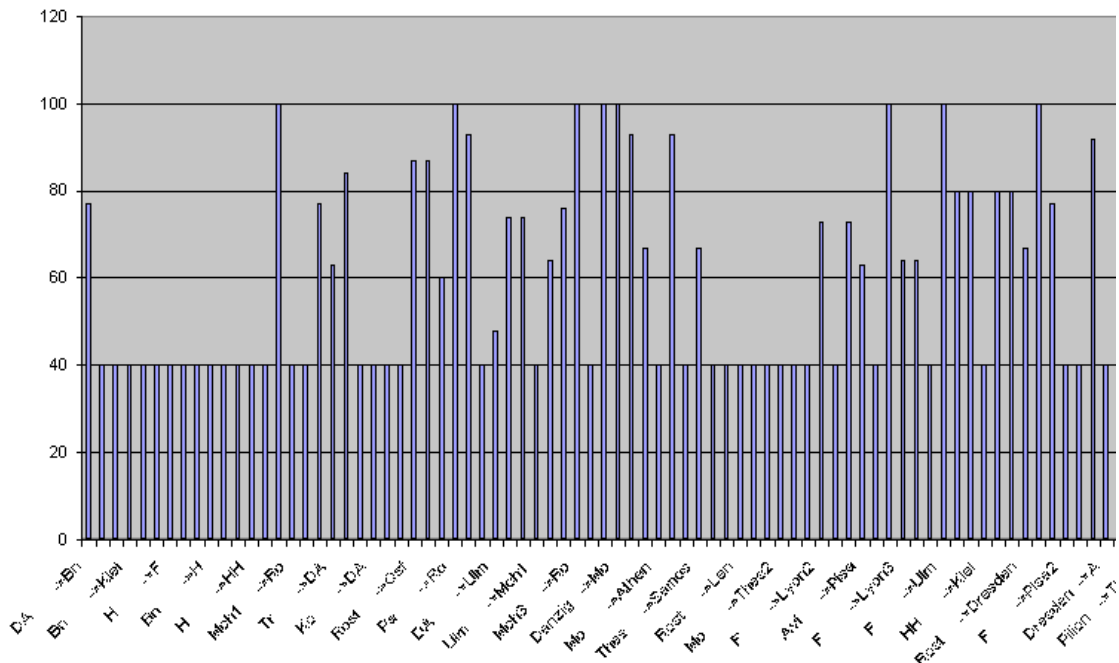


Abb. 6.21 Auslastung der Kanten mit FuzzyPlus-Routing

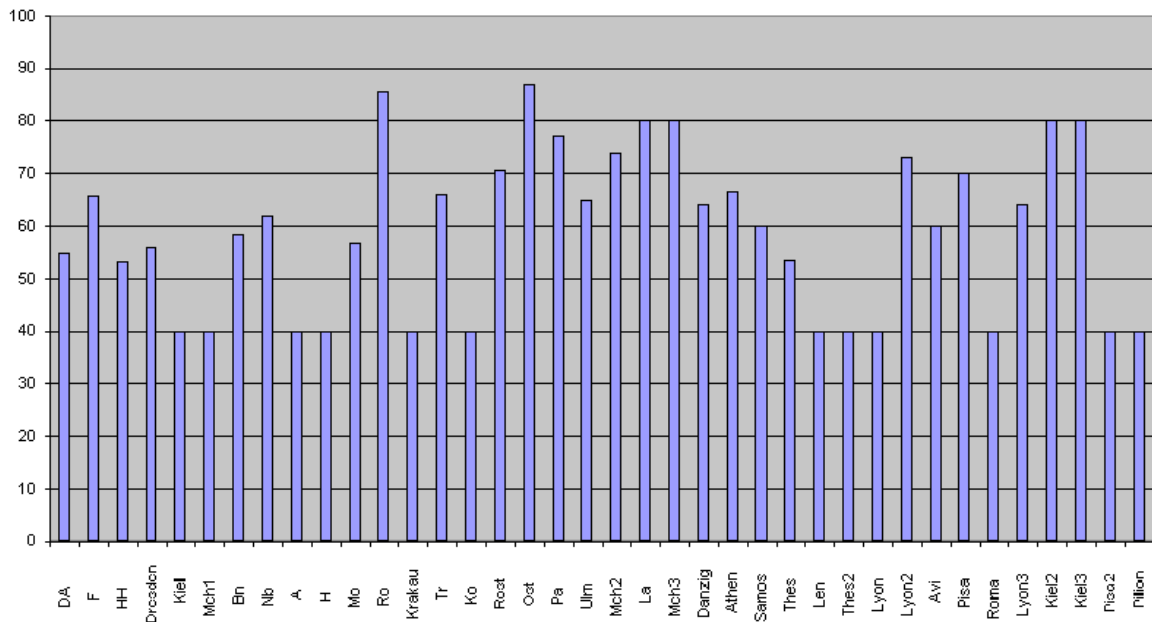


Abb. 6.22 Auslastung der Knoten mit FuzzyPlus-Routing

Derselbe Einsatz wurde noch einmal simuliert, jedoch wurden die Gewichte der Regeln für die Auslastung der Kanten auf 1 gesetzt, die der übrigen Regeln bei 0.5 belassen. Die Auslastung der Kanten ist in Abbildung 6.23 gezeigt.

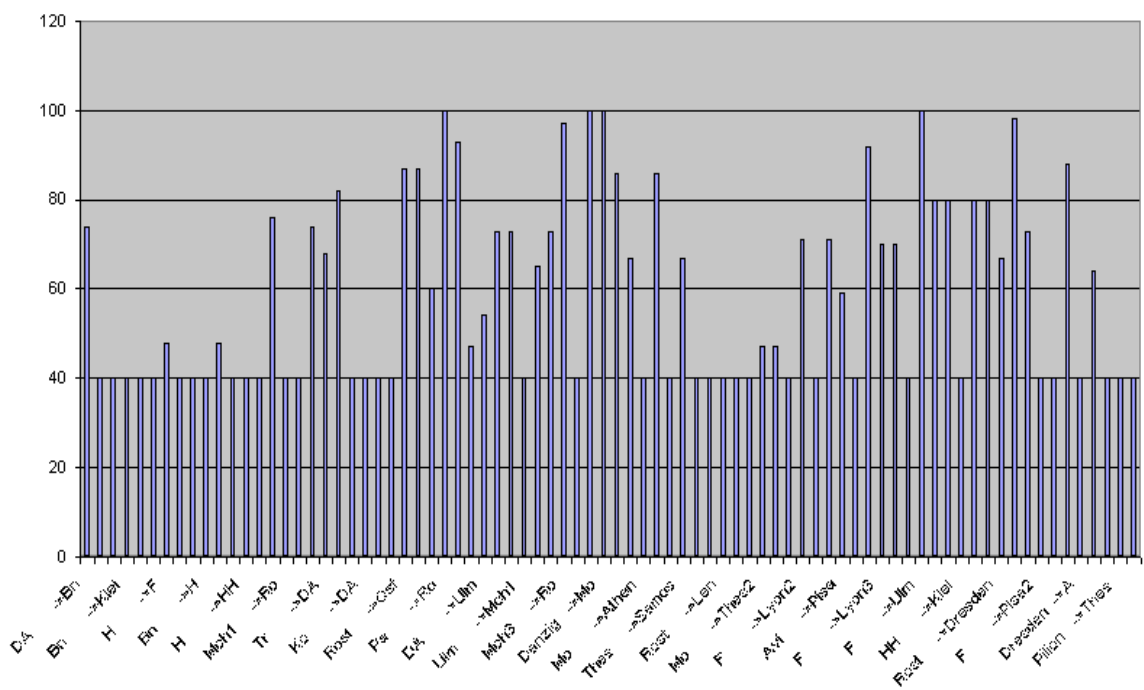


Abb. 6.23 Auslastung der Kanten mit FuzzyPlus-Routing mit doppelt gewichteten Auslastungsregeln

Ein Vergleich mit Abbildung 6.9 zeigt ein verschlechtertes Auslastungsverhalten gegenüber Fuzzy-Routing mit doppelter Gewichtung der Auslastungsregeln. Ein Vergleich mit Fuzzy-Routing, siehe Abbildung 6.7, zeigt deutlich weniger Unterschiede.

2. Kosten der Knoten

Dieses Verfahren verwendet erwartungsgemäß mehr Knoten mit geringeren Kosten als Fuzzy-Routing, wie der Vergleich von Abbildung 6.24 mit Abbildung 6.10 zeigt. Nur der teure Knoten Athen wird weiter verwendet, aus dem selben Grund wie in 6.2.3 unter „2. Kosten der Knoten“ dargelegt.

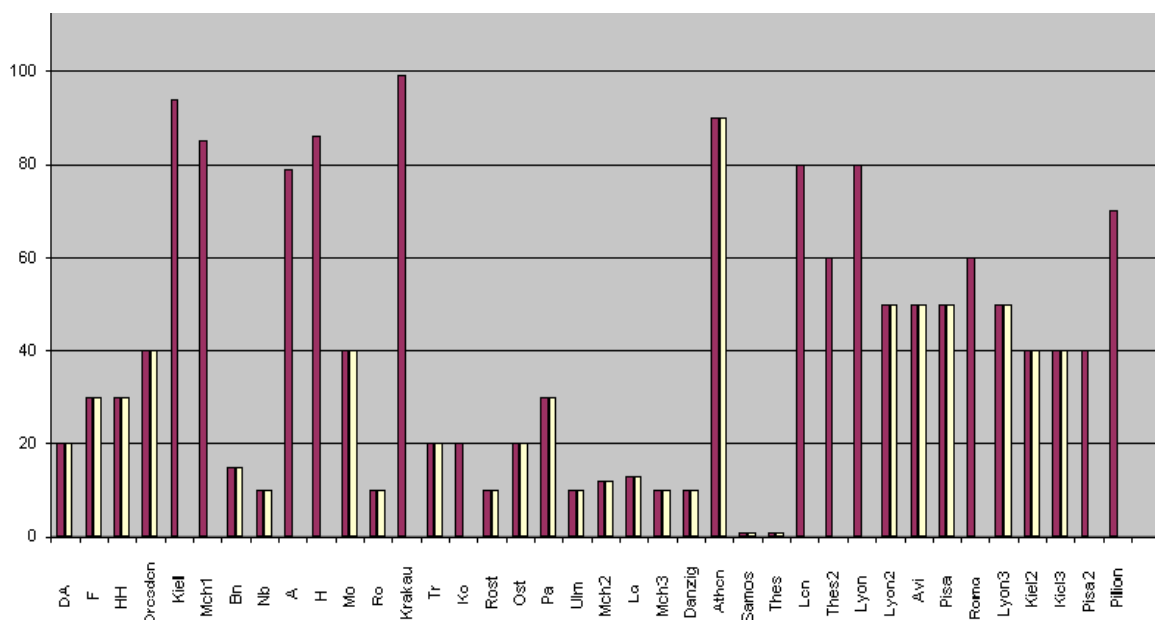


Abb. 6.24 Kosten der Knoten
Knoten mit zwei Säulen sind Knoten, die in einer Route verwendet werden.

3. Bitfehlerhäufigkeit und Verfügbarkeit verwendeter Kanten

Bezüglich dieser Kriterien unterscheidet sich FuzzyPlus-Routing nicht wesentlich von Fuzzy-Routing, wie Abbildung D.1 und Abbildung D.2 in ANHANG D im Vergleich mit den entsprechenden Abbildungen aus 6.2 belegt.

4. Verteilung der Bitfehlerhäufigkeit und Verfügbarkeit

Bezüglich dieser Kriterien unterscheidet sich FuzzyPlus-Routing nicht wesentlich von Fuzzy-Routing, wie die Zahlen in Tabelle 6.10 und Tabelle 6.11 belegen. In Abbildung D.3 und Abbildung D.4 in ANHANG D ist das Verhalten zusätzlich graphisch dargestellt.

6.4 Fuzzy-Neuro-Verfahren

Dieses Kapitel beschreibt die Adaption, die Lernerfolge des Adaptionsverfahren und die Ergebnisse von Fuzzy-Routing mit der adaptierten Regelbasis. Um die verwendeten Begriffe zu klären, wird zunächst der Trainingsalgorithmus skizziert bevor die Ergebnisse des Trai-

nings und des daraus resultierenden Fuzzy-Routings präsentiert werden.

Die zum Training zu verwendenden Routenlisten werden in Datei bereitgestellt. Die einzelnen Zeilen dieser Datei enthalten die Soll-Routen in der Form von Startknoten#Knoten2#...#Zielknoten. Aus jeder dieser einzelnen Routen werden von TransMerkur Trainingsdaten für NEUROTool, also Soll-Werte für die Kanten, erzeugt, wie in 5.4.2 beschrieben.

Der Ablauf des Lernvorgangs in TransMerkur ist wie folgt - siehe auch Abbildung 6.25:

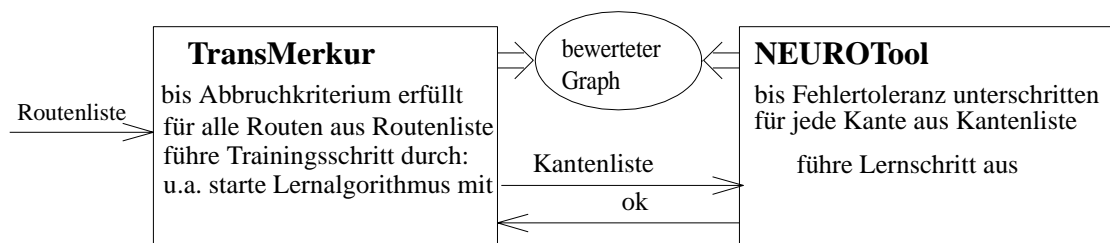


Abb. 6.25 Datenfluß während des Trainings

- NEUROTool wird beauftragt, den Graphen mit dem noch untrainierten neuronalen Netz zu bewerten.
- Die für das Training vorgegebene Routenliste wird gelesen.
- Bis das Abbruchkriterium erfüllt ist, werden für jede Route in der Routenliste die folgenden Schritte ausgeführt:
 - Die Ist-Route wird mit Fuzzy-Routing bestimmt.
 - Ist- und Soll-Route werden verglichen, und die Soll-Werte für die Kanten werden berechnet. Diese Soll-Werte werden in Form von Kantenlisten bereitgestellt.
 - Mit dieser Kantenliste wird der Lernalgorithmus in NEUROTool gestartet. Das Lernen in NEUROTool gilt als beendet, wenn die durchschnittliche Abweichung von Ist- und Soll-Wert die Fehlertoleranz unterschreitet. Während des Lernens hat sich das neuronale Netz verändert, so wird nach dem Ende des Lernens der Graph neu bewertet. Die nächste Route aus der Routenliste kann bearbeitet werden.

Dieses Bearbeiten einer Route aus der Routenliste wird mit **Trainingsschritt** in TransMerkur bezeichnet. Nach jedem Trainingsschritt prüft TransMerkur, wieviele Kanten sich in der Ist-Route verändert haben und bestimmt den prozentualen Anteil falscher Kanten in den Ist-Routen. Der zeitliche Verlauf dieses prozentualen Anteils wird am Ende des Trainings graphisch ausgegeben. Diese Graphik verdeutlicht den Verlauf des Trainingserfolges.

Der Lernvorgang in TransMerkur sollte abgebrochen werden, wenn sich Ist- und Soll-Routen nur noch in zwei oder drei Kanten unterscheiden, die sich in den Parametern ähnlich sind. Diese Ähnlichkeitsprüfung ist nicht ohne Aufwand zu realisieren, denn die Kantenwerte können nicht zum Vergleich der Kanten herangezogen werden, da die Kantenwerte während des Lernvorganges noch nicht die richtigen sind. Folgende Abbruchkriterien haben sich im Test

als erfolgreich erwiesen:

Es wird abgebrochen, wenn

- jede Ist-Route gleich der jeweiligen Soll-Route ist
- oder
- die Routenliste mehr als zehn mal durchlaufen wurde. Die Tests haben gezeigt, daß ein Trainingslauf, der diese Grenze überschreitet, meist ein Problem in den Trainingsdaten hat. Eine oder auch mehrere zum Training vorgeschlagene Routen sind widersprüchlich zu dem verwendeten Regelsystem und bedürfen der genaueren Analyse; siehe hierzu auch 6.4.1.
- oder
- sich nach drei vollständigen Trainingszyklen keine Route mehr geändert hat.

6.4.1 Adaption der Regelgewichte

Das Training benutzt globale Daten, Routen, um lokal Kanten zu adaptieren. Es bestehen also zunächst berechtigte Zweifel, ob ein solches System überhaupt zu trainieren ist. Um das zu überprüfen, wurde der in Abbildung 6.26 dargestellte Graph erzeugt, hier Minigraph genannt.

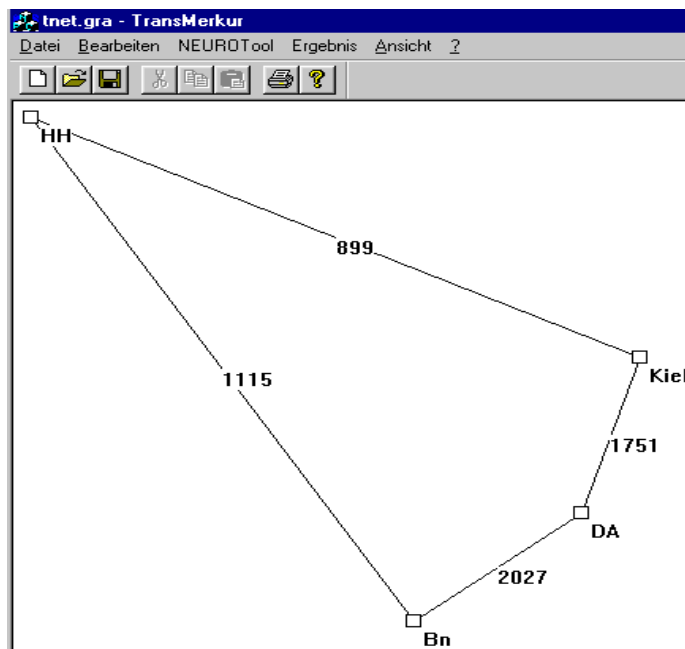


Abb. 6.26 Minigraph

Als Fuzzy-Regelsystem wurde das in 6.1.2.2 beschriebene umstrukturierte Regelsystem verwendet, das auch für die Tests von Fuzzy-Routing und FuzzyPlus-Routing eingesetzt wurde. Jedoch wurde die Längenbewertung den Größenverhältnissen des Graphen angepaßt. Die Fuzzy-Regeln für den Kantenparameter Länge sind mit 0 gewichtet. Die von Fuzzy-Routing gewählte Route von *Bn* nach *Kiel* geht über *HH*. Das System wird mit *Bn-DA-Kiel* trainiert.

Mit dem Gradientenverfahren ist nach 20 Trainingszyklen die Route trainiert und mit Delta-Lernen nach 34 Trainingszyklen. Die Gewichte für die Regeln der Länge sind deutlich erhöht. Auch die Gewichte der anderen Regeln sind leicht verändert, da eine Berücksichtigung des Parameters Länge gleichzeitig eine Veränderung der Bewertung der anderen Parameter bedeutet.

Grundsätzlich ist das System also in der Lage, eine andere Gewichtung der Regeln zu lernen und somit eine andere Gewichtung der Bedeutung der Parameter bei der Kantenbewertung.

Als nächstes wird gezeigt, daß es mit Hilfe der Adaption der Regelgewichte möglich ist, die Fuzzy-Regelbasis so umzugewichten, daß bei der Bewertung der Kanten ein Parameter beachtet wird, der vor dem Training nicht beachtet wurde. Dazu wird der nachfolgend beschriebene Test durchgeführt.

In der für Fuzzy-Routing erstellten Regelbasis werden die Regeln, die die Auslastung betreffen, mit Null gewichtet. Es wird eine Routenliste erzeugt, deren Routen so ausgewählt werden, daß die Auslastung berücksichtigt wird. Dazu ist ein Graph erstellt worden, der mehrere gleichwertige Routen für eine Verbindung zwischen zwei Knoten bietet, wenn die Auslastung nicht bei der Bewertung der Kanten berücksichtigt wird - siehe Abbildung 6.27. Die genauen Daten des Trainingsgraphen können in ANHANG E nachgelesen werden.

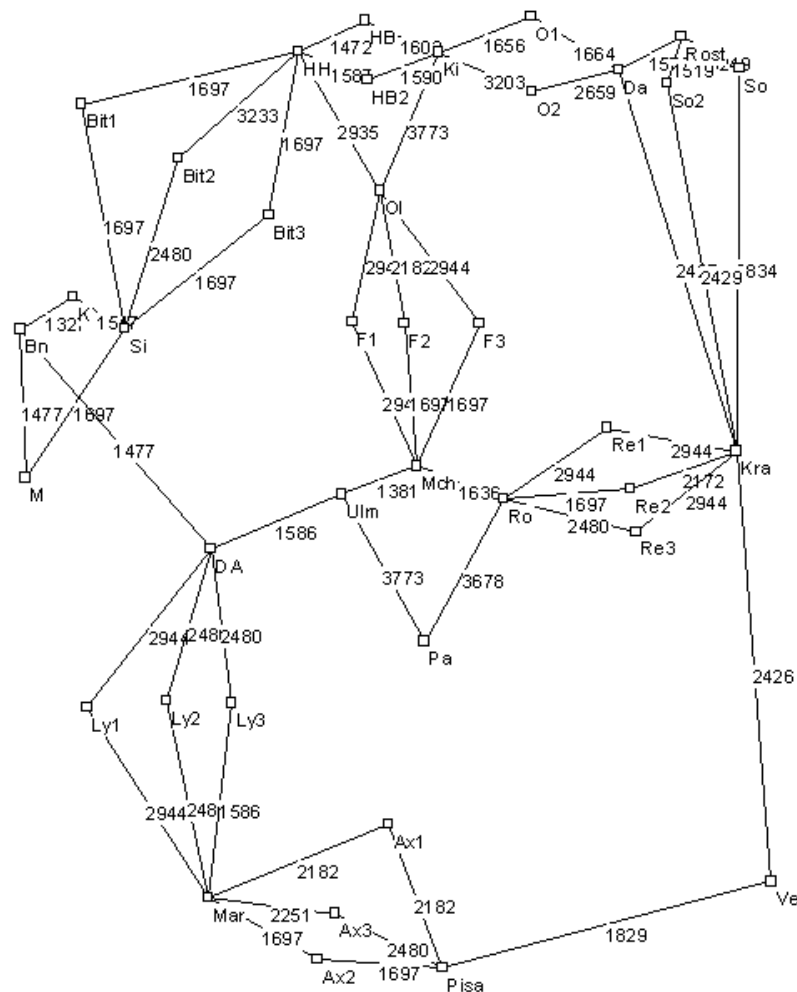


Abb. 6.27 Zum Training verwendeter Graph

Für die Routenliste werden Kanten mit guter Auslastung bevorzugt. Die Qualität dieser Kanten ist etwas schlechter als die zur Verfügung stehenden alternativen Kanten. Mit der folgenden Routenliste wird das neuronale Netz trainiert:

DA#Ly2#Mar

Ro#Re3#Kra

Mar#Ax3#Pisa

Die Auswahl der Trainingsrouten wird am Beispiel der Route *DA-Mar* erklärt. Die Werte der hier betroffenen Kanten sind in Tabelle 6.12 aufgelistet. Die Werte aller anderen Kanten können in ANHANG E in Tabelle E.1 nachgelesen werden. Für *DA-Mar* wird die Route *DA-Ly2-Mar* gewählt. Das Auslastungsverhalten der beteiligten Kanten ist besser als das der Kanten von *DA-Ly3-Mar*, aber die Qualität der Kanten in *DA-Ly2-Mar* ist leicht schlechter.

Die Auswahl der Kanten für das Training suggeriert also eine höhere Gewichtung des Parameters Auslastung als der Parameter, die die Qualität betreffen.

	DA-Ly1	DA-Ly2	DA-Ly3	Ly1 - Mar	Ly2 - Mar	Ly3 - Mar
strategische Bedeutung	0	0	0	0	0	0
physikalische Länge	190	150	147	213	190	185
Auslastung	40	40	4	40	40	4
Verfügbarkeit	40	60	60	40	50	70
Abweichung von der Mindestverfügbarkeit	-20	0	0	-20	-10	10
Bitfehlerhäufigkeit	40	50	50	40	40	40
Abweichung von der maximal erlaubten Bitfehlerhäufigkeit	-10	-10	-10	0	0	0
Kantenwert bestimmt mit System vor dem Training	29,80	24,80	24,80	29,44	24,80	15,86

Tab. 6.12 Kanten und ihre Werte für Routen zwischen DA und Mar

Das Training wurde sowohl mit Delta-Lernen als auch mit Gradientenabstieg durchgeführt. Als Lernrate wurde $\alpha = 0,01$ verwendet. Für die Erzeugung der Kantenliste wurde $\text{delta1} = 0,25$ und $\text{delta2} = 0,35$ verwendet. Die zugelassene Fehlertoleranz im NEUROTool ist 0,1. Mit Delta-Lernen wurden 36 Trainingszyklen gebraucht und mit Gradientenabstieg 15. Abbildung 6.29 zeigt den Trainingsfortschritt für Delta-Lernen und Abbildung 6.28 zeigt den Trainingsfortschritt für das Gradientenabstiegsverfahren. Die entstandenen Systeme sind gleich gut.

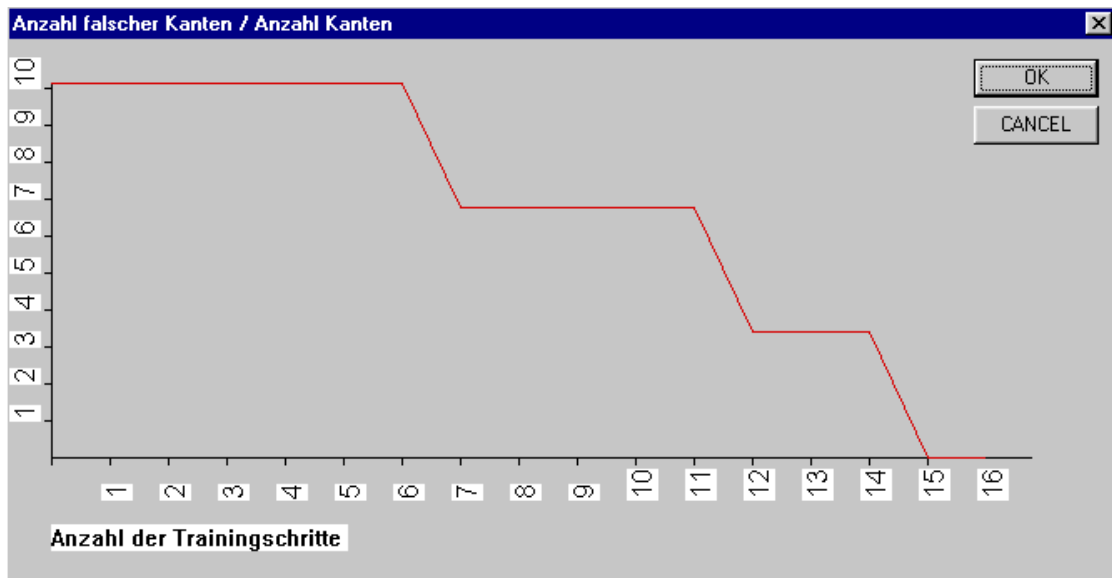


Abb. 6.28 Trainingserfolg mit Gradientenabstiegsverfahren

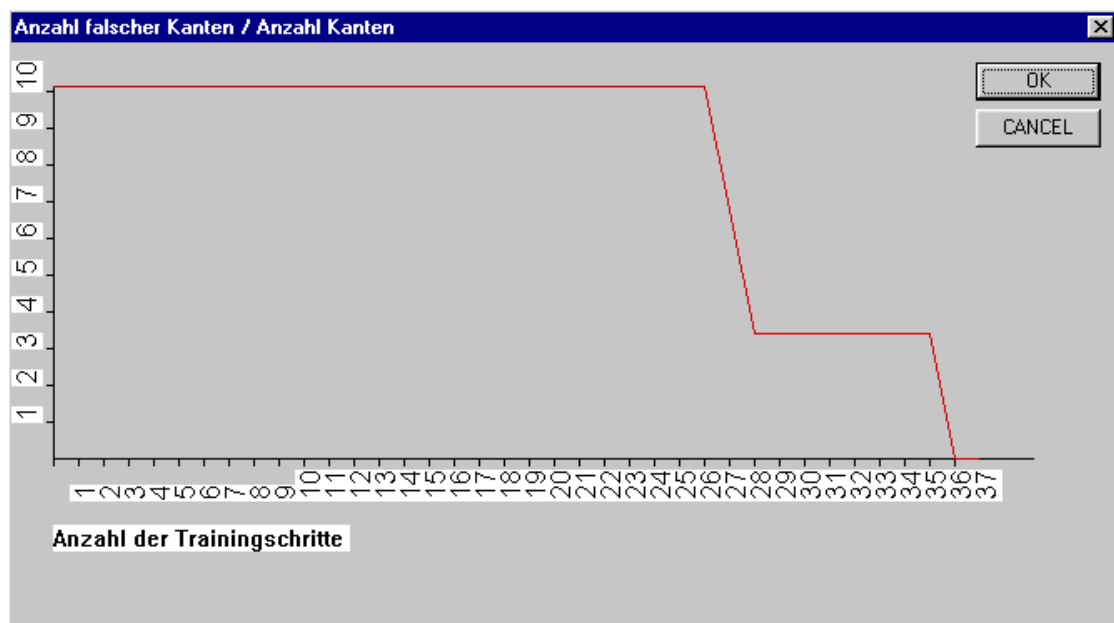


Abb. 6.29 Trainingserfolg mit Delta-Lernen

Im folgenden Text werden die Ergebnisse des Trainings mit Gradientenabstiegsverfahren dargelegt.

Es gibt sechs Regeln, die die Auslastung betreffen. Ihre Gewichte wurden während des Trainings von 0.0 auf die in Tabelle 6.13 dargestellten Gewichte geändert.

Regelbezeichnung	Gewicht nach dem Training
rAusL1	0,45
rAusL2	0,0
rAusL3	0,285
rBeleg1	0,45
rBeleg2	0,466
rBeleg3	0,0

Tab. 6.13 Gewichte der Auslastungsregeln nach dem Training

Das durch das Training entstandene neuronale Netz wird in ein regelbasiertes Fuzzy-System transformiert. Mit diesem wird ein Einsatz auf dem Layer-Netzwerk, auf dem auch das Training durchgeführt wurde, simuliert. Dazu wurden 120 Routenwünsche bearbeitet.

Da das Ziel des Trainings die Beachtung der Auslastungsregeln war, werden die Eigenschaften des Layer-Netzwerkes an seinem Auslastungsverhalten nach einem Einsatz genauer dargestellt:

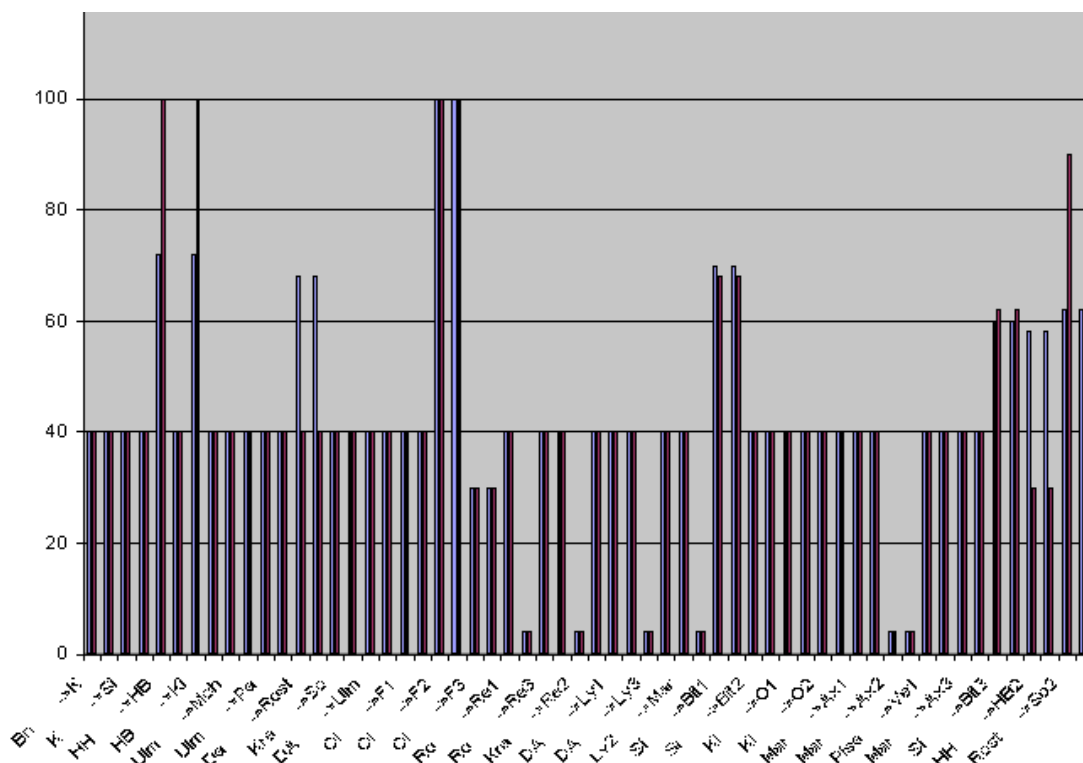


Abb. 6.30 Vergleich der Kantenauslastung des Testgraphen

Abbildung 6.30 zeigt die Auslastung der Kanten des Testgraphen. Die linke Säule stellt

jeweils den Wert der Auslastung der Kante nach einem Einsatz mit dem noch untrainierten regelbasierten Fuzzy-System und die rechte Säule den entsprechenden Wert nach einem Einsatz mit dem trainierten regelbasierten Fuzzy-System dar. Die genauen Werte können in ANHANG E in Tabelle E.3 nachgelesen werden.

Tabelle 6.14 zeigt die Werte für die Auslastung der signifikanten Kanten und erklärt das Verhalten. Wenn eine Ausweichroute existiert, die von der Bewertung her in Frage kommt, wird bei Überschreitung der Toleranzgrenze für die Auslastung der Kanten in der zuerst gewählten Route, die Ausweichroute verwendet. Also beachtet das System nach dem Training die Auslastungsregeln. Die Parameter der Kanten können in ANHANG E in Tabelle E.1 nachgelesen werden.

Kante	Auslastung vor dem Einsatz	Auslastung mit untrainiertem System	Auslastung mit trainiertem System	Kommentar
HH - HB	40	100	72	Ausweichroute (HH-HB2-Ki) wird ab einer bestimmten Auslastung der Original-Route (HH-HB-Ki) mit trainiertem System als besser betrachtet und daher verwendet. Die Kanten der gewählten Ausweichroute sind qualitativ gleich.
HB - Ki	40	100	72	
HH -HB2	30	30	58	
HB2 - Ki	30	30	58	
O1 - F2	40	100	100	Die Kanten der Ausweichrouten (O1 -F1/ F3- Mch) sind auch mit trainiertem System schlechter bewertet, da sie die Qualitätsanforderungen nicht erfüllen.
F2 -Mch	40	100	100	
Rost -So2	30	90	62	Die Kanten der Ausweichroute (Rost-DA-Kra) werden ab einer bestimmten Auslastung der Original-Route (Rost-So2-Kra) mit trainiertem System besser bewertet und daher verwendet. Die Kanten der gewählten Ausweichroute sind qualitativ gleich.
So2 - Kra	30	90	62	
DA - Rost	40	40	68	
DA -Kra	40	40	69	

Tab. 6.14 Vergleich des Auslastungsverhaltens

Es ist möglich, das Training auf einem entsprechend gestalteten Trainings-Layer-Netzwerk durchzuführen und das so entstandene regelbasierte Fuzzy-System auf einem anderen Layer-Netzwerk einzusetzen. Um das nachzuweisen, wurde das trainierte System für das Layer-Netzwerk eingesetzt, mit dem Fuzzy-Routing getestet wurde. Mit dem trainierten Fuzzy-System wurde der in 6.2.3 beschriebene Einsatz simuliert.

Die Abbildung 6.31 zeigt das Auslastungsverhalten der Kanten nach dem Einsatz des trainierten Systems. Das Auslastungsverhalten der Kanten mit dem Original-Regelwerk ist in Abbildung 6.7 abgebildet. Das trainierte System zeigt ein leicht schlechteres Auslastungsverhalten. Dies wird zusätzlich durch die in Tabelle 6.15 dargestellten Werte deutlich. Die Gewichte der Auslastungsregeln im Original-Regelwerk sind alle 0.5, während die Gewichte für das trainierte System die Werte haben, die in Tabelle 6.13 dargestellt sind.

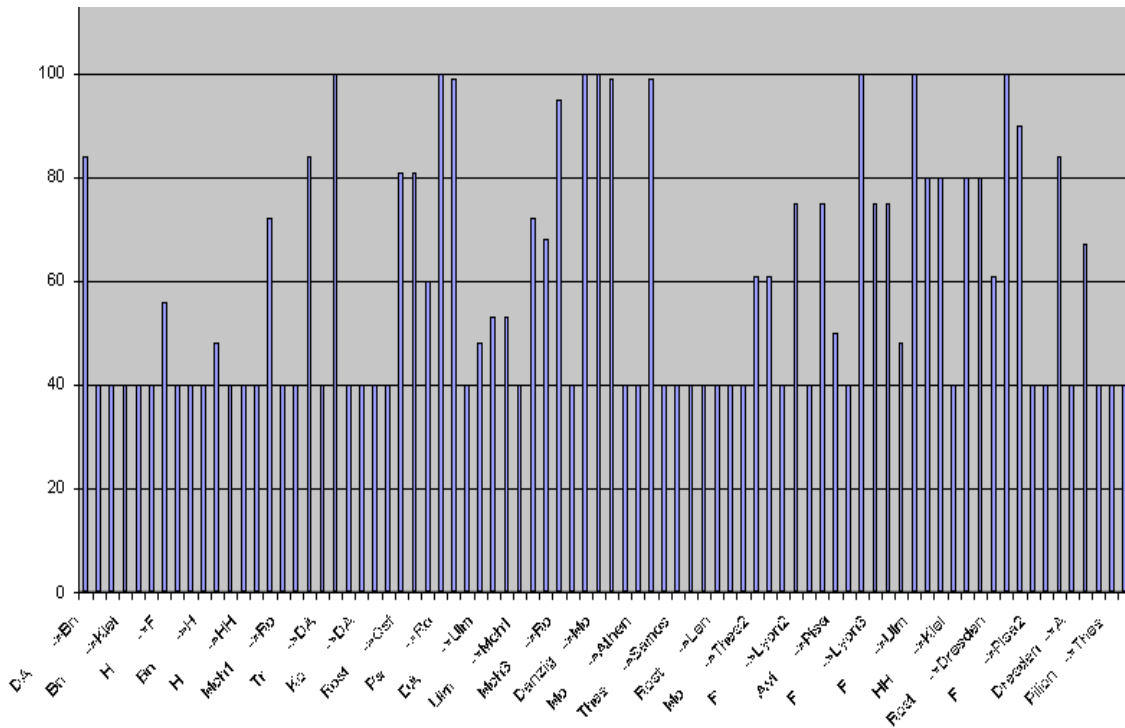


Abb. 6.31 Auslastung der Kanten nach Einsatz des trainierten Systems

Die Abbildung 6.32 zeigt das Auslastungsverhalten der Kanten nach dem Einsatz des untrainierten Systems.

Hier sind die Gewichte der Auslastungsregeln auf 0 gesetzt. Das Auslastungsverhalten ist schlechter.

Fuzzy-System	Anzahl Kanten mit Auslastung über 80%	Anzahl Kanten mit Auslastung von 100/
original Fuzzy-System (Auslastungsregeln mit 0,5 gewichtet)	14	6
trainiertes System	17	7
untrainiertes System (Auslastungsregeln mit 0 gewichtet)	25	12

Tab. 6.15 Vergleich der Systeme bezüglich stark ausgelasteter Kanten

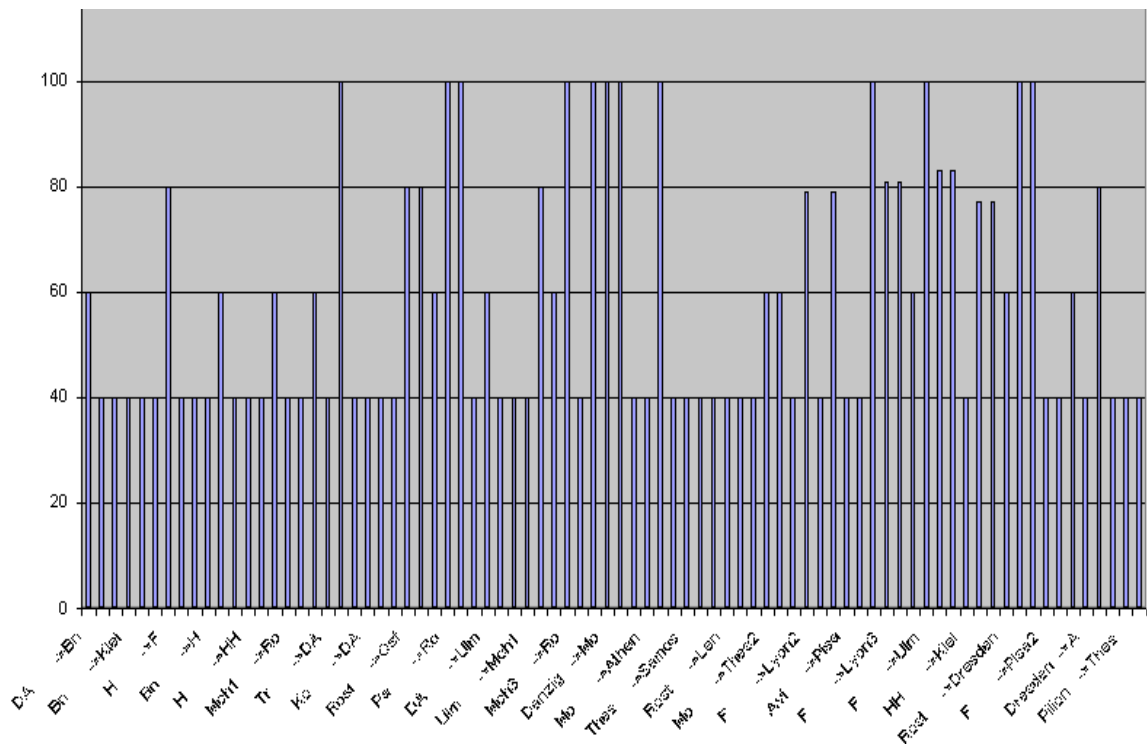


Abb. 6.32 Auslastung der Kanten nach dem Einsatz des untrainierten Systems

6.5 Bewertung

Mit den in den vorangegangenen Kapiteln dargestellten Test ist an künstlichen, für die Aufgabenstellung repräsentativen Layer-Netzwerken gezeigt worden, daß die entwickelten Routing-Verfahren, die an sie gestellten Anforderungen erfüllen.

6.5.1 Fuzzy-Routing

Fuzzy-Routing kann zusammenfassend wie folgt beurteilt werden:

Mit Fuzzy-Routing werden Routen bestimmt, die die Regeln so gut wie in dem Layer-Netzwerk möglich für die Wahl der einzelnen Route einhalten. Alle Parameter werden bei der Routenauswahl wie gewünscht herangezogen.

Da die Regeln des Fuzzy-Systems mit Gewichten versehen sind, kann der Einfluß der einzelnen Regel auf die Wahl justiert werden.

Die Anzahl der Schaltpunkte wird zwar nicht explizit beachtet, jedoch durch die Berechnung der Routenbewertung implizit beachtet, wie die Betrachtungen in 6.2.3 unter „2. Kosten der Knoten“ zeigen.

Der Einfluß von Fuzzy-Routing auf das gesamte Layer-Netzwerk ist positiv:

- das gewünschte Auslastungsverhalten der Knoten und Kanten wird eingehalten
- es werden wirtschaftlich günstige Knoten und Kanten ausgewählt
- die Qualitätskriterien, die an die hochprioritären Übertragungswege gestellt werden, werden nur in wenigen Fällen nicht beachtet, obwohl sie nicht hart geprüft werden.

In diesem Test hatten laut Tabelle 6.11 9,9% der Kanten nicht die geforderte Mindestverfügbarkeit und laut Tabelle 6.10 überschritten 5,5% der Kanten die geforderte maximale Bitfehlerhäufigkeit.

Mit dem Parameter *stratMean* und den dazugehörigen Regeln hat der Benutzer die Möglichkeit, in die Bewertung jeder einzelnen Kante manuell einzugreifen. Er kann die Auswahl gezielt steuern. Die Anforderungen an die Feinsteuerung der Routenbewertung sind so erfüllt.

Die Ergebnisse in 6.2.4 zeigen deutlich die Vorteile von Fuzzy-Routing für das gesamte Layer-Netzwerk.

Bei stark ausgelastetem Layer-Netzwerk ist es jedoch oft nicht möglich, gute Routen zu bestimmen, da dafür keine Ressourcen mehr zur Verfügung stehen.

Um diese Situation zu vermeiden, bietet sich eine Ergänzung des automatischen Routing-Verfahrens an:

Das automatische Routing-Verfahren könnte bei der Überschreitung eines bestimmten Wertes der Auslastung der Kante, den Anwender informieren. Es besteht dann die Möglichkeit mit Hilfe des Parameters *stratMean* die Routen lokal umzuleiten. Der Anwender hat in dieser Situation den Vorteil, daß er die Kante in einen inhaltlichen Zusammenhang zu dem gesamten Layer-Netzwerk stellen kann, indem er der Graphen anschaut, und so Schaltungen geschickt mit Hilfe der Feinsteuerung umlenken kann.

Vorstellbar ist auch eine automatische Sperrung der fast vollen Kanten, durch eine erhebliche Erhöhung des Kantenwertes - siehe Ausblick.

Neue Regeln, die sich auf schon existierende Parameter beziehen, können hinzugefügt wer-

den. Dazu ist es nicht nötig das Programm auszutauschen. NEUROTool wird gestoppt und mit dem veränderten Regelwerk wieder gestartet.

Fuzzy-Routing erfüllt also alle in 3.4 aufgeführten Anforderungen.

6.5.2 FuzzyPlus-Routing

FuzzyPlus-Routing bietet alle Vorteile von Fuzzy-Routing und führt eine zusätzliche Kostenoptimierung durch. Durch die Bestimmung von mehreren Routen entsteht eine leichte Verschlechterung der Performance, die hier jedoch nicht untersucht worden ist.

Mit der in FuzzyPlus-Routing verwendeten Architektur ist es auf Grundlage von Fuzzy-Routing möglich, jeden anderen additiven Parameter zusätzlich zu optimieren. Die Vorteile, die durch Fuzzy-Routing gegeben sind, bleiben grundsätzlich erhalten.

6.5.3 Fuzzy-Neuro-Verfahren

Mit Hilfe der Adaption der Regelgewichte kann ein Merkmal dazugelernt werden, für das jedoch Regeln bereits formuliert sein müssen. In der Praxis kann davon ausgegangen werden, daß eine ungefähre Einschätzung des Parameters vorliegt und so ein einfacher Satz von Regeln formuliert werden kann. Also ist es kein Nachteil, daß die Regeln bereits in einfacher Form vorhanden sein müssen. Die Feinjustierung und die Einpassung in das bereits bestehende Regelwerk wird durch das Training geleistet.

Das Training kann auf einem Trainingsgraphen ausgeführt werden, für den dasselbe Regelwerk gilt wie für den Graphen, für den das trainierte System später eingesetzt werden soll. Das hat den Vorteil, daß das Lernverfahren auf einem kleineren Graphen ablaufen kann. Das ist nützlich, wenn das eigentliche Layer-Netzwerk sehr groß ist, wie in der Praxis üblich.

Bei den Vorbereitungen für den oben beschriebenen Test hat sich gezeigt, daß die Trainingsdaten sehr sorgfältig gewählt werden müssen. Das heißt:

1. Bei der Auswahl der Trainingsrouten muß auf alle Parameter aller ihrer Kanten geachtet werden. Da jede Kante mehrere Parameter hat, kann es vorkommen, daß zwar alle Kanten der Routen, die für das Training gewählt wurden, die Auslastung in der gleichen Weise beachten, aber die implizite Wahl der anderen Parameter widersprüchliche Regeln für deren Wahl suggerieren. So wird dann bei einem Trainingsmuster etwas gelernt, das bei einem der nachfolgenden Muster wieder verlernt werden muß.

und

2. Die Anzahl der Kanten in den Trainingsrouten ist zu beachten. Denn die Anzahl der Hops wird implizit mitbeachtet. Eine Route mit weniger Kanten kann besser sein als eine mit mehr Kanten, obwohl die einzelnen Kantenwerte in der Route mit weniger Kanten höher sind. Bei einem durchgeführten Trainingslauf für den in Abbildung 6.27 gezeigten Graphen wurden dem System elf Routen zum Training angeboten. Zehn dieser Routen wurden in akzeptabler Zeit gelernt. Aber bei der Route *Rost-Da-Kra-Re3-Ro-Mch-Ulm-DA-Ly3-Mar* (9 Kanten, Summe der Kantenwerte: 180,98) blieb der Fehler immer oberhalb der Toleranzgrenze. Eine Analyse ergab, daß es eine alternative Route gibt:

Rost-Da-Kra-Vel-Pisa-Axl-Mar (6 Kanten, Summe der Kantenwerte: 126,24)

Die Kanten der erst genannten Route halten zwar die Regeln besser ein, aber die Kanten sind nicht soviel schlechter, als das dies durch die geringere Anzahl der Summanden in der Bewertung der Route *Rost-Da-Kra-Vel-Pisa-Axl-Mar* ausgeglichen würde.

Das zuletzt genannte Beispiel zeigt, daß das Adaptionungsverfahren also auch zur Analyse von Routen dienen kann. Routen, die sich gar nicht oder nur mit großem Aufwand trainieren lassen, passen nicht in das Wertesystem, das der Betreiber durch die Regeln vorgibt. Diese Routen sollten überprüft werden und gegebenenfalls umgeschaltet werden.

7.1 Rückblickende Beurteilung

Diese Arbeit stellt ein Verfahren zur Routenauswahl in Layer-Netzwerken von Transportnetzen bereit.

Die Aufgabe wird reduziert auf die Bewertung und Auswahl von Routen in einem gerichteten Graphen - ausgehend von der Bewertung der Kanten und Knoten mit jeweils einer Reihe von verschiedenartigen Parametern, die von den technischen und wirtschaftlichen Problemen abhängen. Es zeigt sich, daß ein wesentliches Problem der globalen Bewertung der Routen darin liegt, daß die Berücksichtigung der lokalen Kanten- und Knotenparameter nicht durch eine einfache Funktion beschrieben werden kann. So können die besten Routen nicht einfach an Hand eines aus den Parametern unmittelbar folgendem Distanzmaßes bestimmt werden. Existierende Lösungen zu Kürzeste-Wege-Probleme mit mehreren Parametern lassen sich wegen der Nicht-Additivität einiger Parameter nicht anwenden.

Den Experten in den Schaltzentren der Betreiber gelingt es die Routenauswahl auf Grundlage der Kanten- und Knotenparameter auszuführen, trotz der aufgezeigten Schwierigkeiten. Die Analyse des Expertenverhaltens führt zum hier gewählten Lösungsansatz. Das vorhandene Expertenwissen wird genutzt, um ein regelbasiertes Fuzzy-System zu erstellen, das die Kanten des Layer-Netzwerkes bewertet. Mittels der Bewertung jeder Kante entsteht so ein fuzzy-bewerteter Graph. Auf diesem fuzzy-bewerteten Graphen wird ein Kürzeste-Wege-Algorithmus eingesetzt. So entsteht das Routing-Verfahren Fuzzy-Routing.

Fuzzy-Routing ist mit zwei Defuzzifizierungs-Methoden, der modifizierten Schwerpunkt Methode und der Methode nach Takagi-Sugeno, realisiert und getestet worden. Mit beiden Defuzzifizierungs-Methoden ist die gleiche Wirksamkeit des Routing-Verfahrens zu beobachten.

Mit Fuzzy-Routing wird das Ziel erreicht, alle Parameter der Kanten und Knoten für die Routenauswahl zu berücksichtigen und zwar genau so wie das in den Regeln festgelegt wird. Mit Hilfe eines speziell dazu eingeführten Parameters und einiger zusätzlicher Regeln ist es möglich, ausgewählte Kanten gesondert und abweichend von den allgemein gültigen Regeln zu bewerten, wenn das aus strategischen Gründen notwendig ist. Fuzzy-Routing bietet so eine temporäre und lokale Feinsteuerung der Routenauswahl.

Die Auswirkungen auf das gesamte Layer-Netzwerk nach längerem Einsatz sind wie

gewünscht:

- Die Anforderungen an das Auslastungsverhalten werden beachtet.
- Wirtschaftlich günstige Knoten und Kanten werden gewählt.
- Die Qualitätskriterien werden beachtet.

Wenn sich Anforderungen ändern, die Parameter betreffen, die schon in den Regeln berücksichtigt werden, so kann Fuzzy-Routing leicht daran angepaßt werden. Die Regeln werden in diesem Fall entsprechend den neuen Anforderungen formuliert.

Wenn durch neue Anforderungen Parameter berücksichtigt werden müssen, die bisher noch nicht in den Regeln von Fuzzy-Routing enthalten sind, so kann das Fuzzy-System um die neuen Parameter und die neuen Regeln ergänzt werden.

Das Fuzzy-System hat außerdem den Vorteil, daß die Regeln auch von Fachleuten aus dem Bereich der Telekommunikation, die nur wenig Informatikkenntnisse haben, gelesen, interpretiert und geändert werden können.

Fuzzy-Routing ist der Grundbaustein für weitere Varianten des Routing-Verfahrens. Auf der Menge der k besten, $k \in \mathbb{N}$, mit Fuzzy-Routing bestimmten Routen, kann zusätzlich ein Optimierungskriterium angewandt werden. Der Anwender legt k fest. Je größer k , um so kleiner der Einfluß von Fuzzy-Routing bei der Auswahl der Route. Als solches Optimierungskriterium eignet sich jeder Kantenparameter, der additiv entlang jeder Route ist. So können zum Beispiel die Kosten, die Anzahl der Knoten oder die physikalische Länge zusätzlich optimiert werden. Mit Fuzzy-Routing und den darauf basierenden Routing-Verfahren ist es also möglich Routen gemäß den in 3.4 formulierten Zielsetzungen zu bestimmen. Die Regeln können genau nach den Angaben der Betreiber formuliert werden. Auch der Einsatz zusätzlicher Optimierungskriterien kann nach dessen Ansprüchen hinzugefügt werden.

Die reversible Transformation des regelbasierten Fuzzy-Systems in ein neuronales Netz ermöglicht die Nutzung empirischer Daten. Aus den globalen Daten, den Soll-Routen, werden Soll-Kantenwerte erzeugt, um das Fuzzy-System zu trainieren. Von den vorgestellten Trainingsmöglichkeiten wird der Lernalgorithmus für die Gewichte der Regeln realisiert. Mit diesem Lernalgorithmus wird erfolgreich die Beachtung eines Parameters eingebracht, der vorher schon in den Regeln enthalten ist, aber durch eine Gewichtung der zugehörigen Regeln mit Null nicht beachtet wurde.

Die Erfahrungen aus der Simulation zeigen, daß das Training Daten mit eindeutigen Aussagen benötigt. Werden zum Training Kanten mit Parameterwerten angeboten, die sich widersprechen, so wird das Training sehr schwierig und langsam. Es bleibt noch zu untersuchen, ob eine andere Adaptionmethode in diesem Punkt eine Verbesserung bringt; siehe auch 7.2.2. Es kann davon ausgegangen werden, daß in der Praxis nicht immer ideale Trainingsdaten vorhanden sind. So kann das Training des Fuzzy-Systems für den praktischen Einsatz schwierig werden.

Das Training des Routing-Verfahrens kann jedoch als Analyse-Tool eingesetzt werden, um Routen mit widersprüchlichen Werten der Kantenparameter zu finden.

Für jedes einzelnes Layer-Netzwerk eines Transportnetzes erfüllen die in der Arbeit vorgestellten Routing-Verfahren alle formulierten Zielsetzungen.

7.2 Ansatzpunkte für zukünftige Arbeiten

Die in dieser Arbeit vorgestellten Routing-Verfahren leisten einen Beitrag zur Lösung der Probleme, die bei der Routenauswahl in Transportnetzen vorhanden sind. Es sind jedoch Erweiterungen möglich, die im folgenden kurz dargestellt werden.

7.2.1 Erweiterungen der Anwendung

- Bei stark ausgelastetem Layer-Netzwerk ist es auch für Fuzzy-Routing und seine Varianten schwierig, Routen im Sinne der Regeln zu bestimmen.

Um eine solche Situation im Layer-Netzwerk zu vermeiden, kann wie folgt vorgegangen werden:

1. Nach jeder Schaltung eines Übertragungsweges bestimmt Fuzzy-Routing die Auslastungswerte der beteiligten Kanten und Knoten erneut. Hier wird Fuzzy-Routing um eine Prüfung erweitert. Überschreitet der neu berechnete Auslastungswert einen durch den Anwender festgelegten Betrag, so wird der Anwender mittels einer Warnung informiert. Er hat dann die Möglichkeit, mit Hilfe des Parameters *stratMean* die entsprechenden Kanten bei der weiteren Routenauswahl auszulassen.
Statt den Anwender nur zu informieren, ist eine zweite Erweiterung möglich.
2. Zusätzlich zur Ausgabe der Warnung, wird bei Kanten der Wert *edgeVal* auf den höchst möglichen Wert gesetzt. So wird diese Kante nur noch gewählt, wenn alle anderen verfügbaren Kanten auch so schlecht bewertet werden.

Die Warnung ist in beiden Fällen wichtig, da es sich in jedem Fall um einen Engpaß im Layer-Netzwerk handelt. Nur ein menschlicher Betrachter hat den Überblick über das gesamte Layer-Netzwerk und auch über das Transportnetz und kann entscheiden, was genau zu tun ist.

- In Transportnetzen ist es oft wichtig, Übertragungswege zu schalten, die ganz oder teilweise disjunkt zu anderen Übertragungswegen sind. Daher ist es von Interesse, Routen zu bestimmen, die bestimmte Kanten nicht verwenden, da sie disjunkt zu anderen Routen verlaufen müssen. Eine „Sperrung“ der Kanten durch eine starke Erhöhung des Kantenwertes *edgeVal* kann zum Ziel führen. Die Funktionsfähigkeit dieser Methode ist noch zu testen.
Für Disjunktheitsprüfung reicht es nicht aus, das Layer-Netzwerk der Übertragungsrate des Übertragungswegs zu beachten. Hier sind auch die Führungen der verwendeten Kanäle zu beachten, wie schon in 2.4 dargelegt wurde.
- Wenn Routen für hoch prioritäre Übertragungswege ausgewählt werden, dann dürfen die Anforderungen an die Qualität nicht verletzt werden. Mit Fuzzy-Routing werden zwar meist passende Kanten und Knoten ausgewählt, aber die Beachtung der Qualitätsanforderungen ist nicht garantiert. Um das zu erreichen, muß Fuzzy-Routing um eine „harte“ Überprüfung der Verfügbarkeits- und Bitfehlerhäufigkeitswerte ergänzt werden. Das kann erreicht werden, indem aus

einer mit Fuzzy-Routing erzeugten Lösungsmenge die Route ausgewählt wird, die die Regeln am besten erfüllt und deren Kanten die Qualitätsanforderungen erfüllen.

- Das Verfahren wurde für einen Graphen entwickelt, der hinsichtlich der Bewertung seiner Kanten und Knoten einem Layer-Netzwerk von Transportnetzen entspricht. In 3.1.1 ist eine Reihe von Parametern festgelegt worden, um ein konkretes Verfahren zu entwerfen.

Wenn Fuzzy-Routing oder eine seiner Varianten für ein Transportnetz eingesetzt werden soll, in dem andere Parameter eine Rolle spielen, so können diese ausgetauscht werden. Es ist jedoch Voraussetzung, daß sich die Bewertungen der Kanten und Knoten an Hand dieser Parameter in Regeln formulieren lassen.

- Ein Routing-Verfahren, das zunächst eine Fuzzy-Bewertung des Graphen vornimmt und dann ein Kürzeste-Wege-Verfahren zur Bestimmung der besten Route einsetzt, kann grundsätzlich auf allen Graphen eingesetzt werden, deren Kanten und Knoten mit verschiedenartigen Parametern bewertet sind und deren Bewertung sich in Regeln formulieren läßt. Ein solches Vorgehen bietet sich vor allem bei Bestimmungen von Routen an, bei denen sich die besten Routen nicht einfach an Hand eines aus den Parametern der Kanten und Knoten berechneten Distanzmaßes bestimmen lassen, und die komplizierten Restriktionen unterliegen.

Interessant erscheint der Versuch, Fuzzy-Routing für Planungsprobleme und in der Robotik einzusetzen. Wichtige Zusatzoptimierungen für additive Parameter, wie zum Beispiel Kostenoptimierung, können auch hier wie bei FuzzyPlus-Routing ergänzt werden. Wie die genauen Lösungen aussehen, ist noch zu untersuchen.

7.2.2 Erweiterungen der Technologie

- Das Training der Regelgewichte setzt auf die modifizierte Schwerpunktmethode auf. Die linguistischen Werte der Ausgabevariablen haben trapezförmige Zugehörigkeitsfunktionen. Die Veränderung eines Regelgewichtes w um Δw bedeutet, daß die Zugehörigkeitsfunktion, die diese Regel als Ausgabe hat, auf einer Höhe $(w + \Delta w)\mu$ statt $w\mu$ abgeschnitten wird. Der Schwerpunkt dieser Fläche, die in das Ergebnis eingeht, ändert sich so nur wenig, beziehungsweise gar nicht, wenn die Fläche symmetrisch ist.

Es ist daher anzunehmen, daß das Training des Routingverfahrens effektiver wird, wenn die linguistischen Werte der Ausgabevariablen als Singletons dargestellt werden und statt der modifizierten Schwerpunktmethode die Methode nach Takagi-Sugeno verwendet wird.

- Die Adaption der Eingabe-Variablen und der Ausgabe-Variablen, wie sie in 5.4.3 beschrieben ist, ist noch zu realisieren.
- Die verschiedenen Möglichkeiten der Adaption können kombiniert werden. Es ist noch zu untersuchen, welche Kombinationen sinnvoll sind und wie das genaue Vorgehen aussieht.

Mit einem solchen Ansatz kann zum Beispiel das Problem gelöst werden, das bei der Adaption der Regelgewichte auftritt:

Die Gewichte der Regeln sind aus dem Intervall $[0,1]$. Wenn der Lernalgorithmus zu dem Ergebnis kommt, daß ein Gewicht verkleinert werden muß und mit der vorgesehenen Verkleinerung das Gewicht kleiner Null würde, dann wird das Gewicht im vorgestellten Verfahren auf Null gesetzt. Entsprechend wird beim Überschreiten der oberen Intervallgrenze das Gewicht auf Eins gesetzt.

Ein alternatives Vorgehen kann an dieser Stelle wie folgt aussehen:

Statt das Gewicht zu verändern, wird an dieser Stelle mit einer Adaption der Ausgabevariablen begonnen.

- Das vorliegende Verfahren arbeitet mit linguistischen Variablen, die jeweils drei linguistische Werte haben. Es ist noch zu untersuchen, ob eine Erhöhung der Anzahl der Werte das System verbessert. Sinnvoll erscheint das Aufnehmen neuer linguistischer Werte, wenn eine Analyse des letzten Lernschrittes eine zyklische Veränderung der linguistischen Werte zeigt - wie schon in 5.3 erwähnt.

7.2.3 **Schlußbemerkung**

In dem Gebiet der Telekommunikation werden ständig Neu- und Weiterentwicklungen gemacht. Das gilt auch für den Bereich der Transportnetze.

Die Routenauswahl ist eine zentrale Aufgabe im Betrieb eines Transportnetzes. Sie muß sich nach den Anforderungen der Betreiber gestalten lassen. Technische Weiterentwicklungen können sowohl dazu führen, daß bereits existierende Parameter der Kanten anders zu bewerten sind, aber auch dazu, daß weitere Parameter bei der Routenauswahl zu berücksichtigen sind. Mit dem in dieser Arbeit vorgestellten Verfahren ist es möglich, veränderte Bewertungen zu berücksichtigen. An neu aufzunehmende Parameter werden keine Anforderungen gestellt. Insbesondere müssen sie nicht additiv entlang jeder Route sein.

Diese Arbeit leistet also nicht nur einen Beitrag zur Lösung der Probleme der heutigen Routenauswahl in Transportnetzen, sondern sie liefert auch eine Basis für zukünftige Lösungen in diesem Bereich.

Mit den in 7.2.1 und 7.2.2 vorgestellten Erweiterungsmöglichkeiten liefert sie eine Basis für weitere Fortschritte auf dem Gebiet von Fuzzy-Neuro-Systemen und deren Einsatzmöglichkeiten.

Variablen

Im folgenden sind die linguistischen Variablen definiert. Als Nomenklatur wird die Kommandosprache von NEUROTool verwendet wie sie in [4] definiert ist:

Objekte PH_VAR_IN_O beschreiben die physikalischen Input-Variablen.

Objekte PH_VAR_OUT_O beschreiben die linguistischen Output-Variablen.

PH_VAR_IN_O <objektname> <name> <wert> <ug> <og> <unit>

PH_VAR_OUT_O <objektname> <name> <wert> <ug> <og> <unit>

mit:

<name> string Bezeichnung der Variablen

<wert> float Wert der Variablen

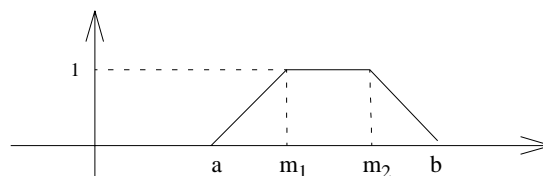
<ug> float untere Grenze des Dynamikbereichs

<og> string Einheit der Variablen

Objekte L_VAR_TRAPEZ beschreiben linguistische Werte mit trapezförmige Zugehörigkeitsfunktionen.

L_VAR_TRAPEZ <objektname> <phys> <a> <m₁> <m₂>

mit:



<phys> string Objektname der zugrundeliegenden physikalischen Variablen

<a> float Parameter a der Zugehörigkeitsfunktion - siehe Abbildung

<m₁> float Parameter m₁ der Zugehörigkeitsfunktion - siehe Abbildung

<m₂> string Parameter m₂ der Zugehörigkeitsfunktion - siehe Abbildung

 string Parameter b der Zugehörigkeitsfunktion - siehe Abbildung

Die Bewertung der Länge einer Kante hängt von der physikalischen Ausdehnung eines Graphen ab. Die hier angegebenen linguistischen Werte für length gelten für das in 6.1.2 beschriebene Layer-Netzwerk.

*/*physikalische Input-Variablen */*

/ Auslastung Knoten A */*

/ linguistische Variable */*

PH_VAR_IN_O ndLsA ndLsA 0 0 100 prozent

/ linguistische Werte */*

L_VAR_TRAPEZ ndLsA_s ndLsA 0 0 20 40 !

L_VAR_TRAPEZ ndLsA_m ndLsA 20 40 60 80 !

L_VAR_TRAPEZ ndLsA_b ndLsA 60 80 100 100 !

/ Auslastung Knoten B */*

/ linguistische Variable */*

PH_VAR_IN_O ndLsB ndLsB 0 0 100 prozent

/ linguistische Werte */*

L_VAR_TRAPEZ ndLsB_s ndLsB 0 0 20 40 !

L_VAR_TRAPEZ ndLsB_m ndLsB 20 40 60 80 !

L_VAR_TRAPEZ ndLsB_b ndLsB 60 80 100 100 !

/ Knotenart Knoten A */*

/ linguistische Variable */*

PH_VAR_IN_O ndCIA ndCIA 1 -0.75 1.75 prozent

/ linguistische Werte */*

L_VAR_TRIANGLE ndCIA_nfs ndCIA -0.75 0 0.75 !

L_VAR_TRIANGLE ndCIA_fs ndCIA 0.25 1 1.75 !

/ Knotenart Knoten B */*

/ linguistische Variable */*

PH_VAR_IN_O ndCIB ndCIB 1 -0.75 1.75 prozent

/ linguistische Werte */*

L_VAR_TRIANGLE ndCIB_nfs ndCIB -0.75 0 0.75 !

L_VAR_TRIANGLE ndCIB_fs ndCIB 0.25 1 1.75 !

/ Kosten KnotenA (prozentualer Anteil der Kosten an den max. Knotenkosten)*/*

/ linguistische Variable */*

PH_VAR_IN_O ndCoA ndCoA 0 0 100 prozent

*/*linguistische Werte */*

L_VAR_TRAPEZ ndCoA_s ndCoA 0 0 20 40 !

L_VAR_TRAPEZ ndCoA_m ndCoA 20 40 60 80 !

```
L_VAR_TRAPEZ ndCoA_b ndCoA 60 80 100 100 !
/* Kosten KnotenB ( prozentualer Anteil der Kosten an den max. Knotenkosten)*/
/* linguistische Variable */
PH_VAR_IN_O ndCoB ndCoB 0 0 100 prozent
/*linguistische Werte */
L_VAR_TRAPEZ ndCoB_s ndCoB 0 0 20 40 !
L_VAR_TRAPEZ ndCoB_m ndCoB 20 40 60 80 !
L_VAR_TRAPEZ ndCoB_b ndCoB 60 80 100 100 !
/* physikalische Länge der Kante */
/* linguistische Variable */
PH_VAR_IN_O length length 0 0 1500 prozent
/*linguistische Werte */
L_VAR_TRAPEZ length_s length 0 0 150 200 !
L_VAR_TRAPEZ length_m length 150 200 300 350 !
L_VAR_TRAPEZ length_b length 300 350 1500 1500 !
/* Auslastung (Belegungsgrad) */
/* linguistische Variable */
PH_VAR_IN_O beGr beGr 0 0 100 prozent
/*linguistische Werte */
L_VAR_TRAPEZ beGr_s beGr 0 0 20 40 !
L_VAR_TRAPEZ beGr_m beGr 20 40 60 80 !
L_VAR_TRAPEZ beGr_b beGr 60 80 100 100 !
/* Verfügbarkeit */
/* linguistische Variable */
PH_VAR_IN_O avai avai 100 0 100 prozent
/*linguistische Werte */
L_VAR_TRAPEZ avai_s avai 0 0 20 40 !
L_VAR_TRAPEZ avai_m avai 20 40 60 80 !
L_VAR_TRAPEZ avai_b avai 60 80 100 100 !
/* Bitfehlerhäufigkeit */
/* linguistische Variable */
PH_VAR_IN_O bitEr bitEr 0 0 100 prozent
/*linguistische Werte */
L_VAR_TRAPEZ bitEr_s bitEr 0 0 20 40 !
L_VAR_TRAPEZ bitEr_m bitEr 20 40 60 80 !
L_VAR_TRAPEZ bitEr_b bitEr 60 80 100 100 !
```

```

/* Abweichung von der Mindestverfügbarkeit */
/* linguistische Variable */
PH_VAR_IN_O deltaAvail deltaAvail 0 -100 100 prozent
/*linguistische Werte */
L_VAR_TRAPEZ deltaAvail_bad deltaAvail -100 -90 -1 0 !
L_VAR_TRAPEZ deltaAvail_good deltaAvail -1 0 20 30 !
L_VAR_TRAPEZ deltaAvail_ecoBad deltaAvail 20 30 100 100 !
/* Abweichung von der max. erlaubten Bitfehlerhäufigkeit */
/* linguistische Variable */
PH_VAR_IN_O deltaBitErr deltaBitErr 0 -100 100 prozent
/*linguistische Werte */
L_VAR_TRAPEZ deltaBitErr_bad deltaBitErr -100 -90 -1 0 !
L_VAR_TRAPEZ deltaBitErr_good deltaBitErr -1 0 20 30 !
L_VAR_TRAPEZ deltaBitErr_ecoBad deltaBitErr 20 30 100 100 !
/* strategische Bedeutung */
/* linguistische Variable */
PH_VAR_IN_O stratMean stratMean 0 0 10 prozent
/*linguistische Werte */
L_VAR_TRAPEZ stratMean_s stratMean 0 0 3 4 !
L_VAR_TRAPEZ stratMean_m stratMean 3 4 7 8 !
L_VAR_TRAPEZ stratMean_b stratMean 7 8 10 10 !

/* physikalische Output-Variablen */
/* Kantenwert */
/* linguistische Variable */
PH_VAR_OUT_O edgeVal edgeVal 1 -11 61 norm
/*linguistische Werte ( s = very good, m = good, b = bad ) */
/* so sind die zu erreichenden Werte zwischen 0 und 50 */
/*willkürlich festgelegter Wertebereich */
L_VAR_TRAPEZ edgeVal_s edgeVal -11 -11 0 20 !
L_VAR_TRAPEZ edgeVal_m edgeVal 10 20 30 40 !
L_VAR_TRAPEZ edgeVal_b edgeVal 25 50 61 61 !

```

Regeln

Im folgenden sind die Regeln aufgelistet in der Kommandosprache von NEUROTool. Die Zeile

```
RULE_O rLength1 rLength1 0.5 edgeVal_s length_s !
```

bedeutet

```
0.5 * IF rLength=length_s THEN edgeVal=edgeVal_s
```

wobei 0.5 das Regelgewicht ist. rLength1 ist der Objektname der Regel.

```
/* Regeln */
```

```
/* betriebliche Regeln */
```

```
/* Kanten mit großer physikalischer Länge sind eher abzulehnen und solche mit kleiner Länge zu bevorzugen */
```

```
RULE_O rLength1 rLength1 0.5 edgeVal_s length_s !
```

```
RULE_O rLength2 rLength2 0.5 edgeVal_m length_m !
```

```
RULE_O rLength3 rLength3 0.5 edgeVal_b length_b !
```

```
/* Kanten zwischen Knoten, die fernschaltbar sind, sind gegenüber den Kanten zwischen nicht fernschaltbaren zu bevorzugen*/
```

```
RULE_O rFs1 rFs1 0.5 edgeVal_b ndCIA_nfs AND ndCIB_nfs
```

```
RULE_O rFs2 rFs2 0.5 edgeVal_b ndCIA_nfs AND ndCIB_fs
```

```
RULE_O rFs3 rFs3 0.5 edgeVal_b ndCIA_fs AND ndCIB_nfs
```

```
RULE_O rFs4 rFs4 0.5 edgeVal_s ndCIA_fs AND ndCIB_fs
```

```
/* Knoten mit hohen Kosten sind abzulehnen und solche mit niedrigen Kosten sind zu bevorzugen.*/
```

```
RULE_O rKost1 rKost1 0.5 edgeVal_s ndCoA_s AND ndCoB_s
```

```
RULE_O rKost2 rKost2 0.5 edgeVal_s ndCoA_s AND ndCoB_m
```

```
RULE_O rKost3 rKost3 0.5 edgeVal_b ndCoA_s AND ndCoB_b
```

```
RULE_O rKost4 rKost4 0.5 edgeVal_m ndCoA_m AND ndCoB_s
```

```
RULE_O rKost5 rKost5 0.5 edgeVal_m ndCoA_m AND ndCoB_m
```

```
RULE_O rKost6 rKost6 0.5 edgeVal_b ndCoA_m AND ndCoB_b
```

```
RULE_O rKost7 rKost7 0.5 edgeVal_b ndCoA_b AND ndCoB_s
```

```
RULE_O rKost8 rKost8 0.5 edgeVal_b ndCoA_b AND ndCoB_m
```

```
RULE_O rKost9 rKost9 0.5 edgeVal_b ndCoA_b AND ndCoB_b
```

```
/* Kaum ausgelastete Knoten sollen nicht mehr genutzt werden, um sie eventuell ganz frei schalten zu können und an einem anderen Ort einsetzen zu können. */
```

```
/*Kanten mit sehr kleiner Auslastung, sind abzulehnen.*/
```

```
/*Knoten und Kanten, die zu ca. 50% ausgelastet sind, sind zu bevorzugen.*/
```

RULE_O rAusL1 rAusL1 0.5 edgeVal_b ndLsA_s AND ndLsB_s

RULE_O rAusL2 rAusL2 0.5 edgeVal_m ndLsA_s AND ndLsB_m

RULE_O rAusL3 rAusL3 0.5 edgeVal_b ndLsA_s AND ndLsB_b

RULE_O rAusL4 rAusL4 0.5 edgeVal_m ndLsA_m AND ndLsB_s

RULE_O rAusL5 rAusL5 0.5 edgeVal_s ndLsA_m AND ndLsB_m

RULE_O rAusL6 rAusL6 0.5 edgeVal_m ndLsA_m AND ndLsB_b

RULE_O rAusL7 rAusL7 0.5 edgeVal_m ndLsA_b AND ndLsB_s

RULE_O rAusL8 rAusL8 0.5 edgeVal_m ndLsA_b AND ndLsB_m

RULE_O rAusL9 rAusL9 0.5 edgeVal_b ndLsA_b AND ndLsB_b

/ Kanten mit niedrigem Belegungsgrad sind eher abzulehnen */*

RULE_O rBeleg1 rBeleg1 0.5 edgeVal_b beGr_s AND beGr_s

/ Kanten, die mittel ausgelastet sind, sind zu bevorzugen */*

RULE_O rBeleg2 rBeleg2 0.5 edgeVal_s beGr_m AND beGr_m

/ Kanten, die fast voll sind, sind abzulehnen */*

RULE_O rBeleg3 rBeleg3 0.5 edgeVal_b beGr_b AND beGr_b

/ Kanten mit kleiner Bitfehlerhäufigkeit sind zu bevorzugen. */*

/ Wenn die Bitfehlerhäufigkeit eher hoch ist, dann wähle Kante nicht */*

RULE_O rBitEr1 rBitEr1 0.5 edgeVal_s bitEr_s !

RULE_O rBitEr2 rBitEr2 0.5 edgeVal_m bitEr_m !

RULE_O rBitEr3 rBitEr3 0.5 edgeVal_b bitEr_b !

/ Kanten mit hoher Verfügbarkeit sind zu bevorzugen.*/*

RULE_O rAvail1 rAvail1 0.5 edgeVal_s avai_b !

RULE_O rAvail2 rAvail2 0.5 edgeVal_m avai_m !

RULE_O rAvail3 rAvail3 0.5 edgeVal_b avai_s !

/ Verfügbarkeit muß größer als die Mindestverfügbarkeit sein */*

/ Ist die Abweichung von der Mindestverfügbarkeit positiv, aber gering (deltaAvali_good), so ist die Kante zu bevorzugen.*

Ist die Abweichung von der Mindestverfügbarkeit negativ (deltaAvali_bad), so ist die Kante abzulehnen.

Ist die Abweichung von der Mindestverfügbarkeit positiv, aber sehr groß (deltaAvali_ecoBad), so ist die Kante abzulehnen./*

RULE_O rdeltaAvail1 rdeltaAvail1 0.9 edgeVal_s deltaAvail_good !

```

RULE_O rdeltaAvail2 rdeltaAvail2 0.9 edgeVal_b deltaAvail_bad !
RULE_O rdeltaAvail3 rdeltaAvail3 0.7 edgeVal_b deltaAvail_ecoBad !
/* Die Bitfehlerhäufigkeit darf nicht größer sein als die dem Kunden garantierte maximale
Bitfehlerhäufigkeit */
und entsprechendes Verhalten wie bei Verfügbarkeit*/
RULE_O rdeltaBitErr1 rdeltaBitErr1 0.9 edgeVal_s deltaBitErr_good !
RULE_O rdeltaBitErr2 rdeltaBitErr2 0.9 edgeVal_b deltaBitErr_bad !
RULE_O rdeltaBitErr3 rdeltaBitErr3 0.7 edgeVal_b deltaBitErr_ecoBad !
/* Knoten mit hoher strategischer Bedeutung sind zu bevorzugen */
/* dient dazu Knoten mit sehr kleiner Auslastung zu nutzen */
/* notwendig für neu aufgebaute Knoten */
RULE_O rStratMean1 rStratMean1 1.0 edgeVal_s stratMean_b !
RULE_O rStratMean2 rStratMean2 1.0 edgeVal_b stratMean_s !
RULE_O rStratMean3 rStratMean3 0.5 edgeVal_b stratMean_m !

```

Regeln nach Umgruppierung

Nach der in 4.2 „Defuzzifizierungsmethode“ beschriebenen Umgruppierung der Regeln entsteht das folgende Regelsystem. Der besseren Lesbarkeit halber ist die Kommandosprache in äquivalente IF - THEN Ausdrücke umformuliert worden.

```

/* Regeln */
/* betriebliche Regeln */
/* Kanten mit großer physikalischer Länge sind eher abzulehnen und solche mit kleiner
Länge zu bevorzugen */
0.5 * IF length = length_s THEN edgeVal = edgeVal_s
0.5 * IF length = length_m THEN edgeVal = edgeVal_m
0.5 * IF length = length_b THEN edgeVal = edgeVal_b
/* Kanten zwischen Knoten, die fernschaltbar sind, sind gegenüber den Kanten zwischen
nicht fernschaltbaren zu bevorzugen*/
0.5 * IF (ndCIA = ndCIA_nfs AND ndCIB = ndCIB_nfs)
      OR (ndCIA = ndCIA_nfs AND ndCIB = ndCIB_fs)
      OR (ndCIA = ndCIA_fs AND ndCIB = ndCIB_nfs)
      THEN edgeVal = edgeVal_b
0.5 * IF ndCIA = ndCIA_fs AND ndCIB = ndCIB_fs THEN edgeVal = edgeVal_s
/* Knoten mit hohen Kosten sind abzulehnen und solche mit niedrigen Kosten sind zu bevor-
zugen.*/
0.5 * IF ndCoA = ndCoA_s AND ndCoB = ndCoB_s
      OR (ndCoA = ndCoA_s AND ndCoB = ndCoB_m)
      OR (ndCoA = ndCoA_m AND ndCoB = ndCoB_s)

```

THEN edgeVal = edgeVal_s

0.5 * **IF** ndCoA = ndCoA_m **AND** ndCoB = ndCoB_m **THEN** edgeVal = edgeVal_m

0.5 * **IF** ndCoA = ndCoA_s **AND** ndCoB = ndCoB_b

OR (ndCoA = ndCoA_m **AND** ndCoB = ndCoB_b)

OR (ndCoA = ndCoA_b **AND** ndCoB = ndCoB_s)

OR (ndCoA = ndCoA_b **AND** ndCoB = ndCoB_m)

OR (ndCoA = ndCoA_b **AND** ndCoB = ndCoB_b)

THEN edgeVal = edgeVal_b

/ Kaum ausgelastete Knoten sollen nicht mehr genutzt werden, um sie evtl. ganz frei schalten zu können und an einem anderen Ort einsetzen zu können. */*

*/*Kanten mit sehr kleiner Auslastung sind abzulehnen.*/*

*/*Knoten und Kanten, die zu ca. 50% ausgelastet sind, sind zu bevorzugen.*/*

0.5 * **IF** ndLsA = ndLsA_m **AND** ndLsB = ndLsB_m **THEN** edgeVal = edgeVal_s

0.5 * **IF** ndLsA = ndLsA_s **AND** ndLsB = ndLsB_m

OR (ndLsA = ndLsA_m **AND** ndLsB = ndLsB_s)

OR (ndLsA = ndLsA_m **AND** ndLsB = ndLsB_b)

OR (ndLsA = ndLsA_b **AND** ndLsB = ndLsB_s)

OR (ndLsA = ndLsA_b **AND** ndLsB = ndLsB_m)

THEN edgeVal = edgeVal_m

0.5 * **IF** ndLsA = ndLsA_s **AND** ndLsB = ndLsB_s

OR (ndLsA = ndLsA_s **AND** ndLsB = ndLsB_b)

OR (ndLsA = ndLsA_b **AND** ndLsB = ndLsB_b)

THEN eedgeVal = edgeVal_b

/ Kanten mit niedrigem Belegungsgrad sind eher abzulehnen */*

0.5 * **IF** beGr = beGr_s **THEN** edgeVal = edgeVal_b

/ Kanten die mittle ausgelastet sind bevorzugen */*

0.5 * **IF** beGr = beGr_m **THEN** edgeVal = edgeVal_s

/ Kanten die fast voll sind abzulehnen */*

0.5 * **IF** beGr = beGr_b **THEN** edgeVal = edgeVal_b

*/*Kanten mit kleiner Bitfehlerhäufigkeit sind zu bevorzugen */*

0.5 * **IF** bitEr = bitEr_s **THEN** edgeVal = edgeVal_s

0.5 * **IF** bitEr = bitEr_m **THEN** edgeVal = edgeVal_m

0.5 * **IF** bitEr = bitEr_b **THEN** edgeVal = edgeVal_b

/ Kanten mit hoher Verfügbarkeit sind zu bevorzugen.*/*

0.5 * **IF** avai = avai_b **THEN** edgeVal = edgeVal_s

0.5 * **IF** avai = avai_m **THEN** edgeVal = edgeVal_m

0.5 * **IF** avai = avai_s **THEN** edgeVal = edgeVal_b

/ Ist die Abweichung von der Mindestverfügbarkeit positiv, aber gering (deltaAvali_good), so ist die Kante zu bevorzugen.*

Ist die Abweichung von der Mindestverfügbarkeit negativ (deltaAvali_bad), so ist die Kante

abzulehnen.

Ist die Abweichung von der Mindestverfügbarkeit positiv, aber sehr groß (deltaAvali_ecoBad), so ist die Kante abzulehnen./**

0.9 * **IF** deltaAvail = deltaAvail_good **THEN** edgeVal = edgeVal_s

0.9 * **IF** deltaAvail = deltaAvail_bad **THEN** edgeVal = edgeVal_b

0.7 * **IF** deltaAvail = deltaAvail_ecoBad **THEN** edgeVal = edgeVal_b

/ Die Bitfehlerhäufigkeit darf nicht größer sein als die dem Kunden garantierte maximale Bitfehlerhäufigkeit*

und entsprechendes Verhalten wie bei Verfügbarkeit/**

0.9 * **IF** deltaBitErr = deltaBitErr_good **THEN** edgeVal = edgeVal_s

0.9 * **IF** deltaBitErr = deltaBitErr_bad **THEN** edgeVal = edgeVal_b

0.7 * **IF** deltaBitErr = deltaBitErr_ecoBad **THEN** edgeVal = edgeVal_b

/ Knoten mit hoher strategischer Bedeutung sind zu bevorzugen */**

dient dazu Knoten mit sehr kleiner Auslastung zu nutzen

*notwendig z.B für neu aufgebaute Knoten */**

1.0 * **IF** stratMean = stratMean_b **THEN** edgeVal = edgeVal_s

1.0 * **IF** stratMean = stratMean_s **THEN** edgeVal = edgeVal_b

0.5 * **IF** stratMean = stratMean_m **THEN** edgeVal = edgeVal_b

ANHANG B*Daten des Layer-Netzwerks*

Die beiden folgenden Tabellen enthalten die Daten der Kanten und Knoten des in 6.1.2 beschriebenen Graphen.

Daten der Kanten

Kantenname	strat-Mean	length	beGr	avai	deltaAvail	bitEr	delta-BitErr	edge-Cost
DA - Bn	5	119	40.00	67	7.00	25	15.00	37.00
Dresden - Kiel	5	182	40.00	27	-33.00	79	-39.00	87.00
Mch1 - DA	5	357	40.00	30	-30.00	71	-31.00	72.00
Bn - Kiel	5	294	40.00	34	-26.00	60	-20.00	74.00
Nb - Kiel	5	247	40.00	4	-56.00	55	-15.00	72.00
A - DA	5	211	40.00	70	10.00	55	-15.00	69.00
H - F	5	452	40.00	27	-33.00	17	23.00	78.00
F - HH	5	516	40.00	77	17.00	40	0.00	58.59
HH - A	5	358	40.00	33	-27.00	6	34.00	74.00
Bn - H	5	268	40.00	33	-27.00	48	-8.00	70.00
Nb - H	5	203	40.00	64	4.00	98	-58.00	68.00
A - Mch1	5	158	40.00	10	-50.00	89	-49.00	102.00
H - HH	5	214	40.00	98	38.00	94	-54.00	78.00
Nb - Mch1	5	119	40.00	70	10.00	10	30.00	67.00
Nb - Ro	5	297	40.00	80	20.00	12	28.00	30.00
Mch1 - Ro	5	248	40.00	30	-30.00	10	30.00	67.00

Tab. B.1 Daten der Kanten des Layer-Netzwerks

Kantenname	strat-Mean	length	beGr	avai	deltaAvail	bitEr	delta-BitErr	edge-Cost
Krakau - Mo	5	262	40.00	30	-30.00	90	-50.00	89.00
Tr - Bn	5	127	40.00	80	20.00	30	10.00	37.00
Tr - DA	5	195	40.00	30	-30.00	80	-40.00	40.00
Tr - F	5	249	40.00	80	20.00	20	20.00	45.00
Tr - Ko	5	133	40.00	80	20.00	60	-20.00	40.00
Ko - DA	5	118	40.00	60	0.00	50	-10.00	40.00
A - H	5	195	40.00	80	20.00	70	-30.00	102.00
Kiel - Rost	5	194	40.00	89	29.00	15	25.00	72.00
Rost - Ost	5	128	40.00	90	30.00	12	28.00	35.00
Ost - Mo	5	262	40.00	70	10.00	15	25.00	50.00
Nb - Pa	5	288	40.00	70	10.00	18	22.00	40.00
Pa - Ro	5	164	40.00	80	20.00	18	22.00	40.00
Dresden - Pa	5	318	40.00	30	-30.00	79	-39.00	55.00
Pa - Mo	5	287	40.00	60	0.00	34	6.00	55.00
DA - Ulm	5	213	40.00	87	27.00	12	28.00	35.00
Ulm - Mch2	5	185	40.00	80	20.00	7	33.00	31.00
Mch2 - Ro	5	264	40.00	65	5.00	40	0.00	31.00
Ulm - Mch1	5	205	40.00	70	10.00	7	33.00	67.00
Ulm - La	5	94	40.00	80	20.00	15	25.00	31.00
La - Mch3	5	109	40.00	60	0.00	30	10.00	31.00
Mch3 - Ro	5	286	40.00	65	5.00	35	5.00	30.00
Dresden-Krakau	5	161	40.00	60	0.00	40	0.00	89.00
Rost - Danzig	5	128	40.00	70	10.00	30	10.00	30.00
Danzig - Mo	5	243	40.00	60	0.00	40	0.00	45.00
Pa - Athen	5	188	40.00	70	10.00	20	20.00	80.00
Mo - Thes	5	203	40.00	60	0.00	40	0.00	40.00
Mo - Athen	5	309	40.00	80	20.00	15	25.00	85.00
Athen - Samos	5	224	40.00	70	10.00	20	20.00	65.00
Thes - Athen	5	209	40.00	10	-50.00	80	-40.00	65.00
Thes - Samos	5	176	40.00	60	0.00	50	-10.00	20.00
Danzig -Krakau	5	75	40.00	40	-20.00	20	20.00	74.00

Tab. B.1 Daten der Kanten des Layer-Netzwerks

Kantenname	strat-Mean	length	beGr	avai	deltaAvail	bitEr	delta-BitErr	edge-Cost
Danzig - Krakau	5	75	40.00	40	-20.00	20	20.00	74.00
Rost - Len	5	146	40.00	50	-10.00	70	-30.00	65.00
Len - Danzig	5	51	40.00	80	20.00	13	27.00	65.00
Len - Mo	5	218	40.00	80	20.00	13	27.00	80.00
Mo -Thes2	5	224	40.00	80	20.00	13	27.00	70.00
Thes2 - Samos	5	180	40.00	80	20.00	13	27.00	50.00
F - Lyon	5	165	40.00	40	-20.00	80	-40.00	75.00
F - Lyon2	5	152	40.00	80	20.00	20	20.00	60.00
Lyon - Avi	5	166	40.00	60	0.00	30	10.00	85.00
Lyon2 - Avi	5	146	40.00	80	20.00	20	20.00	70.00
Avi - Pisa	5	443	40.00	70	10.00	17	23.00	70.00
Pisa - Roma	5	147	40.00	60	0.00	40	0.00	75.00
Pisa - Ro	5	158	40.00	80	20.00	25	15.00	50.00
F - Lyon3	5	184	40.00	80	20.00	20	20.00	60.00
Lyon3 - Avi	5	163	40.00	80	20.00	20	20.00	70.00
DA - H	5	149	40.00	60	0.00	13	27.00	73.00
F - Ulm	5	227	40.00	80	20.00	15	25.00	40.00
HH - Kiel2	5	175	40.00	80	20.00	10	30.00	55.00
Kiel2 - Rost	5	201	40.00	80	20.00	9	31.00	45.00
HH - Kiel	5	194	40.00	30	-30.00	60	-20.00	82.00
HH - Kiel3	5	169	40.00	80	20.00	10	30.00	55.00
Kiel3 - Rost	5	206	40.00	80	20.00	7	33.00	45.00
Rost - Dresden	5	170	40.00	70	10.00	13	27.00	45.00
F -La	5	217	40.00	60	0.00	12	28.00	41.00
F - Pisa	5	482	40.00	51	-9.00	20	20.00	60.00
F - Pisa2	5	433	40.00	60	0.00	30	10.00	55.00
Pisa2 - Roma	5	197	40.00	67	7.00	30	10.00	70.00
DA - Nb	5	317	40.00	67	7.00	35	5.00	35.00
Dresden - A	5	223	40.00	66	6.00	50	-10.00	79.00
Ulm - Mch3	5	181	40.00	60	0.00	30	10.00	30.00
Krakau - Pilion	5	247	40.00	40	-20.00	60	-20.00	104.00

Tab. B.1 Daten der Kanten des Layer-Netzwerks

Kantenname	Strat-Mean	length	beGr	avai	deltaAvail	bitEr	delta-BitErr	edge-Cost
Pilion -Thes	5	250	40.00	40	-20.00	60	-20.00	55.00
Pilion - Samos	5	339	40.00	30	-30.00	70	-30.00	55.00

Tab. B.1 Daten der Kanten des Layer-Netzwerks

Daten der Knoten:

Knotenname	ndCo	ndCl	ndLs
DA	20	1	40.00
F	30	1	40.00
HH	30	1	40.00
Dresden	40	1	40.00
Kiel	94	0	40.00
Mch1	85	0	40.00
Bn	15	1	40.00
Nb	10	0	40.00
A	79	0	40.00
H	86	0	40.00
Mo	40	1	40.00
Ro	10	1	40.00
Krakau	99	0	40.00
Tr	20	1	40.00
Ko	20	1	40.00
Rost	10	1	40.00
Ost	20	1	40.00
Pa	30	1	40.00
Ulm	10	1	40.00
Mch2	12	1	40.00
La	13	1	40.00
Mch3	10	1	40.00
Danzig	10	1	40.00
Athen	90	1	40.00

Tab. B.2 Daten der Knoten des Layer-Netzwerks

Knotenname	ndCo	ndCI	ndLs
Samos	0	1	40.00
Thes	0	1	40.00
Len	80	0	40.00
Thes2	60	1	40.00
Lyon	80	0	40.00
Lyon2	50	1	40.00
Avi	50	1	40.00
Pisa	50	0	40.00
Roma	60	1	40.00
Lyon3	50	1	40.00
Kiel2	40	1	40.00
Kiel3	40	1	40.00
Pisa2	40	1	40.00
Pilion	70	1	40.00

Tab. B.2 Daten der Knoten des Layer-Netzwerks

Der nachfolgende Text enthält ergänzende Ergebnisse zu 6.2.3.

1. Auslastungsverhalten

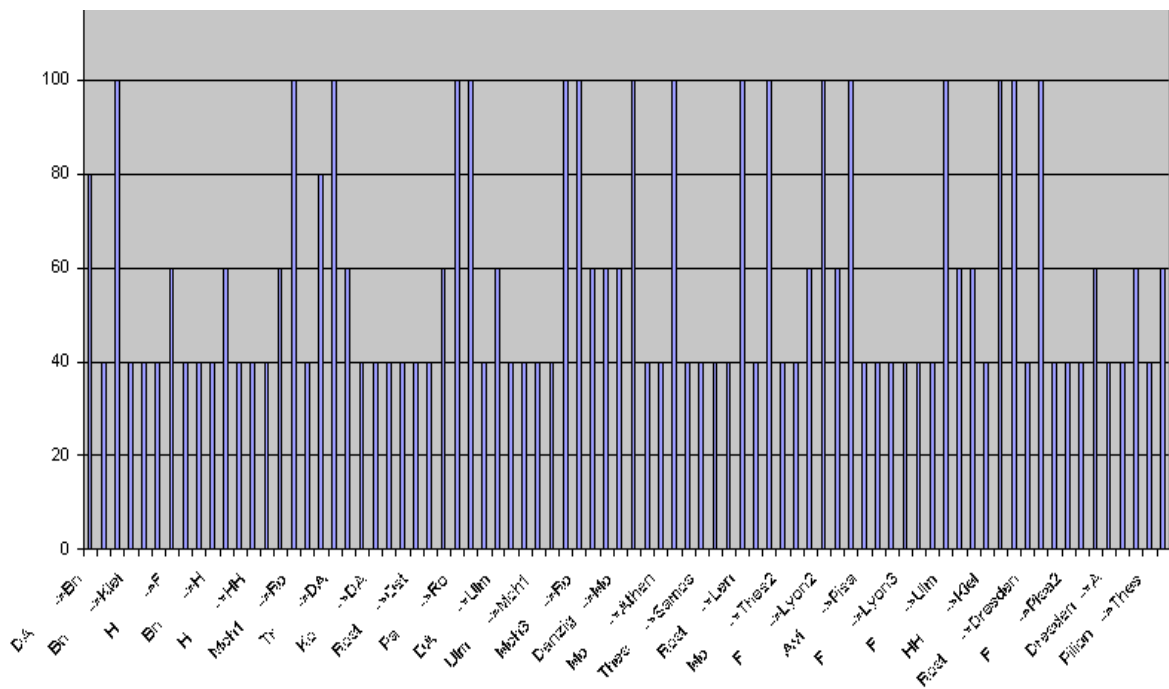


Abb. C.1 Auslastung der Kanten nach Einsatz von Length-Routing

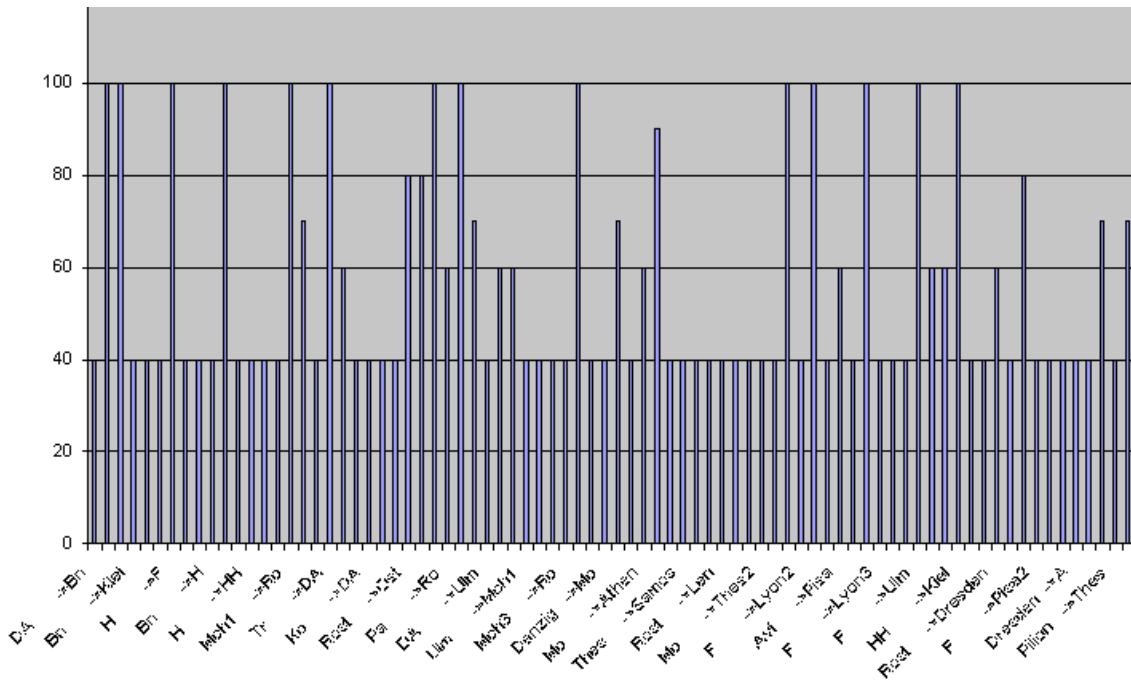


Abb. C.2 Auslastung der Kanten nach Einsatz von Hops-Routing

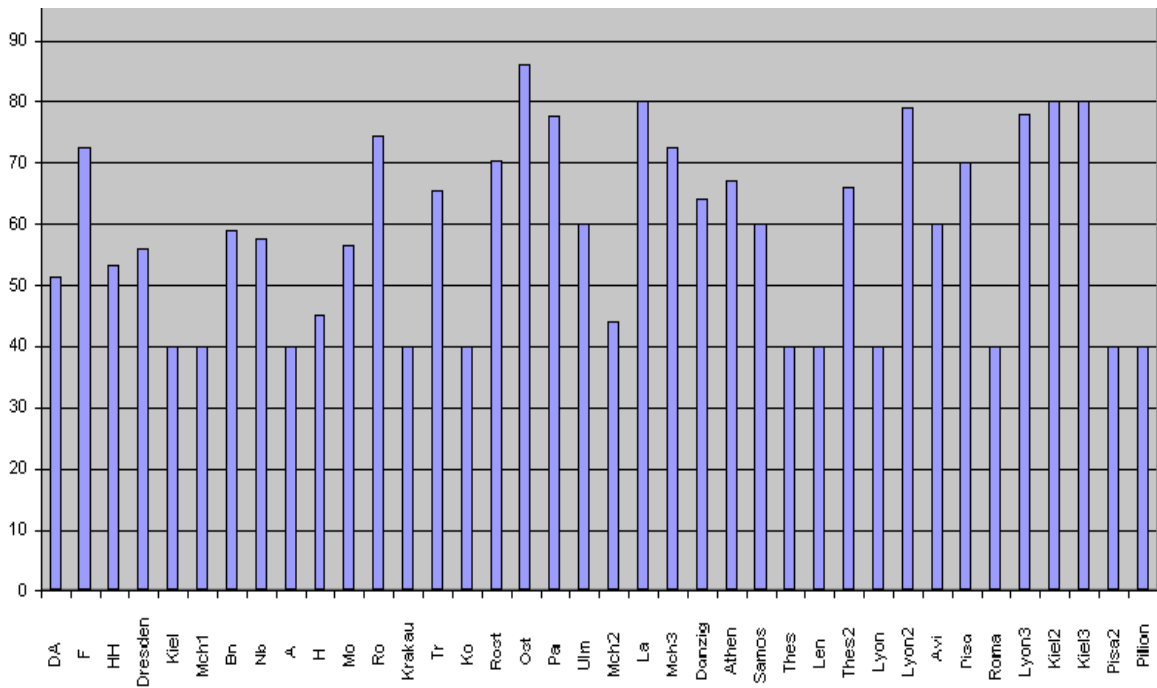


Abb. C.3 Auslastung der Knoten nach Einsatz von Fuzzy-Routing mit gleicher Gewichtung aller Regeln

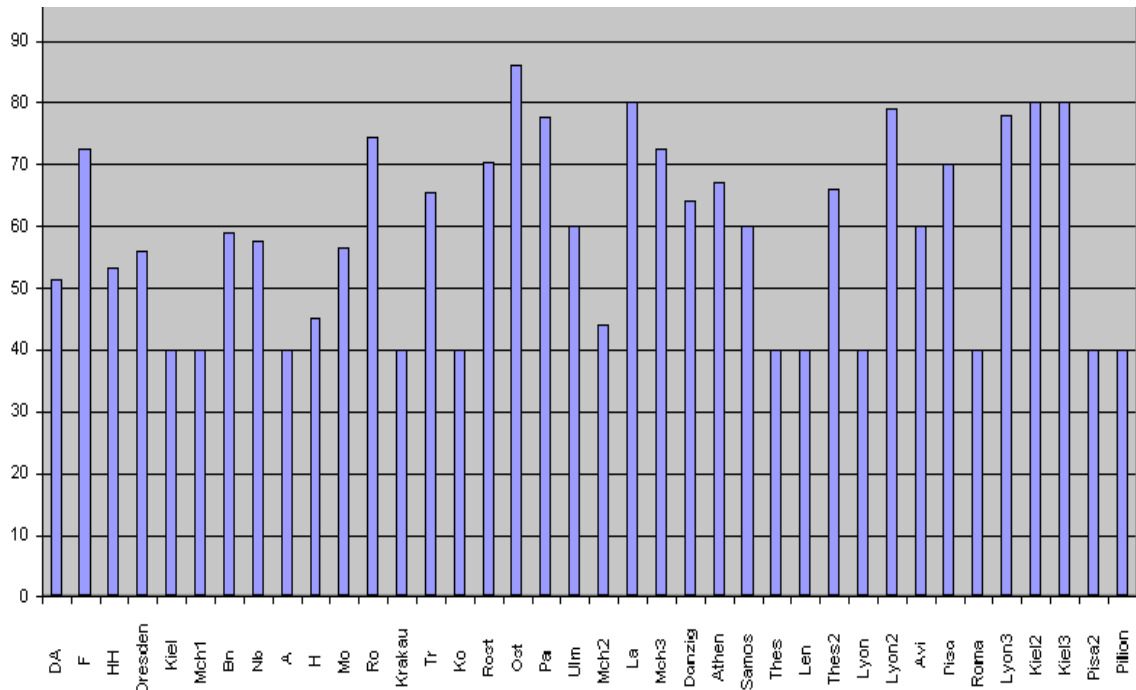


Abb. C.4 Auslastung der Knoten nach Einsatz von Fuzzy-Routing mit doppelter Gewichtung der Regeln für die Auslastung

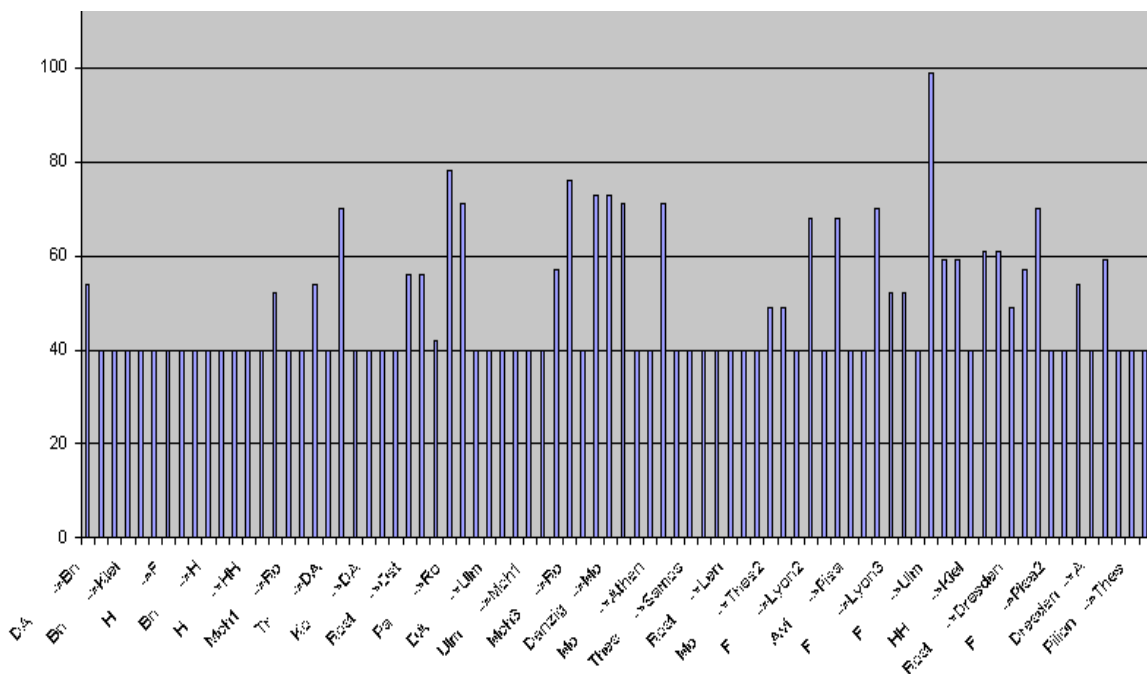


Abb. C.5 Auslastung der Kanten nach 200 Schaltungen mit Fuzzy-Routing

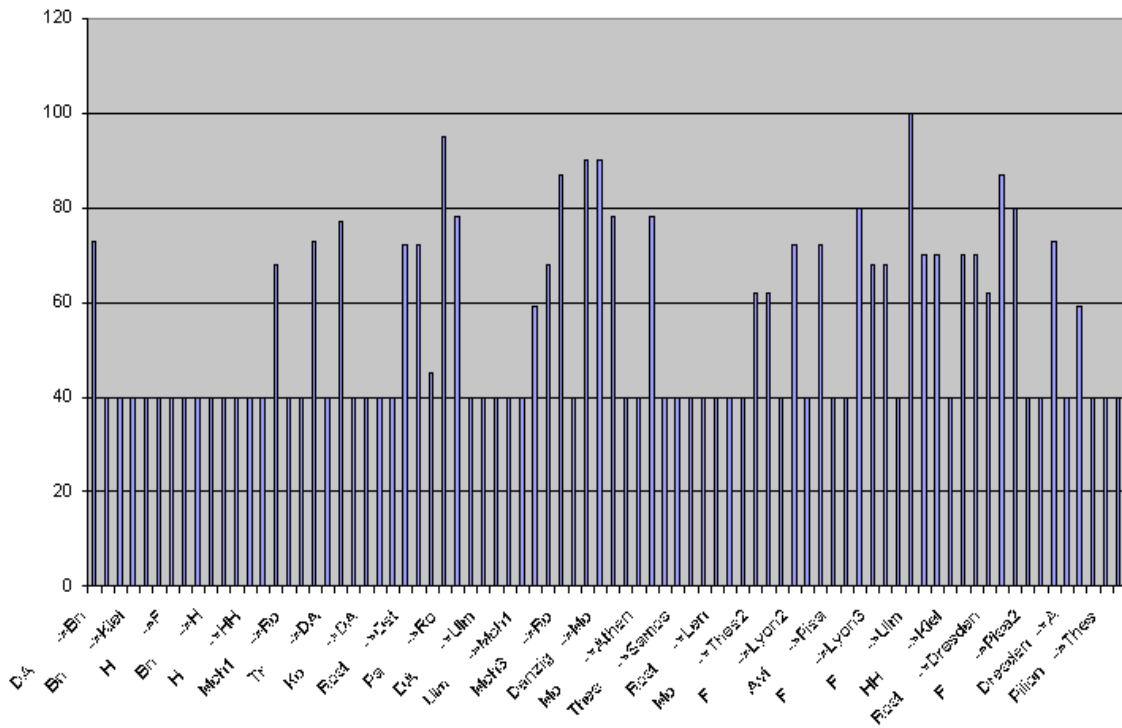


Abb. C.6 Auslastung der Kanten nach 300 Schaltungen mit Fuzzy-Routing

2. Kosten der Knoten

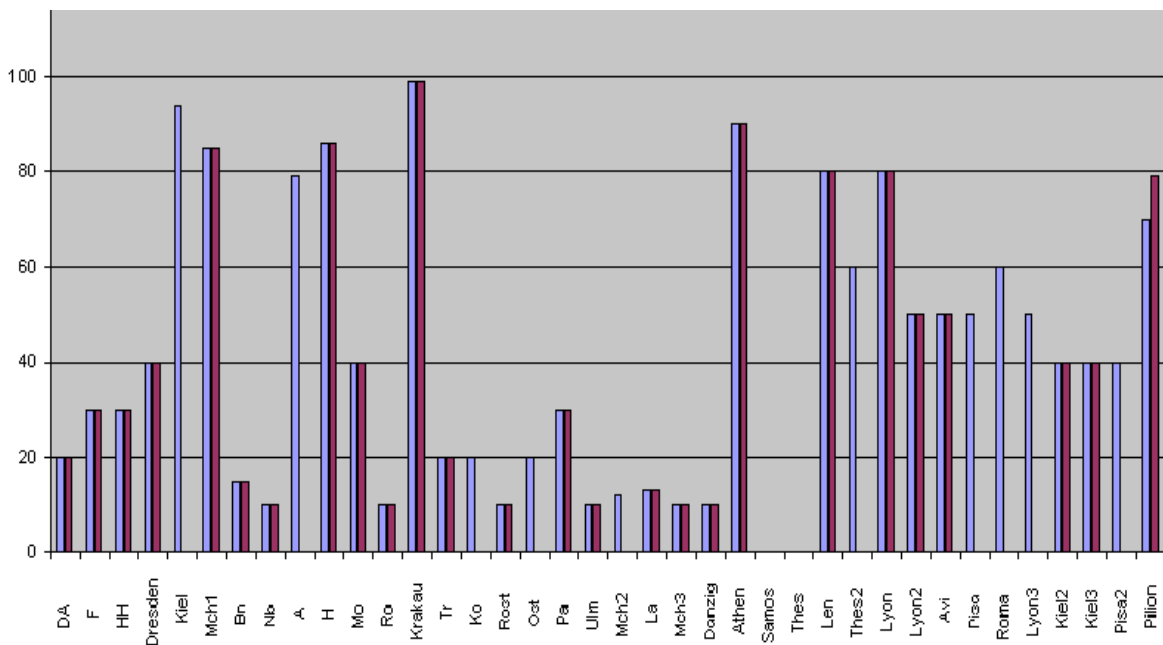


Abb. C.7 Kosten der verwendeten Knoten nach Einsatz von Length-Routing

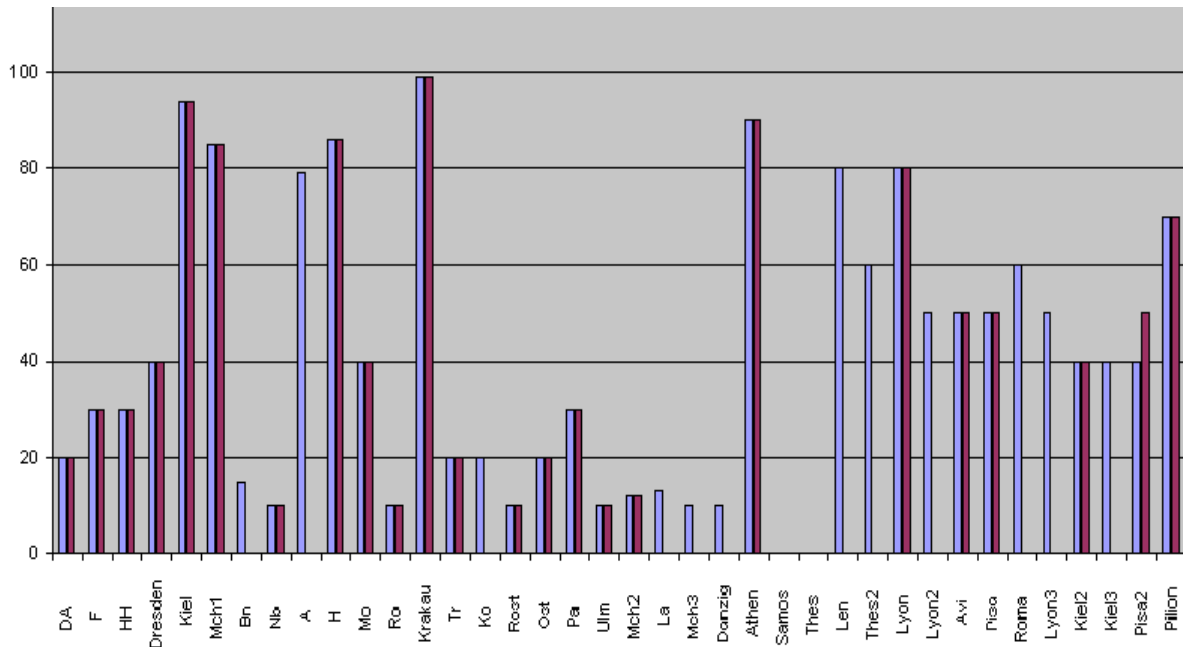


Abb. C.8 Kosten der verwendeten Knoten nach Einsatz von Hops-Routing

3. Bitfehlerhäufigkeit und Verfügbarkeit der verwendeten Kanten

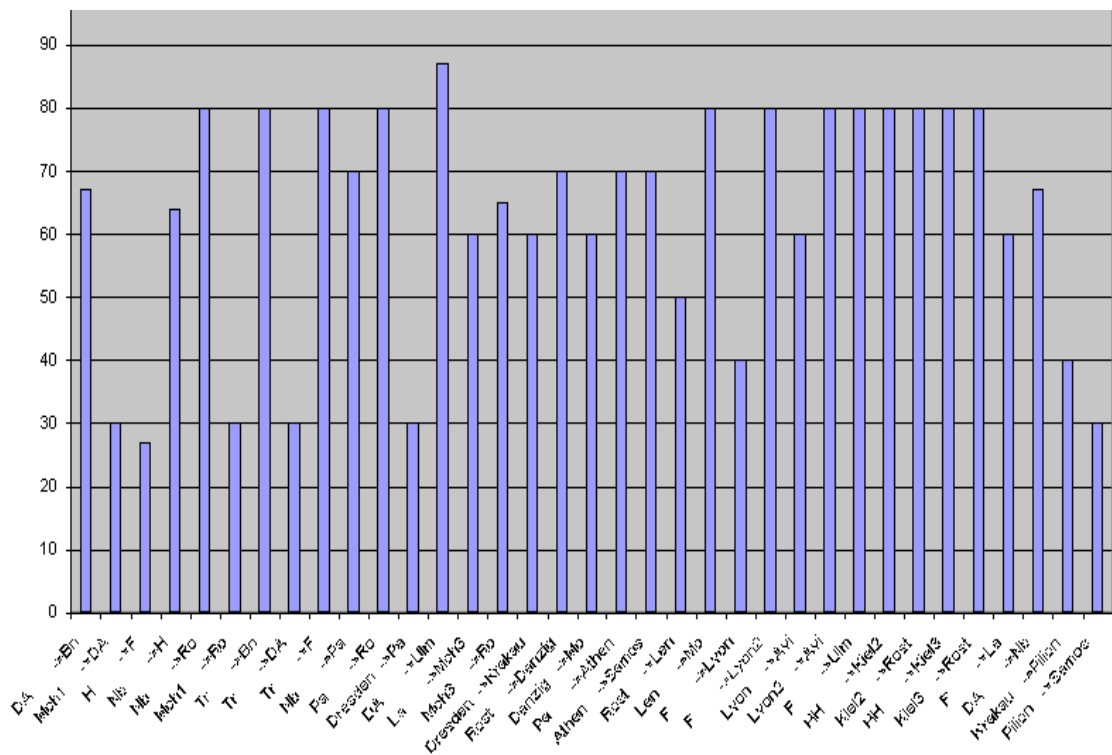


Abb. C.9 Verfügbarkeit der verwendeten Kanten mit Length-Routing

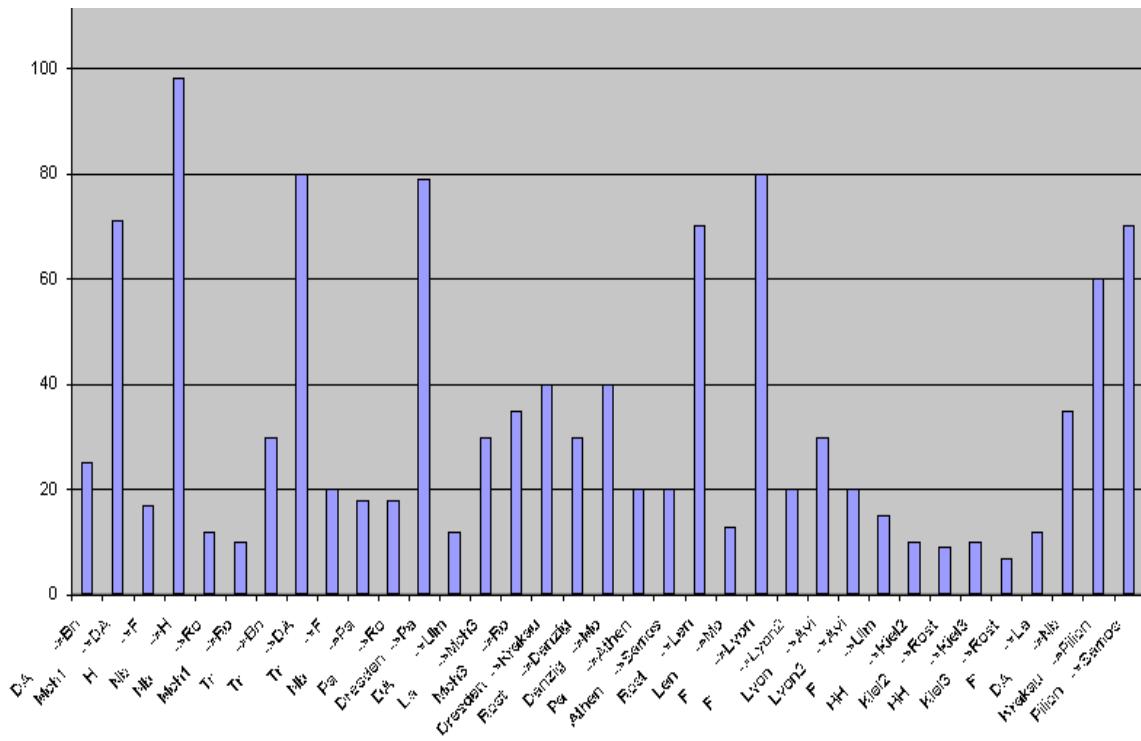


Abb. C.10 Bitfehlerhäufigkeit der verwendeten Kanten mit Length-Routing

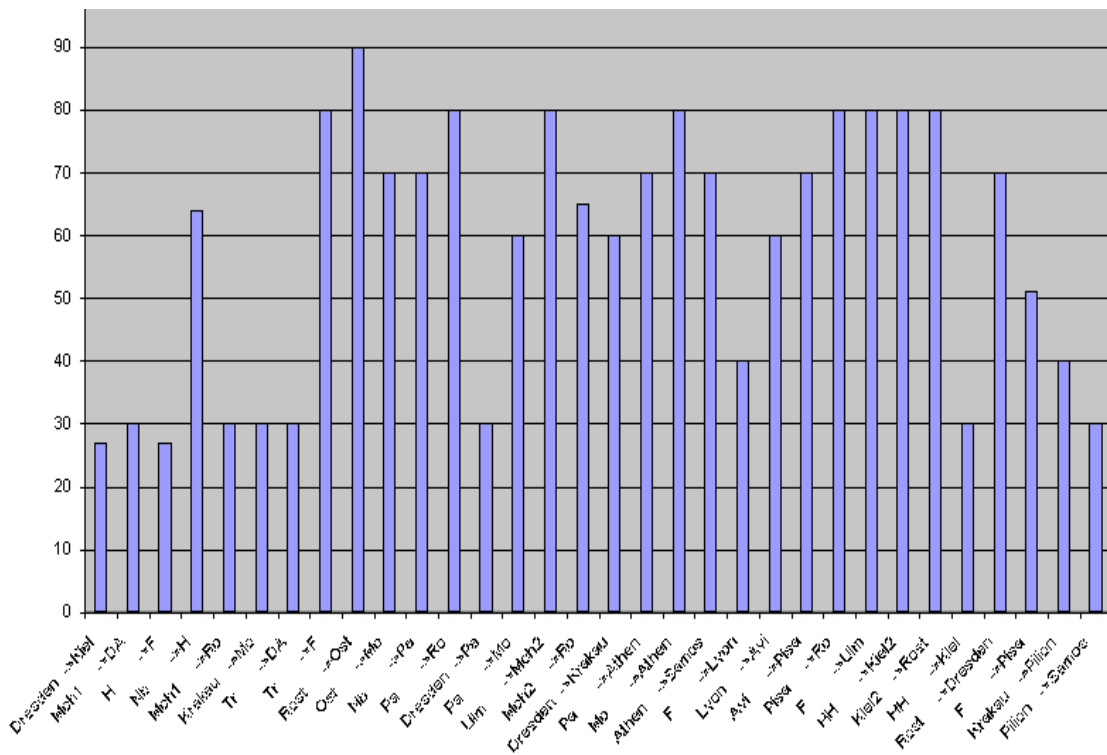


Abb. C.11 Verfügbarkeit der verwendeten Kanten mit Hop-Routing

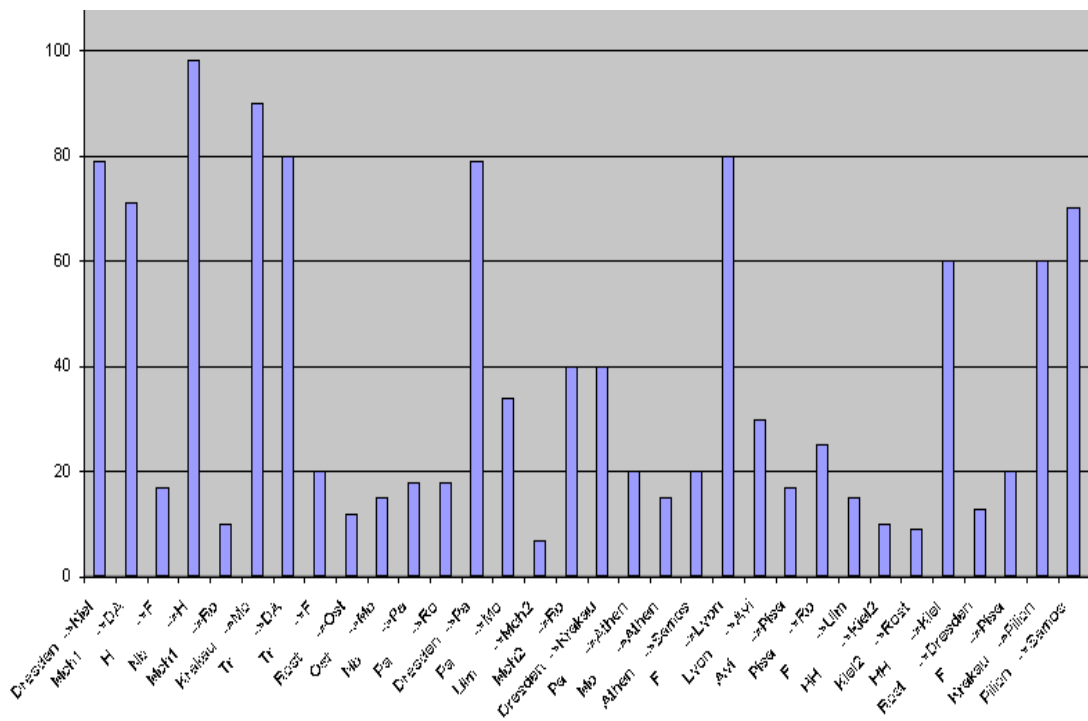


Abb. C.12 Bitfehlerhäufigkeit der verwendeten Kanten mit Hop-Routing

3. Verteilung der Bitfehlerhäufigkeit und Verfügbarkeit

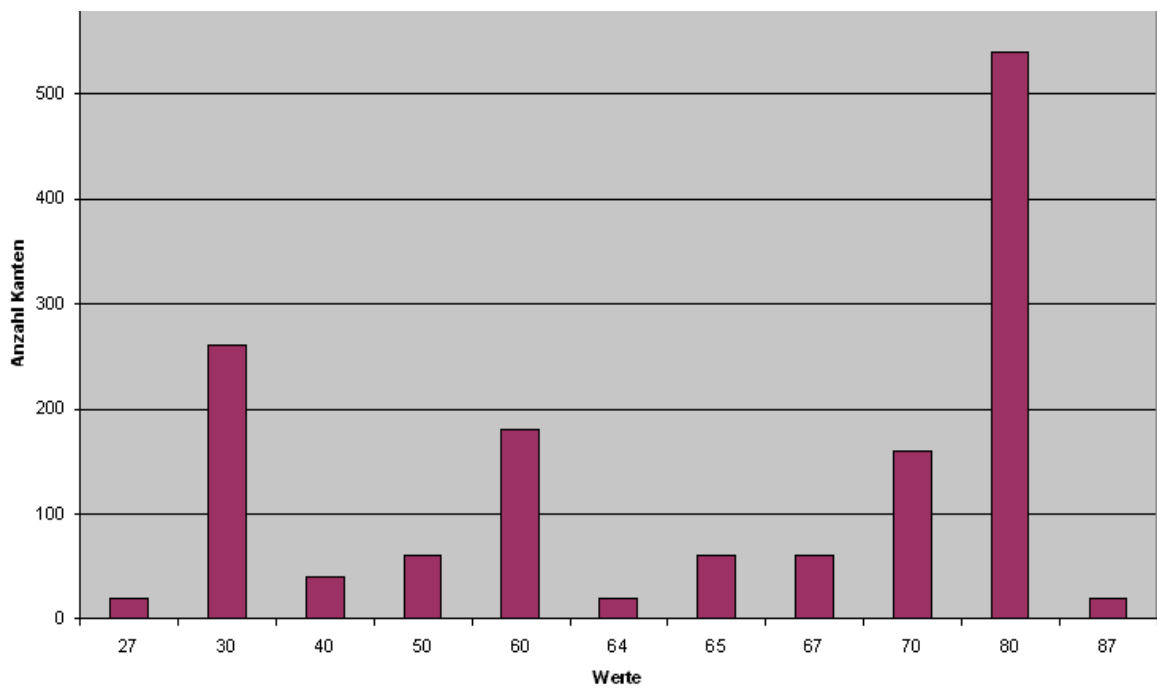


Abb. C.13 Verteilung Verfügbarkeit mit Length-Routing

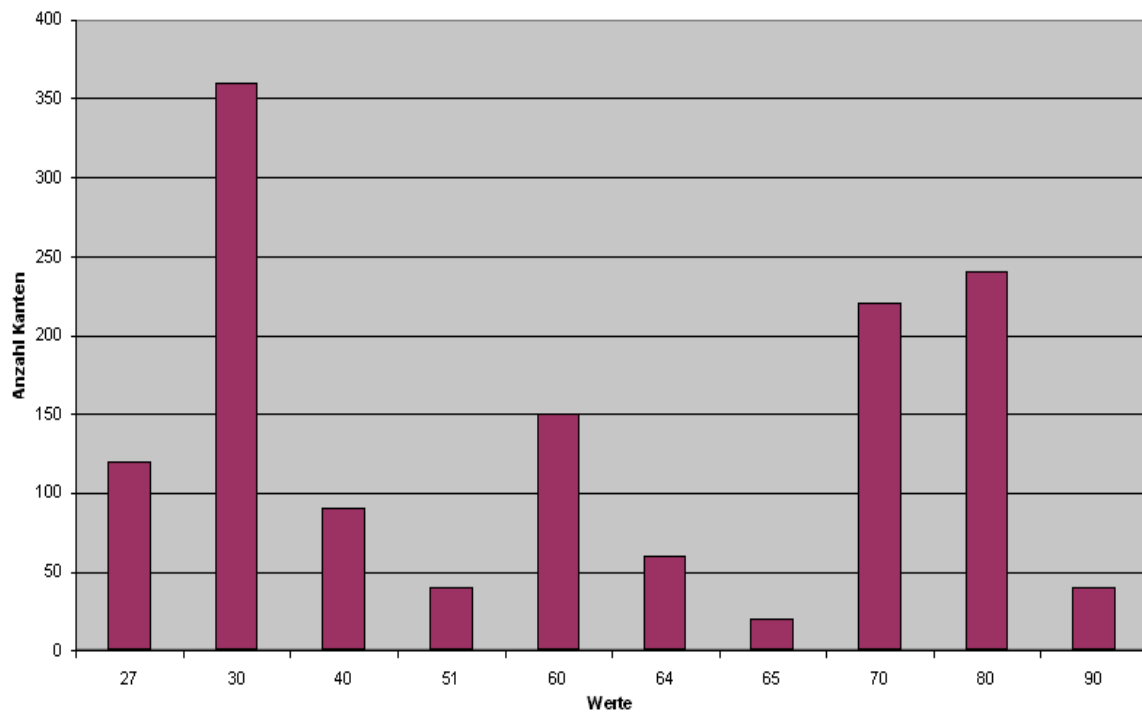


Abb. C.14 Verteilung Verfügbarkeit mit Hop-Routing

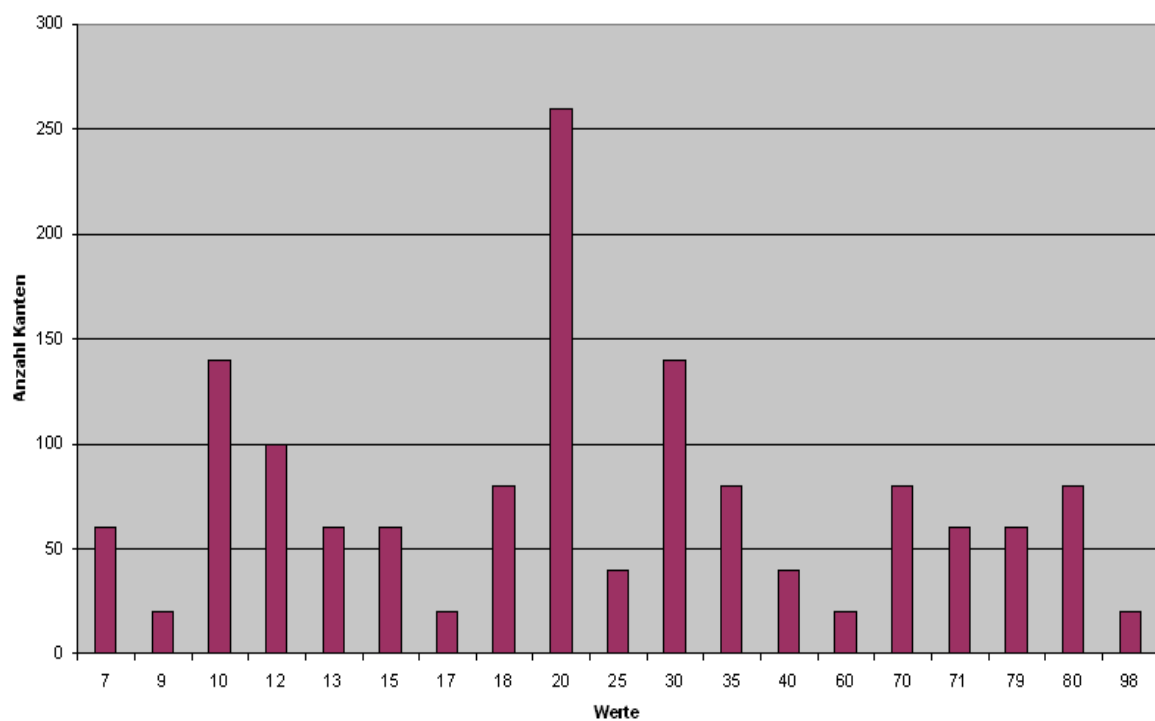


Abb. C.15 Verteilung Bitfehlerhäufigkeit mit Length-Routing

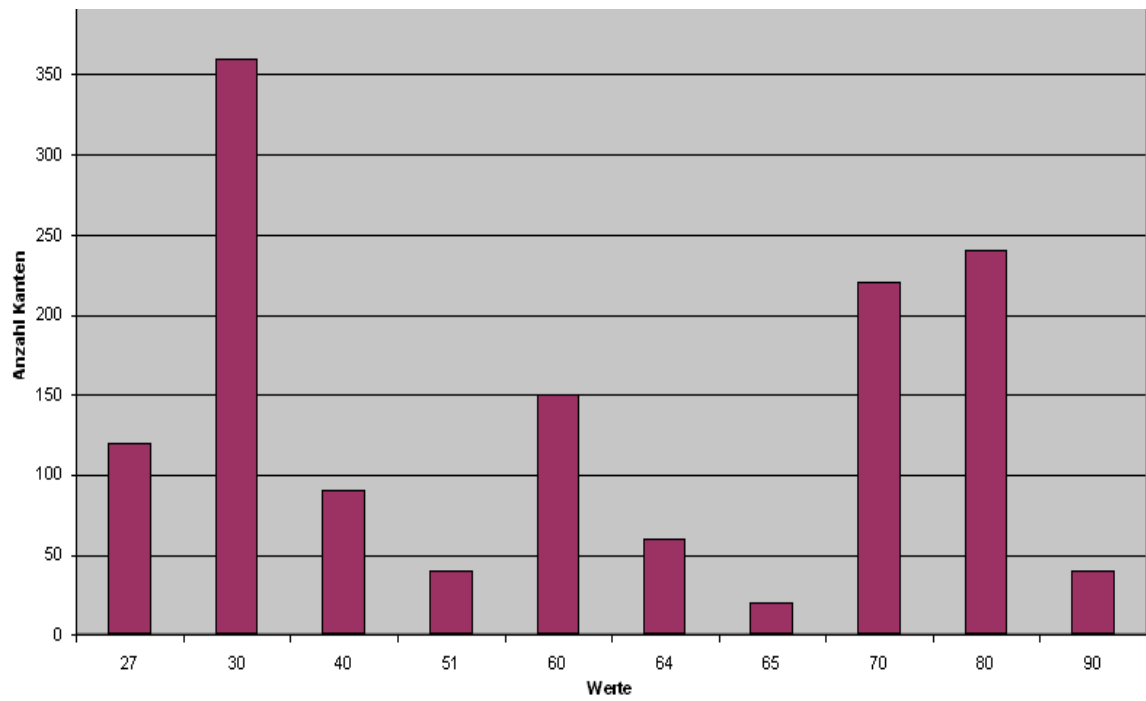


Abb. C.16 Verteilung Bitfehlerhäufigkeit mit Hop-Routing

Der nachfolgende Text enthält ergänzende Ergebnisse zu 6.2.4.

Kante/Anzahl Schaltungen	50	100	150	200	250	300	350	400	450	500
DA ->Bn	40	40	42	54	67	73	74	78	79	82
Dresden->Kiel	40	40	40	40	40	40	40	40	40	40
Mch1 ->DA	40	40	40	40	40	40	40	40	40	40
Bn ->Kiel	40	40	40	40	40	40	40	40	40	40
Nb ->Kiel	40	40	40	40	40	40	40	40	40	40
A ->DA	40	40	40	40	40	40	40	40	40	40
H ->F	40	40	40	40	40	40	45	55	72	92
F ->HH	40	40	40	40	40	40	40	40	40	40
HH ->A	40	40	40	40	40	40	40	40	40	40
Bn ->H	40	40	40	40	40	40	40	40	40	40
Nb ->H	40	40	40	40	40	40	45	55	65	75
A ->Mch1	40	40	40	40	40	40	40	40	40	40
H ->HH	40	40	40	40	40	40	40	40	40	40
Nb ->Mch1	40	40	40	40	40	40	40	40	40	40
Nb ->Ro	40	40	42	52	62	68	69	73	77	86
Mch1 ->Ro	40	40	40	40	40	40	40	40	40	40
Krakau ->Mo	40	40	40	40	40	40	40	40	40	40
Tr ->Bn	40	40	42	54	67	73	74	78	79	82
Tr ->DA	40	40	40	40	40	40	40	47	59	66
Tr ->F	50	60	68	70	73	77	86	97	100	100
Tr ->Ko	40	40	40	40	40	40	40	40	40	40
Ko ->DA	40	40	40	40	40	40	40	40	40	40
A ->H	40	40	40	40	40	40	40	40	40	40
Kiel ->Rost	40	40	40	40	40	40	40	40	40	40
Rost ->Ost	40	40	40	56	69	72	76	86	96	100
Ost ->Mo	40	40	40	56	69	72	76	86	96	100
Nb ->Pa	40	40	40	42	45	45	50	60	70	80
Pa ->Ro	50	60	70	78	85	95	100	100	100	100
Dresden ->Pa	50	60	69	71	74	78	84	94	100	100

Tab. C.1 Auslastung der Kanten bei Einsatz von Fuzzy-Routing in Prozent

Kante/Anzahl Schaltungen	50	100	150	200	250	300	350	400	450	500
Pa ->Mo	40	40	40	40	40	40	40	40	40	40
DA ->Ulm	40	40	40	40	40	40	40	47	63	74
Ulm ->Mch2	40	40	40	40	40	40	40	44	50	51
Mch2 ->Ro	40	40	40	40	40	40	40	44	50	51
Ulm ->Mch1	40	40	40	40	40	40	40	40	40	40
Ulm ->La	40	40	40	40	49	59	69	70	70	70
La ->Mch3	41	42	53	57	60	68	70	70	70	70
Mch3 ->Ro	60	61	72	76	79	87	89	89	89	89
Dresden ->Krakau	40	40	40	40	40	40	40	40	44	54
Rost ->Danzig	50	60	71	73	77	90	100	100	100	100
Danzig ->Mo	50	60	71	73	77	90	100	100	100	100
Pa ->Athen	50	60	69	71	74	78	84	94	100	100
Mo ->Thes	40	40	40	40	40	40	40	40	40	40
Mo ->Athen	40	40	40	40	40	40	40	40	40	40
Athen ->Samos	50	60	69	71	74	78	84	94	100	100
Thes ->Athen	40	40	40	40	40	40	40	40	40	40
Thes ->Samos	40	40	40	40	40	40	40	40	40	40
Danzig ->Krakau	40	40	40	40	40	40	40	40	40	40
Danzig ->Krakau	40	40	40	40	40	40	40	40	40	40
Rost ->Len	40	40	40	40	40	40	40	40	40	46
Len ->Danzig	40	40	40	40	40	40	40	40	40	40
Len ->Mo	40	40	40	40	40	40	40	40	40	46
Mo ->Thes2	40	40	41	49	56	62	66	66	66	66
Thes2 ->Samos	40	40	41	49	56	62	66	66	66	66
F ->Lyon	40	40	40	40	40	40	40	40	40	40
F ->Lyon2	50	60	68	68	69	72	76	79	88	98
Lyon ->Avi	40	40	40	40	40	40	40	40	40	40
Lyon2 ->Avi	50	60	68	68	69	72	76	79	88	98
Avi ->Pisa	40	40	40	40	40	40	40	43	43	43
Pisa ->Roma	40	40	40	40	40	40	40	40	40	40
Pisa ->Ro	40	59	66	70	74	80	92	100	100	100

Tab. C.1 Auslastung der Kanten bei Einsatz von Fuzzy-Routing in Prozent

Kante/Anzahl Schaltungen	50	100	150	200	250	300	350	400	450	500
F ->Lyon3	40	40	42	52	61	68	74	78	79	79
Lyon3 ->Avi	40	40	42	52	61	68	74	78	79	79
DA ->H	40	40	40	40	40	40	40	40	47	57
F ->Ulm	69	79	89	99	100	100	100	100	100	100
HH ->Kiel2	40	40	49	59	65	70	75	80	80	80
Kiel2 ->Rost	40	40	49	59	65	70	75	80	80	80
HH ->Kiel	40	40	40	40	40	40	40	40	40	40
HH ->Kiel3	50	60	61	61	65	70	75	80	90	100
Kiel3 ->Rost	50	60	61	61	65	70	75	80	90	100
Rost ->Dresden	40	40	41	49	56	62	66	66	66	66
F ->La	41	42	53	57	69	87	99	100	100	100
F ->Pisa	40	59	66	70	74	80	92	97	97	97
F ->Pisa2	40	40	40	40	40	40	40	40	40	40
Pisa2 ->Roma	40	40	40	40	40	40	40	40	40	40
DA ->Nb	40	40	42	54	67	73	74	78	82	91
Dresden ->A	40	40	40	40	40	40	40	40	40	40
Ulm ->Mch3	59	59	59	59	59	59	59	59	59	59
Krakau ->Pilion	40	40	40	40	40	40	40	40	44	54
Pilion ->Thes	40	40	40	40	40	40	40	40	40	40
Pilion ->Samos	40	40	40	40	40	40	40	40	44	54

Tab. C.1 Auslastung der Kanten bei Einsatz von Fuzzy-Routing in Prozent

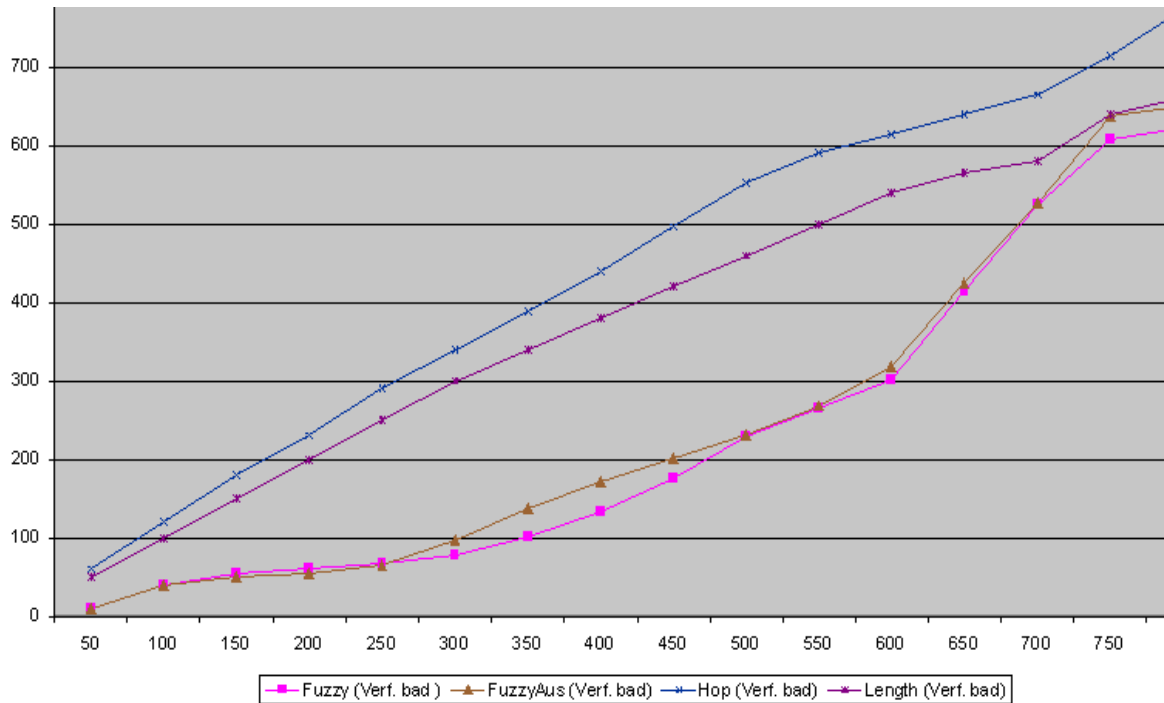


Abb. C.17 Anzahl der verwendeten Kanten, deren Verfügbarkeitswerte niedriger sind als die Mindestverfügbarkeit

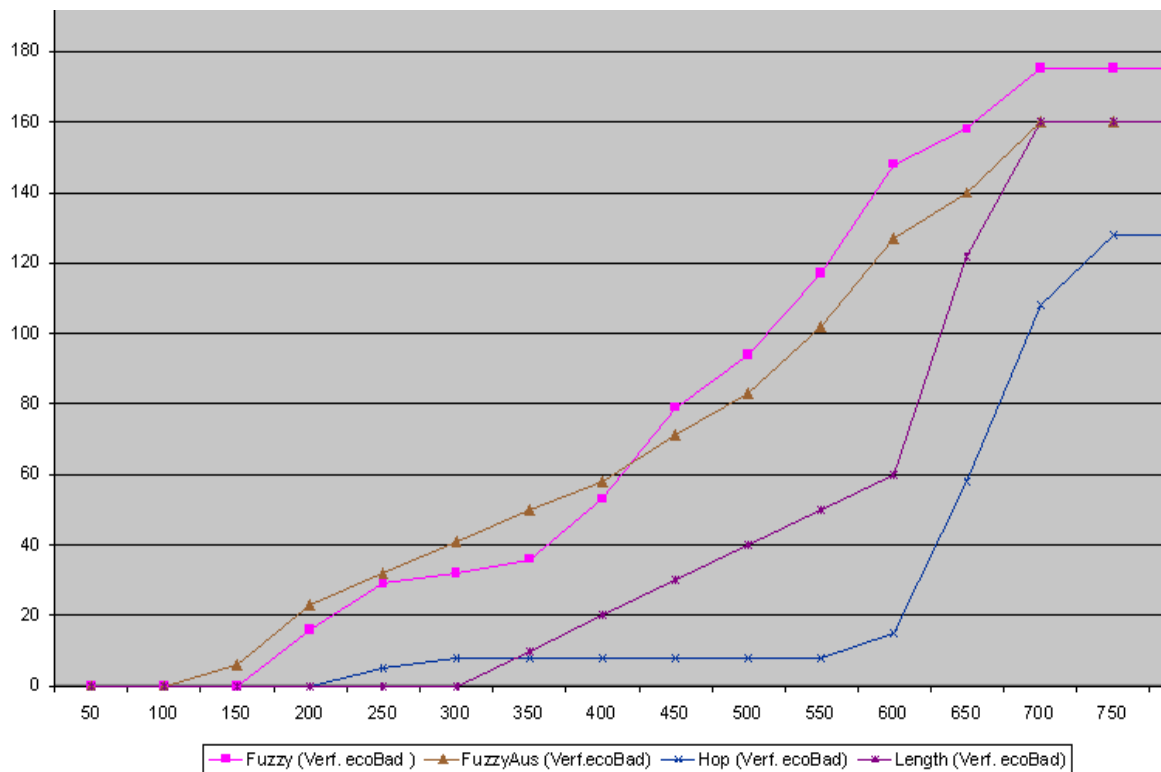


Abb. C.18 Anzahl der Kanten, deren Verfügbarkeitswerte im wirtschaftlich schlechten Bereich liegen

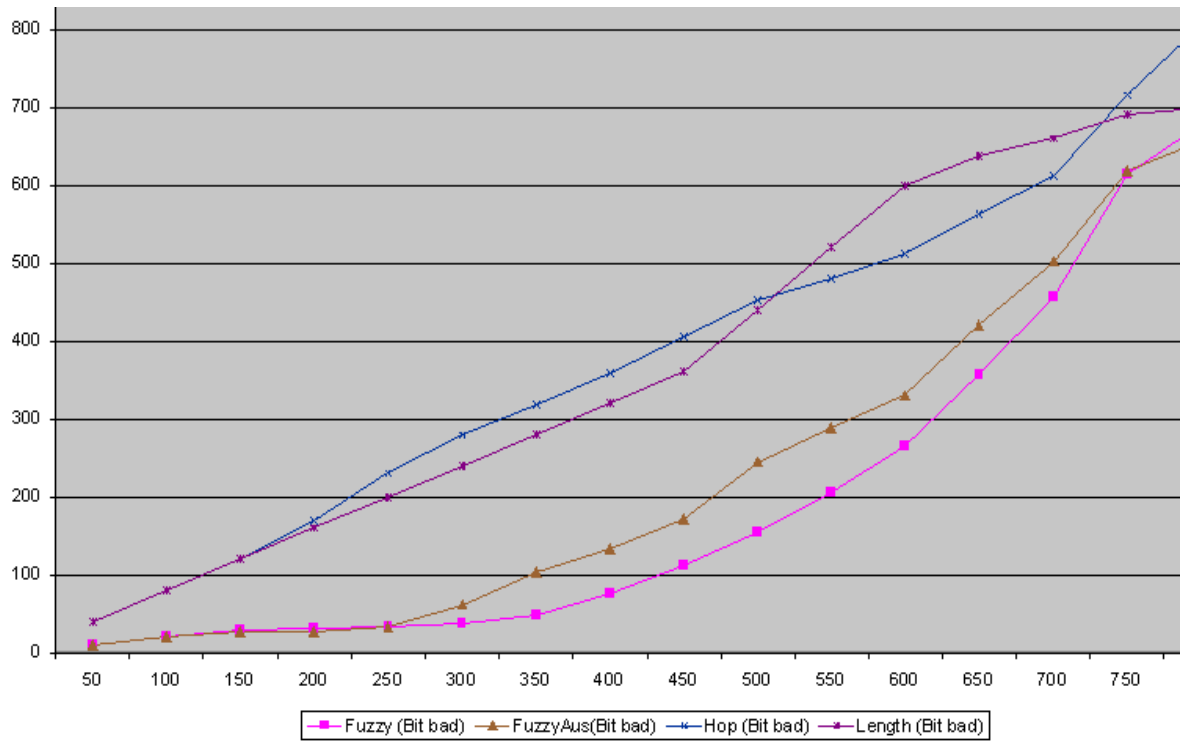


Abb. C.19 Anzahl der verwendeten Kanten, deren Werte für die Bitfehlerhäufigkeit die maximal erlaubte überschreiten

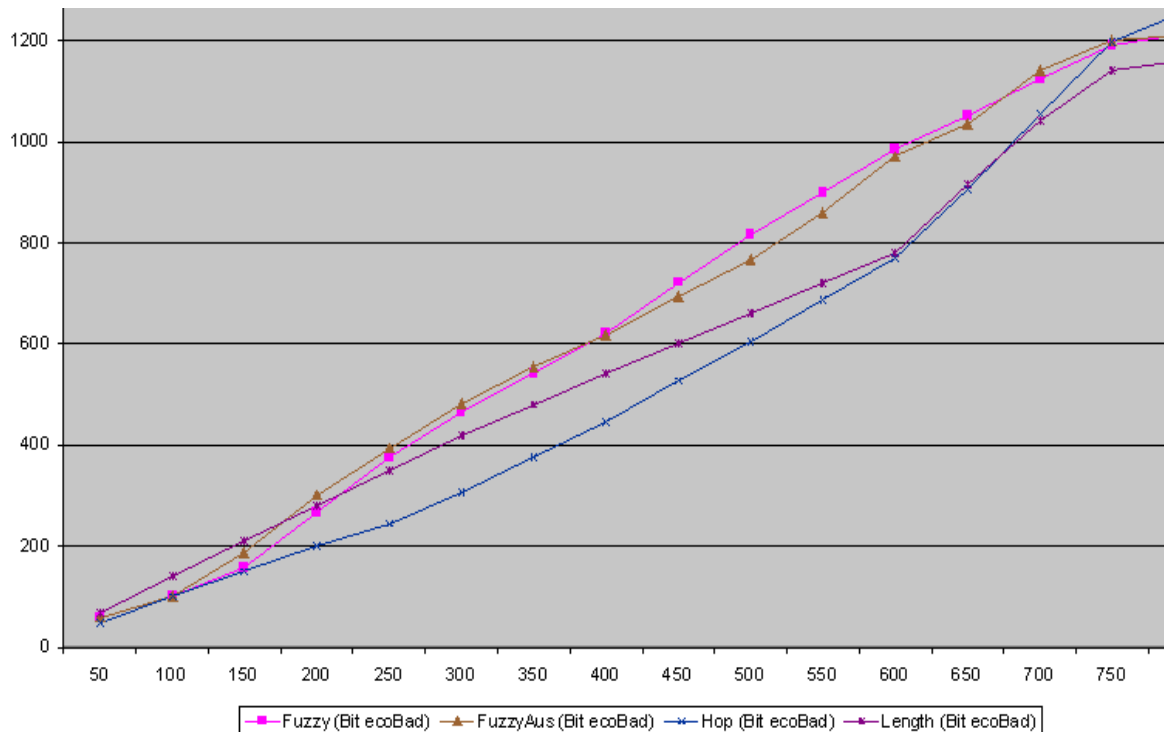


Abb. C.20 Anzahl der verwendeten Kanten, deren Werte für die Bitfehlerhäufigkeit im wirtschaftlich schlechten Bereich liegen

Der nachfolgende Text enthält ergänzende Ergebnisse zu 6.3.1:

3. Bitfehlerhäufigkeit und Verfügbarkeit verwendeter Kanten

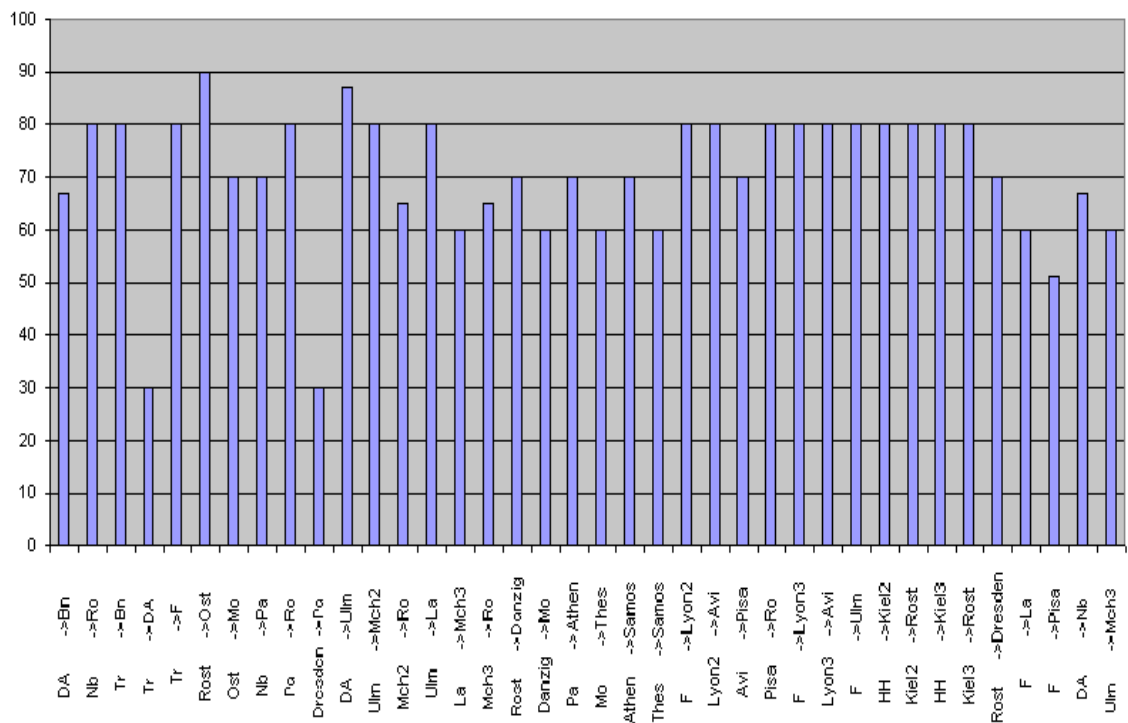


Abb. D.1 Verfügbarkeit verwendeter Kanten mit FuzzyPlus-Routing

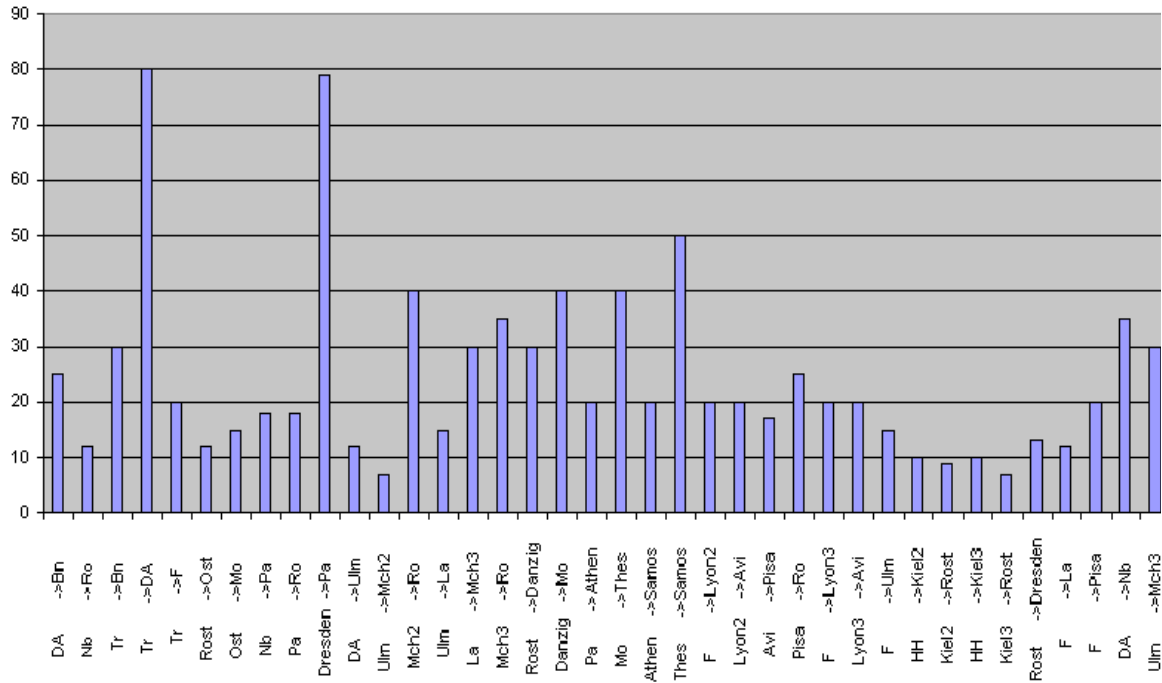


Abb. D.2 Bitfehlerhäufigkeit verwendeter Kanten mit FuzzyPlus-Routing

4. Verteilung der Bitfehlerhäufigkeit und Verfügbarkeit

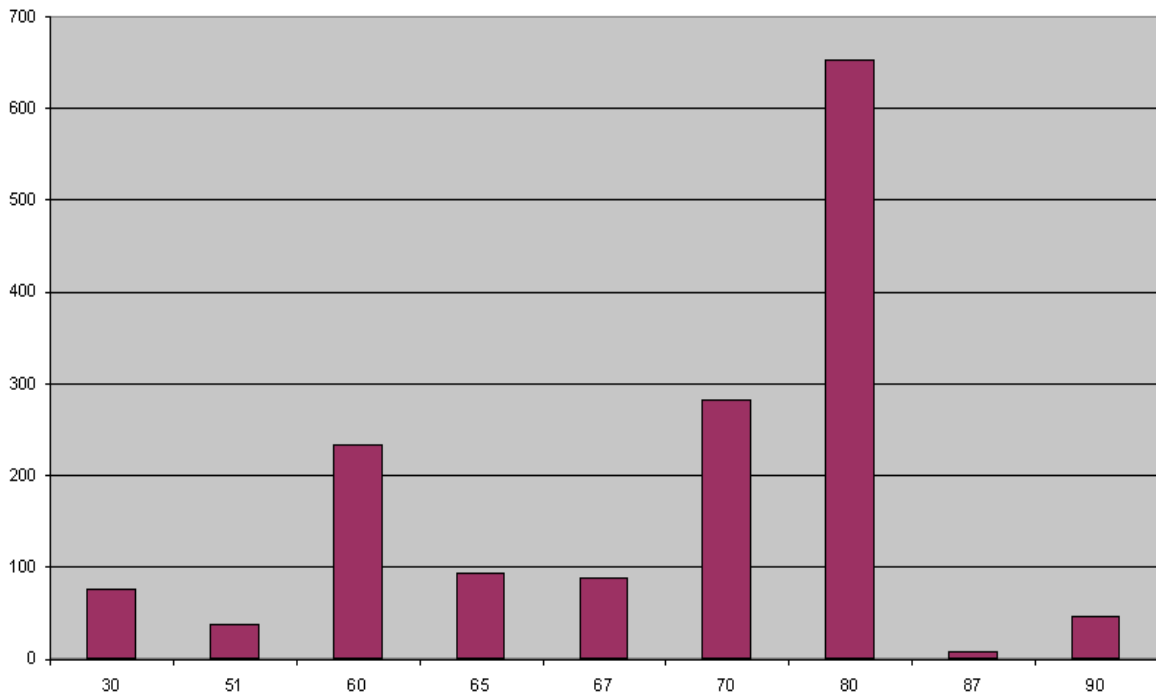


Abb. D.3 Verteilung der Verfügbarkeit mit FuzzyPlus-Routing

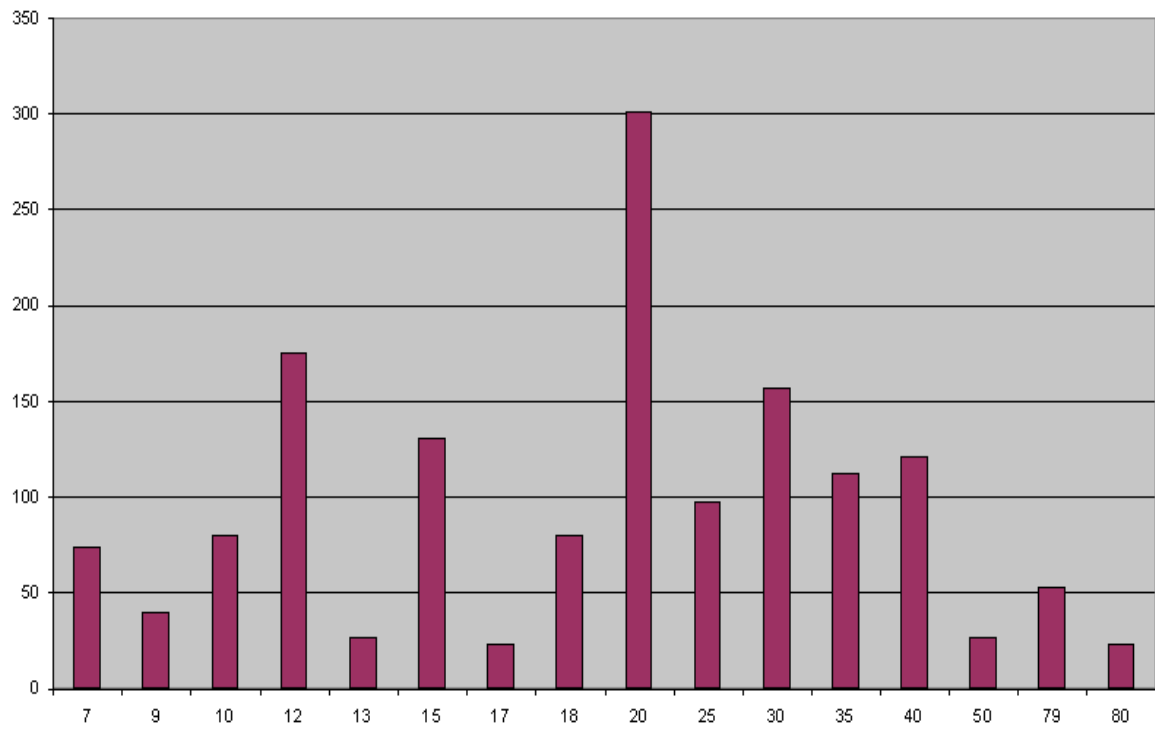


Abb. D.4 Verteilung der Bitfehlerhäufigkeit mit FuzzyPlus-Routing

ANHANG E*Ergebnisse Fuzzy-Neuro-Verfahren*

Die beiden folgenden Tabellen enthalten die Daten der Kanten und Knoten des in 6.4.1 beschriebenen Graphen.

Daten der Kanten

Kantenname	strat-Mean	length	geGr	avai	deltaAvail	bitEr	deltaBitErr
Bn - K	0	58	40	70	10.00	30	10.00
Bn - M	0	140	40	70	10.00	30	10.00
K - Si	0	57	40	60	0.00	40	0.00
M - Si	0	169	40	60	0.00	40	0.00
HH - HB	0	69	40	70	10.00	40	0.00
HH - OI	0	152	40	70	10.00	70	-30.00
HB - Ki	0	75	40	60	0.00	40	0.00
OI - Ki	0	142	40	30	-30.00	50	-10.00
Ulm - Mch	0	75	40	70	10.00	30	10.00
Mch - Ro	0	87	40	60	0.00	40	0.00
Ulm - Pa	0	159	40	40	-20.00	70	-30.00
Pa - Ro	0	154	40	50	-10.00	50	-10.00
Da - Rost	0	67	40	60	0.00	40	0.00
Da - Kra	0	376	40	60	0.00	40	0.00
Kra - So	0	362	40	40	-20.00	60	-20.00
Rost - So	0	62	40	40	-20.00	70	-30.00

Tab. E.1 Daten der Kanten des Trainingsgraphen

Kantename	strat-Mean	length	geGr	avai	deltaAvail	bitEr	deltaBitErr
DA - Ulm	0	134	40	70	10.00	40	0.00
DA - Bn	0	274	40	70	10.00	30	10.00
OI - F1	0	126	40	40	-20.00	40	0.00
F1 - Mch	0	149	40	40	-20.00	40	0.00
OI - F2	0	127	40	60	0.00	40	0.00
F2 - Mch	0	135	40	60	0.00	40	0.00
OI - F3	0	156	30	60	0.00	60	-20.00
F3 - Mch	0	147	30	60	0.00	40	0.00
Ro - Re1	0	118	40	40	-20.00	40	0.00
Ro - Re2	0	120	4	60	0.00	40	0.00
Ro - Re3	0	128	40	60	0.00	50	-10.00
Kra - Re1	0	123	40	40	-20.00	40	0.00
Kra - Re2	0	106	4	60	0.00	40	0.00
Kra - Re3	0	121	40	50	-10.00	40	0.00
DA - Ly1	0	190	40	40	-20.00	40	0.00
Ly2 - DA	0	150	40	60	0.00	50	-10.00
DA - Ly3	0	147	4	60	0.00	50	-10.00
Ly1 - Mar	0	213	40	40	-20.00	40	0.00
Ly2 - Mar	0	190	40	50	-10.00	40	0.00
Mar - Ly3	0	185	4	70	10.00	40	0.00
Si - Bit1	0	215	40	60	0.00	40	0.00
Bit1 - HH	0	211	40	60	0.00	40	0.00
Si - Bit2	0	168	40	60	0.00	60	-20.00
Bit2 - HH	0	152	40	40	-20.00	60	-20.00
Ki - O1	0	94	40	60	0.00	40	0.00
O1 - Da	0	97	40	60	0.00	40	0.00
Ki - O2	0	96	40	40	-20.00	60	-20.00
O2 - Da	0	84	40	60	0.00	80	-40.00
Mar - Ax1	0	183	40	60	0.00	40	0.00

Tab. E.1 Daten der Kanten des Trainingsgraphen

Kantename	strat-Mean	length	geGr	avai	deltaAvail	bitEr	deltaBitErr
Ax1 - Pisa	0	144	40	60	0.00	40	0.00
Mar - Ax2	0	118	4	60	0.00	40	0.00
Ax2 - Pisa	0	118	4	60	0.00	40	0.00
Pisa - Ve1	0	321	40	60	0.00	40	0.00
Kra - Ve1	0	408	40	60	0.00	40	0.00
Mar - Ax3	0	119	40	80	20.00	60	-20.00
Ax3 - Pisa	0	114	40	60	0.00	60	-20.00
Si - Bit3	0	173	30	60	0.00	40	0.00
HH - Bit3	0	157	30	60	0.00	40	0.00
HH - HB2	0	70	30	60	0.00	40	0.00
HB2 - Ki	0	71	30	60	0.00	40	0.00
Rost - So2	0	46	30	60	0.00	40	0.00
So2 - Kra	0	354	30	60	0.00	40	0.00

Tab. E.1 Daten der Kanten des Trainingsgraphen

Daten der Knoten

Knotenname	ndCo	ndCl	ndLs
HH	20	1	36,66
DA	20	1	32,8
Bn	20	1	40
M	20	1	40
K	20	1	40
Si	20	1	38
OI	20	0	38
Ki	20	1	38
HB	20	1	40
Rost	20	1	36,66
So	20	1	40
Da	20	1	40

Tab. E.2 Daten der Knoten des Trainingsgraphen

Knotenname	ndCo	ndCl	ndLs
Kra	50	0	33,42
Ro	20	1	32,8
Mch	20	1	38
Ulm	20	1	40
Pa	20	0	40
F1	20	0	40
F2	20	1	40
F3	20	1	30
Re1	20	0	40
Re2	20	1	4
Re3	20	1	40
Mar	20	1	28
Ly1	20	0	40
Ly2	20	1	40
Ly3	20	1	4
Bit2	0	1	40
Bit1	0	1	40
O2	0	1	40
O1	0	1	40
Pisa	0	1	31
Ax2	0	1	4
Ax1	0	0	40
Ve1	0	1	40
Ax3	0	1	40
Bit3	0	1	30
HB2	0	1	30
So2	0	1	30

Tab. E.2 Daten der Knoten des Trainingsgraphen

Kantenname	Auslastung vor dem Einsatz	Auslastung nach Einsatz des trainierten System	Auslastung nach Einsatz des untrainierten Systems
Bn ->K	40	40	40
Bn ->M	40	40	40
K ->Si	40	40	40
M ->Si	40	40	40
HH ->HB	40	72	100
HH ->Ol	40	40	40
HB ->Ki	40	72	100
Ol ->Ki	40	40	40
Ulm ->Mch	40	40	40
Mch ->Ro	40	40	40
Ulm ->Pa	40	40	40
Pa ->Ro	40	40	40
Da ->Rost	40	68	40
Da ->Kra	40	68	40
Kra ->So	40	40	40
Rost ->So	40	40	40
DA ->Ulm	40	40	40
DA ->Bn	40	40	40
Ol ->F1	40	40	40
F1 ->Mch	40	40	40
Ol ->F2	40	100	100
F2 ->Mch	40	100	100
Ol ->F3	30	30	30
F3 ->Mch	30	30	30
Ro ->Re1	40	40	40
Ro ->Re2	4	4	4
Ro ->Re3	40	40	40
Kra ->Re1	40	40	40
Kra ->Re2	4	4	4

Tab. E.3 Auslastungswerte der Kanten des Trainingsgraphen

Kantenname	Auslastung vor dem Einsatz	Auslastung nach Einsatz des trainierten System	Auslastung nach Einsatz des untrainierten Systems
Kra ->Re3	40	40	40
DA ->Ly1	40	40	40
Ly2 ->DA	40	40	40
DA ->Ly3	4	4	4
Ly1 ->Mar	40	40	40
Ly2 ->Mar	40	40	40
Mar ->Ly3	4	4	4
Si ->Bit1	40	70	68
Bit1 ->HH	40	70	68
Si ->Bit2	40	40	40
Bit2 ->HH	40	40	40
Ki ->O1	40	40	40
O1 ->Da	40	40	40
Ki ->O2	40	40	40
O2 ->Da	40	40	40
Mar ->Ax1	40	40	40
Ax1 ->Pisa	40	40	40
Mar ->Ax2	4	4	4
Ax2 ->Pisa	4	4	4
Pisa ->Ve1	40	40	40
Kra ->Ve1	40	40	40
Mar ->Ax3	40	40	40
Ax3 ->Pisa	40	40	40
Si ->Bit3	30	60	62
HH ->Bit3	30	60	62
HH ->HB2	30	58	30
HB2 ->Ki	30	58	30
Rost ->So2	30	62	90
So2 ->Kra	30	62	90

Tab. E.3 Auslastungswerte der Kanten des Trainingsgraphen

Literaturverzeichnis

-
- [1] Aho, Alfred V., Hopcroft, John E. , Ullman, Jeffrey D.
The Design and Analysis of Computer Algorithms
Addison-Wesley Publishing Company 1974
 - [2] Autorenkollektiv des ITD-Fachausschusses 1.7
Anforderungen an Netzmanagementsysteme bei öffentlichen Netzen
ntz, Heft 6, 1995
 - [3] Bistarelli, Stefano
Soft Constraint Solving and Programming: a general framework
Ph.D. Thesis, Università di Pisa, Dipartimento di Informatica, 2001
 - [4] Bücherl, Angela
Reversible Übertragung von regelbasierten Systemen in adaptiver Form durch
Implementierung als neuronale Netze
Diplomarbeit 1994, Technische Universität München, Prof.Dr.W.Brauer
 - [5] Bothe, Hans-Heinrich
Fuzzy Logic
Springer-Verlag, 1993
 - [6] Cavalieri, S., Di Stefano, A., Mirabella, O.
Neural Strategies to Handle Routing in Computer Networks
International Journal of Neural Systems, Vol. 4, No.3 269-289
 - [7] CCITT, Recommendation M3010, Genf
Principles for a Telecommunications Management Network
October 1992
 - [8] Collett, M., Pedrycz, W.
Application of Neural Networks For Routing in Telecommunications Networks
IEEE Global Telecommunications Conf. , Tech. Program, Conf. Record, Vol.2
Houston, USA, Nov 29 -Dec 2, IEEE 1993

- [9] Digitale Nachrichtenübertragung
Teil 2. Crossconnect- und Multiplextechnik
Siemens-Aktienges., [Abt. Verl.] Berlin, München 1990
- [10] Dijkstra, E.W.
A Note on Two Problems in Connexion with Graphs
Numerische Mathematik 1, 269-271 (1959)
- [11] Driankov, D., Hellendoorn, H., Reinfrank, M.
An Introduction to Fuzzy Control
Springer-Verlag, 1993/1996
- [12] Fullér, Robert
Introduction to Neuro-Fuzzy Systems
Physica-Verlag, 2000
- [13] Frings, Jochen
Eine kurze Einführung in die Netzplanung
Lehrstuhl für Kommunikationsnetze Technische Universität München, Januar
1998
- [14] Gilsdorf, I., Brauer W.
Optimizing Routing Algorithms in Telecommunication Networks with Neural
Networks and Fuzzy Logic
Bernd Reusch (Ed.), Computational Intelligence, Theory and Applications,
International Conference, 6th Fuzzy Days, Dortmund, Germany, May 1999,
Proceedings
Springer, 1999
- [15] Güting, Ralf Hartmut
Datenstrukturen und Algorithmen
B.G.Teuber Stuttgart 1992
- [16] Hellendoorn, Hans, Thomas, Christoph
Defuzzification in Fuzzy Controllers
Journal of Intelligent and Fuzzy Systems, Vol. 1, 109-123 (1993)
1993 John Wiley & Sons, Inc.
- [17] Hellendoorn, H.
Fuzzy Control in Telecommunication
1996 Biennial Conference of the North American Fuzzy Information Society-
NAFIPS
Berkeley, CA, 1996, pp. 444-448
- [18] Hiramatsu, A.
Training Techniques for Neural Network Applications in ATM
IEEE Communications Magazine, October 1995

- [19] Hopfield, J.J.
Neural networks and physical systems with emergent collective computational abilities
Proc.Natl.Acad.Sci.USA
Vol. 79, pp.2554-2558, April 1982
- [20] ITU-Recommendation M.60
Maintenance: Introductuion and General Principles of Maintenance and Maintenance Organisation
Maintenance Terminology and Definitions
March, 1993
- [21] ITU-T Recommendation G.709
General Aspects of Digital Transmission Systems
Synchronous Multiplexing Structure
March, 1993
- [22] ITU-T Recommendation G.803
Architecture of Transport Networks based on the Synchronous Digital Hierarchy (SDH)
March, 1993
- [23] ITU-T Recommendation G.805
Generic functional Architecture of Transport Networks
November, 1995
- [24] Jung, V., Warnecke, H.-J.(Hrsg.)
Handbuch für die Telekommunikation
Springer-Verlag Berlin Heidelberg 1998
- [25] Klir, George J., Yuan, Bo
Fuzzy Sets and Fuzzy Logic
Theory and Applications
Prentice Hall P T R 1995
- [26] Kamoun, Faouzi , Ali, M.K.Mehmet
A Neural Network Shortest Path Algorithm for Optimum routing in packet-switched communications Networks
IEEE Global Telecommunications Conference, GLOBECOM 91, p. 120-4, vol.1 of 3 vol.
- [27] Karsten, M., Schmitt, J., Wolf, L., Steinmetz, R.
Cost and Price Calculation for Internet Integrated Services
Kommunikation in Verteilten Systemen (KiVS), 11. ITG/GI-Fachtagung, Darmstadt, März 1999, Springer

- [28] Lee C.Y.
An Algorithm for Path Connections and Its Applications
IRE Transactions on Electronic Computers, EC-10:346-365
September 1961
- [29] Lee, Byeong Gi, Kang, Minho , Lee, Jonghee
Broadband Telecommunications Technology, 2nd ed.
Artech House, 1996
- [30] McCulloch, W., Pitts, W.
A logical calculus of the ideas immanent in nervous activity
Bulletin of Mathematical Biophysics 5, pp. 115-133, 1943
- [31] Moore E.F.
The Shortest Path Through a Maze
Proceedings of an International Symposium on the Theory of Switching, Part II, pp 285-292
Cambridge, 1959. Harvard University Press
- [32] Nauck, D., Klawonn, F. , Kruse, R.
Neuronale Netze und Fuzzy Systeme
Vieweg, Braunschweig, 1996
- [33] Neuneier, R. , Tresp, V.
Radiale Basisfunktionen, Dichteschätzungen und Neuro-Fuzzy
aus H. Rehkugler und H.G.Zimmerman (Herausgeber)
Neuronale Netze in der Ökonomie
Vahlen, München 1994
- [34] Palm, G.
Informationsverarbeitung in neuronalen Netzen
Informatik-Fachberichte Bd.203, 1989. S.299-302
- [35] Preuß, H.-P. und Tresp, V.
Neuro-Fuzzy
atp - Automatisierungstechnik Praxis 36 (1994) 5
R.Oldenburg Verlag
- [36] Ritter, H., Martinetz, Th., Schulten, K.
Neuronale Netze
Addison-Wesley, Bonn 1994
- [37] Rojas, R.
Theorie der neuronalen Netze
Springer Verlag, Berlin 1993

- [38] Rumelhart, David E. , McClelland, James L. and the PDP Research Group
Parallel Distributed Processing: Explorations in the Microstructure of Cognition
Volume 1: Foundations
MIT Press 1986
- [39] Shier, D.R.
On algorithms for finding the k shortest paths in a network
NETWORKS 9 (1979)
- [40] Shier, D.R.
Iterative Methods for Determining the k Shortest Paths in a Network
Networks, 6: (1976)
- [41] Shier, D.R.
A New Algorithm for Performance analysis of Communication Systems
J. of ACM, Vol 19, July 1972, pp. 423-424
- [42] Siekmann, S.
Neuro-Fuzzy Ansätze in der Finanzanalyse mit Experimenten zur DAX-Prognose
Diplomarbeit, Technische Universität Braunschweig, Prof. Dr. R. Kruse, 1996
- [43] Siemens Business Services
Netzmanager IRONMAN, Netzdaten und Transportnetze bedarfsgerecht anpassen und flexibel managen
Flyer, www.sbs.de/telekommunikation
- [44] Siemens Aktiengesellschaft
The Solution Provider Lexikon
w3.siemens.de/solutionprovider/_online_lexikon
- [45] Sugeno, M., Takagi, T.
Multi-Dimensional Fuzzy Reasoning
Fuzzy Sets and Systems, 9(3): 313-326, 1983
- [46] Sugeno, M.
Introductory Survey of Fuzzy Control
Information Sciences, 36(1985) 59-83
- [47] Takagi, T., Sugeno, M.
Derivation of Fuzzy Control Rules from Human Operators Control Actions
Proc. IFAC Symp. On Fuzzy Information, Knowledge Representation and Decision Analysis. Marseille, July 1983
- [48] Takagi, T. and Sugeno, M.
Fuzzy Identification of Systems and its Applications to Modelling and Control
IEEE Trans. On Systems, Man and Cybernetics, 15(1) (1985), 116-132

-
- [49] Tanenbaum, Andrew S.
Computer-Netzwerke
Wolfram's Fachverlag, 1992, Attenkirchen
- [50] TeleManagement Forum
SLA Management Handbook, GB 917
Public Evaluation/Version 1.5, June 2001
- [51] Widrow, G., Hoff, M.E
Adaptive switching circuits
Institut of Radio Engineers
Western Electronic Show and Convention, Convention Record Part 4
BD: Jahr 1960 Heft: S.96-104
- [52] Wizgall, M.
Planung und Optimieren von Telekommunikationsnetzen
Strategisches Instrument im Wettbewerb
NET, Heft 11,1996
- [53] Wöhlbier, G.(Hrsg.)
Technik der Telekommunikation, Planung von Telekommunikationsnetzen
R.v.Decker's Verlag, G.Schenk, Heidelberg
- [54] Ziegelmann, M.
Constrained Shortest Paths and Related Problems
Dissertation, Universität des Saarlandes, Naturwissenschaftliche-Technischen
Fakultät I, Prof. Dr. K. Mehlhorn, 2001

Index

A

atomare Fuzzy Aussagen 48
Auslastung der Kante 24
Auslastung des Netzknoten 24
Axon 57

B

Bitfehlerhäufigkeit 16
Bitfehlerhäufigkeit der Kante 24

C

CCITT 7
Connections 13

D

Demultiplexen 6
Dendriten 57

E

einfachen RBF-Netzen 60
Erfüllungsgrad einer Fuzzy-Regel 49

F

Fuzzy Logik 43
Fuzzy Menge 44
Fuzzy Regeln 48
Fuzzy-Implikation 46
Fuzzy-Neuro-Verfahren 40
FuzzyPlus-Routing 39
Fuzzy-Routing 38

H

Hochprioritäre Übertragungswege 32
Hop-Routing 83

I

ISDN (Integrated Service Digital Network) 6

K

Kanäle 14

Kante 24

Kanten 14

Knoten 14

Knotenart 24

künstliche neuronale Netze 57

L

LAN 6

Layer-Netzwerk (layer network) 12

Length-Routing 83

linguistische Variable 47

Link Connections 13

Links 12

M

Multiplexen 6

N

Network Connections 13

Netzarchitektur 58

Netzelement 6

Netzknoten 14

NEUROTool 80

P

Parameter der Kante 24

Parameter des Netzknotens 24

Pareto-optimale Kante 29

Pareto-optimale Route 29

Partitionierung (partitioning) 12

plesiochrone digitale Hierarchie (PDH) 7

Pulse Code Modulation Methode (PCM) 7

R

RBF-Netz 59

RBF-Netze 58

Route 14, 24

S

Schalten 14

Schichtung (layering) 12

SCPL, semiringbased constraint logic programing 28

SRCM-Netze 63

SRCM-Neuron 63

strategische Bedeutung 87

strategischen Gesichtspunkte 87

Subnetwork Connections 13

Subnetz (subnetwork) 12

Synchrone Digitale Hierarchie (SDH) 8

T

Telecommunications Management Network (TMN) 10
Telekommunikationsnetz 5
Trail 13
Training des Systems 72
Trainingsschritt 117
TransMerkur 80
Transport Entities 13
Transportnetz 11

U

Übertragungswegen 14

V

Verfügbarkeit 16
Verfügbarkeit der Kante 24

W

WAN 6

Z

Zugehörigkeitsfunktion 44
Zusätzliche linguistische Variablen 87

