# Über den Einsatz von CI-Methoden bei der Ressourcen-Verwaltung in Hochgeschwindigkeitsnetzen

Von der Fakultät für Elektrotechnik der Universität Dortmund
zur Erlangung des akademischen Grades eines

**DOKTOR-INGENIEURS**

genehmigte

**DISSERTATION**

vorgelegt von



## Dipl. - Ing. Detlef Jensen



9. März 2001

| | |
|---|---|
| Referent: | Prof. Dr.-Ing. R.-G. Schehrer |
| Koreferent: | Prof. Dr.-Ing. K. Goser |
| Tag der mündlichen Prüfung: | 9. März 2001 |

## Kurzfassung

In der letzten Zeit gewannen die Hochgeschwindigkeitsnetze in allen Bereichen große Aufmerksamkeit. Nicht zuletzt wurde dieses verstärkte Interesse durch die Entwicklung von leistungsstarken Computer-Plattformen in Verbindung mit verteilten Anwendungen und multimedialen Applikationen - gepaart mit einem steigenden Kommunikationsverhalten - forciert. Diese neuen Dienste, die sich in ihren Anforderungen stark voneinander unterscheiden können, sollen in einem paketorientierten Netz, zusammen mit den bekannten Telefondiensten, integriert werden.
Da die neuen Technologien im Bereich der Vermittlungstechnik enorm leistungsfähig sind und sehr hohe Bandbreiten bereitstellen, bieten sie eine ideale Netzwerklösung für die vielen heterogenen Anwendungen. Die Herausforderung besteht darin, die verfügbaren Ressourcen so zu verwalten, daß keine Überlastsituationen auftreten und jederzeit die Dienstgüte der einzelnen Verbindungen, trotz ihrer unterschiedlichen, teilweise ambivalenten Anforderungen, gewährleistet werden kann. Herkömmliche Steuerungsalgorithmen haben zu wenig Freiheitsgrade, um hier eine effiziente Ausnutzung aller Möglichkeiten garantieren zu können. Methoden aus dem Bereich der Computational Intelligence stellen einen Ansatz dar, um dieses vielschichtige Optimierungsproblem zu lösen.

## Abstract

Highspeed Networking has received widespread attention. Recently this attention has begun to focus because of the deployment of powerful computer platforms in combination with distributed and multimedia applications and an increasing demand for communication. These new network traffic streams with their widely varying characteristics should be integrated into one packetoriented network in combination with the well known telephony services.
Since the emerging technologies provide efficent multiplexing capacity and high bandwidth communications they represent an ideal network solution for heterogeneous computing as well as multimedia applications. Managing the available resources to avoid congestion and grant the guaranteed Quality of Service for connections with dramatic differences in their statistical behavior and partly even contrary requirements pose new challenges very different from the conventional networks. Because a lot of new and multivarious time dependent requirements have to be taken into consideration so that conventional control algorithms cannot guarantee an optimal solution to satisfy the interests of all participating parties. The basic intention of this thesis is to represent methods of the Computational Intelligence to solve this multi-layered optimization problem.

*Ich wollte den Dingen*
*auf den Grund gehen.*
*Seitdem bin ich*
*unterwegs.*

unbekannt


## Danksagung

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Elektronische Systeme und Vermittlungstechnik an der Universität Dortmund.

An dieser Stelle möchte ich mich bei allen bedanken, die mich bei der Erstellung dieser Arbeit begleitet haben. Mein besonderer Dank gilt meiner Familie (Karin, Inga und Isabelle) für die Unterstützung und Rücksichtnahme während der letzten Monate vor der Abgabe der Arbeit.

Herrn Professor Dr. - Ing. Schehrer danke ich recht herzlich für die Betreuung und stete Förderung dieser Arbeit.

Den Kolleginnen und Kollegen des Lehrstuhls danke ich für das gute Arbeitsklima und die gemeinsam verbrachte Zeit.

# Inhaltsverzeichnis


**1 Einleitung** **1**

1.1 Zielsetzung und Aufbau der Arbeit . . . . . . . . . . 3

**2 Kommunikationsnetze**
**Verfahren und Dienste** **4**

2.1 Überblick . . . . . . . . . . 4

2.2 Verkehrsmanagement in Kommunikationsnetzen . . . . . . . . . . 10

2.3 Ziele der dynamisch adaptiven Zugangskontrolle . . . . . . . . . . 21

2.4 Quellenmodelle . . . . . . . . . . 25

**3 Kommunikationsknoten** **36**

3.1 Admission Controller . . . . . . . . . . 37

3.2 Link Control Unit . . . . . . . . . . 39

3.3 Allocation Controller . . . . . . . . . . 41

**4 Konventionelle Kontroll - Verfahren** **42**

4.1 Definition der Eingangslast . . . . . . . . . . 42

4.2 Verfahren zur Bewertung der Simulationsergebnisse . . . . . . . . . . 45

4.3 Policing Controller . . . . . . . . . . 46

4.4 Der Call Admission Controller . . . . . . . . . . 52

**5 Der Fuzzy Controller** **57**

5.1 Grundlagen . . . . . . . . . . 57

5.2 Entwurf des Fuzzy Controllers . . . . . . . . . . 64

5.3 Der Fuzzy Logic basierte Policing Controller . . . . . . . . . . 67

5.4 Simulation des hierarchischen Fuzzy Controllers . . . . . . . . . . 78

**6 Genetische Algorithmen** **89**

6.1 Grundlagen genetischer Algorithmen . . . . . . . . . . 89

6.2 Optimierung der Controller mit Hilfe genetischer Algorithmen . . . . . . . . . . 93

**7 Der Call Admission Controller** **121**

7.1 Die Regelstrategie . . . . . . . . . . 122

7.2 Der Aufbau des Fuzzy Logic basierten Admission Controllers . . . . . . . . . . 123

7.3 Simulation des unscharfen Admission Controllers . . . 126
7.4 Aufbau eines parallelen Admission Controllers . . . 139
7.5 Aufbau eines seriellen Admission Controllers . . . 142
7.6 Optimierung des Admission-Controllers . . . 151
7.7 Bewertung der Admission Controller . . . 152

**8 Neuronale Netze 155**
8.1 Grundlagen künstlicher Neuronaler Netze . . . 156
8.2 Lernverfahren . . . 159
8.3 Lernalgorithmen . . . 160
8.4 Neuronale Netze mit Reinforcement . . . 162
8.5 Simulation . . . 168

**9 Vergleich der Controller 170**
9.1 Policing Controller . . . 170
9.2 Der Call Admission Controller . . . 172

**10 Zusammenfassung und Ausblick 174**
10.1 Zusammenfassung . . . 174
10.2 Ausblick . . . 175

**A Grundlagen 177**
A.1 Auslastung der reservierten Bandbreite bei Anwendung der PR-Methode . 177
A.2 Auslastung der reservierten Bandbreite bei Anwendung der MR-Methode . 178

**B Simulation 180**
B.1 Der Ereigniskalender . . . 180

**C Genetischer Algorithmus 186**
C.1 Anwendungsbeispiel . . . 186
C.2 Beschreibung des GA zur Optimierung der Codierung der Regelbasen . . . 188

**D Policing Controller 190**
D.1 Fitness Funktion 5 . . . 190
D.2 Fitness Funktion 6 . . . 194
D.3 Fitness Funktion 7 . . . 197
D.4 Fitness Funktion 8 . . . 200
D.5 Fitness Funktion 9 . . . 202
D.6 Fitness Funktion 10 . . . 203
D.7 Unscharfe Fitness Funktion . . . 206

**E Call Admission Controller 213**
E.1 Traffic Qualifier . . . 213
E.2 State Qualifier . . . 215

**F Backpropagations-Algorithmus** **233**
F.1 Adaption der Ausgangsgewichtungen . . . . . . . . . . . . . . . . . . . . . . . 233
F.2 Adaption der Verbindungsgewichte der versteckten Layer . . . . . . . . . . . 234

# Kapitel 1

# Einleitung

Die Entwicklung sowohl der Kommunikations- als auch der Rechnertechnik war seit Mitte der 80er Jahre imposant. Damals prognostizierte man, daß die Digitalisierung der Vermittlungstechnik bis ungefähr 2020 flächendeckend in Deutschland durchgeführt wäre. ISDN, die Integration von Daten- und Kommunikationsdiensten in einem Netzwerk, war nur in den Forschungseinrichtungen oder Entwicklungsabteilungen der Telekommunikationsfirmen ein Begriff.
Intel hatte die Entwicklung des 80386 Prozessors abgeschlossen, der 80486 mit einer Taktrate von 33 MHz wurde an Alpha-Kunden ausgeliefert. PCs waren teuer, Applikationen nur spärlich vorhanden, der Datenaustausch zwischen Rechnern erfolgte typischerweise über Modem oder Datenträger.

Beide Bereiche mit ihren unterschiedlichen Ausprägungen entwickelten sich nahezu disjunkt. Es wurde zwischen den Kommunikationsnetzen, die speziell auf die Übertragung von Sprachdaten zugeschnitten waren und deren Anforderungen strikt befolgten, und Systemen zur Übertragung und Verarbeitung von Daten differenziert.
Sprachnetze sind i. a. dadurch gekennzeichnet, daß die Informationen so über Verbindungen übertragen werden, daß die Dienstqualität immer gewährleistet werden kann. Bei den leitungsvermittelten Systemen standen die übertragungstechnischen Einrichtungen (Wähler, Leitungen, ...) dann exklusiv dieser einen Verbindung zur Verfügung. Im Folgenden, beim Übergang zur PCM-Technologie, wurden die Ressourcen durch Anwendung von Zeitmultiplexverfahren mehrfach ausgenutzt. Bei dieser Technik werden den Verbindungen, wiederum unabhängig vom Bedarf, exklusiv Ressourcen in Form von Kanalkapazitäten zur Verfügung gestellt.

Rechnernetze entwickelten sich nahezu unabhängig von den Kommunikationsnetzen zur Übertragung von Sprache. Anfangs wurden sie oft benutzt, um teure Hardware-Komponenten wie z. B. Drucker zu vernetzen, um so eine Mehrfachausnutzung zu ermöglichen und eine größere Wertschöpfung zu erreichen. Im Laufe der Zeit wurden die Applikationen komplexer, so daß man von den Zentralrechnerarchitekturen zu leistungsfähigen dezentralen Lösungen überging. Mit dieser Verteilung war die Möglichkeit, Daten auszutauschen, verknüpft. Aus den sternförmigen Terminalnetzen entwickelten sich Netzwerke, an denen alle

Rechner gleichberechtigt teilnehmen. Diese LANs ermöglichen es, innerhalb eines begrenzten Bereichs eine beschränkte Anzahl von Systemen zu vernetzen. Das Übertragungsprinzip dieser Netze beruht oft auf der Paketvermittlung. Die Übertragung selbst erfolgt nach dem *Best-Effort-Prinzip* und kann in dieser Form keine Dienstqualität garantieren. Es können Verluste von Dateneinheiten oder Verzögerungen in Abhängigkeit von der temporären Lastsituation im Netz auftreten, so daß viele Applikationen nur beschränkt Anwendung finden. Die Leistungsfähigkeit dieser Systeme steigt in Bezug auf die Übertragungsbandbreite stetig an.

Mit dieser rasanten Entwicklung der Gebiete der Informationstechnik, Telekommunikation sowie der Medien ging eine Konvergenz dieser Branchen einher. Applikationen, wie z. B. die Telefonie, die bislang nur in der Kommunikationstechnik verfügbar waren, sind jetzt in der Form von IP-Telefonie im Bereich Rechnernetze verfügbar. Basierend auf dem email-Dienst fand nach dem Umweg über die Voice-Mail bzw. die Video-Mail, die Video-Telefonie und die Video-Konferenz Einzug in den Bereich der Rechnernetze.

Bei dem Wechsel der technologischen Plattform traten und treten immer wieder Probleme auf, die in den speziellen Charakteristika der Dienste und den Randbedingungen, die diese an die Übertragung stellen, begründet sind. Bei der Audioübertragung in Telefonqualität sind durchaus Bitfehlerraten in der Größenordnung von $10^{-3}$ tolerabel, während bei einem Filetransfer eine Fehlerrate von 0 angesetzt werden muß. Auf der anderen Seite ist die Telefonie sensitiv gegenüber Schwankungen in der Verzögerung. Bei der Datenübertragung dagegen spielt dieser Jitter eine untergeordnete Rolle.

Diese Beispiele zeigen, daß die inhärenten Anforderungen, die teilweise sogar konträr sein können, für die Abwicklung der Dienste zwingend erforderlich sind. Eine Steuerung der Datenströme und die Verwaltung der Ressourcen stellt eine schwer überschaubare Aufgabe dar, die noch von der Vielfalt der Dienste, den unterschiedlichen Kombinationsmöglichkeiten und der Anzahl von Nutzern abhängt. Für den Einsatz von deterministischen Verfahren ist eine genaue und vollständige Bestimmung aller Verkehrsparameter der einzelnen Datenströme notwendig. Auf Grund der statistischen Eigenschaften der Verkehrslast können gerade diese Zusammenhänge nicht exakt angegeben werden, so daß Steuerungverfahren zum Einsatz kommen, die präventiv eingreifen. Durch die installierten Mechanismen wird dann im Vorfeld, indem der *worst-case* angenommen wird, sichergestellt, daß die Dienstgüte der Verbindungen gewährleistet werden kann. Nachteilig ist, daß auf Grund dieser Voraussetzung die Ressourcen völlig ineffektiv ausgenutzt werden.

Parallel zu der Enticklung der Informationstechnik und Rechnertechnik sind auf dem Gebiet der Computational Intelligence (CI) grundlegende Verfahren hervorgebracht und beachtliche Fortschritte erzielt worden. Künstliche Neuronale Netze fanden nach einigen Anlaufschwierigkeiten Einzug in die Steuer- und Regelungstechnik und sind ein wesentlicher Bestandteil, wenn Verfahren automatisch, in Abhängigkeit von Erfahrungswerten, optimiert werden sollen. Hier wurden verschiedene Topologien und Lernverfahren in Abhängigkeit von dem Einsatzgebiet entwickelt. Aber nicht nur die Neuronalen Netze, sondern auch die Fuzzy Logic und Genetische Algorithmen sowie hybride Systeme, die eine beliebige Synthese aus den Verfahren darstellen, haben inzwischen theoretisch und auch praktisch bewiesen, daß sie robust bei der Suche in komplexen Optimierungsräumen sind und effektiv

und effizient arbeiten.
Der Einsatz dieser Algorithmen in einem so komplexen Umfeld, mit einem sich ständig ändernden Zustandsraum, der durch die Anforderungen und Randbedingungen der Dienste umrissen wird, wie der Steuerung von Kommunikationssystemen, ist naheliegend.
Daher wurden in dieser Arbeit zur Steuerung des Verbindungsaufbaus und zur Kontrolle der Datenströme in einem Knoten eines Hochgeschwindigkeitsnetzes Verfahren implementiert, die auf Fuzzy Logic und Neuronalen Netzen basieren.
Neben der Analyse von *neuen* Kontrollstrukturen auf der Basis der genannten intelligenten Verfahren, die die technischen Randbedingungen der Dienste und Übertragungssysteme mehr in den Vordergrund stellen , sollen weitere neue Ansätze vorgestellt werden, die die Gewinnoptimierung und die faire Behandlung der verschiedenen Verkehrsklassen untereinander, stärker berücksichtigen.

## 1.1 Zielsetzung und Aufbau der Arbeit

Um den Einsatz der genannten biologisch-nahen Optimierungsverfahren im Bereich der Kommunikationssysteme zu untersuchen, werden in Kapitel 2 zunächst einige strukturelle Merkmale von Hochgeschwindigkeitssystemen, aber auch die Anforderungen und Charakteristika der unterschiedlichen Dienste detaillierter beschrieben. Neben den Anforderungen, die die unterschiedichen Applikationen an das Übertragungssystem stellen, werden konventionelle Verkehrsmanagementverfahren, die in Kommunikationsnetzen eingesetzt werden, vorgestellt. Zur späteren Evaluierung der unterschiedlichen Methoden zur Verkehrssteuerung wird in Kapitel 2.2 das neue Modell eines Kommunikationsknotens präsentiert, mit dem die Einsetzbarkeit und Leistungsfähigkeit der konventionellen und der neuen auf Fuzzy Logic und neuronalen Netzen basierenden Methoden überprüft werden sollen. In den Absätzen 4 und 5 werden dann die konventionellen und die auf Fuzzy Logic basierten Policing Verfahren erläutert sowie deren Verhalten bei einer definierten Verkehrslast überprüft. Im Weiteren werden in Kapitel 6 dann mit Hilfe genetischer Algorithmen diese Controller optimiert. In Kapitel 7 erfolgt die Darstellung Fuzzy Logic basierter Methoden, mit deren Hilfe die Zugangskontrolle geregelt wird. Den Abschluß bildet in Kapitel 8 die vorläufige Untersuchung über den Einsatz von Neuronalen Netzen, die mit Hilfe von Reinforcement Methoden trainiert werden.

# Kapitel 2

# Kommunikationsnetze Verfahren und Dienste

## 2.1 Überblick

Mit der rasanten technologischen Weiterentwicklung der Netzwerktechnik im Allgemeinen und dem stetigen Wachstum des Internets im speziellen sowie der großen Anzahl neuer Applikationen ist eine Konvergenz der bestehenden Netze verbunden. Aus dieser Entwicklung

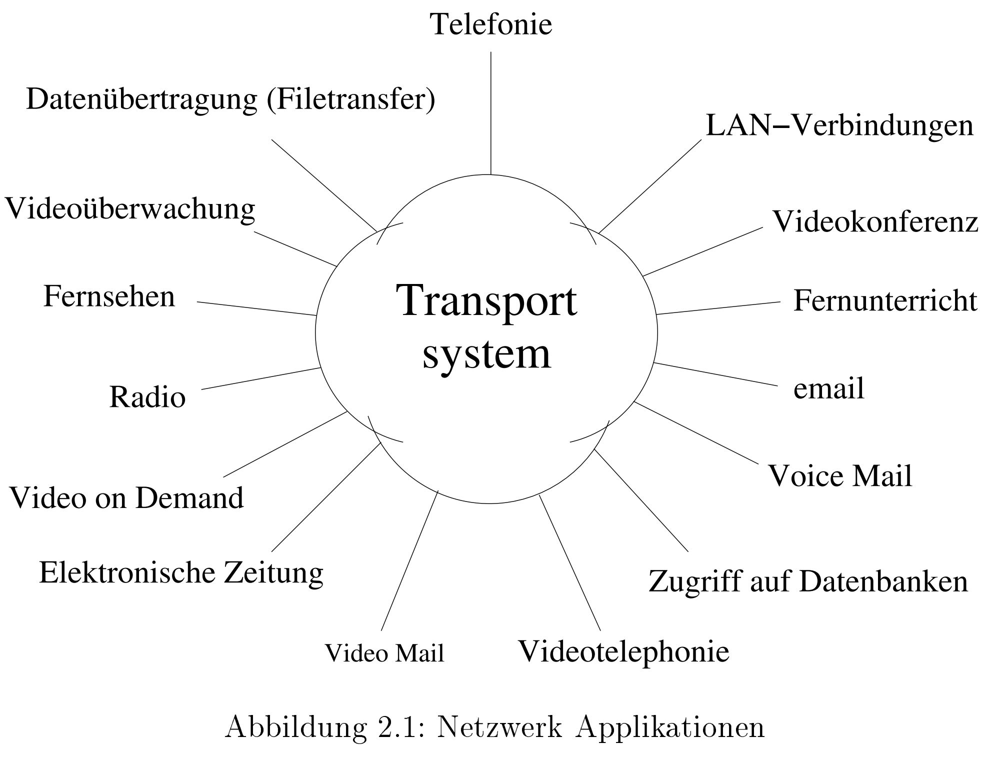



Abbildung 2.1: Netzwerk Applikationen

resultiert, daß Sprach-, Video-, Daten- und Multimediadienste in all ihren Ausprägungen

gemeinsam über *ein* Medium mit einer den Applikationen adäquaten Qualität abzuwickeln sind.
Das Spektrum dieser Dienste wird in Abbildung 2.1 gezeigt. Es reicht von der normalen Telefonie über Videotelefonie, Konferenzen, Datenaustausch bis hin zur Koppelung von LANs und der Übertragung von TV. Schon eine grobe Betrachtung der Dienste zeigt die vielfältigen Charakteristika und Übertragungsanforderungen. Tabelle 2.1 stellt einen kleinen Bereich möglicher Dienste dar und gibt einen Überblick über deren Dienstparameter. Die Anforderungen fallen in allen Bereichen unterschiedlich aus. Die Bandbreite der Applikationen variiert im Intervall von einigen kbits/s bis zu mehreren hundert Mbit/s. Verlustraten und Verzögerungen von Informationseinheiten sind tolerabel bis nicht tolerierbar. Einige Verkehrsarten, wie der interaktive Austausch von Daten oder die Übertragung von Videosequenzen, zeichnen sich als sehr burstbehaftet aus. Es wechseln sich Intervalle ab, in denen viele Informationen anfallen, die dann unter Berücksichtigung der QoS-Parameter übertragen werden müssen, mit Phasen, in denen weniger Daten zu transportieren sind. Andere hingegen benötigen während der gesamten Übertragung eine konstante Übertragungsrate oder dulden keine Schwankungen in der Verzögerung. Der Burstfaktor, der häufig auch in der deutschsprachigen Literatur als *Burstiness* bezeichnet wird, wird durch das Verhältnis von maximaler zu durchschnittlicher Bandbreite (Gl. 2.1) festgelegt und ist, wie in der Tabelle 2.1 dargestellt, für jeden Dienst eine charakteristische Kenngröße.

$$B = \frac{BW_{max}}{BW_{\oslash}} \tag{2.1}$$

Die Übertragung von Sprache stellt bis jetzt noch den größten Anteil der angeforderten Verbindungen dar und macht deshalb trotz des geringen Bandbreitenbedarfs die größte Übertragungskapazität aus. Neben diesen Anforderungen muß auch Multimediaverkehr übertragen werden. Unbeantwortet bleibt die Frage, welche Dienste in der Zukunft in welchem Umfang genutzt werden [28] und welche neuen Dienste in absehbarer Zukunft eine Nachfrage auslösen können.

### Audioübertragung

In typischen paketorientierten Kommunikationssystemen, wird die Sprache digitalisiert, codiert und in Pakete eingebunden. Die Übertragung dieser Einheiten stellt wie bei anderen Echtzeitdiensten eine strenge Anforderung an die Verzögerung und an den Jitter. Übertragungszeiten $> 0.25s$ wirken sich negativ auf die Interaktion zwischen den Teilnehmern aus. Weiterhin beeinflussen Echos, die bei ab einer Verzögerung von $> 250ms$ hörbar auftreten, die Qualität der Verbindung negativ. Sie können dann nur durch zusätzlichen technischen Aufwand[1] unterdrückt werden.
Der Burstfaktor für unkomprimierte Sprache ist 1. Das heißt, daß, obwohl während des Sprechens immer wieder Pausen auftreten, diese auch mit der vollen Bandbreite übertragen werden. Detaillierte Untersuchungen [70] haben ergeben, daß die Sprachpausen $60\% - 65\%$

[1] Echokompensation

| | mittlere Bandbreite [Mbit/s] | durchschnittliche Verzögerung [s] | Bitfehler-rate | Burst-faktor |
|---|---|---|---|---|
| Audio | | | | |
| Telefonqualität | | | | |
| unkomprimiert | 0.064 | $< 0.25$ | ca. $10^{-3}$ | 1 |
| komprimiert | 0.006 - 0.032 | $< 0.25$ | ca. $10^{-3}$ | 2 - 10 |
| CD-Qualität | 1.4 | $< 0.25$ | $< 10^{6}$ | 2 - 10 |
| Video | | | | |
| unkomprimiert | 40 - 100 | $< 1$ | ca. $10^{-3}$ | 1 |
| komprimiert | 0.4 - 150 | $< 1$ | $< 10^{-6}$ | ca. 5 - 20 |
| Bildtelefon | 0.2 - 2 | $< 0.25$ | $< 10^{-6}$ | ca. 5 |
| Datentransfer | | | | |
| LAN-Kopplung | $< 1000$ | 1 - 1000 | 0 | 5 - 100 |
| Echtzeit Daten | $< 1000$ | 0.001 - 1 | 0 | 5 - 20 |

Tabelle 2.1: Qualitätsparameter einiger Übertragungsdienste

in jeder Richtung einer Sprachverbindung ausmachen. Zur Steigerung der Effizienz von Übertragungssystemen werden vielfach Verfahren eingesetzt, um die Pausen herauszufiltern und um die Sprache zu komprimieren. Das Resultat zeigt sich in dem Burstfaktor, der dann zwischen 2 und 10 variiert.

### Datenübertragung

Datenapplikationen sind generell intolerant gegenüber Fehlern und setzen daher eine völlige Integrität voraus. Zur Sicherstellung der fehlerfreien Übertragung sind deshalb unterschiedliche Mechanismen in den Kommunikationsprotokollen installiert. Sie reichen von Verfahren zur Fehlererkennung und Behebung bis hin zur wiederholten Sendung von fehlerhaft empfangenen Daten. Im Gegensatz zu den meisten anderen Applikationen unterliegen diese Dienste bezüglich der Verzögerung und des Jitters keinen besonderen Beschränkungen, da bei den Kommunikationspartnern keine Synchronisation erforderlich ist und somit auch keine konstanten Zeitraten benötigt werden. Die Burstiness variiert in Abhängigkeit von den Anwendungen und kann Werte bis zu 100 annehmen.

### Videoübertragung

Im Gegensatz zur Sprachübertragung, bei der nur kurze voneinander unabhängige Samples übermittelt werden, ist die Übertragung von Videosequenzen erheblich komplexer. Einerseits enthält ein Bild, das aus vielen Pixeln, die sich in Helligkeit und Farbe unterscheiden können, aufgebaut ist, sehr viel mehr Daten. Andererseits entstehen auch, um die Informationsmenge zu reduzieren, in Abhängigkeit von der eingesetzten Codiertechnik, Korrelationen zwischen den Bildern [67]. Bei einer paketorientierten Übertragung werden diese komprimierten Daten - fragmentiert und über eine Vielzahl von kleineren Einheiten verteilt - übermittelt. Die Auswirkungen des Verlustes einer solchen Dateneinheit oder das Auftreten eines Bitfehlers auf die Videoqualität hängt dann entscheidend von den enthaltenen Informationen ab und kann einen Verlust der Phasenorientierung oder der Synchronisation zur Folge haben, was sich in einer Unschärfe oder in Form von Störungen wie einer Welle, einer Anzahl von verzerrten Linien oder aber in einer Nebelwolke manifestieren kann. Wichtig ist deshalb zur Aufrechterhaltung der geringsten Videoqualität, daß Synchronisationssignale und rudimentäre Bildinformationen übertragen werden. Weitergehende Informationen, die eine größere Bildqualität liefern, sind nicht obligatorisch. Diese Unterteilung resultiert darin, daß die Datenpakete vor ihrer Übertragung informationsabhängig klassifiziert und priorisiert werden müssen. Pakete, die essentielle Informationen enthalten, müssen infolgedessen mit einer hohen, alle anderen mit einer niedrigeren Piorität versehen werden. Durch Einsatz dieser Techniken beträgt die tolerierbare Bitfehlerrate $\approx 10^{-3}$. Der Burstfaktor ist 1 für die unkomprimierte Übertragung von Videosequenzen und variiert zwischen 5 und 20 in Abhängigkeit vom eingesetzten Komprimierungsverfahren. Ebenso wie bei der Audioübertragung bestehen hier strikte Anforderungen an die Verzögerung. So sind die Übertragungszeiten auf maximal $1s$ beschränkt.

**Multimedia Applikationen**

Multimedia-Applikationen sind eine Obermenge der oben aufgeführten Dienste. Sie sind eine beliebige Kombination der Dienste, so daß Informationen mit Hilfe von Text, Grafik und Video simultan dargestellt und vermittelt werden können. Bei diesen Anwendungen sind mehrere Basisdienste miteinander verknüpft. Es sind neue Dienste denkbar, bei denen Standbilder oder Texte mit Musik oder Sprache hinterlegt sind. Andere Applikationen verknüpfen Video und Audio oder vereinigen alle genannten Dienste, d. h. Text, Standbilder, Video und Audio. Neben den servicespezifischen Anforderungen an die Übertragung, die natürlich auch weiterhin bestehen, kommen durch die Synthese der unterschiedlichen Informationsformen weitere verschärfte Anforderungen in Bezug auf die Synchronisation zum Tragen.
Der Bereich erstreckt sich im einfachsten Fall von der groben Verknüpfung verschiedener Objekte, wie zum Beispiel die laufende Übertragung von Standbildern - Audio und Daten - bis hin zur äußerst präzisen Synchronisation der Sprache zur Lippenbewegung des Sprechers. Diese sog. Lippensynchronisation toleriert Verzögerungen zwischen -90 bis zu 120 ms. Das Maß für die Verknüpfung der Objekte wird *Skew* genannt.

### 2.1.1 Übertragungsverfahren

Durch die stetige Zunahme des Verkehrsaufkommens in den Netzen auf Grund der ständig wachsenden Anzahl von Benutzern bildet die installierte Übertragungsbandbreite zwangsläufig einen nicht zu übersehbaren Engpaß. Die Erhöhung der Bandbreite zur Entlastung der Verbindungen bietet sich vordergründig als die Lösung des Problems an. Tieferge-

hende Untersuchungen zeigen aber, daß die Ursachen nicht allein auf dem vergrößerten Verkehrsaufkommen beruhen, vielmehr haben sich auch das Profil und die Anforderungen des Verkehrs verändert. Darüber hinaus ist die Integration dieser Dienste mit den unterschiedlichen Anforderungsspektren und unter Berücksichtigung der Randbedingungen, die Dienstqualität und eine effiziente Auslastung gewährleisten zu können, in ein Netzwerk problematisch. Der Austausch von Informationen über ein Netzwerk ist im Allgemeinen, wie Tabelle 2.1 entnommen werden kann, burstbehaftet. Dieser Faktor variiert applikationsabhängig zwischen 1 bei der Übertragung von Audiodaten und Werten > 100 bei der Kopplung von Local Area Networks.
Bei der synchronen[2] Übertragung der Daten wird für jeden Dienst eine feste Bandbreite während der gesamten Übertragungszeit reserviert. Die Belegung der Bandbreite erfolgt indem jeder Verbindung, wie in Abbildung 2.3 dargestellt, ein oder mehrere Kanäle exklusiv zugeordnet werden. Diese Reservierung erfolgt unabhängig vom temporären Lastmuster der Verkehrsquelle und richtet sich nach der maximal benötigten Bandbreite. Bei einer stark burstbehafteten Verkehrslast wie in Abb. 2.2 verdeutlicht, würden diese, auf Basis der Spitzenlast, reservierten Ressourcen nur für Bruchteile der Verbindungszeit benötigt.

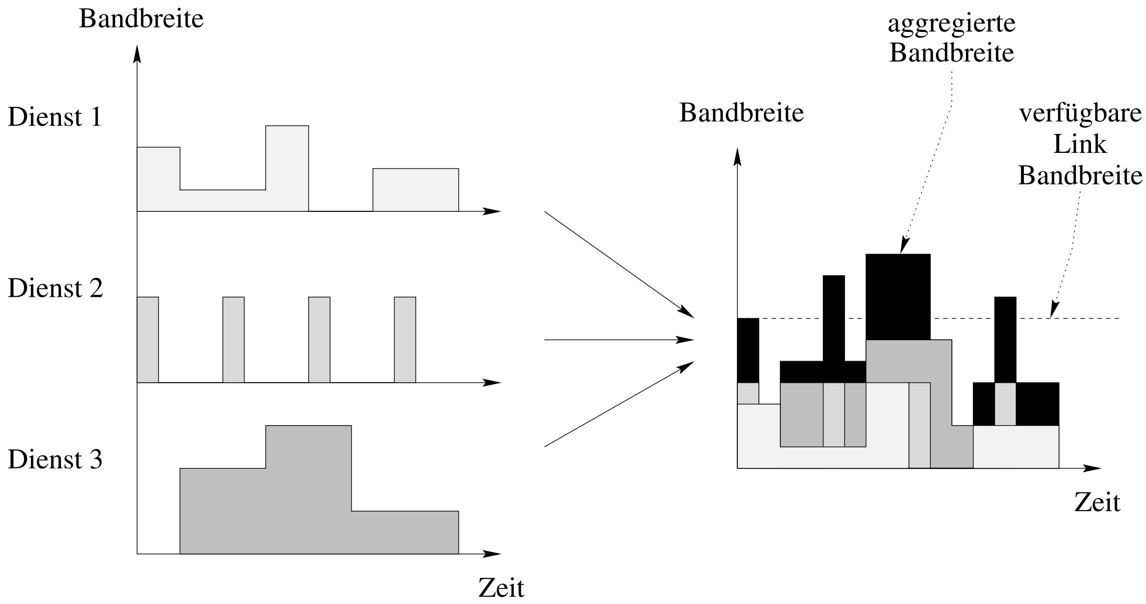



Abbildung 2.2: Prinzip des statistischen Multiplexens

Bei detaillierterer Betrachtung der Darstellung 2.2 zeigt sich weiterhin, daß für jeden Dienst soviel Bandbreite reserviert werden müßte, daß letztendlich nur *eine* Verbindung aufgebaut werden könnte.
Zusammenfassend kann festgestellt werden, daß dieses synchrone Übertragungsverfahren stets die gewünschte Dienstqualität bereitstellen kann. Nachteilig ist aber, daß Ressourcen

[2] In diesem Fall liegt eine feste Kopplung zwischen dem Übertragungskanal und der Verbindung vor.

die auf Grund des burstbehafteten Verkehrs nicht genutzt werden, nicht anderen Verbindungen zur Übertragung von Daten zur Verfügung gestellt werden können. Weiterhin kann bei ungünstigen Konstellationen der Fall auftreten, daß freie Übertragungsbandbreite keinen weiteren Verbindungen mehr zugeordnet werden kann.
Die Folge ist, daß ein System, das auf diesem Übertragungsverfahren basiert, vielfach nicht effizient betrieben werden kann.
Anders bei der asynchronen Übertragung von Daten. Hier liegt keine feste Kopplung zwi-

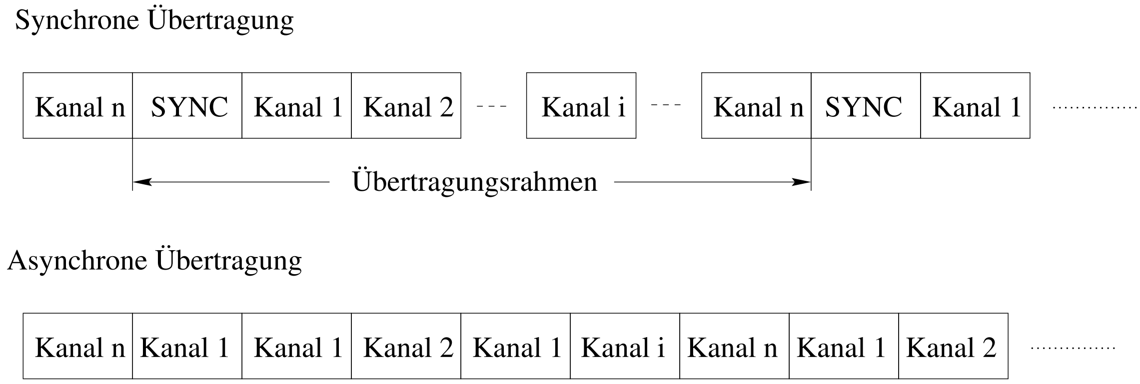



Abbildung 2.3: Schematische Darstellung synchroner und asynchroner Übertragungsverfahren

schen einem Übertragungsrahmen und einem Verbindungskanal vor (Abb. 2.3). Bandbreite wird bei diesem Verfahren nur belegt, wenn Daten zur Übertragung anstehen. Die Dienste arbeiten also mit einer variablen Bitrate. Durch den Einsatz dieser Methode werden Ressourcen zur Übertragung von Daten anderer Verbindungen freigesetzt. Abbildung 2.2 zeigt, daß so durch den Einsatz eines Multiplexers drei Verbindungen gleichzeitig aufgebaut und auch betrieben werden können. Die Effizienz konnte so erheblich gegenüber dem synchronen Übertragungsverfahren verbessert werden.
Problematisch bei diesem Verfahren ist, daß die Reservierung der Übertragungskapazitäten für die einzelnen Verbindungen auf einer statistischen Annahme über das Verkehrsverhalten eines Dienstes beruht.
Der Vorsatz einer jeden Netzplanung ist es, die netzinternen Übertragungswege und vermittlungstechnischen Einrichtungen möglichst optimal auszunutzen und damit eine hohe Auslastung des gesamten Netzes zu erhalten. Um dieses Ziel zu erreichen, werden die Netzressourcen gegebenenfalls überbucht (Abb. 2.2). Von einer Überbuchung spricht man, wenn die Summe aller Bandbreiten der eingerichteten Verbindungen größer ist als die maximale Übertragungsrate der Leitung selbst. In bestimmten Situationen, wenn die Bandbreite mehrerer Verbindungen gleichzeitig erhöht werden, entsteht eine transiente Überlast im Netz, um eine zufriedenstellende Übertragung zu gewährleisten. Hier müssen dann weitergehende Strategien greifen.
Die vorhandenen Ressourcen sind dann möglichst gerecht zwischen den verschiedenen Verbindungen aufzuteilen. Zur Sicherstellung, daß die vorhandenen Übertragungswege mög-

lichst wirksam ausgenutzt werden und eine Bandbreite je nach Bedarf und Möglichkeit den Anwendungen zur Verfügung steht, müssen *Verkehrsmanagementverfahren* eingesetzt werden.

## 2.2 Verkehrsmanagement in Kommunikationsnetzen

Um eine solche Vielzahl an Diensten mit ihren vielseitigen und dynamischen, teilweise sogar entgegengesetzten Charakteristika und Anforderungen an die Übertragung unterstützen und gleichzeitig effizient und kostenoptimal arbeiten lassen zu können, muß das Transportsystem eine *große Flexibilität* und eine *überdurchschnittliche Leistungsfähigkeit* in allen aufgeführten Bereichen aufweisen. Um darüber hinaus auch noch die individuellen Belange der Nutzer zu berücksichtigen und sich auf neue (zukünftige) Verkehrsklassen und Applikationen einstellen zu können, muß sich das eingesetzte Verkehrsmanagement durch folgende Eigenschaften auszeichnen.

- Einfachheit
  Die eingesetzten Algorithmen zur Verkehrssteuerung und zum Management müssen möglichst einfach gestaltet sein und ohne großen Aufwand implementiert werden können.

- Adaptivität
  Auf Grund des stetig wechselnden Verkehrsprofils und der universellen Einsetzbarkeit der Managementsysteme ist die schnelle Anpassung an neue Anforderungen essentiell.

- Robustheit
  Wegen der vielen unterschiedlich stringenten Anforderungen der Applikationen muß ein hohes Maß an Ausfallsicherheit gegeben sein. Es ist deshalb zwingend notwendig, daß auch noch nach Ausfall von Kontrollmechanismen der Datenverkehr rudimentär geregelt wird.

- Kontrollierbarkeit
  Der Austausch von Daten aber auch Überlastsituationen und deren Abbau müssen überwacht, kontrolliert und gesteuert werden, so daß eine optimale Auslastung der Ressourcen möglichst ohne Leistungseinschränkungen erfolgen kann.

Eine grundlegende Aufgabe des Verkehrsmanagements besteht in der Kontrolle und Überwachung der Verkehrslast. Um zu verhindern, daß es nicht zu einer Verminderung der Dienstqualität oder nachhaltigen Beeinflussung von aufgebauten Verbindungen kommt, werden im Wesentlichen zwei Mechanismen installiert, die zu verschiedenen Zeitpunkten, d. h. mit unterschiedlichen Ansätzen in den Kommunikationsablauf eingreifen. Man unterscheidet hier die *Verkehrssteuerung am Netzzugang* und die *netzinterne Verkehrssteuerung* sowie *präventive* und *rückgekoppelte* Verfahren.

### 2.2.1 Feedback Control

Netzinterne Mechanismen regulieren den Verkehrsfluß an der Netzzugriffseinheit auf der Basis des gegenwärtigen Netzzustands, d. h., daß diese Mechanismen erst dann greifen, wenn bestimmte Lastsituationen im Netzwerk vorliegen. Aktionen werden als Reaktion auf bestimmte unerwünschte Zustände eingeleitet. Man spricht deshalb von *rückgekoppelten Verfahren* [78].
Für die Bewertung dieses Zustandes und zum Erkennen von anstehenden Konflikt- und Überlastsituationen ist ein Feedback von den Netzwerkmanagementinstanzen unbedingt erforderlich. Die zeitlichen Randbedingungen für die Übertragung und Auswertung dieser Indikatoren sind zu sehr eingeschränkt, um eine effektive Kontrolle zu etablieren. Sie müssen eine Überlastsituation rechtzeitig anzeigen, sollten aber auch transiente Situationen, die bei burstbehaftetem Multimediaverkehr vorkommen, auch als temporäre Zustände bewerten können. Darüber hinaus besteht durch diese Rückkopplung inhärent die Gefahr, daß Instabilitäten auf Grund von Schwingungen und Überreaktionen auftreten.

### 2.2.2 Verkehrssteuerung am Netzzugang

Die Zugangskontrolle wird eingesetzt, um beim Aufbau einer Verbindung festzustellen, ob genügend Netzkapazitäten zur Verfügung stehen, um den Dienst abwickeln zu können. Die Entscheidung beruht zum einen auf den Verkehrs- und Qualitätsparametern des zu vermittelnden Dienstes und zum anderen auf dem gegenwärtigen Zustand des Netzes. Dieser Parametersatz dient dann im Folgenden dazu, die aufgebauten Verbindungen zu überwachen und den reibungslosen Ablauf der Kommunikation sicherzustellen.
Im Gegensatz zu den rückgekoppelten Kontrollverfahren, bei denen die Automatismen erst dann greifen, wenn Verluste oder Überlastsituationen auftreten, wird bei einem präventiven Ansatz durch den Einsatz geeigneter Mechanismen versucht, diese unakzeptablen Netzzustände, die die Dienstqualität nachhaltig beeinflussen, schon im Vorfeld auszuschließen. Dieser Ansatz ist gerade in Netzen, die auf einer verbindungsorientierten Übertragung basieren, sehr effektiv. Die Entscheidung, ob eine neue Verbindung zwischen zwei Teilnehmern aufgebaut werden kann, wird dabei von dem Status der Route, über die die Daten übertragen werden sollen, und von den Anforderungen des Dienstes abhängig gemacht. Es wird im Vorfeld abgeklärt, ob eine neue Verbindung etabliert werden kann, ohne daß eine Beeinträchtigung der bestehenden Verbindungen zu erwarten ist. Die Bewertung der Dienste erfolgt an Hand spezifischer Parameter, die im sog. *Verkehrsvertrag* verankert sind.

#### Verkehrsvertrag

Dieser Verkehrsvertrag wird zwischen dem Benutzer und dem Betreiber des Netzes abgeschlossen. Er beschreibt die Lastcharakteristika sowie die Qualitätsanforderungen des Dienstes und dient dazu, das Netzwerk effizient zu betreiben. Die Belegung der Ressourcen erfolgt an Hand der vereinbarten Parameter ebenso wie das Management - Überwachung und Steuerung - der Datenströme und die Tariffierung. Für die vielfältigen Aufgaben werden die Parameter in zwei Klassen aufgeteilt.

**Source Traffic Descriptor**

Die in dem Source Traffic Descriptor zusammengefaßten Kenngrößen beschreiben die inhärenten Charakteristika der Verkehrsquellen. Im Wesentlichen umfassen sie die maximale, minimale und durchschnittliche Paketdatenrate, die Burstiness sowie die Burst- und die Paketlänge.

**Paketübertragungsrate**

Die Paketübertragungsrate läßt sich aus der Anzahl gesendeter Informationseinheiten in einem angemessenen Zeitraum bestimmen. Da bei vielen der vorgestellten Applikationen die Daten nicht kontinuierlich sondern in Bursts erzeugt werden, treten unterschiedliche variable Bitraten auf. Man unterscheidet die maximale Transferrate und die minimale Übertragungsrate. Aus beiden Werten läßt sich unter Berücksichtigung weiterer relevanter Lastparameter, wie der Burstiness, eine mittlere Paketübertragungsrate ableiten.

**Mittlere Paketübertragungsrate**

Für jede Verbindung durch das Netz wird eine Datenübertragungsrate festgelegt, die unabhängig von der physikalischen Geschwindigkeit der Anschlußleitung ist. Diese garantierte mittlere Datenübertragungsrate wird als *Sustainable Cell Rate (SCR)* bezeichnet. Die SCR kann Werte zwischen 0 kbit/s (Sonderfall) und der *maximalen Anschlußleitungsgeschwindigkeit* haben. Es handelt sich um einen Mittelwert, der sich aus einem burstartigen Datenverkehr, über einen bestimmten vorgegebenen Zeitraum betrachtet, ergibt.

**Burstlänge**

Für jede einzelne Verbindung wird außer den Übertragungsraten noch eine *vereinbarte Datenmenge*, die sogenannte *Committed Burst Size* ($B_c$), und eine *zusätzliche Datenmenge*, die *Excess Burst Size* ($B_e$), festgelegt. Durch $B_c$ wird die Datenmenge festgelegt, die in einem gewählten Zeitintervall von einem Teilnehmer in das Netz gesendet werden kann und vom Netz transparent zum Ziel transportiert wird.

$B_e$ beschreibt die maximale Anzahl an Daten, die in dem Zeitintervall zusätzlich zu $B_c$ gesendet werden kann, ohne daß diese Daten am Netzeingang verworfen werden. Diese Daten werden jedoch vom Eingangsknoten mit einer gegenüber den konformen Paketen verminderten Priorität gekennzeichnet, die in Überlastzuständen ein einfaches Verwerfen ermöglicht.

**QoS Descriptor**

Neben der Beschreibung der Übertragungsanforderungen durch die physikalischen Randbedingungen wird eine Verbindung noch durch die Qualitätsanforderungen, die zur Abwicklung der Dienste notwendig sind, charakterisiert. Die Parameter zur Beschreibung der sog. *Quality of Service* (QoS) sind in dem sog. QoS Descriptor verankert. Diese Angaben sind deshalb notwendig, damit das Netzmanagement für die Berücksichtigung der

speziellen Randbedingungen ein abgestuftes an die Dienste angepaßtes Leistungsangebot zur Verfügung stellen kann. Die Übertragungsqualität ist nicht allein von der zur Verfügung stehenden Bandbreite abhängig. Sie wird dadurch erreicht, daß diese Ressourcen entsprechend der Anforderungen der Applikationen und der Randbedingungen des Netzmanagements verwaltet werden. Diese Verwaltung und Kontrolle muß auf der einen Seite gewährleisten, daß die für einen Dienst reservierte Bandbreite für Applikationen, die nach dem „Best-Effort-Prinzip" arbeiten, nicht mehr zur Verfügung stehen.
Wichtige Indikatoren für die Bewertung der Übertragungsqualität sind die Verzögerung von Paketen, Schwankungen der Verzögerungszeiten und die verschiedenen Verlustraten.

**Verzögerung**
Die Verzögerung zwischen zwei Punkten eines Netzes ist das Zeitintervall zwischen dem Senden des ersten Bits einer Informationseinheit durch die Datenquelle und dem Empfangen des letzten Bits an dem Zielpunkt. Die gesamte Verzögerung ergibt sich durch die Summe der Verarbeitungs- und Übertragungszeiten in den einzelnen Netzwerkinstanzen, die durchlaufen werden. Die Daten werden codiert, paketiert, übertragen, gespeichert, vermittelt und im Empfänger wieder reassembliert und decodiert. Auf diesem Weg durch das Netz kann die Verzögerung der Dateneinheiten - lastabhängig - unterschiedlich lang sein. Aus der Differenz von Minimalwert und Maximalwert ergibt sich die *Verzögerungsschwankung*.

**Verzögerungsschwankung**
Die Verzögerungsschwankung oder Jitter kann auf der Empfangsseite durch den Einsatz großer Buffer, mit denen es möglich ist, die Pakete kontrolliert zu verzögern, ausgeglichen werden. Daten, die unterschiedlich lange Laufzeiten hatten, werden entsprechend ihrer Ankunftszeiten im Speicher abgelegt und dann wieder mit der konstanten Taktrate ausgelesen. Mit der Entkopplung der Eingangs- und Ausgangsseite durch den Speicher geht aber auch gleichzeitig eine weitere Verzögerung der Pakete einher. Das bedeutet, daß es nicht immer möglich ist, Verzögerungsschwankungen auszugleichen. In Fällen, in denen die maximale Verzögerung überschritten wird, müssen die Pakete absorbiert werden.

**Fehlerrate**
Im Allgemeinen ist eine Fehlerrate definiert als das Verhältnis aus der Anzahl innerhalb eines Zeitintervalls geeigneter Länge fehlerhafter Informationseinheiten zur Summe aller Informationseinheiten. Abhängig von der Größe der informationstragenden Einheiten kann zwischen der Bitfehler- und der Paketverlustrate, d. h. von der Verlustrate von Informationsblöcken, unterschieden werden.

- Bitfehlerrate (Bit Error Rate - BER)
  Die Bitfehlerrate ist das Verhältnis zwischen der Anzahl fehlerhafter empfangener Bits und der Summe aller empfangenen Bits. Die Bitfehlerrate wird hauptsächlich durch das Übertragungssystem bestimmt[3] und wird aus diesem Grund bei dieser

[3] Sie hängt z. B. von dem eingesetzten Medium und den Einsatzbedingungen ab.

Untersuchung nicht zur Beschreibung der Dienstqualität herangezogen.

- Paketverlustrate
  Die Paketverlustrate ergibt sich aus dem Verhältnis zwischen der Anzahl fehlerhafter oder verworfener Pakete und der Summe aller übertragenen Pakete. Auftreten können diese Verluste auf Grund unzureichender Ressourcen in den Knoten eines Netzwerkes infolge transienter Überlastungen oder Hardware-Fehler. Die Auswirkungen eines Paketverlustes auf die Applikationen ist unterschiedlich und hängt von spezifischen Charakteristika wie z. B. dem Kompressionsverfahren und Netzwerk-spezifischen Eigenarten wie der Paketgröße ab.

Basierend auf diesen Parametersätzen erfolgt das Management der unterschiedlichen Dienste auf verschiedenen Ebenen im Kommunikationsprotokoll.

### 2.2.3 Call Admission Control

Ein wesentlicher Aspekt beim Aufbau von Verbindungen stellt die sog. Connection Acceptance Control bzw. Connection Admission Control (CAC) dar. CAC beschreibt dabei die Aktionen, die in den Netzwerkinstanzen während der Verbindungsaufbauphase abwickelt werden müssen, um im Vorfeld sicherzustellen, daß die Qualität der bestehenden Dienste nicht oder nur im tolerierbaren Maß beeinträchtigt wird. Bei einem Verbindungsaufbauversuch wird ein sog. Verkehrsvertrag mit den Netzinstanzen ausgehandelt. In dieser Vereinbarung werden die Verkehrscharakteristika und die Übertragungsqualität des Dienstes durch den Benutzer beschrieben. Der Netzbetreiber muß während der Signalisierungsphase prüfen, ob das Netz entlang der Übertragungsstrecke genügend Ressourcen zur Verfügung stellen kann, um einerseits die geforderte Dienstqualität abzudecken und andererseits die Qualität der bestehenden Verbindungen auch weiterhin zu gewährleisten. Die Verbindung wird erst dann geschaltet, wenn die Übertragungs- und Qualitätsanforderungen entlang des gesamten Weges durch das Netz, erfüllt werden können. Wenn die Verbindung aufgebaut werden kann, wird dies durch eine „confirm“ Meldung signalisiert. Können die Anforderungen nicht erfüllt werden, wird der Verbindungswunsch mit einer „reject“ Nachricht abgewiesen.

#### Funktionsweise der Call Admission Control

Wie in Abbildung 2.4 dargestellt, setzt sich der Admission Controller aus zwei funktionalen Einheiten zusammen.
Der *Traffic Qualifier* wertet die servicespezifischen Kenngrößen aus dem Verkehrsvertrag aus und nimmt eine Klassifikation des Verkehrsverhaltens vor. Im einfachsten Fall kann dies die zu reservierende Übertragungsbandbreite sein. Das Ergebnis dieser Einstufung spiegelt dann die Anforderung, die der Dienst an das Übertragungssystem stellt, wider. Die zweite Einheit, der *State Qualifier*, verarbeitet die systemspezifischen Zustandsgrößen und ermittelt so ein Maß für die Auslastung des Knotens und der Übertragungseinrichtungen entlang der Verbindungsstrecke zum Zielteilnehmer.

Beide Kenngrößen werden daraufhin in der *Decision Unit* miteinander verglichen. Kann die Anforderung durch die noch freien Reserven abgedeckt werden, kommt eine Verbindung zustande. Dem anfordernden Teilnehmer wird dies durch eine *confirm-Meldung* signalisiert. Im negativen Fall erfolgt eine Anzeige durch eine *reject-Mitteilung.*

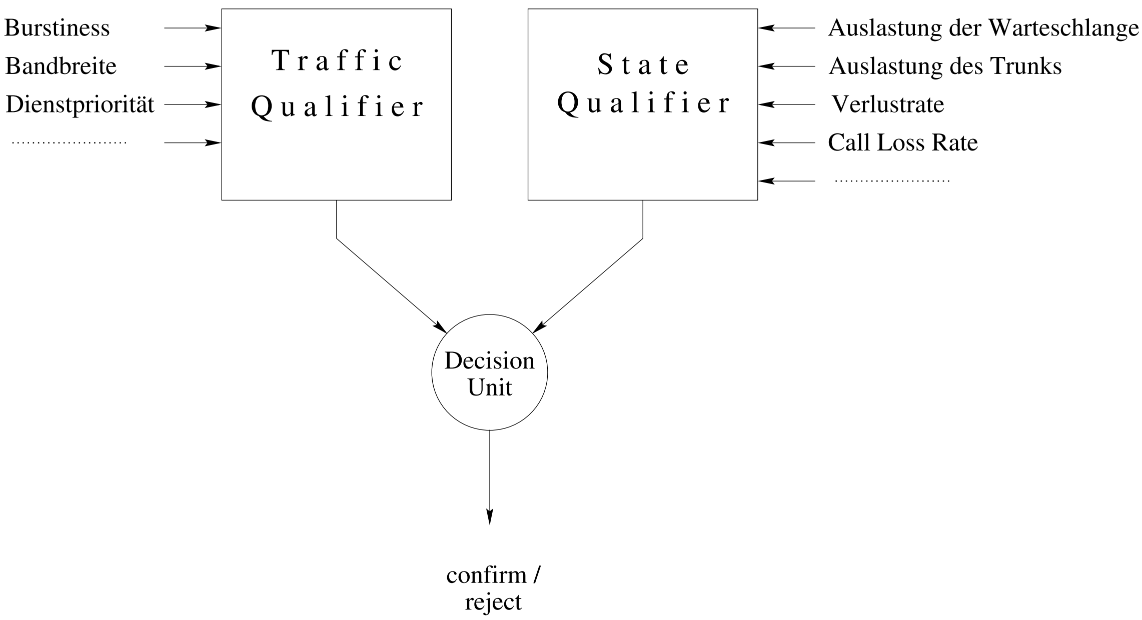



Abbildung 2.4: Funktionaler Aufbau eines Call Admission Controllers

### 2.2.4 Statische Zugangskontrolle

Bei diesen konventionellen Verfahren muß der Kontroller auf der Basis Qualitätsanforderungen auf der einen Seite und dem Netzwerkstatus, der sich in der Auslastung der Ressourcen reflektiert, darüber entscheiden, ob eine Verbindung aufgebaut werden kann. Bei diesen Verfahren kann eine Verbindung nur dann aufgebaut werden, wenn genügend Ressourcen zur Verfügung stehen, um die gewünschten Qualitätsanforderungen abzudecken und gleichzeitig die bestehenden Verbindungen mit der zugesicherten Dienstgüte abzuwickeln.

#### Peak Reservation Methode (PR)

Ein Verfahren, um den Zugang zum Netz zu begrenzen, stellt die Peak-Rate Reservation Methode dar. Bei dieser Methode wird eine Reservierung auf Basis der maximal möglichen Bandbreite eines Dienstes vorgenommen. Bei jedem Verbindungswunsch wird in dem *State Qualifier* die Auslastung ermittelt, d. h. es wird die bereits reservierte Bandbreite ermittelt. In der *Decision Unit* wird dann geprüft, ob die bereits reservierte Bandbreite zuzüglich der angeforderten maximalen Übertragungsbandbreite kleiner oder gleich der nominalen Übertragungskapazität des angeschlossenen Links ist. Trifft dies zu, wird die Verbindung geschaltet. Ist die Summe der Bandbreiten größer als die Übertragungskapazität, wird der Verbindungsaufbau unterbrochen und der Verbindungswunsch abgelehnt. Dieses Verfahren läßt sich durch den Zusammenhang 2.2 beschreiben.

$$Decision_{Admission} = \begin{cases} accept & : \text{ für } C > BW_{max,k} + \sum_{i=1}^{N} BW_{max,i} \\ reject & : sonst \end{cases} \tag{2.2}$$

Bei dieser Gleichung stellt C die maximale Übertragungskapazität des Links dar. N ist die Anzahl der aufgebauten Verbindungen, deren jeweilige maximale Bandbreiten durch die Parameter $BW_{max,1}...BW_{max,N}$ repräsentiert werden. $BW_k$ ist die maximale Bandbreite des Dienstes, der aufgebaut werden soll. Bei der Anwendung dieses Verfahrens wird für die einzelnen Verbindungen ähnlich wie bei der Leitungsvermittlung soviel Bandbreite exklusiv reserviert, daß keine Paketverluste auf Grund von Speicherüberläufen auftreten können. Die Auslastung der reservierten Bandbreite hängt bei diesem Reservierungsverfahren von der Burstiness (B), also dem Verhältnis zwischen der maximalen und der mittleren Bandbreite, sowie der Verweilzeit in dem Zustand, in dem die Daten mit der maximalen Übertragungsrate erzeugt werden, ab. Dieser Zusammenhang[4] wird durch Gleichung 2.3 beschrieben.

$$BW_{Utilization} = \frac{\frac{T_{max}}{T_{ges}} \cdot (B-1) + 1}{B} \tag{2.3}$$

Der Quotient von $T_{max}$ und $T_{ges}$ in Gl. 2.3 ist das sog. Activity Ratio. Es beschreibt den Anteil der Zeit in dem Daten mit der maximalen Bandbreite übertragen wurden, bezogen auf die gesamte Übertragagungszeit.

Ausgehend von diesem Zusammenhang verdeutlicht Abbildung 2.5 anschaulich die Abhängigkeit der Nutzung der reservierten Bandbreite von der Burstiness eines Dienstes und dem Activity Ratio. Für den Fall, daß die reservierte der durchschnittlichen Bandbreite entspricht - entweder gilt $B = 1$ oder $T_{max} = T_{ges}$ - wird die Bandbreite vollständig genutzt. Der Wirkungsgrad beträgt 100%. Ist $B > 1$, sinkt der Auslastungsgrad monoton mit dem Activity Ratio. Auf Grund dieser Abhängigkeit stellt das Peak Reservation Verfahren für stark burstbehaftete Dienste mit nur kurzen Verweilzeiten in den Zuständen mit einer hohen Datenrate, die keine stringenten Anforderungen an die Verlustrate stellen, eine suboptimale Lösung dar. Ein anderes Verfahren, bei dem die effiziente Auslastung der Bandbreite gegenüber der Verlustrate präferiert wird, ist die *Minimal Bandwidth Reservation Methode*.

### Minimal Bandwidth Reservation Methode (MR)

Diese Methode zeichnet sich dadurch aus, daß eine Reservierung von Übertragungskapazität auf Basis der *minimalen* Bandbreite unter der Randbedingung, daß $BW_{min} \overset{!}{>} 0$, eines Dienstes vorgenommen wird. Bei jedem Verbindungswunsch wird geprüft, ob die bereits reservierte Bandbreite zuzüglich der gewünschten minimalen Übertragungsbandbreite kleiner

[4]Herleitung in Anhang A.1

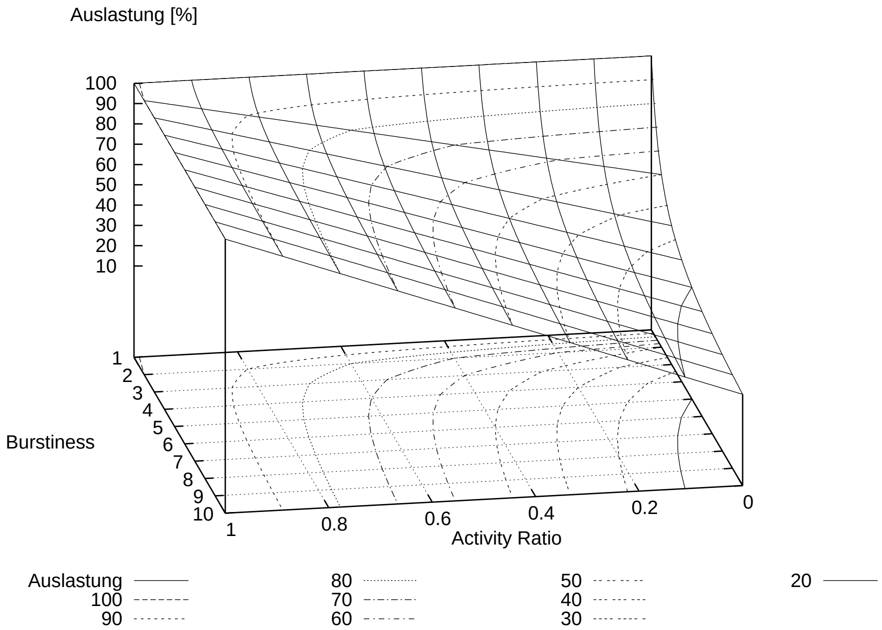



Abbildung 2.5: Auslastung der reservierten Bandbreite bei Verwendung der Peak Reservation Methode

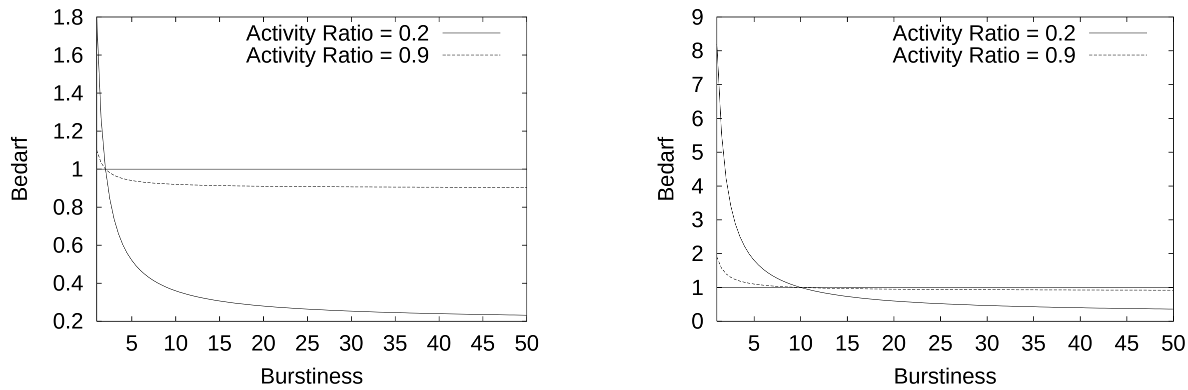
oder gleich der nominalen Übertragungskapazität des angeschlossenen Links ist. Trifft dies zu, wird die Verbindung geschaltet. Ist die Summe der Bandbreiten größer als die Übertragungskapazität wird der Verbindungsaufbau unterbrochen und der Verbindungswunsch abgelehnt. Dieses Verfahren läßt sich durch den Zusammenhang 2.4 beschreiben.

$$Decision_{Admission} = \begin{cases} accept & : \text{ für } C > BW_{min,k} + \sum_{i=1}^{N} BW_{min,i} \text{ mit } BW_{min,k} > 0 \\ reject & : sonst \end{cases} \tag{2.4}$$

Bei dieser Gleichung stellt C die maximale Übertragungskapazität des Links dar. N ist, wie beim PR Verfahren, die Anzahl der aufgebauten Verbindungen, deren jeweilige minimale Bandbreiten durch die Parameter $BW_{min,1}...BW_{min,N}$ repräsentiert werden. $BW_k$ ist die minimale Bandbreite des Dienstes, der aufgebaut werden soll. Die Auslastung der reservierten Bandbreite hängt bei diesem Reservierungsverfahren von der Burstiness (B), dem Verhältnis von maximaler zu minimaler Bandbreite sowie der Verweilzeit in dem Zustand in dem die Daten mit der minimalen Übertragungsrate erzeugt werden, ab. Dieser Zusammenhang wird durch Gleichung 2.5[5] beschrieben.

$$BW_{Bedarf} = \frac{T_{min}}{T_{ges}} + \frac{1}{B} \cdot \frac{BW_{max}}{BW_{min}} \left(1 - \frac{T_{min}}{T_{ges}}\right) \tag{2.5}$$

[5]Herleitung in Anhang A.2




(a) $\frac{BW_{max}}{BW_{min}} = 2$ (b) $\frac{BW_{max}}{BW_{min}} = 10$

Abbildung 2.6: Minimal Bandwidth Reservation

Abbildung 2.6 zeigt die Abhängigkeit des Bedarfs an Übertragungskapazitäten von der Burstiness und dem Activity Ratio für $\frac{BW_{max}}{BW_{min}} = 2$ und $\frac{BW_{max}}{BW_{min}} = 10$. Der Verlauf der Kurven kann grob in zwei Bereiche getrennt werden. Für den Fall, daß $BW_{\oslash} > BW_{min}$ ist der Bedarf an Ressourcen $> 1$. Der Link wird in dieser Betriebsweise überbucht. Die Überbelegung wird bei einem geringen Burstfaktor entscheidend durch das Verhältnis von $BW_{max}$ und $BW_{min}$ geprägt. Es treten Verluste und Verzögerungen auf. Erst wenn $BW_{\oslash} \leq BW_{min}$ und der Dienst stark burstbehaftet ist, ist der Bedarf an Übertragungskapazitäten geringer als die reservierte Bandbreite, so daß die QoS dann gewährleistet werden kann. Dieser Fall kann auftreten, wenn das Zustandsmodell der Verkehrsquelle neben den aktiven Phasen auch Stati aufweist, in denen keine Übertragung stattfindet.
Neben diesen Verfahren, die die Randbereiche darstellen, existieren in der Literatur weitere Algorithmen, bei denen die Reservierung der Ressourcen entlang der Verbindungsstrecke mit Durchschnittswerten erfolgt. Bei der Equivalent Bandwidth Methode (EB) geht man von einer für einen Dienst charakteristischen Verkehrslast aus. Die sog. äquivalente Bandbreite, die zur Reservierung der Ressourcen herangezogen wird, läßt sich dann aus service- und knotenrelevanten Parametern ableiten. So fließen in die Berechnung maximale und durchschnittliche Bitrate des Dienstes, die verfügbare Bandbreite des Links sowie die maximal tolerierbare Cell Verlustrate und der verfügbare Speicherplatz der Knotens ein. Bei der Weighted Variance Methode (WV) fließt darüber hinaus in die Entscheidung, ob eine Verbindung aufgebaut werden kann, die Varianz zwischen der mittleren und maximalen Transferrate des neuen Dienstes ein, so daß auf Grund des nicht unerheblichen Rechenaufwandes, Einschränkungen der Dynamik auftreten können.
Im Wesentlichen bedeutet das, daß die Parameter, die diesen Verfahren zu Grunde liegen, empirisch oder durch komplexe Berechnungen ermittelt werden müssen. Bei dem sich stetig ändernden Lastprofil der angebotenen Dienste zieht das eine permanente, zeitaufwendige Adaption nach sich.

#### Zusammenfassung

Wird für die Reservierung entlang des Pfades die Spitzenbitrate zu Grunde gelegt, erfolgt kein statistisches Multiplexen der Quellen. Es werden genügend Warteplätze für Pakete und Übertragungsbandbreite vorgehalten, damit auch sporadisch auftretende Verkehrsschwankungen problemlos aufgefangen werden können. Zellenverluste können in diesem Fall vernachlässigt werden. Im Gegensatz dazu führt eine Belegung mit einer kleineren Bitrate als der Maximalbitrate, mit der eine Datenquelle senden kann, zu einer Überbuchung des Links (Abbildung 2.2). Das simultane Auftreten von Perioden, in denen Quellen mit ihrer Spitzenbitrate Dateneinheiten emittieren, führt dazu, daß Verluste oder Verzögerungen, d. h. Einschränkungen in der QoS, auftreten.
Die in diesem Bereich eingesetzte Strategie muß einen Kompromiß darstellen, der sowohl die Belange der Nutzer als auch die der Service-Provider zufriedenstellend berücksichtigt. Sie muß ein Maximum an Verbindungen zulassen, muß aber auch die den Benutzern vertraglich garantierte Qualität liefern können.

### 2.2.5 Dynamisch adaptive Zugangskontrolle

Die dynamische Zugangskontrolle, die einen wesentlichen Teil dieser Untersuchung darstellt, bietet mehr Freiheitsgrade bei der effizienteren Auslastung des Netzes als statische Verfahren. Bei diesem Ansatz wird nicht allein auf der Basis der Verkehrsparameter und der momentanen reservierten Bandbreite entschieden, ob eine Verbindung aufgebaut werden kann.
Es werden darüber hinaus weitere Kenngrößen, die den Entscheidungsprozeß wesentlich beeinflussen können, sowie deren Abhängigkeiten untereinander und deren zeitlicher Verlauf berücksichtigt.
Als Einflußgrößen, die das Regelverhalten nachhaltig beeinflussen, haben sich folgende Kenngrößen herauskristallisiert.

#### Abweichung von der deklarierten Bandbreite $BW_\Delta$

$BW_\Delta$ ist die prozentuale Abweichung der tatsächlichen von der deklarierten Bandbreite. Die Berücksichtigung dieser relativen Abweichung der effektiven Bandbreite, mit der ein Dienst Daten von der im Rahmen des Verkehrsvertrages beim Verbindungsaufbau deklarierten Bandbreite generiert, sorgt dafür, daß der Controller auf Irregularitäten im Bedarfsfall entsprechend reagieren kann. Diese Kenngröße gibt an, um welchen Faktor sich die tatsächliche und die vereinbarte Bandbreite unterscheiden können, ohne daß die übrigen konformen Verbindungen über die vereinbarten Toleranzen hinaus, beeinflußt werden. Er kann aber auch in der Weise interpretiert werden, daß die im Verkehrsvertrag festgeschriebene Bandbreite im Folgenden um diesen Faktor herabgesetzt werden kann, ohne daß eine negative Beeinflussung der einzelnen Verbindungen zu erwarten ist. Diese verminderte effektive Bandbreite ermöglicht dann eine erhöhte Auslastung der vorhandenen Ressourcen, indem weitere Verbindungen aufgebaut und betrieben werden können.

#### Dienstpriorität $P_S$

Die statische Priorität $P_S$ berücksichtigt die stringente Abhängigkeit des Dienstes von relevanten Parametern. Sie beschreibt auf der einen Seite die Stärke der Bindung an die servicespezifischen Charakteristika wie z. B. die Echtzeitfähigkeit, dient aber auch zur Durchsetzung der gewünschten Optimierungsstrategie.
Kann ein Dienst Verzögerungen oder den Verlust von Informationseinheiten auf Grund von Redundanzen oder speziellen Sicherungsverfahren in gewissen Grenzen tolerieren, so kann die Priorität herabgesetzt werden. Im Gegensatz dazu bekommen Dienste, die sich durch eine harte Echtzeitanforderung ausweisen, eine sehr hohe Priorität, damit Verlust und Verzögerung von Zellen dieser Quelle so gering wie erforderlich bleiben.

#### Auslastung der Dienstwarteschlange $Q_U$

Der Einfluß der Auslastung der Warteschlange eines Dienstes $Q_U$ führt dazu, daß eine Quelle in dem Fall, daß viele Warteplätze belegt sind, auf Grund eines temporären Engpasses in Folge einer Überlastsituation, bevorzugt behandelt werden kann. Der Auslastungsgrad führt im Wesentlichen zu einer adäquaten Vergrößerung der Priorität des Dienstes. Bei der Anpassung dieses Stellenwertes müssen wiederum die Belange und Anforderungen der anderen Verbindungen einbezogen werden, um Beeinträchtigungen zu vermeiden oder nur in zulässigem Rahmen zuzulassen.

#### Systemauslastung $S_U$

In Anlehnung an die Auslastung der servicespezifischen Warteschlange gibt die Auslastung des gesamten Speichers die Möglichkeit, den Knotenzustand im Allgemeinen zu beschreiben und zu beurteilen. Im direkten Vergleich mit dem Parameter $Q_U$ kann die spezifische Lastsituation relativiert werden. Ist die Systemauslastung gering im Verhältnis zur Auslastung einer dienstspezifischen Warteschlange, liegt eine lokale Überlastung vor. Als Folge davon kann eine Priorisierung und damit bevorzugte Abarbeitung dieser Warteschlange erfolgen. In dem Fall, daß die Kenngrößen $S_U$ und $Q_U$ nahezu gleich sind, liegt eine globale[6] (knotenweite) Überlastsituation vor. Es erfolgt in diesem Fall keine bevorzugte Behandlung einzelner Warteschlangen.

#### Auslastung der abgehenden Links $T_U$

Die Auslastung der abgehenden Links stellt einen relevanten Indikator dar, um eine optimale Zuteilung der zur Verfügung stehenden Bandbreite gewährleisten zu können. Ist der Ausgang des Knotens nahezu ausgelastet, wird die zur Verfügung stehende Bandbreite also in hohem Umfang ausgenutzt, kann die geringe noch verfügbare Bandbreite dann nur unter wenigen, auf Grund ihrer Parameter privilegierte Quellen, aufgeteilt werden. Ist der

[6] Global bedeutet in diesem Fall, daß sich die Überlastsituaton nicht auf eine Verbindung sondern auf alle Verbindungen des Konotens bezieht.

Knoten jedoch lediglich gering ausgelastet, so daß die verfügbare Bandbreite ausreichend groß ist, können mehr oder sogar alle Quellen ihre Daten übertragen.

**Verlustrate**

Eng verwoben mit den oben beschriebenen Größen sind die Verlustraten. Es muß, abhängig von dem Punkt an dem die Verluste auftreten, zwischen der Datenverlustrate (LR) und der sog. Call Loss Rate (CLR) unterschieden werden.

**Datenverlustrate (LR)**
Bei dem vorliegenden Knotenmodell und der Zielsetzung dieser Untersuchung treten Datenverluste nur an der Warteschlange auf. Verluste auf Grund von Störungen an den Übertragungseinrichtungen werden vernachlässigt. Bedingt durch die beschränkte Speichertiefe, die sich von der maximalen Zeitverzögerung, die ein Dienst tolerieren kann, ableiten läßt, kann in Hochlastsituationen hier ein Überlauf auftreten. Die Verlustrate ist durch das Verhältnis der Anzahl der verworfenen Daten zu der Gesamtzahl aller Daten, die der Warteschlange übergeben werden, definiert.

**Call Loss Rate (CLR)**
Die Call Loss Rate tritt auf, wenn bei einem Verbindungswunsch, in Abhängigkeit von der Bewertungsstrategie, festgestellt wird, daß die Ressourcen soweit erschöpft sind, daß die Verbindungsparameter, ohne nachhaltige Beeinträchtigung der bereits bestehenden Verbindungen, auf Dauer nicht gewährleistet werden können. Rechnerisch ergibt sich die CLR aus dem Verhältnis der abgelehnten Verbindungen zu der Gesamtzahl aller Verbindungswünsche.

## 2.3 Ziele der dynamisch adaptiven Zugangskontrolle

Bei diesen neuen Verfahren wird die Entscheidung, ob eine Verbindung aufgebaut werden kann, von unterschiedlichen Größen abhängig gemacht. Allgemein läßt sich das Verfahren dann durch die Beziehung 2.6 beschrieben.

$$Decision_{Admission} = f(t, T_U(t), Q_U(t), S_U(t), LR(t), CLR(t)) \tag{2.6}$$

Es werden also Verfahren eruiert, bei denen die Entscheidung, ob eine Verbindung geschaltet wird, von der momentanen Auslastung der Warteschlangen und des Kanals abhängt. Weiterhin werden Strategien vorgestellt, die auf der Auswertung der Verlustrate (LR) und der Call Loss Rate (CLR) beruhen.
Neben diesen mehr an dem System orientierten Ansätzen, die technisch meßbare Verbesserungen erreichen, existieren Methoden, bei denen spezielle Randbedingungen optimiert werden sollen.

### Mehrwertdienste

Es ist denkbar, daß auf Grund wirtschaftlicher Aspekte *weniger profitable* Dienste in einem bestimmten Rahmen abgewiesen werden, um Ressourcen für Verbindungen, die einen höheren Deckungsbeitrag erwirtschaften, vorzuhalten.
Bei dieser Strategie steht neben den technischen Randbedingungen die Gewinnoptimierung bei der Entscheidung über den Verbindungsaufbau im Vordergrund.

### Fairness

Ein weiterer Aspekt, der berücksichtigt werden kann, manifestiert sich in der fairen Behandlung der Verkehrsklassen untereinander. Diese Fairness kann so definiert werden, daß alle Verkehrsklassen gleich behandelt werden, d. h. dieselbe Zugangsblockierung erfahren. In Hochlastsituationen kann wie oben beschrieben aus technischen Gründen nicht jeder Verbindungswunsch realisiert werden. Das bedeutet, daß Dienste mit einer gewissen Wahrscheinlichkeit, die sich aus der Anzahl der Verbindungswünsche, die nicht vermittelt wurden zur Gesamtzahl aller Verbindungswünsche errechnen läßt, abgelehnt werden. Diese sog. CLR ist unter anderem eine Funktion des Angebotes und des Bandbreitenbedarfs, so daß sich anforderungsbedingt drastische Unterschiede bei den korrespondierenden Verlustraten ergeben können.
Fairness bedeutet in diesem Fall, daß die Entscheidungsstrategie eine Egalisierung der Verlustraten zwischen den einzelnen Diensten bewirkt. Ist die Verbindung allerdings etabliert, erfolgt durch die CAC, die einen open-loop-Algorithmus darstellt, keine Überwachung der vereinbarten Parameter. Die Überprüfung des tatsächlichen Verkehrsprofils mit dem vereinbarten erfolgt durch die sog. Policing Controller.

## 2.3.1 Policing Control

Diese sog. *Policing-Verfahren* überwachen den Nutzerzugang auf der Basis des genannten *Verkehrsvertrags*, der zwischen dem Benutzer und dem Netzmanagement beim Aufbau der Verbindung vereinbart wird. In Fällen, in denen diese Parameter verletzt werden, z. B. auf Grund von Fehlfunktionen der Netzwerkeinrichtungen oder mißbräuchlichem Verhalten, müssen die Netzressourcen bestehender Verbindungen geschützt werden. Die UPC [7] oder auch *Traffic Policing* fassen alle Aktionen, die zur Überwachung und Steuerung des Verkehrs am Netzzugangspunkt notwendig sind, zusammen. Pakete, die den Verkehrsvertrag verletzen, können absorbiert oder gespeichert und dann zu einem späteren Zeitpunkt - vertragskonform - übertragen werden. Weiterhin ist es möglich, diese Verbindungen bei der Abrechnung mit Aufschlägen zu versehen. Weitere Maßnahmen sind denkbar.
In der Literatur ist eine Unzahl von Verfahren entworfen und eruiert worden [17, 62, 81, 21]. Es zeichnet sich ab, daß die Policing Algorithmen im Wesentlichen folgende Anforderungen erfüllen müssen.

[7] Usage Parameter Control

- Fehlerzustände müssen schnell und sicher erkannt und lokalisiert werden. Die Kontrollreaktionen sollten innerhalb sog. kürzester Latenzzeiten *(Echtzeit)* eingeleitet werden, um zu verhindern, daß Speicher überlaufen oder knappe Ressourcen durch den nicht vereinbarten Verkehrsstrom belegt werden.
- Die Verfahren sollten einfach und leicht zu implementieren sein.
- Verbindungen, die konform zum abgeschlossenen Verkehrsvertrag arbeiten, dürfen durch die Kontrollaktionen nicht negativ beeinflußt werden.
- Da die Verkehrscharkteristiken von den Diensten und diese wiederum von den lokalen Nutzern abhängen, sollten die eingesetzten Systeme adaptiv sein.
- Die Verfahren sollten die Fähigkeit besitzen, sich auf neue Verkehrsarten einzustellen.
- Wenn genügend freie Ressourcen vorhanden sind und keine bestehenden, vertragskonformen Verbindungen beeinflußt werden, sollten diese Verbindungen weiterhin übermittelt und die gemachten Vereinbarungen erweitert werden.

Der bekannteste Mechanismus zur Überwachung der Paketübertragungerate ist das sog. Leaky-Bucket Verfahren.

### Leaky Bucket

Die Idee dieses Verfahrens beruht darauf, den Datenstrom in einer Warteschlange zu verwalten, die mit einer konstanten Datenrate abgearbeitet wird. Werden zu viele Daten angeliefert, läuft die Warteschlange über. Die Pakete werden verworfen.
Bei Anwendung des Leaky-Bucket-Algorithmus werden keine Pakete, die konform zum Verkehrsvertrag sind, verworfen. Dieses Verfahren kommt im B-ISDN zum Einsatz und ist in der Recommendation I.371 [32] fixiert.
Vom ATM-Forum wird eine Variante des Leaky-Bucket-Verfahrens eingeführt [3]. Der *Generic Cell Rate Algorithm* [8] (Abbildung 2.7) oder der *Continuous-State Leaky Bucket Algorithm* orientieren sich nicht an der Paketrate sondern an den Ankunftszeitpunkten der Pakete. Der Datenstrom ist nach diesem Verfahren konform, wenn die Variable X' den Grenzwert L nicht überschreitet. Initialisiert werden die Variablen beim Empfang des ersten Paketes $t_a$ einer Verbindung mit $X = 0$ und $LCT = t_a(1)$, so daß X' = 0 gilt. Im Folgenden ändert sich X' bei jeder Ankunft eines Paketes um den Betrag $I - \Delta$. Das Intervall I ist eine dienstspezifische Größe mit der Abweichungen von der vereinbarten Übertragungsrate berücksichtigt werden. $\Delta$ ist die Zeit zwischen zwei aufeinanderfolgenden Paketankünften. Ist X'(t) $<$ 0, so sendet der Dienst mit einer Transferrate, die kleiner als die vereinbarte ist. Das Paket ist somit konform zum Vertrag. Wird die Übertragungsrate überschritten (X'(t)$\geq$ 0), muß in einem weiteren Schritt sichergestellt werden, daß der Betrag der Überschreitung nicht größer als ein vorgegebener Grenzwert ($L$) ist. Bei Überschreitung von L ist das Paket nicht konform zum Verkehrsvertrag und wird verworfen.

[8] GCRA

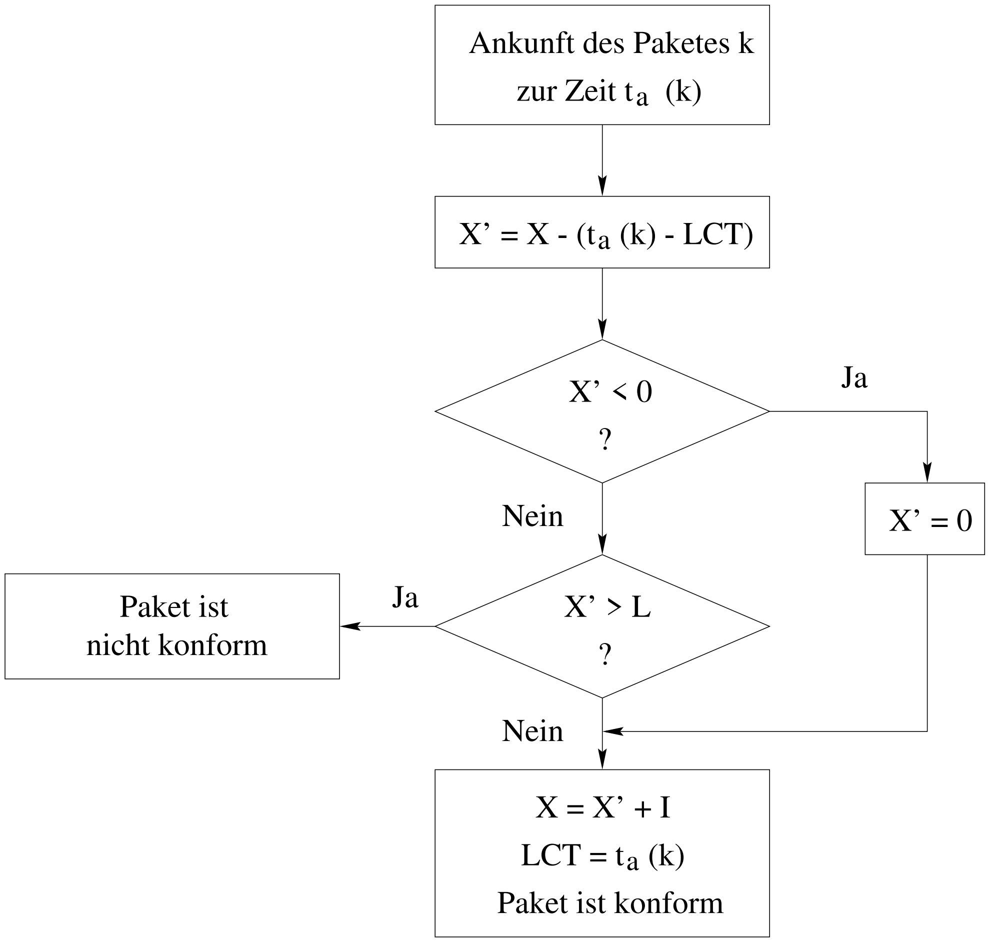



Abbildung 2.7: Generic Cell Rate Algorithm

Dieses Verfahren wurde für die Simulation implementiert. Es dient als Referenz, um die Leistungsfähigkeit der im Folgenden vorgestellten, auf Fuzzy-Logic und Neuronalen Netzen basierten Mechanismen, abschätzen und vergleichen zu können.

## 2.4 Quellenmodelle

Um die Leistungsfähigkeit und die Qualität von Netzen abschätzen zu können, müssen die besonderen Kennzeichen der unterschiedlichen Dienste, die vermittelt werden sollen, sowie deren Anforderungen an das Übertragungssystem bekannt sein. Da es nicht sinnvoll erscheint, für jeden dieser Dienste ein eigenes Modell zu entwickeln, wird durch eine geeignete Zusammenfassung und Reduktion der charakteristischen Parameter eine Einteilung in zwei Verkehrstypen vorgenommen. Zur qualitativen Beschreibung der oben genannten Dienste wurde grob zwischen sog. Diensten mit variabler bzw. konstanter Bitrate differenziert.

- VBR-Dienste (Variable Bit Rate)
  Diese Dienste arbeiten mit einer variablen Bitrate, die sich entweder durch eine Variation der Bitraten im Datenfluß während einer Verbindung auszeichnet, oder dadurch, daß sich während der Dauer einer Verbindung Aktivitäts- und Ruhephasen oder Zeitabschnitte mit einer verringerten Aktivität abwechseln. Der Burstfaktor ist immer größer als 1.

- CBR Quellen (Constant Bit Rate)
  Dienste dieser Verkehrskategorie zeichnen sich dadurch aus, daß sie während ihrer gesamten Verbindungszeit permanent über eine feste Übertragungsbandbreite verfügen können. Diese Bandbreite entspricht der maximalen Nachrichtenübertragungsrate. Die Burstiness dieser Quellen ist eins. Selbst Pausen werden mit der maximalen Bandbreite „übertragen“ .

Zur Nachbildung der unterschiedlichen Verkehrscharakteristika zum Einsatz in Simulationen existieren in der Literatur viele unterschiedliche Ansätze. In dem folgenden Abschnitt soll das dieser Untersuchung zugrunde liegende Modell zur Erzeugung einer künstlichen Verkehrslast dargestellt werden.

### 2.4.1 Modelle

Auf Grund der Integration der unterschiedlichsten Dienste (Daten, Video, Audio, Multimedia) in ein *gemeinsames* Kommunikationssystem spielt die Nachbildung des Lastverhaltens einer oder mehrerer Quellen in Bezug auf die Sicherung der Dienstqualität, der Dimensionierung und Optimierung des *gesamten* Übertragungssystems als auch zur Validierung von Zugriffs-, Kontroll- und z. B. Routingalgorithmen eine essentielle Rolle. Zur simulativen Untersuchung dieser vielfältigen Aufgaben müssen die realen Quellen in ihrem Verhalten nachgebildet werden. Die Modelle sollten sehr flexibel sein und auf definierten stochastischen Prozessen beruhen, um reproduzierbare Ergebnisse zu erzeugen. Es sollte möglich sein, sowohl zufälligen Verkehr als auch gezielte Lastmuster von realen Quellen nachzubilden. Für den letzteren Fall müssen genügend Parameter zur Verfügung stehen, um die Belange des echten Verkehrsmusters nachbilden zu können. Dabei ist es relevant, daß das Verkehrsmodell folgende Randbedingungen erfüllt:

- Simulation einer realistischen Verkehrslast von einer großen Anzahl von Quellen, die dann zu einer Verkehrsklasse zusammengefaßt werden.
- Mit Hilfe des Quellenmodells sollte es möglich sein, sowohl die kurzzeitigen als auch die mittel- und langfristigen Eigenschaften des Verkehrs von realen Quellen reproduzieren zu können.
- Der Lastgenerator muß in der Lage sein, ein breites Spektrum unterschiedlicher Verkehrsmuster und einen Mix zu erzeugen.
- Bei vielen Diensten besteht eine Korrelation zwischen den übertragenen Informationseinheiten, so daß es unumgänglich ist, diese Beziehungen nachzubilden.
- Es müssen genügend Parameter zur Verfügung stehen, um eine Übereinstimmung zwischen künstlicher und realer Verkehrslast zu erzielen.

Die Forschung auf diesem Gebiet ist wegen der Vielfalt und Variabilität des Dienstverhaltens noch nicht abgeschlossen. [25, 46, 49].

### Quellenmodellierung

Zur Beschreibung des Verhaltens eines Dienstes und zur Modellierung der Datenströme einer Verkehrsquelle wird ein *hierarchisch* aufgebautes Quellenmodell eingesetzt. Mit diesem System ist es dann möglich, die unterschiedlichen Kommunikationsphasen nachzubilden. So können sowohl der Beginn und das Ende einer Verbindung als auch das statistische Verhalten eines Dienstes und einer unterschiedlichen Anzahl von realen Quellen reproduziert werden. Bei dem gewählten Ansatz wird das Kommunikationsverhalten eines Systems in *drei* funktionale Ebenen differenziert (Abbildung 2.8).

**Verbindungsebene**
Die Verbindungsebene stellt die oberste Schicht des in Abbildung 2.8 gezeigten Quellenmodells dar. Sie beschreibt die zeitlichen Zusammenhänge zwischen dem Auf- und Abbau einer Verbindung, der Belegungsdauer, sowie die Zeitintervalle zwischen aufeinanderfolgenden Verbindungswünschen für die einzelnen Verkehrsklassen. Die Funktionalität dieser Ebene kann durch vier charakteristische Zeiten bzw. Zeitpunkte beschrieben werden.

- Relevant ist der Zeitpunkt, wann eine Verbindung aufgebaut wurde.
- Die Verbindungsdauer oder auch Belegungsdauer beschreibt den Zeitraum, über den eine Verbindung besteht und Informationen zwischen den Teilnehmern ausgetauscht werden können.
- Der Zeitpunkt an dem die Verbindung abgebaut ist.
- Der Zeitraum zwischen zwei Verbindungsaufbauwünschen *einer* Verkehrsklasse

Der Ankunftsprozeß wird durch einen Poisson Prozeß mit exponentieller Verteilung der Zeitintervalle charakterisiert.

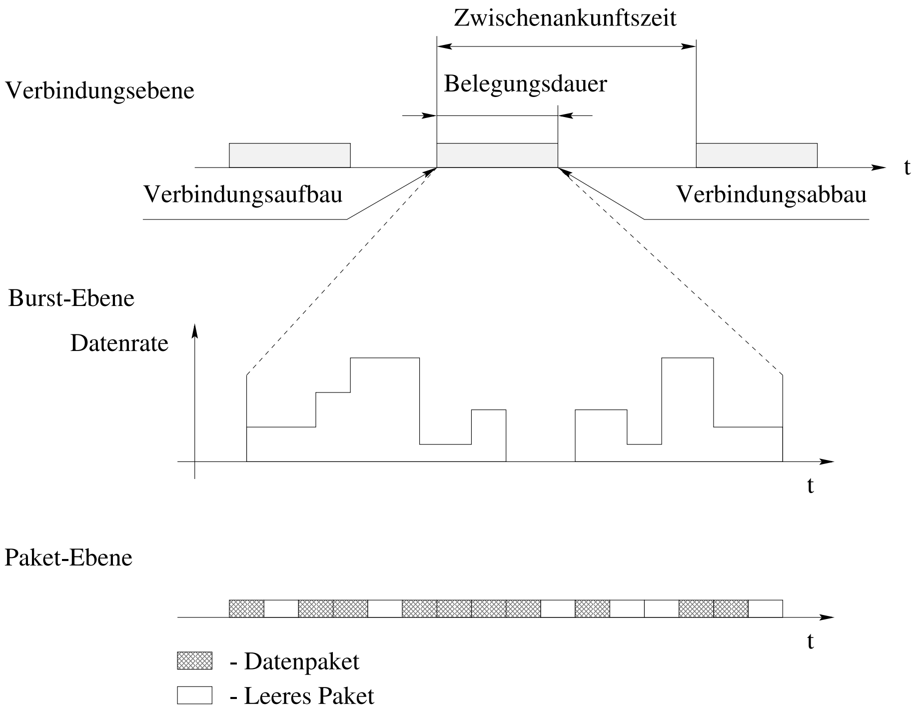



Abbildung 2.8: Hierarchisches Modell einer Verkehrsquelle

**Burst-Ebene**
Während einer Verbindung können dann, in Abhängigkeit von der Verkehrsklasse, Phasen auftreten, in denen die Quellen unterschiedlich aktiv. Das in Abbildung 2.8 dargestellte Lastverhalten auf der Burstebene zeigt, daß sich Intervalle in denen Daten mit unterschiedlicher Intensität erzeugt werden, abwechseln. Diese sogenannten Bursts bestehen aus einer geometrisch verteilten Anzahl von Datenpaketen mit einer konstanten Zwischenankunftszeit.

**Paket-Ebene**
Auf dieser Ebene wird der Ankunftsprozeß der Dateneinheiten in Abhängigkeit von der eingeprägten Bitrate und der Paketlänge nachgebildet.
Zur Verdeutlichung der Zusammenhänge zeigt Abbildung 2.9 eine detailliertere Auflösung der Paketebene. Erkennbar sind zwei Bursts der mittleren Länge $TB_i$. Diese Bursts wiederum sind in konstante Intervalle $\Delta T_i$ aufgeteilt. Die Zeitspanne $\Delta T_i$ berechnet sich aus der Übertragungszeit eines Paketes zuzüglich einer übertragungsbedingten Verzögerungszeit. $\Delta T_i$ ist nur abhängig von der Paketgröße und der Datentransferrate des Übertragungskanals. Sie ist konstant für das betrachtete System und unabhängig vom jeweiligen Zustand einer Quelle.

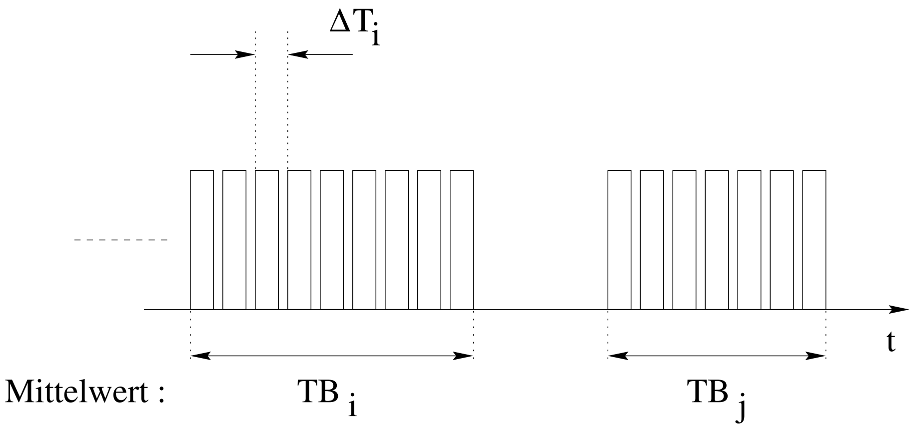



Abbildung 2.9: Zell-Ebene

**Zustandsdiagramm des Quellenmodells**

Zur Beschreibung der Zusammenhänge zwischen den Kommunikationsebenen und zur Nachbildung der charakteristischen Eigenschaften von einzelnen Verkehrsquellen und Gruppen existieren unterschiedliche mathematische Verfahren und Methoden. Eine mögliche Differenzierung der Ansätze kann an Hand des Zeitverhaltens erfolgen.

Es existieren hier *zeitkontinuierliche* Verfahren, wie die *M*arkoff *M*odulierten *P*oisson *P*rozesse (MMPP) und *zeitdiskrete, deterministische* Methoden, deren Ankunftsprozesse durch eine beliebige Verteilungsfunktion charakterisiert werden können und deren Abarbeitungsstrategie deterministisch ist [2, 52, 57, 64]. Diese Prozeßklasse wird unter dem allgemeinen Begriff *G*eneral *M*odulated *D*eterministic *P*rocess (GMDP) zusammengefaßt. Die Stärken beider Ansätze liegen darin, daß zum einen der bursthafte Charakter der Quellen erfaßt, zum anderen die Korrelation zwischen aufeinanderfolgenden Informationseinheiten, wie z. B. bei Videodiensten, beschrieben werden kann [46].

Bei dem hier gewählten Verfahren wird das statistische Verhalten der hierarchischen Verkehrsquelle nach Abbildung 2.8 mit Hilfe von *M*arkoff *M*odulated *D*eterministic *P*rocesses (MMDP), die eine Unterklasse der GMDP darstellen, nachgebildet. Der MMDP ist ein stochastischer Prozeß, bei dem die Zellrate, mit der eine Quelle arbeitet, durch die verschiedenen Zustände einer zeitdiskreten irreduziblen Markoff-Kette vorgegeben wird. Die Umsetzung erfolgt mit Hilfe eines endlichen Automaten mit N diskreten Zuständen[9], die direkt mit den Aktivitäten einer Quelle assoziiert sind. In jedem dieser Stati generiert die Quelle Datenpakete mit einer deterministischen Zwischenankunftszeit $\Delta T_i$ (Abbildung 2.9), so daß die maximale Zellrate in einem Status mit

$$CR_i = \frac{1}{\Delta T_i} \tag{2.7}$$

gegeben ist. Der Index i reflektiert dabei die Statusnummer. Die Anzahl $X_i$ der Informationseinheiten, die in einem Zustand erzeugt werden, ist *geometrisch* verteilt. Die Wahr-

[9]Im Folgenden werden die Begriffe Zustand und Status synomym benutzt

scheinlichkeit, daß $X_i = n$, wird durch die Verteilungsfunktion 2.8 beschrieben.

$$P(X_i = n) = \begin{cases} p(1-p)^{n-1} & : \quad f\ddot{u}r\ n = 1, 2, ...; 0 < p < 1 \\ 0, & : \quad sonst \end{cases} \tag{2.8}$$

Die mittlere Anzahl Pakete, die während eines Bursts übertragen werden, kann nach [2] mit Hilfe der Beziehung 2.9 berechnet werden .

$$X_i = E[X_i^n] = \frac{1}{1-p} \tag{2.9}$$

Neben diesem Zusammenhang gilt auch weiterhin, daß sich die durchschnittliche Anzahl der übertragenen Pakete aus dem Quotienten der Burstlänge $TB_i$ und der mittleren Übertragungszeit $\Delta T_i$ berechnen läßt.

$$X_i = \frac{TB_i}{\Delta T_i} \tag{2.10}$$

Der Parameter p kann dann wie folgt bestimmt werden.

$$p = 1 - \frac{1}{X_i} = 1 - \frac{\Delta T_i}{TB_i} \tag{2.11}$$

Die Erzeugung geometrisch verteilter Zufallszahlen erfolgt mittels Gleichung 2.12 aus einer gleichverteilten Zufallszahl U über die inverse Funktion.

$$X_i = \left\lfloor \frac{ln\ U}{ln\ (1-p)} \right\rfloor + 1 = \left\lfloor \frac{ln\ U}{ln\ \frac{\Delta T_i}{TB_i}} \right\rfloor + 1 \tag{2.12}$$

Die zustandsspezifischen Zellraten werden in einer Diagonalmatrix

$$\Lambda = diag[\lambda_1, \lambda_2, ..., \lambda_N] \tag{2.13}$$

zusammengefaßt.
Die Statusübergangswahrscheinlichkeiten des Automaten werden durch eine quadratische Matrix $\Omega$ der Ordnung N beschrieben.

$$\Omega = \begin{bmatrix} 0 & \omega_{12} & \cdots & \omega_{1N} \\ \omega_{21} & 0 & & \vdots \\ \vdots & & \ddots & \vdots \\ \omega_{N1} & \cdots & \cdots & 0 \end{bmatrix} \tag{2.14}$$

Die Elemente $\omega_{ij}$ geben die Wahrscheinlichkeit an, daß nach Ablauf der Aufenthaltszeit in einem Status $i$ ein Wechsel in den Zustand $j$, mit $i \neq j$ erfolgt. Die Übergangswahrscheinlichkeit $\omega_{ii}$ mit $i \in \{1, 2, ..., N\}$, die angibt, daß der Zustand nicht verlassen wird, ist 0. Ein Übergang in einen anderen Zustand ist obligat.
Um die Komplexität des Modells zu reduzieren, erfolgt für die weiteren Betrachtungen eine

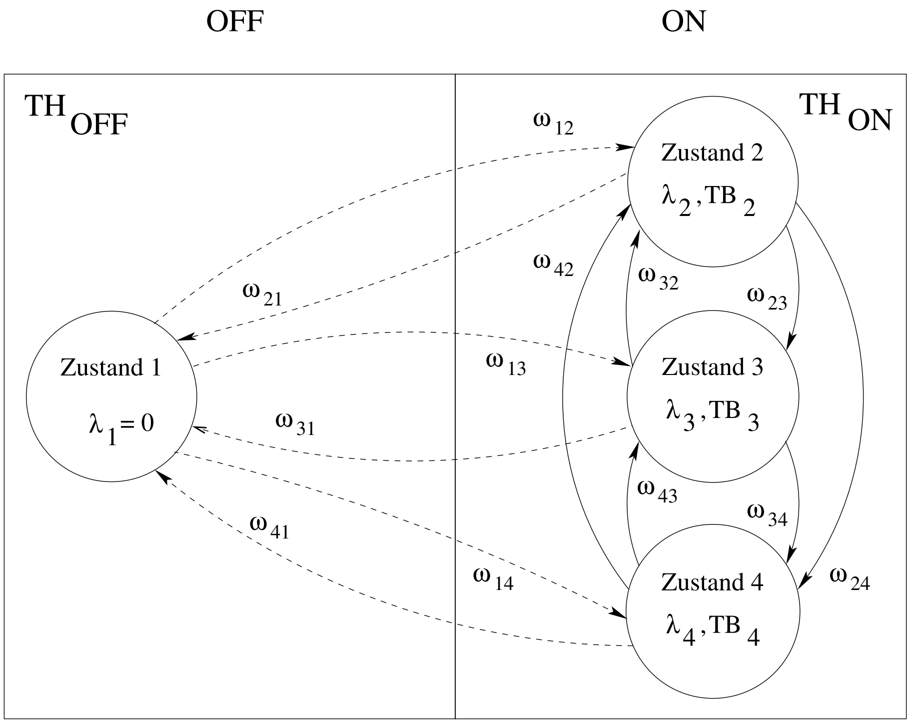



Abbildung 2.10: Allgemeines Zustandsmodell einer Verkehrsquelle

Reduktion des Systems auf einen endlichen Zustandsraum mit maximal *vier* Zuständen. Das resultierende Modell ist in Abbildung 2.10 dargestellt.

Im Wesentlichen ist dieses Modell in die zwei Bereiche „OFF" und „ON" gegliedert, die das Verhalten des Dienstes auf der *Verbindungsebene* reflektieren. Kennzeichnend für den Zeitraum in dem die Quelle inaktiv ist, ist der Zustand 1. Die aktive Phase in der die Quelle mit unterschiedlicher Intensität Daten erzeugt wird durch die Zustände i = 2, 3 und 4 beschrieben. In der Ruhephase ist der korrespondierende Dienst nicht aktiviert, es besteht also keine *aktive* Verbindung zwischen Teilnehmern. Während der Aktiv-Phase werden dann, entsprechend der Regeln der unterschiedlichen Stati, Informationseinheiten erzeugt. Die Unterteilung in diese beiden Bereiche erlaubt es also, die *Verbindungsebene* zu berücksichtigen. Der Ankunftsprozeß einer neuen Verbindung wird durch einen Poisson Prozeß, bei dem die Zwischenankunftszeiten exponentiell mit einer dienstspezifischen Ankunftrate $\lambda_S$ (Gl. 2.15) verteilt sind, nachgebildet.

$$P\{T_A \leq t\} = 1 - e^{-\lambda_S t} \tag{2.15}$$

Die mittlere Länge dieser Zeitabschnitte wird durch die Verweilzeiten (Holdingtimes) $TH_{ON}$ und $TH_{OFF}$ beschrieben.

Die *Burstebene* wird durch die Zustände 2, 3 und 4 sowie die Zustandsübergänge innerhalb des ON-Bereichs realisiert. Wie in Abbildung 2.10 dargestellt, werden in den einzelnen Zuständen, während der Burstdauer $TB_i$, die Datenpakete mit unterschiedlichen Bitraten

$\lambda_i$ erzeugt. Befindet sich das System in dem Zustand $i$, gibt $\omega_{ij}$ die Übergangswahrscheinlichkeit vom Zustand $i$ in den Zustand $j$ unter der Bedingung, daß $i \neq j$, an. Ein Wechsel in einen anderen Zustand nach Ablauf der Burstlänge ist obligat. Daraus resultiert, daß $\omega_{ii} = 0$.

Die konkrete Basis zur exakten Beschreibung dieser Prozesse bilden gemäß der Theorie der GMDP die im Folgenden beschriebenen Matrizen $\Omega$ (2.16) und $\Lambda$ (2.17).

$$\Omega = \begin{bmatrix} 0 & \omega_{12} & \omega_{13} & \omega_{14} \\ \omega_{21} & 0 & \omega_{23} & \omega_{24} \\ \omega_{31} & \omega_{32} & 0 & \omega_{34} \\ \omega_{41} & \omega_{42} & \omega_{43} & 0 \end{bmatrix} \tag{2.16}$$

$\Omega = \{\omega_{ij}\}$ ist eine quadratische Matrix vierter Ordnung, die die Übergänge und die korrespondierenden Übergangswahrscheinlichkeiten zwischen den Phasen $\{1, 2, 3, 4\}$ beschreibt. Die Matrix $\Lambda$, eine Diagonalmatrix der Ordnung 4, gibt die Datenraten an, mit denen die Quellen in den unterschiedlichen Phasen arbeiten. Vereinbarungsgemäß stellt der Zustand 1 die OFF-Phase dar, so daß $\lambda_1 = 0$.

$$\Lambda = diag[0, \lambda_2, \lambda_3, \lambda_4] \tag{2.17}$$

Neben der Anpassung der Verkehrslast mit Hilfe der oben beschriebenen Parameter bietet die Anzahl der signifikanten Phasen einen weiteren Freiheitsgrad bei der Nachbildung der charakteristischen Eigenschaften von Verkehrsquellen. Eine konventionelle Sprachquelle kann durch *eine* konstante Übertragungsrate und eine exponentiell verteilte Belegungsdauer charakterisiert werden. Das heißt, daß zur Berschreibung dieser Verkehrslast ein Modell, das aus zwei Phasen (ON - OFF) besteht, ausreichend ist. Bei komplexeren Systemen, bei denen der Verkehr burstbehaftet ist, muß die Anzahl der Stati erhöht werden. Bekanntlich setzen sich Sprachsignale aus alternierenden Aktivitäts- und Ruheintervallen zusammen. Um die Übertragungseffizienz zu steigern, werden deshalb Sprachpausen unterdrückt, also nicht übertragen. Das Lastverhalten kann dann durch ein *dreiphasiges* Modell beschrieben werden. Für die Realisierung von Videoquellen, die einen extrem burstbehafteten Charakter aufweisen oder bei denen eine Korrelation zwischen aufeinanderfolgenden Informationseinheiten besteht, kann das 4-Phasen Modell zur Nachbildung herangezogen werden.

### CBR Quelle

Mit Hilfe des allgemeinen Zustandsmodells einer Verkehrsquelle, wie sie in Abbildung 2.10 gezeigt wird, können unterschiedliche Verkehrsarten und Verhaltensmuster eingeprägt werden. Durch die Reduzierung des Systems auf *zwei* Phasen, ergibt sich das in Abbildung 2.11 dargestellte ON/OFF Modell, mit dem das Verhalten einer CBR - Quelle auf Verbindungs- und Zellebene nachgebildet werden kann. Die Aktivitäten der Quelle werden in Bezug auf die Verbindungsdauer durch die Bereiche OFF und ON beschrieben. In der ON(Verbindungs)- Phase wird ein kontinuierlicher Strom von Datenpaketen erzeugt. Während der OFF-Periode ist keine Verbindung aufgebaut, so daß keine Daten übertragen werden. Die Länge der inaktiven Phase ist exponentiell verteilt. Bei der aktiven Periode

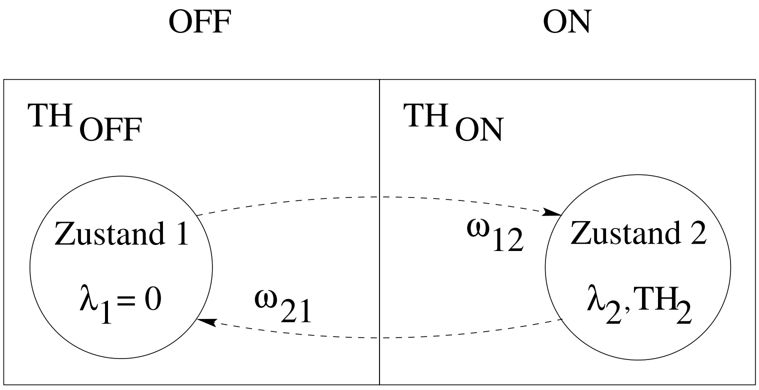



Abbildung 2.11: Modell einer CBR - Quelle

einer Quelle, liegt an dieser Stelle eine geometrische Verteilung vor, da neue Nachrichtenpakete nach Ablauf von festen Zeitintervallen $\Delta T_i$ erzeugt werden, die Zeit also nicht kontinuierlich sondern diskret ist. Die Übergangswahrscheinlichkeiten werden in diesem Fall durch die Matrix 2.18 gegeben.

$$\Omega = \begin{bmatrix} \omega_{11} & \omega_{12} \\ \omega_{21} & \omega_{22} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{2.18}$$

Nach Ablauf der Verweilzeit in den Phasen 1 bzw. 2 ist der Übergang in den komplementären Zustand obligatorisch. Die entsprechenden Übergangswahrscheinlichkeiten haben den Wert 1. Die Matrix 2.19 beschreibt die Bitraten in den Zuständen 1 und 2.

$$\Lambda = \begin{bmatrix} 0 & 0 \\ 0 & \lambda_2 \end{bmatrix} \tag{2.19}$$

Da im inaktiven Zustand keine Daten erzeugt werden, also $\lambda_1 = 0$, degeneriert das ursprüngliche MMDP System hier zu einem sogenannten Interrupted Deterministic Process (IDP). Die in der Aktiv-Phase möglichen Bitraten, mit denen die Quellen senden können, sind der Tabelle 2.2 zu entnehmen [46].

| Dienst | $\lambda_2$ |
|---|---|
| Telefonie | 64 kbit/s |
| Viedeotelefonie (geringe Qualität) | 128 kbit/s |
| Videokonferenz | 2 Mbit/s |
| konventionelle Videocodecs | 34 Mbit/s |
| Standard TV | 140 Mbit/s |

Tabelle 2.2: CBR Übertragungsraten

**VBR**

Durch die Reduzierung des Systems auf eine Konstellation mit *drei* Zuständen ergibt sich das in Abbildung 2.12 dargestellte ON/OFF Modell, mit dem das Verhalten einer VBR-Quelle auf Verbindungs-, Burst- und Zellebene nachgebildet werden kann. Wie bei der

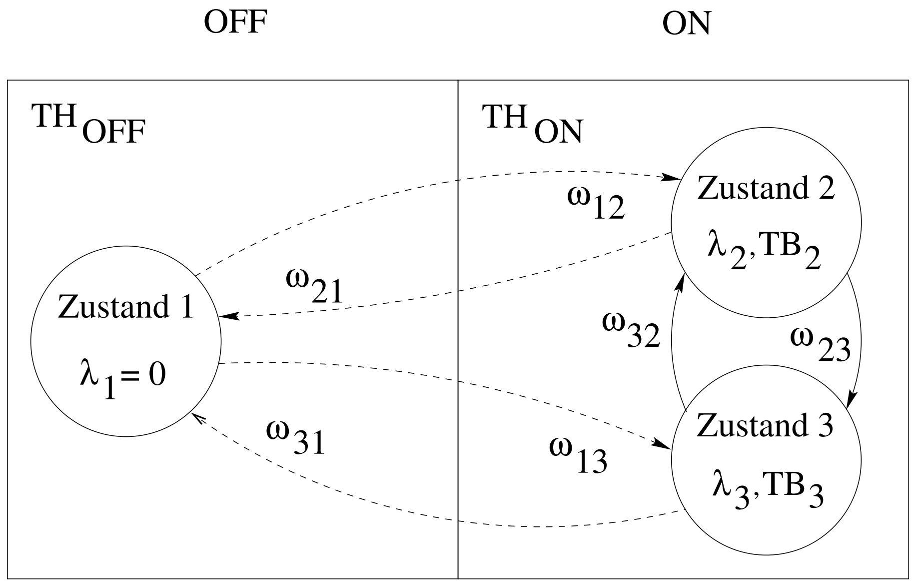



Abbildung 2.12: Modell einer VBR - Quelle mit 3 Zuständen

CBR-Quelle wird das Verhalten auf der Verbindungsebene durch die Parameter der ON- und OFF-Bereiche festgelegt. Die Übergangswahrscheinlichkeiten $\omega_{ij}$ lassen sich mit den Zusammenhängen 2.20 beschreiben.

$$\begin{aligned}
\omega_{11} &= \omega_{22} = \omega_{33} = 0 \\
\omega_{12} &= \frac{TH_2}{TH_2 + TH_3} \\
\omega_{13} &= \frac{TH_3}{TH_2 + TH_3} \\
\omega_{21} &= \omega_{31} = \frac{TH_{OFF}}{TH_{ON} + TH_{OFF}} \\
\omega_{23} &= \omega_{32} = \frac{TH_{ON}}{TH_{ON} + TH_{OFF}}
\end{aligned} \tag{2.20}$$

Es besteht im Wesentlichen eine Abhängigkeit von den Statushaltezeiten. Matrix 2.21 definiert die Bitraten in den einzelnen Stati.

$$\Lambda = diag[0, \lambda_2, \lambda_3] \tag{2.21}$$

Diese Quelle generiert eine Last mit zwei unterschiedlichen Intensitäten.

**Multimediale Verkehrslast**

Eine Konstellation mit *drei* aktiven Zuständen, wie sie Abbildung 2.10 dargestellt ist, dient dazu, das Verhalten komplexerer Datenströme nachzubilden. Weiterhin wird auch hier zwischen einem inaktiven und einer aktivem Bereich, der sich in drei Zustände mit unterschiedlichen Profilen zeigt, differenziert. Mit diesem Quellentyp kann eine stark burstbehaftete Verkehrslast eingeprägt werden, mit der es möglich ist, multimediale Dienste zu simulieren.

$$\omega_{11} = \omega_{22} = \omega_{33} = \omega_{44} = 0 \tag{2.22}$$

$$\omega_{1j} = \frac{TH_j}{\sum\limits_{n=2}^{k} TH_n} \tag{2.23}$$

$$omega_{21} = \omega_{31} = \omega_{41} = \frac{TH_{OFF}}{TH_{ON} + TH_{OFF}} \tag{2.24}$$

Für die übrigen Elemente gilt

$$\omega_{ij} = \frac{TH_{ON}}{TH_{ON} + TH_{OFF}} \frac{TH_j}{\sum\limits_{n=2, n\neq i}^{k} TH_n} \; mit \; i,j \in \{2,...,k\} \; und \; k = 4 \tag{2.25}$$

Die Bitraten in den einzelnen Zuständen werden durch die Matrix $\Lambda$ beschrieben.

$$\Lambda = diag[0, \lambda_2, \lambda_3, \lambda_4] \tag{2.26}$$

**Beispiel**

Zur Verdeutlichung des Quellenmodells und der verschiedenen Parameter soll das Lastmuster VBR-Quelle mit 4 Zuständen nachgebildet werden. Die in der Tabelle 2.3 dargestellten Parameter sind in Anlehnung an [46] charakteristisch für Dienste, die den interaktiven Datenaustausch und LAN-Verbindungen unterstützen. Daraus ergibt sich dann mit den

| Parameter | Status | | | |
|---|---|---|---|---|
| | 1 | 2 | 3 | 4 |
| mittlere Verweilzeit [$s$] | 60 | 0.14 | 0.04 | 0.02 |
| Bitrate $\lambda_i$ [bit/s] | 0 | $1 \cdot 10^6$ | $2 \cdot 10^6$ | $1 \cdot 10^7$ |
| $TH_{ON}$ [s] | 240 | | | |

Tabelle 2.3: Paramter der 4-phasigen VBR-Quelle

Gleichungen 2.22 bis 2.25 die Matrix $\Omega$

$$\Omega = \begin{bmatrix} 0 & 0.7 & 0.2 & 0.1 \\ 0.2 & 0 & 0.54 & 0.26 \\ 0.2 & 0.7 & 0 & 0.1 \\ 0.2 & 0.63 & 0.17 & 0 \end{bmatrix} \tag{2.27}$$

und die Matrix $\Lambda$

$$\Lambda = diag[0, 1 \cdot 10^6, 2 \cdot 10^6, 1 \cdot 10^7] \tag{2.28}$$

Abbildung 2.13 zeigt einen Ausschnitt aus dem simulativ ermittelten Lastmuster einer VBR-Quelle unter Verwendung der obigen Parameter. Im Wesentlichen sind die vier Zustände mit den unterschiedlichen Bitraten $\lambda_1, ..., \lambda_4$ sowie die abgestuften Verweilzeiten in den einzelnen Phasen zu erkennen.

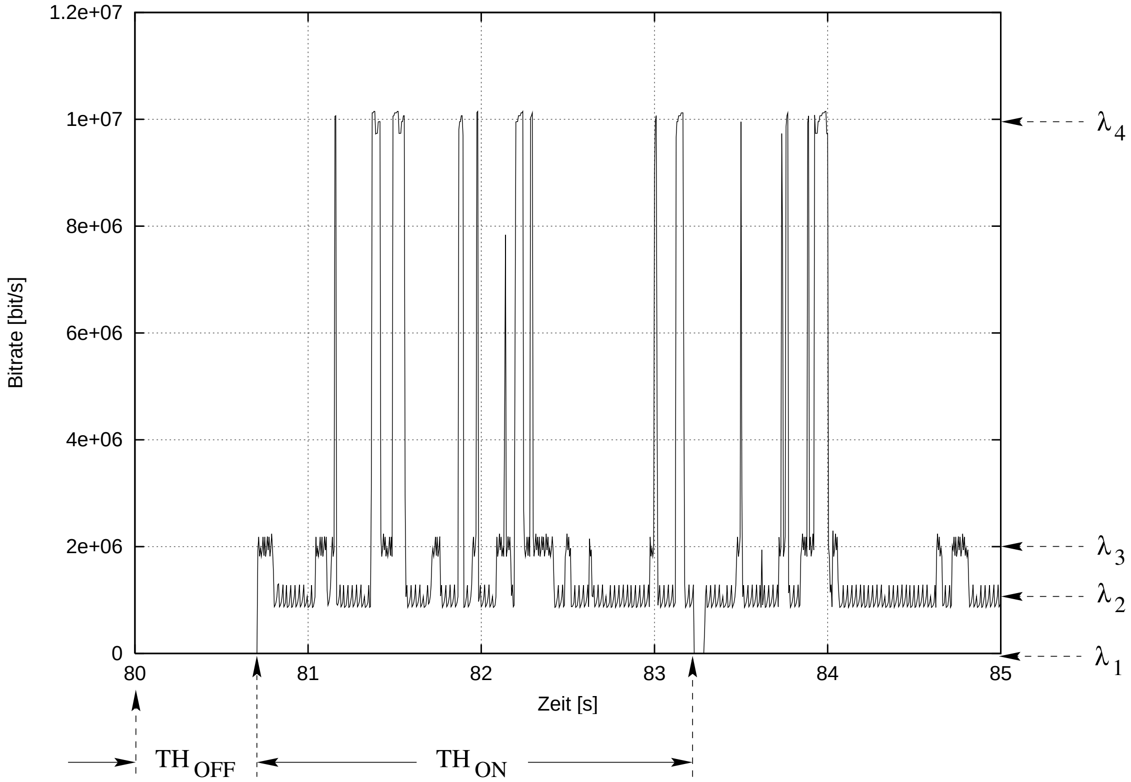



Abbildung 2.13: Beispiel eines simulativ erzeugten Verkehrsmusters

# Kapitel 3

# Kommunikationsknoten

Um die in dem Kapitel 2.2 detailliert beschriebenen Verkehrsparameter und Gütekriterien, die die Zuweisung der Ressourcen und die Behandlung der Verbindung im Wesentlichen beeinflussen, zu berücksichtigen und deren Einfluß auf andere Dienste und den gesamten Durchsatz in einem Zugangsknoten aufzeigen zu können, wurde ein Modell eines neuen Kommunikationsknotens mit der in Abbildung 3.1 dargestellten Struktur entwickelt. Die Abbildung zeigt den grobgranularen Aufbau einer Zugangseinheit. Eingangsseitig sind, we-

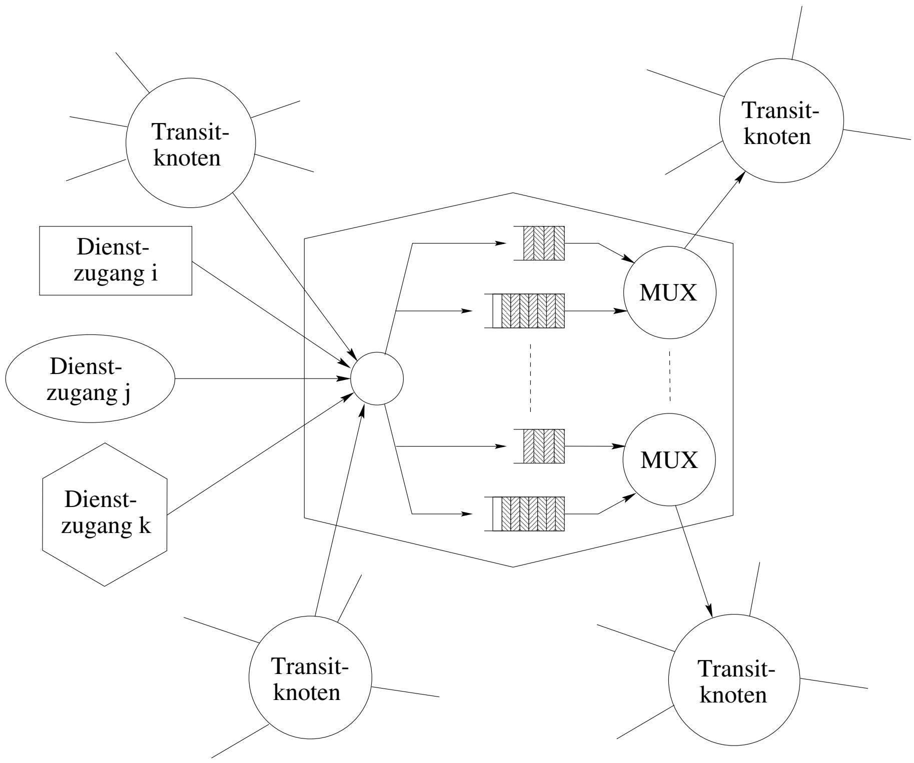



Abbildung 3.1: Aufbau eines Kommunikationsknotens

gen der Einbettung des Knotens in ein multimediales Umfeld, diverse Dienste, die sich in ihren Verkehrsparametern sowie QoS-Anforderungen stark unterscheiden können, angedeutet. Die Palette reicht, wie in Kapitel 2.1 beschrieben, von konventionellen isochronen Gesprächsverbindungen bis hin zu stark burstbehafteten und verbindungslosen Datenübertragungsdiensten. Am Ausgang des Knotens können weitere Transiteinheiten - zum Aufbau eines komplexen Netzes - aber auch Endteilnehmer - angeschlossen sein. Die grundlegende Struktur reflektiert schon im Vorfeld, daß das Systemkonzept für eine Vielzahl unterschiedlicher Teilnehmer, die bedient, und Dienste, die hier abgewickelt werden können, ausgelegt ist.
Der interne Aufbau wird, wie eine detaillierte Auflösung des Aufbaus des Kommunikationsknotens nach Abbildung 3.2 zeigt, im Wesentlichen durch drei funktionale Einheiten dominiert:

- Admission Controller
  Der Admission Controller (Abschnitt 3.1) entscheidet darüber, ob bei Bedarf eine neue Verbindung aufgebaut werden kann. Die Entscheidung basiert auf Zustandsgrößen, die die Auslastung des Netzes reflektieren, und den dienstspezifischen Übertragungsanforderungen. Kommt eine Verbindung zustande, werden diese Anforderungen in dem sog. Verkehrsvertrag vereinbart.

- Link Control Unit
  Die Link Control Unit (Abschnitt 3.2) überwacht die Datenströme einer Verbindung auf Grund der im Verkehrsvertrag vereinbarten Qualitäts- und Übertragungsparameter.

- Allocation Controller
  Der Allocation Controller (Abschnitt 3.3) ist zuständig für die Abschätzung der Übertragungsreserven entlang des Verbindungswegs durch das Netzwerk von der Datenquelle bis zur Datensenke.

## 3.1 Admission Controller

Der Admission Controller regelt den Zugang von Verbindungen zum Netz auf der Basis der Verkehrsparameter eines Dienstes und dem Zustand sowohl der lokalen als auch globalen Informationen über die Auslastung der verfügbaren Ressourcen entlang des Übertragungsweges vom Sender bis zum Empfänger.
Quellen, die Daten absetzen möchten, signalisieren das dem sogenannten *Admission Controller* mit einer *request-Meldung*. Die Signalisierung umfaßt die in Kapitel 2.2 genannten Dienst- und Güteparameter. An Hand dieser Charakteristika und zusätzlicher Statusinformationen vom *Allocation Controller* und dem *Cell Level Controller* wird dann bestimmt, ob die gewünschte Verbindung bereitgestellt werden kann. Dem Dienst wird durch eine *reject-Meldung* mitgeteilt, daß eine Verbindung mit den geforderten Parametern nicht zur Verfügung gestellt werden kann. Im Gegensatz dazu indiziert eine *confirm-Meldung*, daß

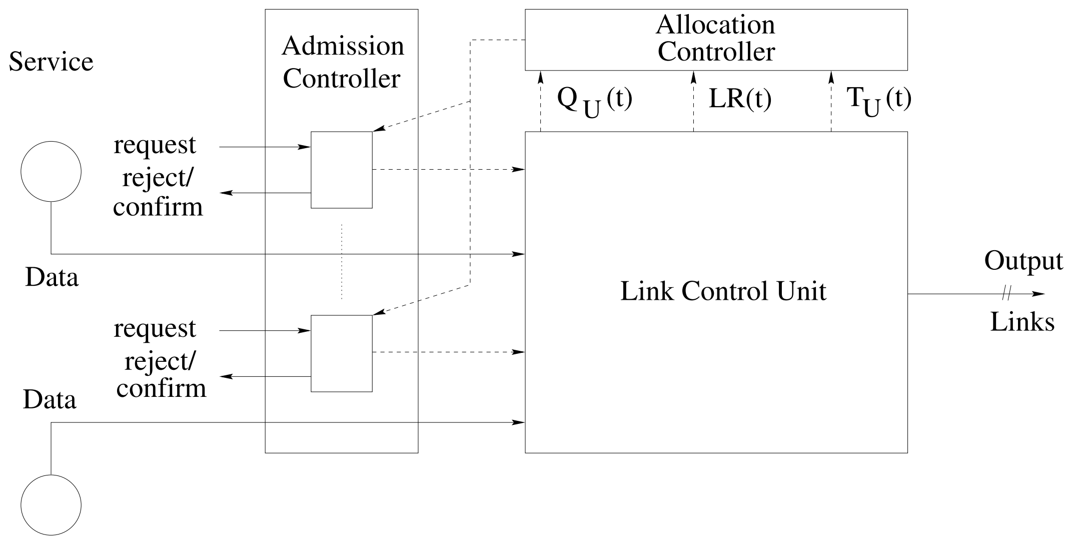



Abbildung 3.2: Struktur eines Zugangspunktes

genügend Ressourcen bereitstehen, um eine Verbindung mit den angegebenen Rahmenbedingungen zu schalten. Wesentlicher Bestandteil dieser Signalisierung stellt die Bedingung dar, daß die Serivicequalität der bereits bestehenden Verindungen nicht über die vereinbarten Toleranzen hinaus nachteilig beeinflußt wird. In diesem Falle werden die Nutzerdaten zur transparenten Übertragung an die *Link Control Unit* (Abbildung 3.3) übergeben.
Neben diesen beschriebenen essentiellen Aufgaben des Admission Controllers, die auf der Auswertung der temporären technischen Zustände basieren, können hier aber auch noch weiterreichende mehrdimensionale und dynamische Entscheidungsstrategien implementiert werden.

### 3.1.1 Entscheidungsstrategien

Bei speziellen Optimierungen von Kostenfunktionen kann es möglich sein, daß eine Verbindung nicht zustande kommt, obwohl genügend Übertragungskapazitäten zur Verfügung stehen, um die geforderte Dienstqualität garantieren zu können.

#### Mehrwertdienste

Es ist denkbar, daß auf Grund wirtschaftlicher Aspekte *weniger profitable* Dienste in einem bestimmten Rahmen abgewiesen werden, um Ressourcen für Verbindungen, die einen höheren Deckungsbeitrag erwirtschaften, vorzuhalten.

Bei dieser Strategie steht neben den technischen Randbedingungen die Gewinnoptimierung bei der Entscheidung über den Verbindungsaufbau im Vordergrund.

### Fairness

Dieses Verfahren ist so ausgerichtet, daß alle Dienste gleich behandelt werden.
In Hochlastsituationen kann wie oben beschrieben aus technischen Gründen nicht jeder Verbindungswunsch realisiert werden. Das bedeutet, daß Dienste mit einer gewissen Wahrscheinlichkeit, die sich aus der Anzahl der Verbindungswünsche, die nicht vermittelt wurden, zur Gesamtzahl aller Verbindungswünsche errechnen läßt, abgelehnt werden. Diese sog. CLR ist unter anderem eine Funktion des Angebotes und des Bandbreitenbedarfs, so daß sich anforderungsbedingt drastische Unterschiede bei den korrespondierenden Verlustraten ergeben können.
Fairness bedeutet in diesem Fall, daß die Entscheidungsstrategie eine Egalisierung der Verlustraten zwischen den einzelnen Diensten bewirkt.

## 3.2 Link Control Unit

Die Link Control Unit (LCU) ist in Abbildung 3.3 detailliert dargestellt. Sie übernimmt die transparente Übertragung, d. h. das Einketten der Informationseinheiten in die Warteschlangen, die Auswertung der Steuerinformationen und die Weiterleitung der Daten. Darüber hinaus überwacht und steuert sie die Datenströme der verschiedenen Dienste sowie der abgehenden Links. Diese Kontrolle basiert auf lokalen Kenngrößen wie den vereinbarten Verkehrsparametern und dem Auslastungsgrad der vorhandenen Ressourcen.
Am Eingang dieser Einheit wird für jede Eingangslast die Abweichung der tatsächlichen von der zu Beginn der Übertragung im Verkehrsvertrag vereinbarten Bandbreite bestimmt und an den Policing Controller übermittelt. Zusätzlich werden auch die aktuelle Länge der Warteschlange eines jeden Dienstes, die gesamte Systemauslastung sowie die Auslastung des Kanals an den Controller weitergereicht. Mit Hilfe unterschiedlicher Algorithmen werden aus diesen Daten dann die Bedienwahrscheinlichkeiten $P_B$ der Dienste ermittelt, aus denen dann die Steuerinformationen für den Multiplexer abgeleitet werden. Der Multiplexer am Ausgang des Systems schaltet die Datenströme sequentiell auf die entsprechenden Ausgangslinks. Die Bedienstrategie der einzelnen Datenpfade kann dabei nach unterschiedlichen Abarbeitungskriterien erfolgen. Einige wichtige werden im Folgenden kurz aufgezählt.

### FCFS

Bei der Abarbeitung der Warteschlangen nach dem First Come First Serve Prinzip werden die Nachrichten entsprechend ihrer Ankunftszeit übertragen. Es erfolgt weder eine Priorisierung eines Dienstes, noch hat die Auslastung der Warteschlangen eine Auswirkung auf die Reihenfolge der Abarbeitung. Ein gezieltes Verzögern der Daten innerhalb der möglichen Toleranzen, um andere Warteschlangen abzuarbeiten, ist nicht möglich.

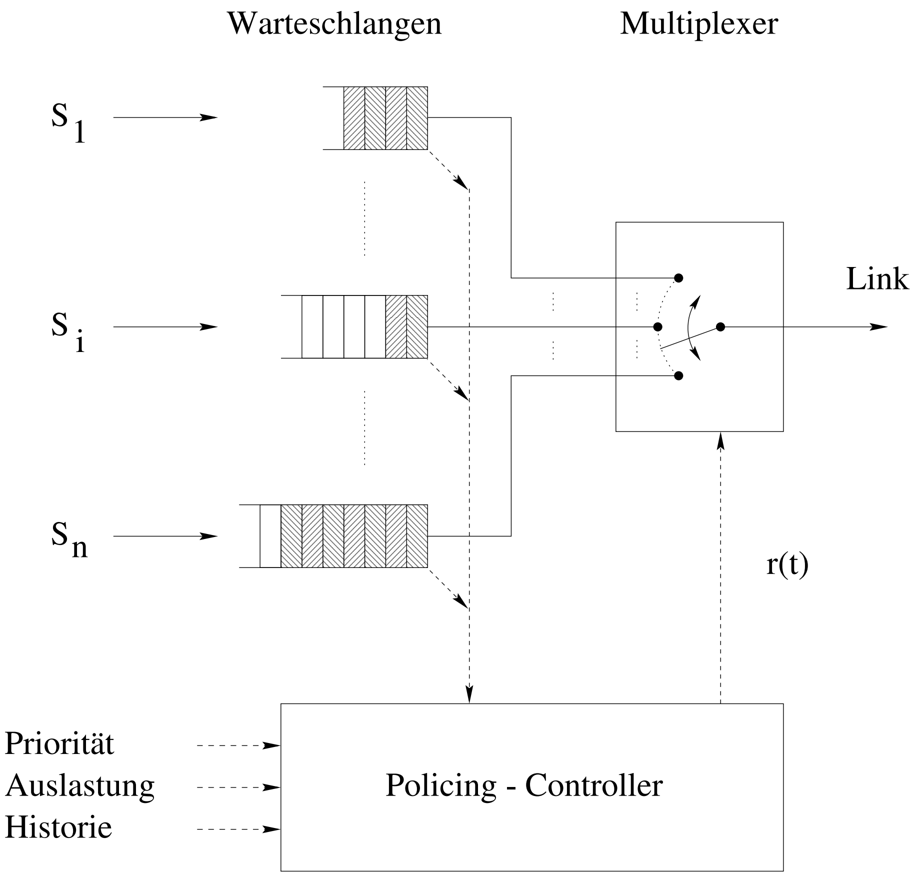



Abbildung 3.3: Struktur der Link Control Unit

### Round Robin

Bei diesem Verfahren werden die Warteschlangen sukzessive bedient. Die Ankunftszeit bzw. die Wartezeit einer Nachricht wird bei der Abarbeitung nicht berücksichtigt. Wie bei dem FCFS Verfahren erfolgt auch hier weder Priorisierung noch hat man die Möglichkeit, Daten systematisch zu verzögern.

### Dynamisch adaptive Bedienung

Mit Hilfe der zeitabhängigen Zustandsinformationen über die Auslastung der Ressourcen und des Verkehrsverhaltens der unterschiedlichen Dienste kann unter Berücksichtigung der gewünschten Bedienstrategie eine Bedienwahrscheinlichkeit der einzelnen Dienste innerhalb der assoziierten Toleranzen abgeleitet werden. Durch dieses Verfahren soll erreicht werden, daß die Ressourcen unter Beibehaltung der vereinbarten Randbedingungen effizient genutzt werden. Mit Hilfe dieser speziellen Struktur des Kommunikationsknotens soll zum einen eine *faire* Bandbreitenzuteilung, zum andern eine optimale Systemauslastung erreicht werden. Eine Zuteilung der zur Verfügung stehenden Bandbreite kann als *fair* be-

zeichnet werden, wenn Dienste, die ihre vereinbarten Parameter überschreiten, durch dieses Fehlverhalten die anderen Verbindungen des Knotens nicht oder nur im zulässigen Maß beeinflussen. Bei einer hohen Auslastung des Netzes soll die notwendige Beschränkung der Bandbreite nur die Verbindungen betreffen, die für diese Überlastung des Netzes verantwortlich sind. Diese Beschränkung der Bandbreite kann jedoch unterbleiben, solange die Toleranzen der vereinbarten Parameter der übrigen vertragskonformen Dienste nicht überschritten werden. In diesem Fall kann der Dienst die vereinbarte Bandbreite überschreiten, d. h. solange ungehindert übertragen, bis die Ressourcen erschöpft sind.
Um diesen Anforderungen genüge zu tun, wird die Bearbeitung der Pakete, die in den Warteschlangen temporär abgelegt sind, vom Systemzustand und der Lastcharakteristik abhängig gemacht. Aus den genannten Größen wird eine Bedienpriorität ermittelt, die von der Länge der Warteschlangen, der Auslastung des Kanals, der Priorität des Dienstes und der Konformität der Verkehrslast mit den vereinbarten Verbindungsparametern abhängig ist.

$$P_B = f(P_S, B_\Delta, T_U(t), Q_U(t)) \tag{3.1}$$

Stehen in mehreren Warteschlangen Daten zur Übertragung an, werden aus lokalen Zustandsinformationen Bedienwahrscheinlichkeiten für die betreffenden Dienste ermittelt. Die höchste Wahrscheinlichkeit impliziert nach dem Prinzip „Winner Takes It All“ , daß dieser Dienst dann bevorzugt behandelt wird.

Der Aufbau und die Struktur des Controllers sowie die differenzierte Bewertung der unterschiedlichen Eingangsgrößen mit verschiedenen Regelstrategien werden in den folgenden Kapiteln detaillierter ausgeführt.

## 3.3 Allocation Controller

Mit Hilfe des Allocation Controllers werden sowohl die Auslastung der Ressourcen des betrachteten Zugangsknotens als auch die Übertragungsreserven entlang des Verbindungswegs durch das Netzwerk von der Datenquelle bis zur Datensenke bstimmt. Die vorliegende Untersuchung beschränkt sich allerdings nur auf den Zugangsknoten, so daß hier im Wesentlichen die Auslastung der Warteplätze und die Belegung des Kanals ermittelt werden.

# Kapitel 4

# Konventionelle Kontroll - Verfahren

Die Modellierung und Simulation von Verfahren und Prozessen stellt eines der wesentlichen Werkzeuge zur Unterstützung der Entwicklung und zur Evaluierung von Methoden in der Kommunikationstechnik dar. Bei diesen Verfahren erfolgt mit der Modellbildung eine zielorientierte Vereinfachung der Realität durch Abstraktion. Um die Leistungsfähigkeit des in Kapitel 3 beschriebenen Kommunikationsknotens zu ermitteln und die Einsetzbarkeit der verwendeten Kontroll- und Regelstrategien detailliert beurteilen zu können, wurde ein umfangreiches ereignisgesteuertes Simulationsprogramm in der Programmiersprache C++ unter dem Betriebssystem Linux implementiert. Bei dem eingesetzten Verfahren beschränkt sich die Analyse des Systems auf die *diskreten* Zeitpunkte, in denen Zustandsänderungen auftreten. Ausgehend von einem Initialereignis, werden die Zeitpunkte von Folgeereignissen, die Aktionen auslösen, die dann weitere Ereignisse zur Folge haben, berücksichtigt. Die Zeitpunkte werden in einem Kalender abgelegt, der chronologisch abgearbeitet wird. Der Aufbau der Ereignistabelle ist im Anhang B.1 beschrieben. Auf diese Art wird sichergestellt, daß im Gegensatz zu der zeittreuen Simulation das System nur zu relevanten Zeitpunkten, d. h., wenn ein Zustandswechsel erfolgt, analysiert wird. Zweck dieser Simulation ist es, verschiedene Regelansätze für die Policing- und CAC Algorithmen zu testen und miteinander zu vergleichen. Zur Bewertung der Güte der Verfahren werden unterschiedliche Größen herangezogen.
Neben dem Simulationsprogramm des Knotenmodells wurde eine erhebliche Anzahl von Programmen zur Bearbeitung und Aufbereitung der Ein- und Ausgangsdaten implementiert.

## 4.1 Definition der Eingangslast

### 4.1.1 Lastprofil

Um das Regelverhalten zu überprüfen, wurde das in Abbildung 4.1 dargestellte Lastprofil eingeprägt. Als maximale Länge der Warteschlangen wurde für alle fünf Dienste Warteplätze für jeweils der Wert 400 Pakete eingerichtet. Die Bandbreite des abgehenden Kanals beträgt 155 Mbit/s. Die Datenpakete haben eine Dimension von 53 Bytes.

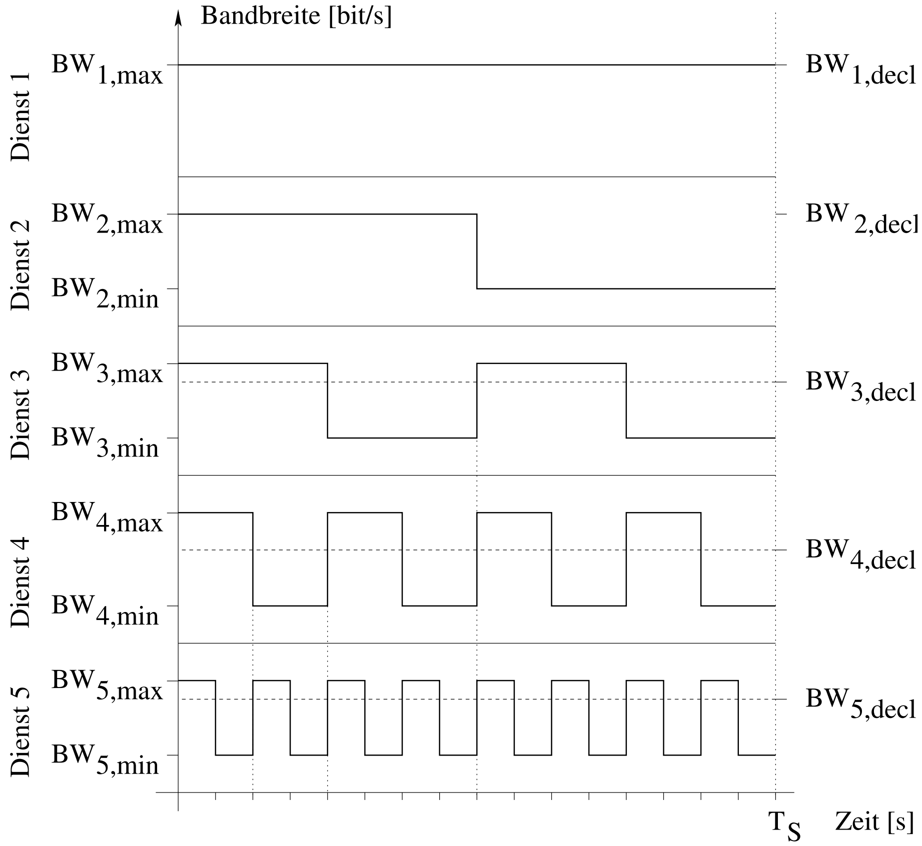



Abbildung 4.1: Lastprofil der Verkehrsquellen

### 4.1.2 Eingangslast zur Untersuchung der Policingverfahren

Für die Simulation des Regelverhaltens des Policing-Controllers wurde der Simulationszeitraum $T_S$ auf 1.6 Sekunden beschränkt. Die spezifischen Parameter der Verkehrslast, die deklarierte Bandbreite $BW_{decl}$, die Spitzenbandbreite $BW_P$, der Burst-Faktor $B$ und die Dienstpriorität $P_S$ sind in der Tabelle 4.1 angegeben. Zur Untersuchung wurden ausschließlich Quellen mit zwei Stati (ON - OFF Quellen) verwandt. Die Bitrate in der ersten Phase beträgt deshalb immer 0 Mbit/s. Die Verweildauern in den entsprechenden Zuständen werden durch die Parameter $T_{H,1}$ und $T_{H,2}$ erfaßt.
Der Dienst 1 bildet mit der sehr hohen Bandbreite, die während der gesamten Simulation benötigt wird, und der hohen Eingangspriorität eine Grundlast. Die übrigen Quellen sind durch einen deutlich geringeren Bandbreitenbedarf charakterisiert. Gemäß Tabelle 4.1 überschreitet Dienst 2 die Spitzenbandbreite, die sich rechnerisch aus deklarierter Bandbreite und Burstfaktor ergibt, nicht, während die übrigen Dienste ihre vereinbarte Spitzenbandbreite verletzen. Die Eingangsprioritäten liegen zwischen gering (jeweils 0.3 für die

| Dienst | Bandbreite [Mbit/s] | | | Burst- | Verweilzeit [s] | | Priorität |
|---|---|---|---|---|---|---|---|
| | min | max | decl | faktor | $TB_1$ | $TB_2$ | |
| 1 | 0 | 96 | 96 | 1 | 0 | 1.6 | 0.9 |
| 2 | 0 | 28 | 10 | 2 | 0.8 | 0.8 | 0.5 |
| 3 | 0 | 10 | 2.5 | 4 | 0.4 | 0.4 | 0.3 |
| 4 | 0 | 25 | 5 | 2 | 0.2 | 0.2 | 0.3 |
| 5 | 0 | 35 | 8 | 2 | 0.1 | 0.1 | 0.6 |

Tabelle 4.1: Verkehrslast I

Dienste 3 und 4) und mittel (0.5 für Dienst 2 und 0.6 für Dienst 5). Durch den Verlauf des vorgegebenen Lastprofils kann jede mögliche Kombination der deklarierten Bandbreite und der Spitzenbandbreite aller fünf Dienste erfaßt werden.
Eine Verkürzung des Beobachtungszeitraumes, die im Hinblick auf die damit verbundene Verringerung der realen Simulationszeit wünschenswert ist, wäre durch Verkleinerung der einzelnen Intervalle unter Beibehaltung des Teilungsverhältnisses möglich. Nachteilig wäre allerdings, daß dann die Einschwingzeiten, die sich bei jedem Zustandswechsel eines Dienstes ergeben, in keinem Verhältnis zum kleinsten Zustandsintervall stehen würden.

### 4.1.3 Eingangslast zur Untersuchung der CAC-Algorithmen

Für die Abschätzung der Leistungsfähigkeit der CAC-Algorithmen muß auf Grund der Randbedingungen, die durch das Minimal Reservation Verfahren vorgegeben wird - die minimale Bitrate muß $> 0Bit/s$ sein[1] - ein weiteres Verkehrsprofil festgelegt werden. Neben den in der Tabelle 4.2 gegebenen Werte für die vier unabhängigen Verkehrsparameter deklarierte Bandbreite $BW_{decl}$, Spitzenbandbreite $BW_P$, Burst-Faktor $B$ und Dienstpriorität $P_S$ wurde eine Simulationszeit von $T_S = 10s$ zugrunde gelegt. Diese Verlängerung wurde notwendig, um den Zugangsknoten mit einer repräsentativen Anzahl von Verbindungswünschen zu belasten. Verbunden mit der Verlängerung der Simulationszeit war leider eine erhebliche Steigerung der Programmlaufzeit.

| Dienst | Bandbreite [Mbit/s] | | Burst- | Verweildauer [s] | | Priorität |
|---|---|---|---|---|---|---|
| | min | max | faktor | $TB_1$ | $TB_2$ | |
| 1 | 1 | 5 | 5 | 1.6 | 1.6 | 0.9 |
| 2 | 0.5 | 4 | 8 | 0.8 | 0.8 | 0.5 |
| 3 | 0.5 | 9 | 18 | 0.4 | 0.4 | 0.2 |
| 4 | 0.5 | 6 | 10 | 0.2 | 0.2 | 0.7 |
| 5 | 1 | 6 | 3 | 0.1 | 0.1 | 0.6 |

Tabelle 4.2: Verkehrslast II

[1] Wie in Abschnitt 2.2.4 erläutert

## 4.2 Verfahren zur Bewertung der Simulationsergebnisse

### 4.2.1 Kenngrößen

Für die Bestimmung der Güte eines Algorithmus stehen viele unterschiedliche charakteristische Größen und Auswertungsverfahren zur Verfügung:

#### Verlustraten

Die Beschreibung der unterschiedlichen relevanten Verlustraten ist in Abschnitt 2.2.5 erfolgt. Im Allgemeinen kann jedoch fixiert werden, daß das Auftreten von Verlusten die Qualität stark beeinträchtigt. Die Verlustraten stellen daher ein essentielles Bewertungskriterium dar.

#### Verzögerung ($T_D$)

Die Verzögerung ist ebenfalls ein relevanter Faktor zur Beurteilung der Leistungsfähigkeit des Kommunikationsknotens. Er hängt von der Auslastung der Warteschlangen, der Belastung der Systemressourcen und der Bearbeitungsstrategie ab.

#### Durchsatz ($\gamma$)

Der Durchsatz beschreibt das Verkehrsvolumen, das durch den Kommunikationsknoten erfolgreich bearbeitet werden konnte. Er berechnet sich aus den Informationseinheiten, die in einem festgelegten Zeitraum geeigneter Länge übertragen werden konnten.

#### Power

Der Durchsatz ($\gamma$) und die durchschnittliche Verzögerung $T_D$ werden in der Kenngröße Power zusammengefaßt.

$$P = \frac{\gamma}{T_D} \tag{4.1}$$

#### Auslastung

Die Auslastung ist ein Indikator für die Nutzung der Systemressourcen. Man unterscheidet zwischen der Auslastung der dienstspezifischen Warteschlange, die die temporäre Lastsituation eines Dienstes reflektiert ($Q_U$) und der Inanspruchnahme aller Warteplätze eines Knotens, die dann Auskunft über die allgemeine Lastsituation ($S_U$) gibt. Neben der Nutzung des Speichers stellt die Auslastung des Kanals $T_U$ eine relevante Kenngröße zur Beurteilung der untersuchten Verfahren dar.

### 4.2.2 Bewertung

Die Beurteilung und der Vergleich der Verfahren untereinander kann auf der Basis unterschiedlich gewonnener Werte basieren.

#### Spitzenwerte

Diese Werte dienen zur Bestimmung der aktuellen Kenngrößen sowie der Berechnung der Parameter während der Simulation. Die Spitzenwerte sind für die Bewertung und den Vergleich der unterschiedlichen Verkehrssteuerungsverfahren nur bedingt von Bedeutung. Für einen objektiven Vergleich müßten alle Prozesse exakt synchronisiert sein, so daß alle Zustände immer in derselben Reihenfolge durchlaufen würden. In solch einem Falle könnten die Verfahren dann zu jedem Zeitpunkt miteinander verglichen werden.

#### Mittelwerte

Aussagen über die Qualität der eingesetzten Verfahren lassen sich dann einfacher über die Bildung von Mittelwerten machen. In der vorliegenden Untersuchung werden deshalb für die in Abschnitt 4.2.1 genannten Parameter die Integrale über die Funktionen über den gesamten Simulationslauf mit Hilfe der Riemannschen Summe nach Gl. 4.2 approximiert. Eine Skalierung erfolgt, indem diese Summe auf die maximale Fläche bezogen wird.

$$\bar{X} = \frac{\sum_{i}^{N}(t_i - t_{i-1}) \cdot Parameter(t)}{Parameter_{max} \cdot T_S} \tag{4.2}$$

Die so ermittelte Kennzahl ist ein Maß für die Güte des Controllers. Die Interpretation wiederum ist dann von der Bedeutung des basierten Parameters abhängig. Bei Verlustraten und Verzögerungen weist ein geringer Wert, dagegen bei Power, Durchsatz und Auslastung eine höhere Kennzahl die gute Qualität aus.

## 4.3 Policing Controller

Um die Leistungsfähigkeit der Policing-Verfahren beurteilen zu können, wurde der Policing Controller zur Überwachung des Datenflusses mit Hilfe des sog. Generic Cell Rate Algorithm realisiert. Bei diesem Verfahren wird durch die Verwendung von Zählern in den einzelnen Datenpfaden der Quellen deren Bandbreite begrenzt. Nach jedem gesendeten Datenpaket wird der assoziierte Sendezähler inkrementiert. In regelmäßigen Intervallen, die sich direkt aus der im Verkehrsvertrag ausgehandelten Übertragungsrate ergeben, wird der Zähler dekrementiert. Falls jedoch der Schwellenwert überschritten wird, werden die überzähligen Datenpakete verworfen.
Grundlage für die simulative Untersuchung der Policing Controller bildet das in Abb. 4.1 dargestellte Verkehrsmuster in Verbindung mit den Daten aus Tabelle 4.1. Der Netzzugang

wird mit dem Peak Reservation Verfahren abgewickelt. Die Warteschlangen werden nach der FCFS-Strategie bedient.
In den Abbildungen sind aus Gründen der Übersichtlichkeit nur die Mittelwerte angegeben. Die Vertrauensintervalle sind sehr klein und rangieren in einem Bereich $\leq 10\%$ der Mittelwerte.

### 4.3.1 Verlustraten

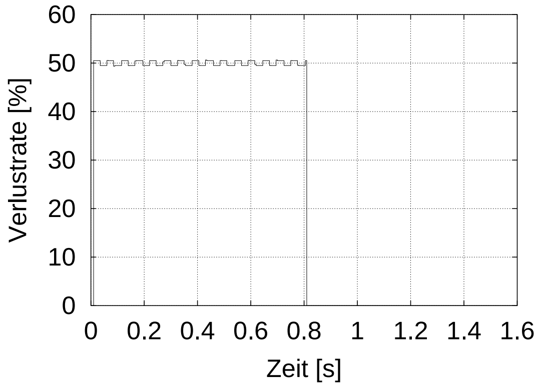
Die ermittelten Verlustraten für die Dienste 1 und 3 beliefen sich während des gesamten Simulationszeitraums auf 0%. Die übrigen Verläufe sind in den Abbildungen 4.2 und 4.2(c) wiedergegeben. Es wird ersichtlich, daß bei diesem Verfahren ein Überschreiten der vereinbarten Vertragsparameter in jedem Fall zu Verlusten führt, unabhängig davon, in welchem Maße die Dienste ihre deklarierte Bandbreite einhalten oder nicht. So wird schon im Vorfeld sichergestellt, daß keine Konflikte durch eine Verletzung der vereinbarten Parameter auftreten können. Durch dieses restriktive Verfahren wird die Dienstqualität der bestehenden Verbindungen in jedem Falle sichergestellt.

### 4.3.2 Auslastung der Warteschlangen

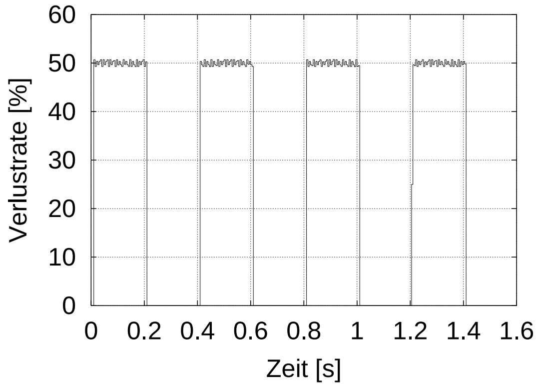
Dem Vorteil, daß die einmal vereinbarten Vertragsparameter uneingeschränkt zugesichert werden können, steht die stark eingeschränkte Ausnutzung der Übertragungskapazitäten gegenüber. Die Abbildungen 4.3 sowie 4.4 zeigen den Verlauf der Spitzenwerte der Auslastung für die dienstspezifischen Warteschlangen. Die Nutzung der Systemressourcen ist gering. Die Spitzenwerte bei der Auslastung der Warteschlangen sind in jedem Fall kleiner als 3.5%.

### 4.3.3 Kanalauslastung

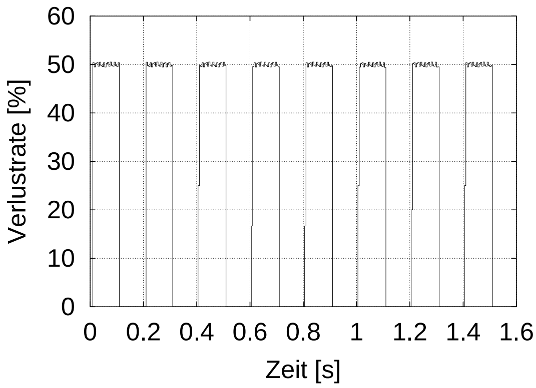
In Abbildung 4.5 ist die Auslastung des angeschlossenen Links dargestellt. Es ist ersichtlich, daß die Kanalkapazitäten nur zu einem Teil genutzt werden. Der Kanal wird während des gesamten Simulationszeitraums weit unter der möglichen Auslastungsgrenze betrieben. Es stehen zu jedem Zeitpunkt noch genügend Reserven zur Verfügung, um weiteren Verkehr anderer Verbindungen zu bewältigen.




(a) Dienst 2

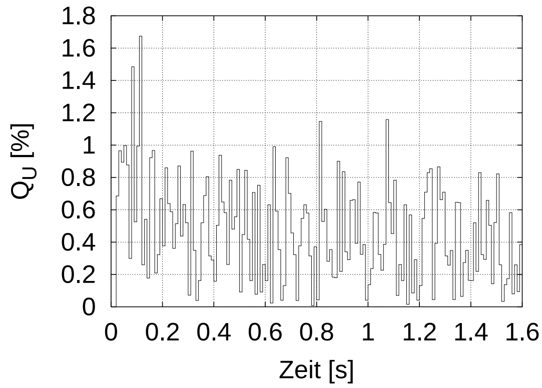



(b) Dienst 4

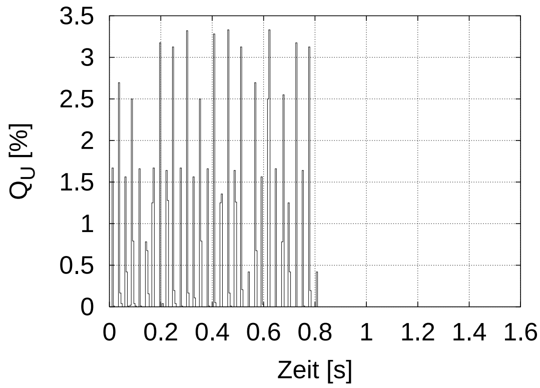



(c) Dienst 5

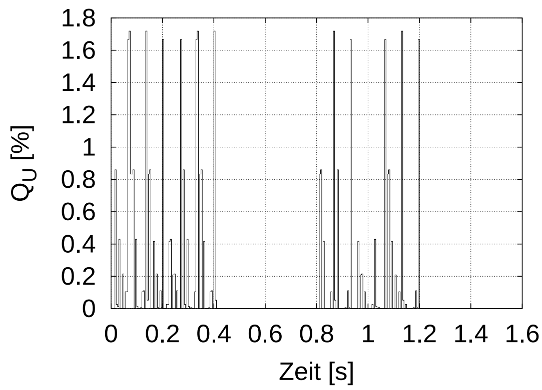
Abbildung 4.2: Verlustraten



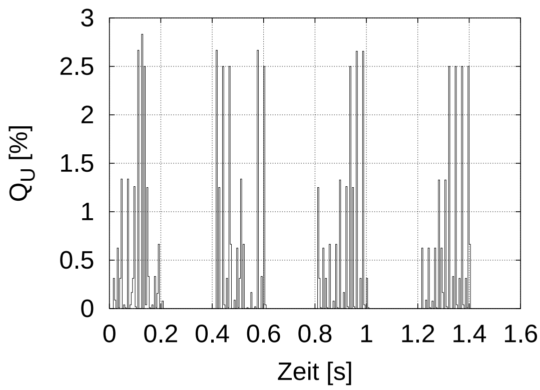

(a) Dienst 1




(b) Dienst 2




(c) Dienst 3




(d) Dienst 4

Abbildung 4.3: Auslastung der Warteschlangen

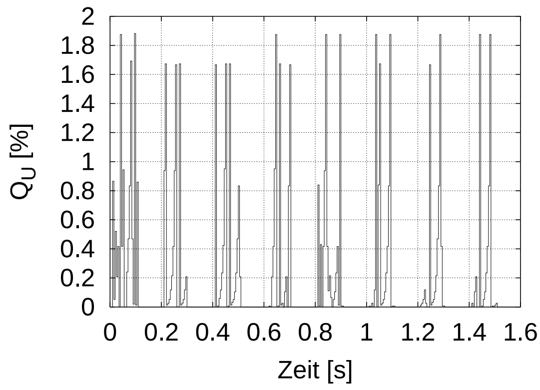



Abbildung 4.4: Auslastung der Dienstwarteschlange 5

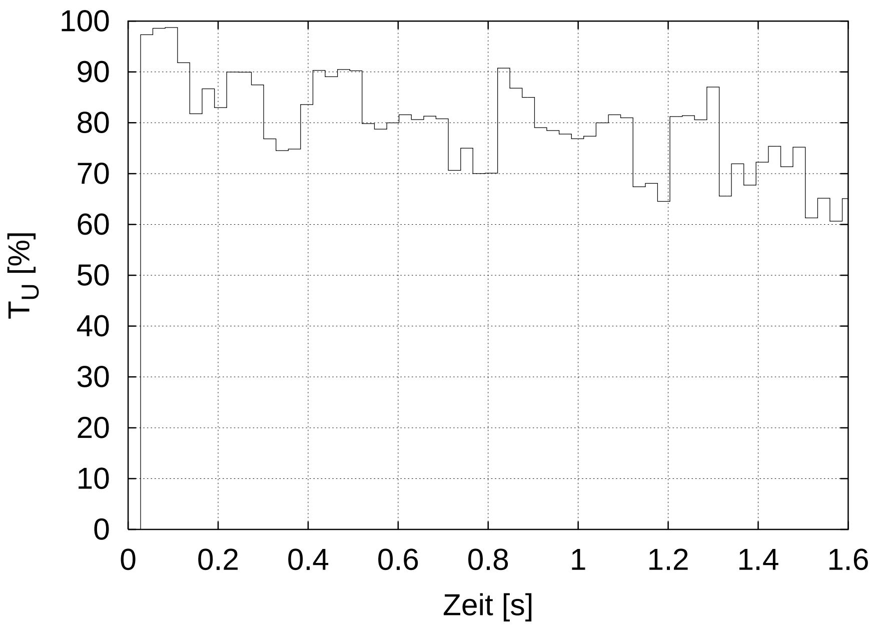



Abbildung 4.5: Kanalauslastung

### 4.3.4 Bewertung des Policing Controllers

Tabelle 4.3 dokumentiert die für die Auslastung und Verluste ermittelten Bewertungskennzahlen. Die Belegung der Warteplätze ist immer kleiner 0.4% für Dienst 1, die Nutzung der Ressourcen der übrigen Dienste ist noch geringer. Die Kennzahlen für die Dienste 1 und 3 belaufen sich während des gesamten Simulationszeitraums erwartungsgemäß auf 0%. Bei den übrigen Diensten treten Verluste auf, obwohl der Kanal noch erhebliche Übertragungskapazitäten aufweist. Nach Tabelle 4.3 wird die zur Verfügung stehende Bandbreite nur zu 79% genutzt. Es stehen demnach noch genügend Reserven zur Verfügung, um weiteren Verkehr anderer Verbindungen zu bewältigen. Auch hier wird ersichtlich, daß bei diesem Verfahren ein Überschreiten der vereinbarten Vertragsparameter in jedem Fall zu Verlusten führt, unabhängig davon, in welchem Maße die Dienste ihre deklarierte Bandbreite einhalten oder nicht.

Bei der Anwendung dieses Verfahrens kann ein Überschreiten der vereinbarten Band-

| Dienst | Auslastung | Verlustrate |
|---|---|---|
| 1 | 0.004 | 0.00 |
| 2 | 0.002 | 0.25 |
| 3 | 0.001 | 0.00 |
| 4 | 0.002 | 0.25 |
| 5 | 0.001 | 0.25 |
| Mittelwert | 0.2% | 15% |
| Kanalauslastung | 0.793 | — |

Tabelle 4.3: Leistungskennzahlen des GCRA

breite auf relativ einfache Art und Weise verhindert werden. Durch die starre Festlegung der Periode für die Dekrementierung des Zählers kann das Verfahren jedoch nicht flexibel auf Schwankungen der Bandbreite reagieren. Die mögliche Zuteilung, der von einer Quelle temporär nicht genutzten Ressourcen durch einen anderen Dienst, kann bei diesem Kontrollverfahren nicht ausgenutzt werden. Systemweit manifestierte sich eine Verlustrate von 15%, die durchschnittliche Nutzung der Warteplätze ist mit $\approx 0.2\%$ eher gering.
Die Auslastung des Kanal ist mit ca. 80% unkritisch.
Die Auswertung demonstrierte nachhaltig, daß der vorgestellte Policing Controller in der Lage ist, die im Verkehrsvertrag vereinbarten Dienstparameter stringent zu überwachen. Es wurde aber auch sichtbar, daß die unterlagerte Strategie übertrieben restriktiv ist. Aktuelle Zustandsinformationen über die Auslastung und Verluste werden nicht berücksichtigt. Die Folge ist, daß die Ressourcen nur ungenügend genutzt werden.

Daher soll im Rahmen dieser Arbeit in Abschnitt 5.3 ein adaptiver auf Fuzzy Logic basierter Policing-Controller entwickelt werden, mit dessen Hilfe eine flexible Zuteilung der zur Verfügung stehenden Bandbreite unter Ausnutzung der vorhandenen Ressourcen möglich ist.

## 4.4 Der Call Admission Controller

Grundlage für die simulative Untersuchung der Call Admission Controller bildet das in Abb. 4.1 dargestellte Verkehrsmuster in Verbindung mit den Daten aus Tabelle 4.2. Bei der Simulation wird das Angebot linear mit einem Inkrement von 5 Erlang für alle Dienste gesteigert. Die Überwachung der Datenströme erfolgt in allen Fällen mit Hilfe des in Abschnitt 4.3 beschriebenen GCRA Verfahrens. Die Warteschlangen werden nach der FCFS-Strategie bedient.
In den folgenden Abbildungen sind aus Gründen der Übersichtlichkeit nur die Mittelwerte angegeben. Die ermittelten Vertrauensintervalle sind $\leq 10\%$ der entsprechenden Mittelwerte.

### 4.4.1 Das Peak Reservation Verfahren

Die Abbildungen 4.6 und 4.7 zeigen die simulativ ermittelte Call Loss Rate sowie die Auslastung der Warteschlangen und des abgehenden Kanals bei Verwendung des Peak Reservation Verfahrens. Die Verlustrate ist, da auf Grund der Reservierung der Übertragungsressourcen auf der Basis der maximalen

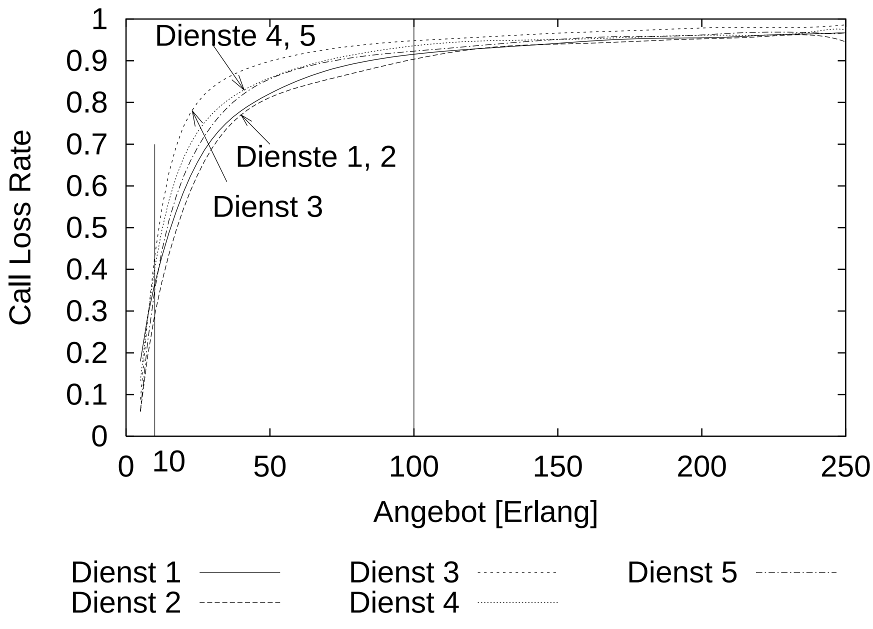



Abbildung 4.6: Call Loss Rate

Übertragungsrate kein Überlauf auftritt, Null. Die CLR nimmt schnell große Werte an. Schon ab 50 Erlang ist der Wert für alle Dienste auf > 80% (Abb. 4.6) angestiegen. Im Bereich mittlerer Last, zwischen 10 Erlang und 100 Erlang, unterscheiden sich die Verläufe der Kennlinien auf Grund der Reservierungsbandbreite. Dienst 3 hat in diesem Bereich wegen

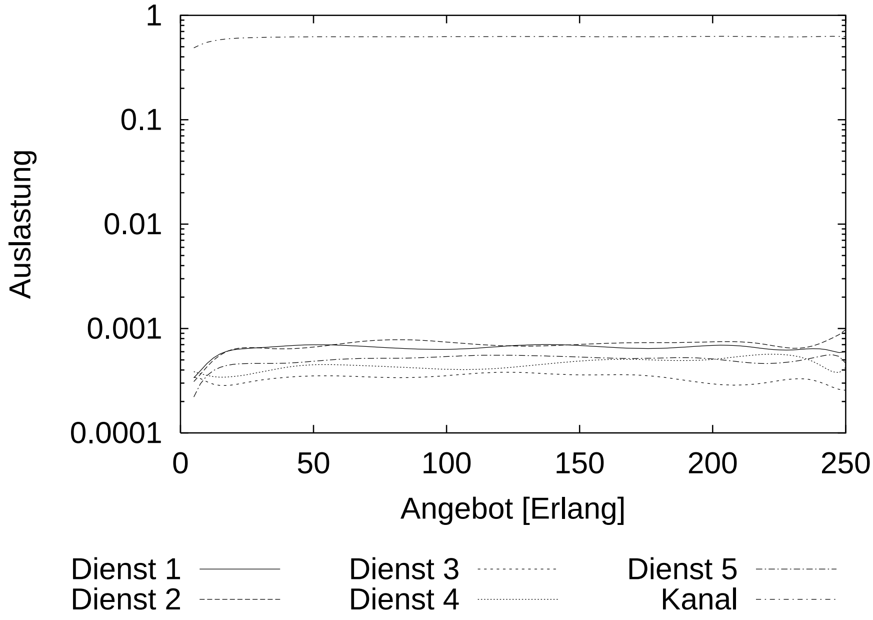



Abbildung 4.7: Auslastung der Systemressourcen

einer maximalen Belegungsbandbreite von 9 Mbit/s die größten Verluste. Die Bandbreitenreserve sinkt temporär auf kleinere Werte. Durch die feinere Granularität der Bandbreiten der übrigen Dienste ist dann auch die Abstufung der CLR schlüssig zu erklären.
Eng korreliert mit diesem Verhalten ist die Auslastung des Systems. Abbildung 4.7, zeigt sowohl die Auslastung der Warteschlangen als auch die Belegung des Links. Die Nutzung des Speichers ist minimal und für alle Dienste < 0.08%. Der Kanal wird mit einer Auslastung von < 64% weit unter seinen optimalen Bedingungen betrieben.

### 4.4.2 Minimal Reservation

Bei dem Minimal Reservation Verfahren (MR) wird die Bandbreite auf der Basis der geringsten Übertragungsrate ungleich Null reserviert. Auf Grund dieses Ansatzes kann die Dienstqualität nicht gewährleistet werden. Im Gegensatz zu dem Peak Reservation Verfahren wird hier aber die verfügbare Bandbreite vollständig genutzt. Die Verfahren stellen also die Grenzen für alle weiteren Verfahren dar und dienen somit zur Bestimmung der Leistungsfähigkeit.
In den Abbildungen 4.8 bis 4.10 ist der Verlauf der Kennlinien bei Verwendung des Minimal Reservation Verfahrens wiedergegeben.
Die CLR nimmt nur langsam zu. Bei einer Last von 50 Erlang beträgt ihr Wert für die Dienste 1 und 5 ca. 15%. Für die übrigen beträgt sie ca. 10%. Weiterhin zeichnet sich deutlich ab, daß die Dienste in zwei Gruppen, mit unterschiedlichen Verläufen der CLR, aufgesplittet werden. Der Abstand zwischen den Kennlinien beträgt ca. 20% und läßt sich

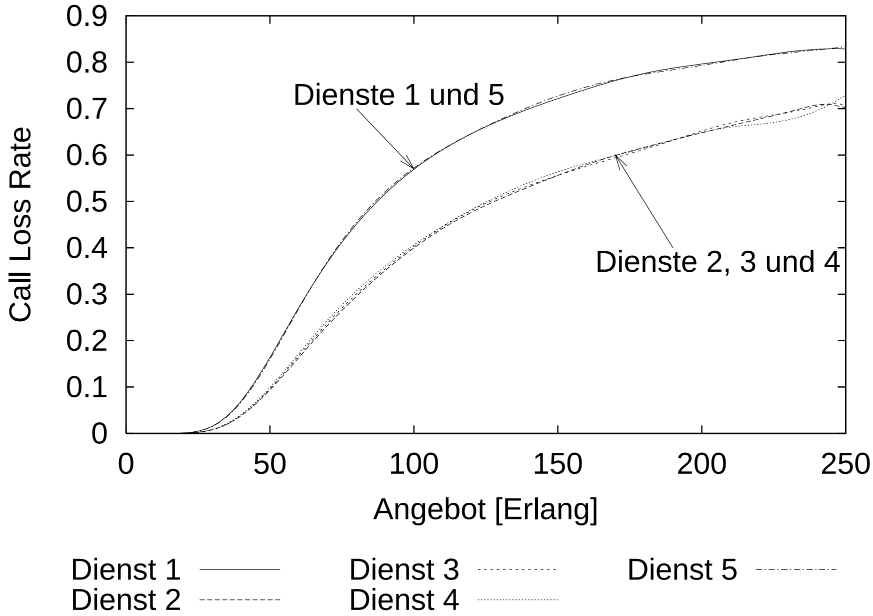



Abbildung 4.8: Call Loss Rate

damit begründen, daß die Dienste 1 und 5 mit einer größeren minimalen - für die Reservierung relevanten - Bandbreite arbeiten als die Dienste 2, 3 und 4. Der Kanal wird schon bei einem Angebot von 15 Erlang mit 95% Auslastung an seiner Grenze betrieben. Die Reserven, um beliebig weitere Verbindungen zu schalten, sind erschöpft. Vor diesem Hintergrund und mit Gl. 2.4 ist ersichtlich, daß Dienste mit einer feineren Granularität in Bezug auf die Übertragungsrate eher aufgesetzt werden können als solche mit einer größeren minimalen Bandbreite.

Die Verlustrate von Informationseinheiten ist für alle Dienste sehr hoch. Sie liegt zwischen 68% und 85%. Die Verlustrate der Dienste 1 und 5 ist ab 50 Erlang wegen der gesteigerten CLR in diesem Bereich kleiner als bei den Diensten 1, 2 und 3.

Da die benötigte Übertragungskapazität nur an Hand der minimalen Bandbreite bewertet wird, der Verkehr der Dienste aber burstbehaftet ist, ist die Auslastung der Speicherkapazität der Warteschlangen so groß, daß die Informationseinheiten absorbiert werden.

### 4.4.3 Bewertung der Call Admission Controller

Die untersuchten CAC-Strategien wiesen sich dadurch aus, daß die Reservierung der Übertragungsbandbreite entweder auf der maximalen oder aber der minimalen Bandbreite eines Dienstes basierte. Diese Verfahren verkörpern zwei extrem konträre Ansätze. Auf der einen Seite treten, beim Einsatz des PR-Verfahrens keine Informationsverluste auf. Die Auslastung der Ressourcen ist aber außerordentlich gering. Die CLR steigt schnell an.

Bei dem Einsatz der MR-Methode ist die CLR im Vergleich zur PR-Methode deutlich ge-

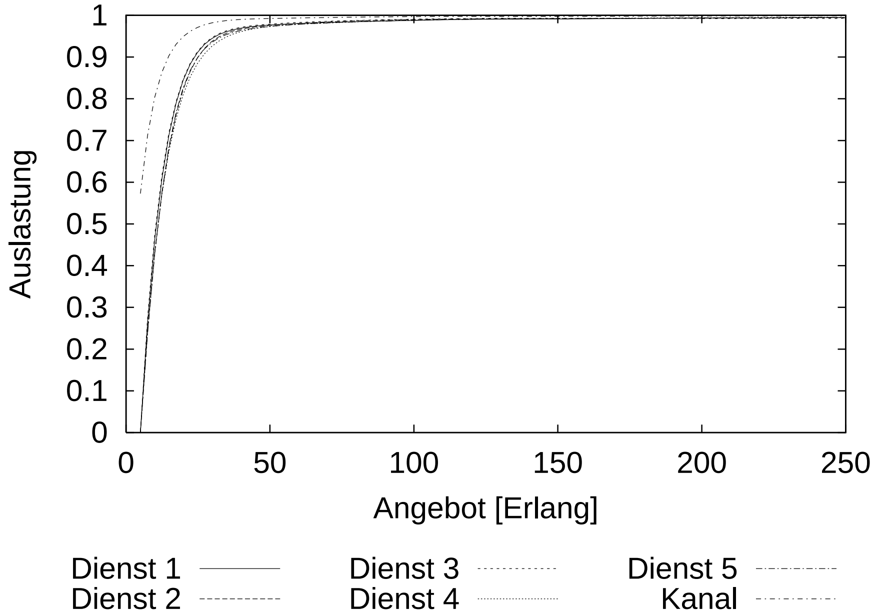



Abbildung 4.9: Auslastung

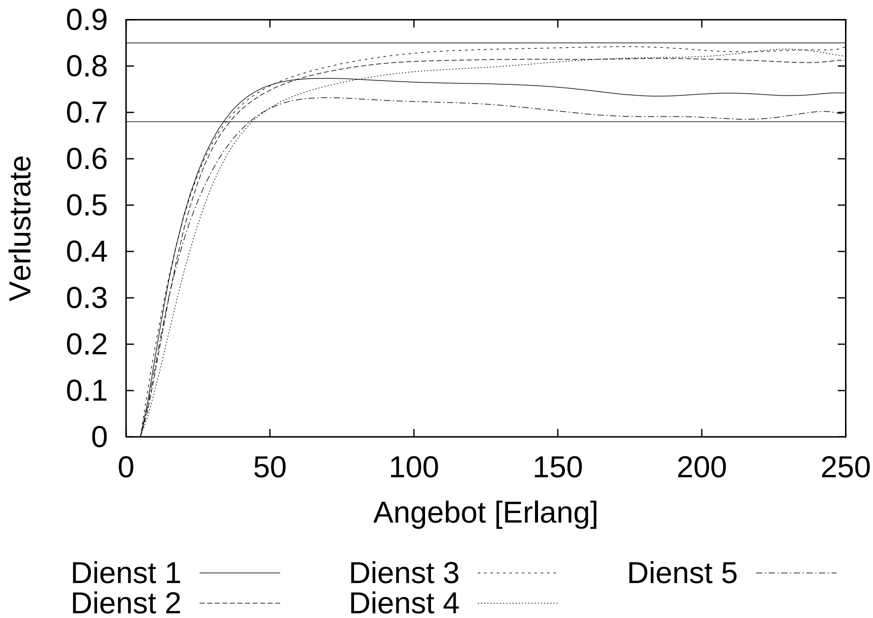



Abbildung 4.10: Minimal Bandwidth Reservation

ringer. Das System ist deshalb schon bei einem geringen Angebot vollständig überlastet, so daß im gesamten Verlauf nicht unerhebliche Datenverluste auftreten.

Die Simulationsergebnisse dokumentieren weiterhin, daß die Dienste auf Grund ihres unterschiedlichen Bandbreitenbedarfs nicht gleichbehandelt werden. Dienste mit einer hohen Spitzenbitrate weisen beim PR-Verfahren erheblich größere Verluste auf als die übrigen Dienste. Die Behandlung der Dienste ist *nicht fair*.
Die Ergebnisse dokumentieren, daß mit den eingesetzten exakten Verfahren das System „eindimensional“ optimiert werden kann. Bei den untersuchten Strategien waren das die Minimierung der Datenverluste (PR) bzw. die Maximierung des Durchsatzes (MR). Beide Verfahren sind nicht adaptiv, so daß der Einfluß temporärer Lastsituationen nicht berücksichtigt wird.
Um diese offensichtlichen Nachteile zu umgehen und um weitere differenziertere Merkmale einzuprägen, soll in dem folgenden Kapitel ein neuartiges Verfahren vorgestellt werden bei dem die Überwachung und Steuerung der Datenströme mit Hilfe Fuzzy Logic basierter Methoden erfolgt.

# Kapitel 5

# Der Fuzzy Controller

Fuzzy Systeme sind numerische Annäherungsverfahren, die die Beziehung zwischen Eingängen und Ausgängen eines Systems verallgemeinern. Aber anders als bei den Neuronalen Netzen können die Verknüpfungen zwischen Systemgrößen sowie deren Transformation durch feste Regeln beschrieben werden. Der Hauptvorteil der Fuzzy Logic Controller gegenüber klassischen Reglern liegt darin begründet, daß das *qualitative* Wissen und die Erfahrung über einen Prozeß in sog. linguistische Variablen und Regeln eingebracht und nicht durch ein *quantitatives* Regelwerk beschrieben wird. Auf Grund dieses Vorteils, daß ein Fuzzy Controller keine exakten Informationen benötigt, ist er in Bereichen, die mit Unsicherheiten und Ungenauigkeiten assoziiert sind, sowie in nichtlinearen Systemen gut einsetzbar. Ein Fuzzy Controller hat gegenüber konventionellen Methoden folgende Vorteile:

- Ein Fuzzy Controller kann relativ schnell und einfach implementiert werden.
- In einen Fuzzy Controller kann Expertenwissen eingebracht werden.
- Der Ausgang eines Fuzzy Controllers wird dynamisch adaptiert.

Auf Grund dieser Eigenschaften, die gerade für diesen Einsatzbereich essentiell sind, soll im Folgenden eine kurze Einführung in die Grundlagen der Fuzzy Logic gegeben werden. Eine weiterführende Vertiefung zur Fuzzy Logic findet sich in [11, 9, 22, 42, 43, 66]

## 5.1 Grundlagen

### 5.1.1 Fuzzy Sets

In der klassischen Algebra wird eine Menge M auf einer Grundmenge G so definiert, daß man festlegt, welche Elemente x der Grundmenge zur Menge M gehören sollen ($x \in M$) oder nicht in M enthalten sind ($x \notin M$). Diese Zugehörigkeit kann durch eine charakteristische

Funktion beschrieben werden.

$$\mu_M = \begin{cases} 1 & : \quad für\ x \in M \\ 0 & : \quad sonst \end{cases} \tag{5.1}$$

Durch $\mu_M$ wird die Grundmenge G auf die Menge [0;1] abgebildet.

$$\mu_M : G \rightarrow [0;1] \tag{5.2}$$

In der konventionellen Logik sind die Mengen also durch charakteristische Funktionen ausgezeichnet, die nur zwei Werte annehmen können. Dieser Zweiwertigkeit entspricht die Alternative: *Ein Element ist Teil einer Menge M oder nicht.* Eine weitere Möglichkeit existiert nicht. Bei der Fuzzy Logic wird diese Zweiwertigkeit erweitert, um eine genauere Klassifikation der Elemente zu ermöglichen.

**Unscharfe Mengen**

Die Ergänzung besteht in der Einführung *unscharfer* Mengen, den *Fuzzy* Mengen. Für Elemente einer Grundmenge existiert dann definitionsgemäß eine kontinuierliche graduelle Zugehörigkeit, die sich von der Nichtmitgliedschaft ($x \notin M$) bis zur vollen Mitgliedschaft ($x \in M$) erstreckt. Durch die Erweiterung des Wertebereiches der charakteristischen Funktion auf alle reellen Zahlen zwischen 0 und 1 ist es möglich, $\mu_M(x)$ als Zugehörigkeitsgrad des Wertes x zu einer Menge zu interpretieren. $\mu_M$ bildet die Grundmenge G auf das Einheitsintervall $[0,1]$ ab.

$$\mu_m : G \rightarrow [0,1] \tag{5.3}$$

Die Funktion $\mu(x)$ wird daher als Zugehörgikeitsfunktion bezeichnet. Mathematisch läßt sich die unscharfe Menge mit Gleichung 5.4 beschreiben.

$$A = \{(x, \mu_A(x)) | \forall x \in X, \mu_A \in [0,1]\} \tag{5.4}$$

Es existieren vielfältige Fuzzy-Mengen, deren Einsatz in Applikationen von der Beschreibungsform abhängig ist. Während die Verwendung von Gauß-förmigen oder quadratischen Fuzzy-Sets (sog. S-Zugehörigkeitsfunktionen) relativ viel Rechnerleistung benötigt, erfordert die Darstellung der Mengen in diskreter Form[1], abhängig vom Detaillierungsgrad, viel Speicherplatz. Für den praktischen Einsatz haben sich trianguläre und trapezförmige Fuzzy-Mengen mit linearen Flanken etabliert, da sie in Bezug auf Rechenzeit und Speicherplatzbedarf eine optimale Lösung darstellen. Die Abbildung 5.1 zeigt die Modellierung eines Fuzzy-Sets durch eine dreieck- und trapezförmige Menge. Die Beschreibung dieser Fuzzy-Sets ist unkompliziert und kann z. B. mit Hilfe der sog. LR-Referenzfunktionen, wie in Absatz 5.1.1 gezeigt, erfolgen. Die Beschreibung der Sets beschränkt sich bei Trapezen auf vier und bei triangulären Formen auf drei Parameter.
Die Mengen oder auch *linguistischen Werte* dienen der Bewertung einer physikalischen Größe, die als *linguistische Variable* bezeichnet wird.

[1]Die Zugehörigkeitsfunktionen sind in Form von „look up-Tabellen“ im Speicher abgelegt

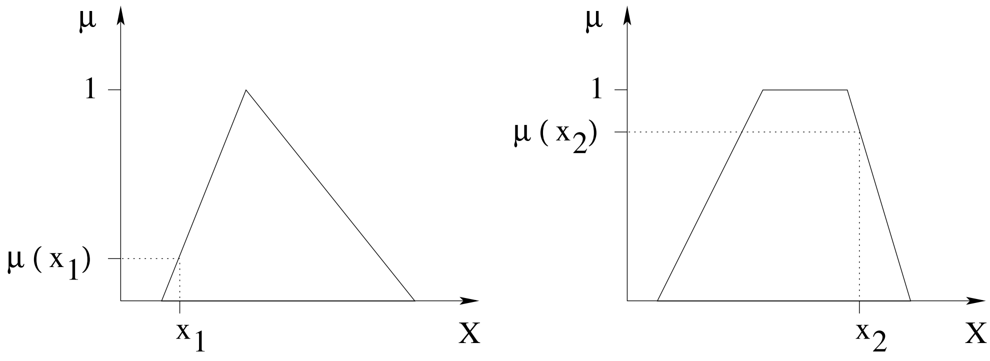



Abbildung 5.1: Trianguläre und trapezförmige Fuzzy-Mengen

### Linguistische Variablen

Ein weiteres wesentliches Kennzeichen der Fuzzy Logic besteht darin, daß Kenngrößen durch die *linguistischen Variablen* ersetzt werden. Der Wertebereich dieser Größen wird nicht wie bei klassischen Systemen exakt angegeben sondern durch eine *qualitative, linguistische* Beschreibung erfaßt. Diese *linguistischen Terme* zeichnen sich durch eine sprachliche Unschärfe aus und werden infolgedessen mit unscharfen Mengen beschrieben. Das heißt, daß durch die Fuzzy Sets eine Abbildung der linguistischen Werte auf numerische erfolgt. Für die vollständige Beschreibung einer Variablen ist allerdings ein ganzer Satz dieser Fuzzy-Sets notwendig. Die linguistische Variable „Verlustrate“ nach Abbildung 5.2 kann z.

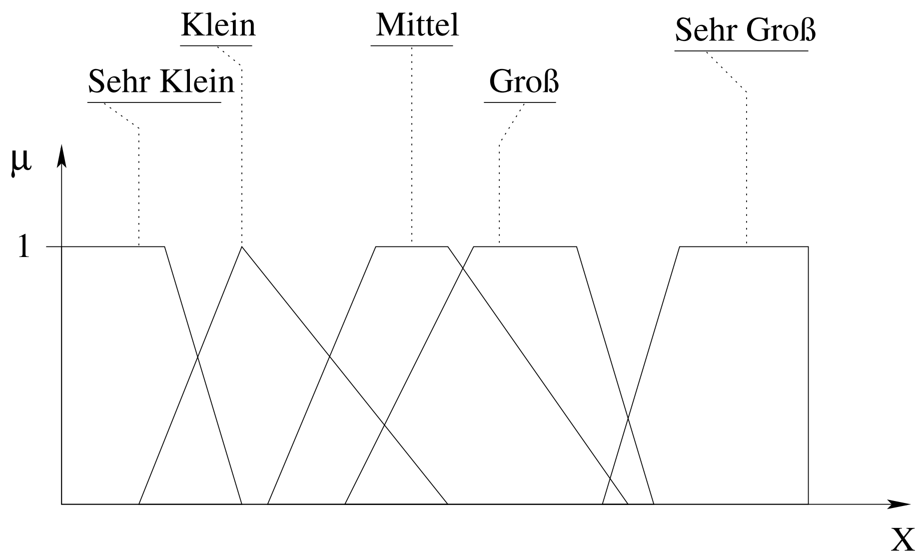



Abbildung 5.2: Beschreibung des Wertebereichs der linguistischen Variablen *Verlustrate* mit den linguistischen Termen *Sehr Klein, Klein, Mittel, Groß, Sehr Groß*

B. mit den Termen „Sehr Klein“ , „Klein“ , „Mittel“ , „Groß“ und „Sehr Groß “ qualifiziert werden. Die Transformation von scharfen numerischen Eingangswerten auf den Bereich der unscharfen linguistischen Terme erfolgt durch die Fuzzyfizierung mit Hilfe der beschriebenen Zugehörigkeitsfunktionen. Die Dimensionierung, d. h. die Festlegung der Anzahl der Terme, sowie die Definition der Kontur der Zugehörigkeitsfunktionen ist Aufgabe des Entwicklers und enthält das Wissen um die Systemgrößen und deren Abhängigkeiten.

### Unscharfes Schließen

Diese Terme werden mit den linguistischen Regeln, die eine qualitative, im Wesentlichen unscharfe Beschreibung eines Prozesses oder Sachverhaltes liefern, verknüpft. Diese Verarbeitungsvorschriften bestehen aus zwei Teilen, der Prämisse und der Schlußfolgerung und bilden eine Verarbeitungsvorschrift entsprechend der in Abbildung 5.3 gezeigten Form. In

***WENN***
( die Verlustrate Klein ) ***UND*** ( die Verzögerung Sehr Klein ist )
***DANN***
( ist die Qualität Groß )

Abbildung 5.3: Unscharfe Regel

dem abgebildeten Zusammenhang werden die linguistischen Variablen Verlustrate und Verzögerung in der Implikation miteinander verkettet. Sie werden durch die unscharfen Terme *Klein* bzw. *Sehr Klein* qualifiziert. Die Verknüpfung dieser Eingangswerte erfolgt ebenfalls durch unscharfe Operatoren, wie z. B. *UND* bzw. *ODER*. Neben diesen Operatoren existieren eine Reihe anderer Verknüpfungsmöglichkeiten [42], deren Einsatz von den jeweiligen Applikationen abhängig ist.
Die resultierende unscharfe Ausgangsgröße ist die Variable Qualität, die den Wert *Groß* annimmt.

### Fuzzy-Inferenz System

Nachdem die Grundlagen kurz umrissen wurden, soll im Folgenden der Ablauf innerhalb eines Fuzzy-Controllers beschrieben werden. Allgemein wird das Übertragungsverhalten des FC durch eine linguistische, qualitative Form gegeben. Erst durch die Definition der eingesetzten Methoden, Parameter sowie dem Aufbau einer applikationsspezifischen Wissensbasis wird dieses Verhalten auch quantitativ festgelegt. Die Abbildung 5.4 gezeigte Inferenzmaschine verarbeitet die Zugehörigkeitsvektoren basierend auf dem vorgegebenen Regelwerk weiter - manipuliert also das gespeicherte Wissen -, um so eine Lösung für das vorliegende Problem zu produzieren. Der in Abb. 5.4 gezeigte Referenzaufbau eines Fuzzy

Controllers gliedert sich in vier Einheiten auf. Die zentrale Wissensbasis besteht aus der applikationspezifischen Datenbasis und dem Regelwerk zur Beschreibung der Zusammenhänge zwischen den unterschiedlichen Systemgrößen. Detailliert enthält sie die Beschreibung der Zugehörigkeitsfunktionen sowohl der Eingangs- als auch der Ausgangsgrößen des Controllers, die Regelbasen, Angaben über Operatoren zur Verknüpfung der unscharfen Mengen und Parameter, die Einfluß auf die Verfahren zur Gewinnung der scharfen Ausgangsgrößen haben. Die Regeln werden in der Inferenz-Einheit benutzt, um die fuzzyfizierten Eingangsgrößen zu bearbeiten. Das immer noch unscharfe Ergebnis wird dann mit Hilfe des Defuzzyfizierers auf einen numerischen Wert abgebildet.

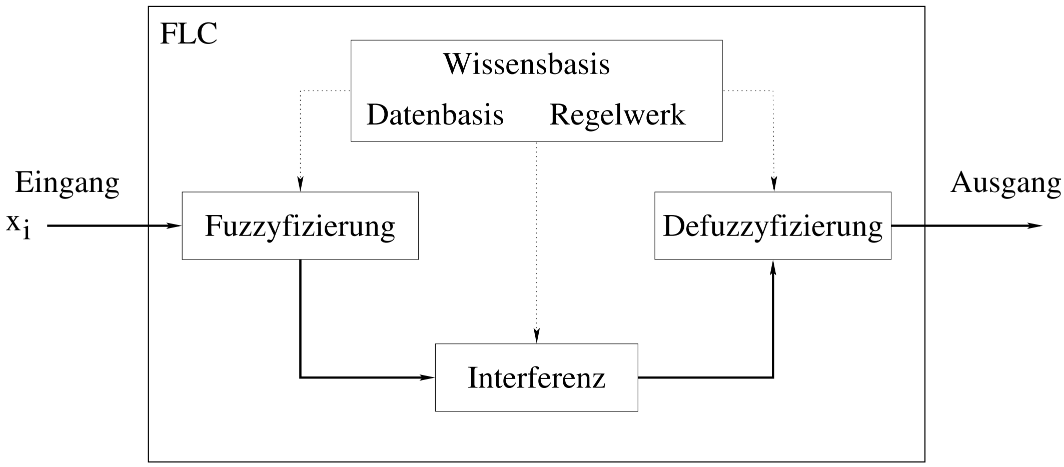



Abbildung 5.4: Prinzipieller Aufbau eines Fuzzy Logic Controllers

**Fuzzyfizierung**
In dem Fuzzyfizierer werden die scharfen Eingangswerte mit Hilfe der Zugehörigkeitsfunktionen (Abb. 5.5) in einen korrespondierenden Fuzzy-Vektor transformiert, der die Zugehörigkeitsgrade zu den einzelnen unscharfen Mengen einer Variablen enthält. Die Dimension dieses Vektors ist durch die Anzahl der linguistischen Terme vorgegeben. Der fuzzyfizierte Eingangswert der Systemgröße Verlustrate „X = Verlustrate “ (Abb.5.5) kann dann wie folgt charakterisiert werden. Für einen Wert $X = x_1$ kann die Verlustrate in einem relativ umfangreichen Bereich von klein bis groß, eher aber als durchschnittlich interpretiert werden. Der Vektor der Zugehörigkeitsgrade hat folgendes Aussehen.

$$\begin{aligned} Z &= (\mu_{SehrKlein}(x_1), \mu_{Klein}(x_1), \mu_{Mittel}(x_1), \mu_{Groß}(x_1), \mu_{SehrGroß}(x_1)) \\ &= (0, 0.24, 1, 0.4, 0) \end{aligned}$$

Die Definition der Zugehörigkeitsfunktionen erfolgt, wie schon früher erwähnt, mit Hilfe der sog. LR-Darstellung. Bei diesem Verfahren werden die Zugehörigkeitsfunktionen aus

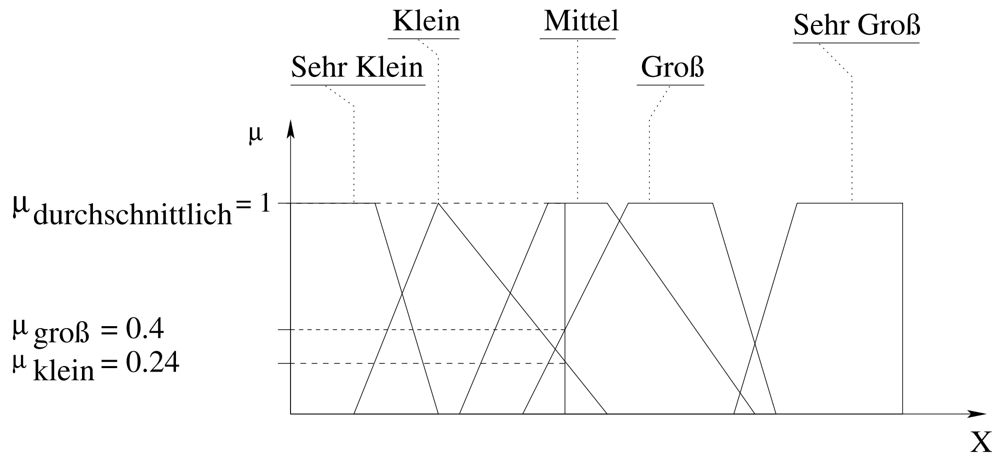



Abbildung 5.5: Fuzzyfizierung eines scharfen Eingangswertes

mehreren Geradenabschnitten, die durch wenige Parameter bestimmt sind, zusammengesetzt, so daß es relativ einfach ist, trianguläre und trapezförmige Funktionen aufzubauen. Die Abbildung 5.6 zeigt die für die Beschreibung von dreieck- sowie trapezförmigen Membership-Funktionen notwendigen Parameter. Die mathematische Darstellung der Zusammenhänge erfolgt mit den Gleichungen 5.5 und 5.6.

$$\mu_{Dreieck} = \begin{cases} L\left(h\frac{\gamma-x}{\gamma-a}\right) & : \ f\ddot{u}r\ x \in [a, \gamma] \\ R\left(h\frac{x-\gamma}{b-\gamma}\right) & : \ f\ddot{u}r\ x \in [\gamma, b] \\ 0 & : \ sonst \end{cases} \tag{5.5}$$

$$\mu_{Trapez} = \begin{cases} L\left(h\frac{\gamma-x}{\gamma-a}\right) & : \ f\ddot{u}r\ x \in [a, \gamma] \\ h & : \ f\ddot{u}rx \in [\gamma, \delta] \\ R\left(h\frac{x-\delta}{b-\delta}\right) & : \ f\ddot{u}r\ x \in [\delta, b] \\ 0 & : \ sonst \end{cases} \tag{5.6}$$

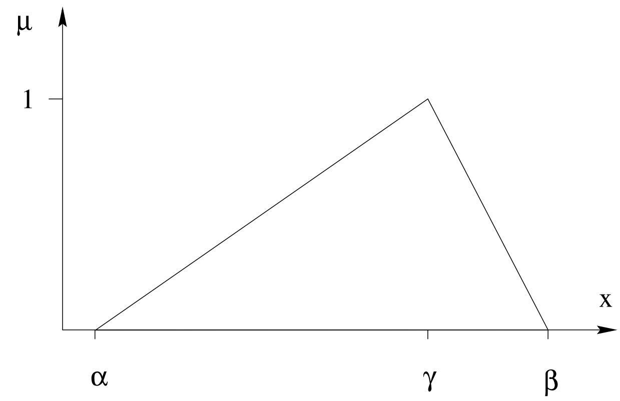
**Inferenz**

In dieser Einheit werden die fuzzyfizierten Eingangssignale mit Hilfe des Regelwerkes ausgewertet. Die grundlegenden Operationen zur Bearbeitung der unscharfen Mengen sind die „Vereinigung“ und der „Durchschnitt“. Die Vereinigung von zwei Mengen A und B resultiert in einer Menge C, die wiederum unscharf ist.

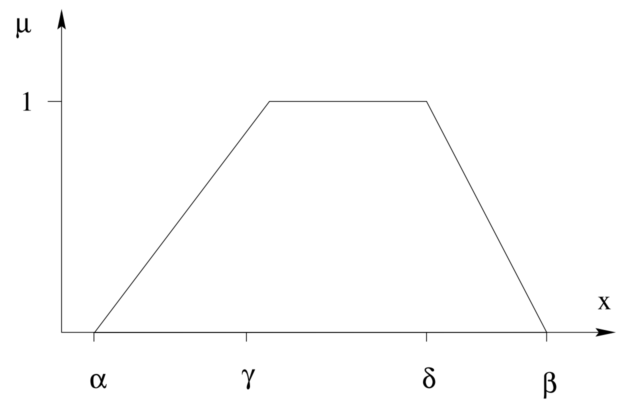
$$C = A \cup B \tag{5.7}$$

Die Berechnung der resultierenden Vereinigungsmenge erfolgt mit Hilfe der Beziehung 5.8:

$$\mu_C(x) = max\{\mu_A(x), \mu_B(x)\}, \forall x \in X \tag{5.8}$$




(a) Dreieck




(b) Trapez

Abbildung 5.6: Kennwerte stückweise linearer Zugehörigkeitsfunktionen

Analog dazu kann der Durchschnitt von zwei Mengen wie folgt beschrieben werden:

$$C = A \cap B \tag{5.9}$$

Die Bestimmung der Durchschnittsmenge erfolgt mit dem Minimum-Operator (5.10):

$$\mu_C(x) = min\{\mu_A(x), \mu_B(x)\}, \forall x \in X \tag{5.10}$$

**Defuzzyfizierung**
Der *Defuzzyfier* faßt die unscharfen Ausgabevariablen zusammen, bewertet sie und ermittelt nach einem vorgegebenen Verfahren aus diesen Werten einen scharfen Ausgabewert des FLC.

$$o_i = defuzz(\mu_i) \tag{5.11}$$

Auch hier existieren unterschiedliche Methoden mit spezifischen Vor- und Nachteilen [42]. Die in dieser Arbeit verwendete Schwerpunktmethode [2] hat die Vorteile, daß sie ein hinreichend glattes Übertragungsverhalten bei einem akzeptablen Rechenaufwand liefert. Für die Ermittlung des Schwerpunktes einer triangulären Zugehörigkeitsfunktion wird Formel 5.12 benutzt, während der Schwerpunkt einer trapezförmigen Zugehörigkeitsfunktion mit Hilfe der Formal 5.13 berechnet wird.

$$COG_{Dreieck} = \frac{\alpha + \beta + \gamma}{3} \tag{5.12}$$

$$COG_{Trapez} = \frac{\beta^2 + \delta^2 - \alpha^2 - \gamma^2 + \beta \cdot \delta - \alpha \cdot \gamma}{3 \cdot (\beta + \delta - \alpha - \gamma)} \tag{5.13}$$

Ist der Schwerpunkt einer Überlagerung mehrerer Fuzzy Sets zu ermitteln, werden die mit dem jeweiligen Erfüllungsgrad $\mu_i$ gewichteten Schwerpunkte $x_i$ aller aktiven Terme aufaddiert und durch die Summe der Erfüllungsgrade dividiert [13]:

$$COG_{Gesamt} = \frac{\sum_i^n x_i \cdot \mu_i}{\sum_i^n \mu_i} \tag{5.14}$$

## 5.2 Entwurf des Fuzzy Controllers

Fuzzy Controller werden zur Realisierung von Erfahrungsstrategien und zur modellfreien[3] Regelung eingesetzt. Fachleute können dann auf Grund ihres Wissens um die Beziehungen zwischen dem Systemzustand und den resultierenden Steuergrößen sowie der Dynamik ohne tiefgründige regelungstechnische Kenntnisse die Zusammenhänge verbal formulieren. Diese Erfahrungen reflektieren sich in der Gestalt der Zugehörigkeitsfunktionen und der Regelbasis.
Im Folgenden soll deshalb der grundsätzliche Ablauf bei der Konzeption eines Fuzzy Reglers skizziert werden (Abb. 5.7).

**Systemanalyse**
In der ersten Phase werden Informationen über das zu steuernde System, in dem vorliegenden Fall über den Zugangsknoten, gesammelt. Die Zusammenhänge zwischen den Systemgrößen werden ebenso wie die Probleme extrahiert und analysiert, was gegebenenfalls zu einer Reduzierung der Wissensbasis führen kann. Dieser Schritt bildet die Grundlage für die Festlegung der Ein- und Ausgangsgrößen sowie der linguistischen Terme und der Regelbasen.

[2] Center of Gravity - COG

[3] Dies bedeutet, daß kein mathematisches Modell des Prozesses erforderlich ist.

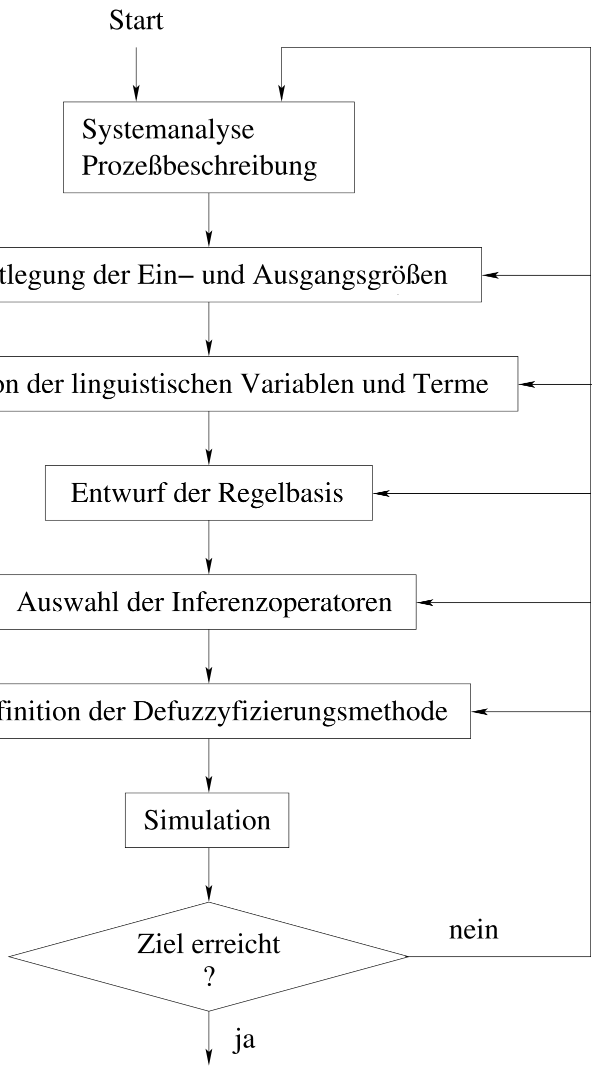



Abbildung 5.7: Entwurfsphasen eines Fuzzy Controllers

**Festlegung der Ein- und Ausgangsgrößen**
Die Anzahl der Ein- und Ausgangsgrößen ist im Wesentlichen durch den Aufbau des Knotenmodells vorgegeben und beschränkt sich auf die Größen, die an den Warteschlangen, am Link und im Multiplexer verfügbar sind. Neben diesen Werten, die direkt aus der Simulation abgeleitet werden können, dienen die Vorgabewerte, die während des Verbindungsaufbaus ausgetauscht werden und die Verkehrslast sowie die QoS beschreiben, als zusätzliche Eingaben. Relevant für den späteren Einsatz ist die Tatsache, daß der erforderliche Realisierungsaufwand maßgeblich von der Anzahl der Eingangsgrößen abhängig ist.

**Definition der linguistischen Variablen und Terme**

Eine wesentliche Aufgabe ist die Bestimmung der Anzahl, Form, Lage und der Einflußbereich der Zugehörigkeitsfunktionen.

Beim Entwurf eines Fuzzy Controllers muß beachtet werden, daß die Anzahl der Freiheitsgrade eines unscharfen Reglers mit der Zahl der linguistischen Terme zunimmt[4]. Je höher die Zahl der Terme ist, umso genauer kann das Systemverhalten angepaßt werden. Allerdings zieht eine große Anzahl von Termen einen erhöhten Aufwand bei der Realisierung nach sich. Der Umfang der Regelbasis wächst exponentiell mit der Anzahl der Terme. Zur Beschreibung des Wertebereichs der Variablen sollte die Anzahl der Terme auf ein Minimum beschränkt werden. In der Praxis übliche Werte liegen zwischen zwei und sieben Terme.

Neben der Anzahl hat auch die Kontur der Membership-Funktionen einen Einfluß auf die Güte des unscharfen Reglers. Es existieren vielfältige Formen, deren Einsatz in Applikationen von der Beschreibungsform abhängig ist. Während die Verwendung von Gauß-förmigen oder quadratischen Fuzzy-Sets (sog. S-Zugehörigkeitsfunktionen) relativ viel Rechnerleistung benötigt, erfordert die Darstellung der Mengen in diskreter Form[5] abhängig vom Detaillierungsgrad, viel Speicherplatz. Für den praktischen Einsatz haben sich trianguläre und trapezförmige Fuzzy-Mengen mit linearen Flanken etabliert, da sie in Bezug auf Rechenzeit und Speicherplatzbedarf eine optimale Lösung [42] darstellen. Neben diesen Vorteilen weisen sie auch eine größere Empfindlichkeit gegenüber Werteschwankungen als etwa Funktionen mit Glockenform auf. Die Auflösung kann einfach über die Einflußbreite, d. h. über die Steilheit der Flanken, eingestellt werden. In den Bereichen, wo eine hohe Auflösung erforderlich ist, zeichnen sich die Zugehörigkeitsfunktionen durch eine kleine Einflußbreite aus.

Darüber hinaus spielen die Überlappung der Funktionen und die Festlegung der Randmengen eine wesentliche Rolle für den Betrieb des Reglers. Die Überdeckung benachbarter Zugehörigkeitsfunktionen hat Einfluß auf die Stabilität des Systems. Bei einer nur schwachen Überschneidung ist vielfach eine Schwingung um den stationären Zustand des Systems zu beobachten. Ist im Gegenteil die Überlappung zu groß, resultiert dies in einem Überschwingen des Systems [53]. Um gute Ergebnisse zu erreichen, muß dem Überdeckungsgrad benachbarter Funktionen Aufmerksamkeit geschenkt werden. Nachdem die Membership-Funktionen festgelegt sind, müssen die Zusammenhänge zwischen den Systemgrößen formuliert werden.

**Entwurf der Regelbasis**

Die Regeln enthalten das eigentliche Wissen über das Verhalten des Reglers. In ihnen sind die Verfahrensanweisungen, abgeleitet aus der Systemanalyse, für den unscharfen Regler festgeschrieben.

Die Anzahl der Regeln steigt exponentiell mit der Zahl der linguistischen Terme, so daß

[4] Die Unschärfe des Systems wird jedoch verringert, weil der gesamte Zustandsraum detaillierter erfaßt werden kann.

[5] Die Zugehörigkeitsfunktionen sind in Form von „look up-Tabellen“ im Speicher abgelegt

unmittelbar deutlich wird, daß es nahezu unmöglich ist, einen Regelraum, der von mehr als zwei Eingangsgrößen aufgespannt wird, auszunutzen. Um die Beziehungen zwischen den Systemgrößen eindeutig und überschaubar zu gestalten, werden in dieser Untersuchung nur hierarchisch strukturierte Fuzzy Controller mit zwei Eingängen und einem Ausgang verwendet. Die Anzahl der Regeln beträgt dann, wenn vorausgesetzt wird, daß die Variablen mit fünf Termen beschrieben werden, *25*.
Mit diesem Ansatz ist es möglich, eine *vollständige*[6], *konsistente*[7] Regelbasis *ohne Redundanzen*[8] aufzubauen.

**Auswahl der Inferenzoperatoren**
Für die UND und ODER Verknüpfungen werden hier nur die MIN- bzw. MAX-Operatoren eingesetzt.

**Definition der Defuzzyfizierungsmethode**
Zur Ermittlung der scharfen Stellgröße am Ausgang des Reglers, muß eine geeignete Defuzzyfuzzierungsmethode festgelegt werden. Bei den untersuchten Verfahren wurde die Flächenschwerpunktmethode eingesetzt. Hierbei dient die Projektion des Schwerpunktes der Ausgangsfuzzymenge auf die Abszisse als scharfe Stellgröße.
Ausschlaggebend für die Wahl dieses Verfahrens war, daß der Rechenzeitaufwand gegenüber anderen Verfahren relativ gering ist. Weiterhin haben Untersuchungen [12] gezeigt, daß diese Transformationsstrategie ein sehr stabiles Reglerverhalten[9] [10] aufweist.

## 5.3 Der Fuzzy Logic basierte Policing Controller

Nach der groben Beschreibung der grundlegenden und für die Arbeit relevanten Abläufe der Fuzzy Logic soll in dem folgenden Kapitel ein Policing Controller vorgestellt werden, der durch den Einsatz eines Fuzzy Controllers realisiert wird.

### 5.3.1 Definition der Zielsetzung

Die Systembeschreibung nach Abschnitt 3.2 bildet die Grundlage für die Entwicklung einer Regelstrategie. Auf Grund der beschriebenen Zusammenhänge umfaßt die Aufgabe der Regelstrategie folgende Anforderungen:

- Verminderung der systemweiten Paketverlustrate

[6] Zu jeder Kombination von Ein- und Ausgabewerten existiert mindestens eine aktive Regel
[7] Es wird angenommen, daß die Regelbasis frei von Widersprüchen ist.
[8] Auf Grund der überschaubaren Größe wird die Möglichkeit, daß eine Regel mehrmals definiert wird, minimiert.
[9] Bei den Berechnungsschritten kommt es nur zu geringen Sprüngen der Stellgrößen
[10] Die Flächenschwerpunktmethode zeigt auch bei widersprüchlichen Reglen ein stabileres Verhalten als andere Defuzzyfizierungsmethoden

- Begrenzung der Paketverluste in Abhängigkeit von der Dienstpriorität
- Optimale Nutzung der Speicherkapazität des Zugangsknotens
- Bestmögliche Auslastung des Links

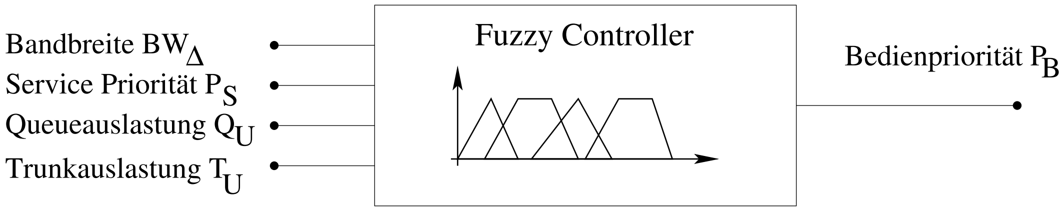



Abbildung 5.8: Fuzzy Policing Controller

Um diesen Anforderungen gerecht zu werden, ist die Stellgröße des Fuzzy Controllers sowohl von dem lokalen Knotenzustand als auch von der temporären Lastcharakteristik der Dienste abhängig. Das Ausgangssignal des Controllers (Abb. 5.8), die Bedienpriorität $P_B$, gibt dabei an, mit welcher Priorität eine Warteschlange bedient werden soll. Stehen in mehreren Warteschlangen Daten zur Übertragung an, entscheidet die höchste Wahrscheinlichkeit nach dem Prinzip „Winner Takes It All“ , welcher Dienst dann bevorzugt behandelt wird. Diese Information wird aus den lokalen Zustandsinformationen wie der Überschreitung der deklarierten Bandbreite $BW_\Delta$, der Dienstpriorität $P_S$ sowie der Auslastung der Warteschlangen $Q_U$ und der Belegung der Übertragungsbandbreite $T_U$ abgeleitet.
Wie oben beschrieben stellt neben der Wahl der Ein- und Ausgangsgrößen die Regelbasis einen zentralen Bestandteil des Fuzzy Controllers dar. Mit Hilfe der Regeln werden die Beziehungen zwischen den einzelnen Parametern festgelegt. Um die Dimension handhabbar für Simulationen zu halten und um Abhängigkeiten zwischen den einzelnen Parametern hervorzuheben und die Zusammenhänge anschaulich und nachvollziehbar zu gestalten, wurde ein *hierarchisch strukturierter Fuzzy Controller* eingesetzt.
Bei einem konventionellen, *eben* aufgebauten System mit vier Eingangsvariablen, die jeweils durch fünf bzw. drei Fuzzy Sets qualifiziert werden und einer Ausgangsvariablen, die ebenfalls durch fünf linguistische Terme definiert wird, würde die Regelbasis mit Hilfe einer Hypermatrix mit insgesamt $Z_{eben} = 5 \cdot 5 \cdot 5 \cdot 3 = 375$ Zuständen beschrieben werden.

### 5.3.2 Struktur des Policing Controllers

Bei dem hier verwendeten Ansatz (Abb. 5.9) setzt sich der Policing-Controller aus drei Fuzzy-Controllern ($FC_1$, $FC_2$ und $FC_3$) zusammen, die die unterschiedlichen Eingangsgrößen miteinander verknüpfen. Die Regelbasen können in diesem Fall durch *drei zweidimensionale Matrizen* beschrieben werden. Zwei dieser Matrizen haben einen Umfang von jeweils $5 \cdot 5 = 25$ Zuständen, die dritte des in Abbildung 5.9 dargestellten Controllers $FC_3$

kann durch nur $5 \cdot 3 = 15$ Zustände[11] vollständig erfaßt werden. Der gesamte Zustandsraum umfaßt $Z_{hierarchisch} = 65$ Zustände. Durch Anwendung einer hierarchischen Struktur

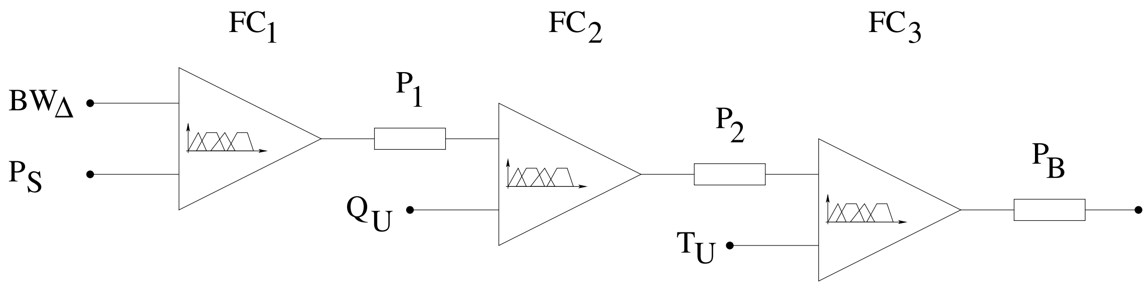



Abbildung 5.9: Dreistufiger Aufbau der Fuzzy Logic basierten Policing Controllers

ergibt sich eine Einsparung an Zuständen von ca. 83 % gegenüber einem ebenen Aufbau. Diese doch erhebliche Reduktion schlägt sich im Wesentlichen auf den Speicherbedarf und die Verarbeitungsgeschwindigkeit sowie auf die Überschaubarkeit und Handhabbarkeit des Fuzzy Controllers nieder. Weitere Punkte betreffen die Konsistenz der Regeln sowie die Sicherstellung der Abdeckung aller denkbaren Zustände des Prozesses.

Als Eingangsgrößen wurden in diesem Ansatz die folgenden Variablen gewählt.

$BW_\Delta$

$BW_\Delta$ ist ein Maß für die Abweichung der aktuellen Bandbreite von der deklarierten Bandbreite. Die Berechnung der Eingangsgröße $BW_\Delta$ erfolgt mit der Beziehung 5.15.

$$BW_\Delta = \begin{cases} \frac{BW - BW_{decl}}{BW} & : \; f\ddot{u}r \; BW - BW_{decl} > 0 \\ 0 & : \; sonst \end{cases} \tag{5.15}$$

$BW$ ist die aktuelle Datenrate, die am Eingang der Warteschlange bestimmt wird. $BW_{decl}$ stellt, die im Verkehrsvertrag zu Beginn der Übertragung ausgehandelte Bandbreite dar. Durch die Anwendung der Beziehung 5.15 ist sichergestellt, daß $BW_\Delta$ immer im Intervall $[0, 1]$ liegt. Ist $BW_\Delta > 0$, wird der Dienst nicht vertragskonform betrieben. $BW_\Delta = 0$ impliziert, daß die deklarierte Bandbreite eingehalten wird.

$P_S$

$P_S$, ein Wert zwischen 0 und 1, stellt eine applikationsspezifische statische Priorität dar. Sie berücksichtigt im Wesentlichen die stringente Abhängigkeit eines Dienstes von den relevanten Übertragungs- und Qualitätsparametern.

[11] Die Auslastung der Übertragungsressourcen des Kanals wird nur durch drei linguistische Terme beschrieben

$P_1, P_2$

$P_1$ und $P_2$ sind interne Hilfsgrößen.

$Q_U$

Die Auslastung der Warteschlange $Q_U$ ergibt sich aus dem Verhältnis der aktuellen Belegung der Warteplätze zur maximalen Länge der Warteschlange

$$Q_U = \frac{Q_{aktuell}}{Q_{max}} \tag{5.16}$$

Für den Wertebereich dieser Variablen gilt: $Q_U \in [0, 1]$.

$T_U$

$T_U$ beziffert die Auslastung des abgehenden Übertragungskanals. Sie berechnet sich aus der aktuellen genutzten Bandbreite und der maximalen Datenrate des Kanals.

$$T_U = \frac{T_{aktuell}}{T_{max}} \tag{5.17}$$

Für den Wertebereich dieser Variablen gilt: $T_U \in [0, 1]$.

$P_B$

Der Ausgangswert des Controllers $P_B$ gibt an, mit welcher Priorität eine Warteschlange bedient werden soll. Stehen in mehreren Warteschlangen Daten zur Übertragung an, werden die assoziierten Prioritäten $P_B$ bestimmt. Ein großer Wert bedeutet dabei, daß diese Daten vorrangig zu behandeln sind.

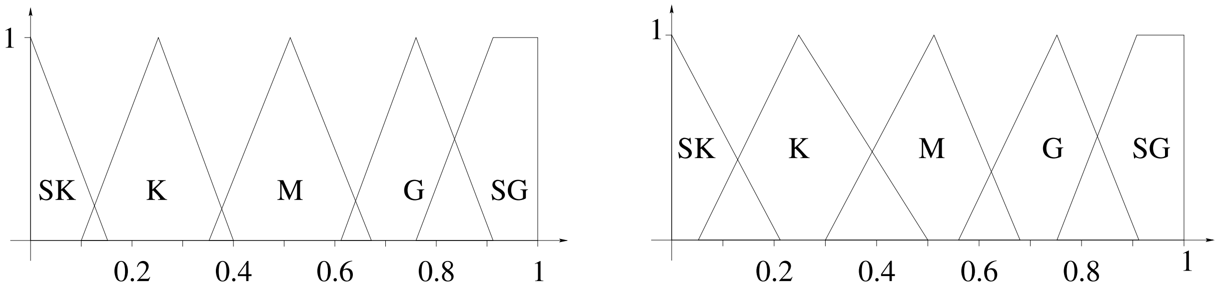
### 5.3.3 Linguistische Terme und Zugehörigkeitsfunktionen

Zur vollständigen Beschreibung des Reglers und seines Verhaltens müssen die Fuzzy-Variablen durch linguistische Werte mit den assoziierten Zugehörigkeitswerten charakterisiert werden. Die linguistischen Terme für die drei Eingangsvariablen $P_S$ (Abb. 5.10(a)), $BW_\Delta$ (Abb. 5.10(b)) und $Q_U$ (Abb. 5.11(a)) sowie der beiden temporären Prioritäten $P_1$, $P_2$ und der Ausgangsgröße $P_B$(Abb. 5.11(b)) werden mit Hilfe von fünf Sets differenziert. Die Termbasis setzt sich aus den Termen

Sehr Klein (SK) - Klein (K) - Mittel (M) - Groß (G) - Sehr Groß (SG)

zusammen. Unterschiede bestehen lediglich in der Lage und der Einflußbreite.

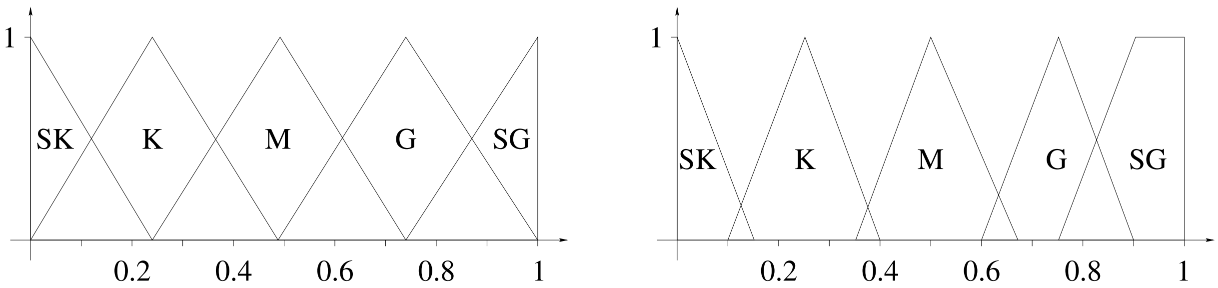
Der Auslastung des Kanals $T_U$ (Abb. 5.12) sind drei Terme zugeordnet. Es wird unterschieden, ob der Kanal nur geringfügig genutzt wird, so daß die Lastsituation als nicht kritisch (NK) eingeschätzt werden kann. Daneben existieren der normale Betriebsbereich




(a) Dienstpriorität $P_S$

(b) Abweichung von der deklarierten Bandbreite $BW_\Delta$

Abbildung 5.10: Darstellung der linguistischen Variablen $P_S$ und $BW_\Delta$




(a) Auslastung $Q_U$

(b) $P_1$,$P_2$ und $P_B$

Abbildung 5.11: Darstellung der linguistischen Variablen $Q_U$, $P_1$ und $P_2$

(N) und ein Zustand mit einer außerordentlich hohen Belastung der Ressourcen, der als kritisch (K) bezeichnet werden kann. Auffallend ist die große Einflußbreite des Terms NK, der so gewählt wurde, weil der Betrieb des Kanals in diesem Bereich weit unterhalb des optimalen Arbeitspunktes liegt. Eine hohe Auflösung ist hier nicht erforderlich.

Bei dem vorliegenden Entwurf finden ausschließlich trapez- und dreieckförmige Zugehörigkeitsfunktionen Anwendung. Trapezförmige Sets werden bei diesem ersten Ansatz nur an den Rändern der linguistischen Variablen benutzt. Der Wahl der Form und Parameter $\alpha, \beta, \gamma$ sowie $\delta$ der linguistischen Terme liegen die Erfahrung des Designers des Fuzzy Controllers zugrunde.

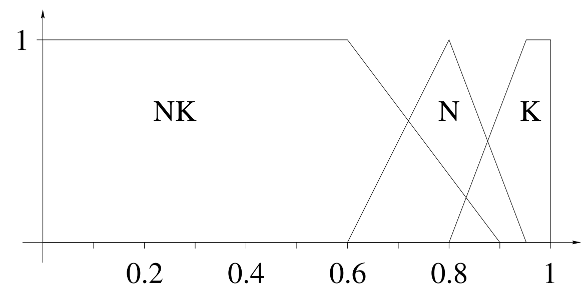



Abbildung 5.12: Linkauslastung $T_U$

### 5.3.4 Regelbasen

Nach der Beschreibung der linguistischen Terme und der Zugehörigkeitsfunktionen im vorausgegangenen Kapitel wird im folgenden Abschnitt das Regelverhalten des dreistufigen Controllers festgelegt. Die Definition erfolgt mit den in den Tabellen 5.1 - 5.3 beschriebenen Beziehungen zwischen den Systemgrößen.

### 5.3.5 Fuzzy Controller 1

Tabelle 5.1 zeigt die Regelbasis des Controllers $FC_1$. Im Fuzzy Controller 1 wird mit $BW_\Delta$

| $P_1$ | | $BW_\Delta$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_S$ | SK | SK | SK | SK | SK | SK |
| | K | K | K | SK | SK | SK |
| | M | M | M | K | SK | SK |
| | G | G | G | M | K | SK |
| | SG | SG | SG | G | M | K |

Tabelle 5.1: Regelbasis für FuzzyController $FC_1$

und der für den Dienst charakteristischen konstanten Priorität $P_S$ die temporäre Priorität $P_1$ bestimmt. Die Verknüpfung der Eingangsgrößen erfolgt mit dem „UND-Operator", so daß Regeln in der folgenden Form zu lesen sind.

$$\text{WENN } (BW_\Delta == Mittel) \text{ UND } (P_S == Klein)$$
$$\text{DANN } (P_1 == SehrKlein)$$

Aus der Tabelle 5.1 kann abgeleitet werden, daß eine sehr kleine Abweichung von der deklarierten Bandbreite keinen Einfluß auf die Priorität $P_1$ hat. Anders bei größeren Abweichungen. In diesem Fall ist die temporäre Priorität $P_1$ immer kleiner als die Dienstpriorität. Die Höhe der Abstufung hängt von dem Ausmaß der Überschreitung der deklarierten Bandbreite ab. Genauere Zusammenhänge sind aus dem Kennlinienfeld ersichtlich.

#### Kennfelddarstellung des Übertragungsverhaltens des Fuzzy Controllers 1

In Abb. 5.13 ist der Verlauf der temporären Priorität $P_1$ in Abhängigkeit von den beiden Eingangsgrößen $P_S$ und der relativen Bandbreitenüberschreitung $BW_\Delta$ dargestellt. Die gezeigte Ausgangsgröße $P_1$ wird für die Darstellung mit der Schwerpunktmethode nach Gleichung 5.14 ermittelt. Fehler und Ungenauigkeiten, die durch die Defuzzyfizierung einfließen, sind für den Betrieb vernachlässigbar, da die internen Signale nur in fuzzyfizierter

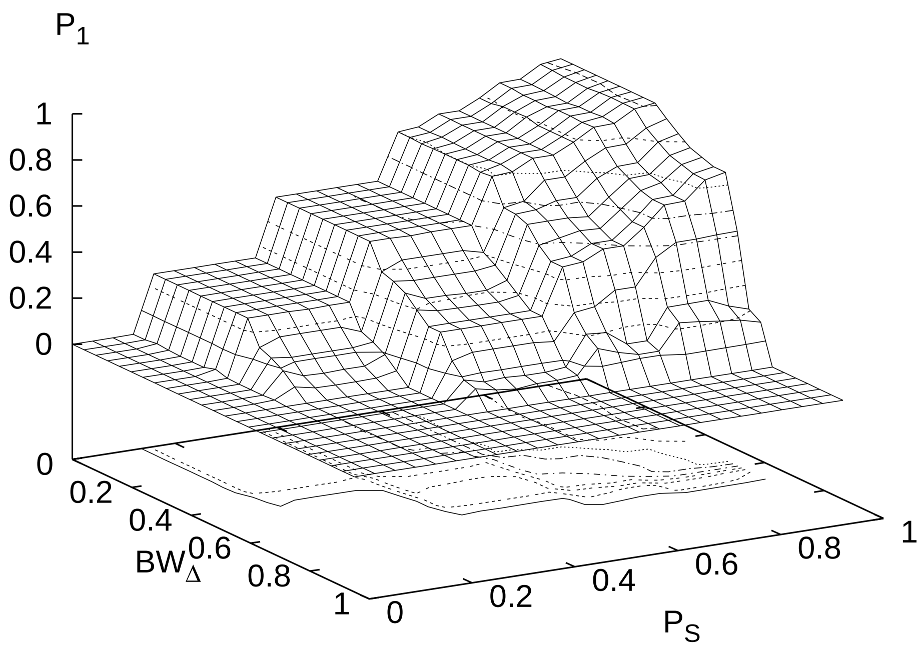



Abbildung 5.13: Abhängigkeit der temporären Priorität $P_1$ von der Dienstpriorität $P_S$ und der Bandbreitenüberschreitung $BW_\Delta$

Form vorliegen und auch verarbeitet werden.

Das ermittelte Kennlinienfeld weist Plateaus auf, die mit ansteigender Dienstpriorität durch steile Übergänge[12] gekennzeichnet sind. Die Änderung von $P_1$ in Abhängigkeit von $BW_\Delta$ vollzieht sich mehr oder weniger fließend.
An den markanten Knickpunkten kann man Diskontinuitäten erkennen, die das Reglerverhalten negativ beeinflussen können. Gemäß der Regeln in Tabelle 5.1 ist im Bereich großer $BW_\Delta$ und für kleine Prioritäten $P_S$, $P_1 = 0$. In den übrigen Gebieten ist die Funktion $P_1 = f(BW_\Delta)$ für ein gegebenes konstantes $P_S$ monoton. Auffallend sind die ausgeprägten Plateaus. In diesen Bereichen reagiert der Fuzzy Controller auf Grund seiner Unschärfe nicht auf Änderungen der Eingangsgrößen. Für diese Fälle muß geprüft werden, ob nicht Oszillationen oder Verschiebungen, die eine Instabilität zur Folge haben, auftreten können.

Die Auswertung der Kennlinienfelder ist nicht trivial und bietet vielfach die Möglichkeit zur Fehlinterpretation. Probleme, die beim Design von Fuzzy Controllern einen wesentli-

[12]Solche Felder sind in der Literatur als sog. mehrdimensionale Multirelaischarakteristika bekannt.

chen Einfluß haben, betreffen die Stabilität des Algorithmus und die vollständige Erfassung des Regelraumes. Die Auswirkungen schlagen sich dann in unterschiedlicher Ausprägung in diesen Diagrammen nieder.
Eine Möglichkeit, um sicherzustellen, daß der Regelraum vollständig ausgeschöpft wird, ist, daß zur Erzeugung des Kennlinienfeldes dem Fuzzy Controller zufällig ermittelte Eingangstupel $(BW_\Delta, P_S)$ aus dem zulässigen Wertebereich übergeben werden. Bei Anwendung dieses Verfahrens ergab sich auch nach mehrfacher Wiederholung dieselbe, eindeutige Kennliniencharakteristik. Das bedeutet, daß das Übertragungsverhalten nur von den aktuellen und nicht von Werten, die in der Vergangenheit ermittelt wurden, abhängig ist. Dieses statische Verhalten ist ein Hinweis für die vollständige Erfassung der Betriebszustände mit dem aufgestellten Regelwerk. Die Stabilität ist, wie oben erwähnt, ein weiteres Problemfeld dieser Regler und muß, gegebenenfalls, gesondert betrachtet werden.

### 5.3.6 Fuzzy Controller 2

Mit dem $FC_2$ wird aus der temporären Priorität $P_1$ und der Auslastung der spezifischen Warteschlange $Q_U$ die temporäre Priorität $P_2$ ermittelt. In diesem Teilcontroller sorgt eine

| $P_2$ | | $Q_U$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_1$ | SK | SK | SK | K | M | M |
| | K | K | K | K | M | M |
| | M | M | M | M | G | G |
| | G | M | M | G | G | SG |
| | SG | M | M | G | SG | SG |

Tabelle 5.2: Regelbasis für Fuzzy Controller $FC_2$

sehr kurze Warteschlange für ein Absinken der Priorität $P_2$ im Vergleich zur temporären Priorität $P_1$. Bei einer höheren Auslastung der verfügbaren Warteplätze steigt $P_2$, um eine Bedienung der Warteschlange zu forcieren.

#### Kennfelddarstellung des Übertragungsverhaltens des Fuzzy Controllers 2

In Abb. 5.14 ist die Abhängigkeit der temporären Priorität $P_2$ von $P_1$ und $Q_U$ dargestellt. Der Verlauf bestätigt das beschriebene Verhalten des Controllers. Bei einer geringen Auslastung der Warteschlangen ($Q_U < 0.2$) und für $P_1 < 0.8$, ist $P_2 \leq P_1$.
Eine gut ausgelastete Warteschlange hingegen führt zu einem kontinuierlichen Anstieg der Priorität $P_2$.

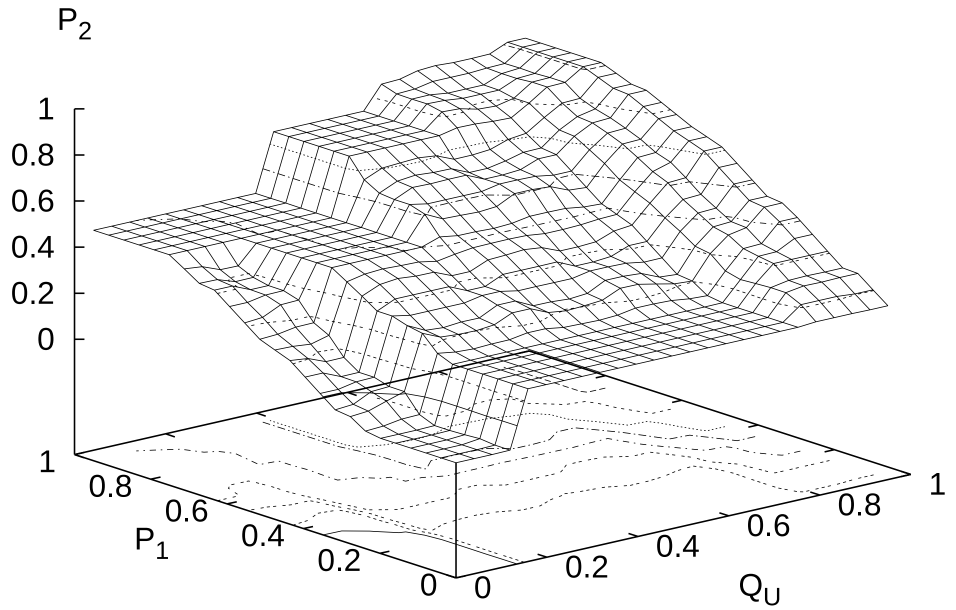



Abbildung 5.14: Abhängigkeit der temporären Priorität $P_2$ von der Priorität $P_1$ und der Auslastung der Dienstwarteschlange $Q_U$

### 5.3.7 Fuzzy Controller 3

Aus der temporären Priorität $P_2$ und der Auslastung des Kanals $T_U$ wird im Teilcontroller $FC_3$ die Ausgangsgröße $P_B$, die festlegt, mit welchem Nachdruck eine Warteschlange bedient werden sollte, ermittelt. Für das Regelverhalten dieses Teilcontrollers gilt, daß bei einem Betrieb des Kanals im unkritischen bzw. normalen Bereich die Bedienpriorität mit einer temporären Priorität $P_2$, die die Werte „Sehr Klein“ und „Klein“ annimmt, angehoben wird. Erst im kritischen Betriebsbereich des Links wird die Bedienpriorität dieser Dienste herabgesetzt, so daß nur Dienste, die sich durch eine „Sehr Große“ Priorität $P_2$ auszeichnen, abgearbeitet werden.

#### Kennfelddarstellung des Übertragungsverhaltens des Fuzzy Controllers 3

Abbildung 5.15 zeigt das Regelverhalten des Fuzzy Controllers 3, dessen Regelbasis in Tabelle 5.3 wiedergegeben ist. Bei niedriger bzw. normaler Linkauslastung kann man im Wesentlichen zwei Verhaltensweisen ablesen. Ist die Priorität $P_2 \leq 0.45$, ist $P_B = const. = 0.45$. Im zweiten Bereich ($P_2 > 0.45$) steigt $P_B$ als eine Funktion von $P_2$ monoton an.
Wird der Kanal dagegen im kritischen Bereich $T_U > 0.7$ betrieben, führt das bis auf den

| $P_B$ | | $T_U$: NK | $T_U$: N | $T_U$: K |
|---|---|---|---|---|
| $P_2$ | SK | M | K | SK |
| | K | M | K | SK |
| | M | M | M | K |
| | G | G | G | M |
| | SG | SG | SG | SG |

Tabelle 5.3: Regelbasis für Fuzzy Controller $FC_3$

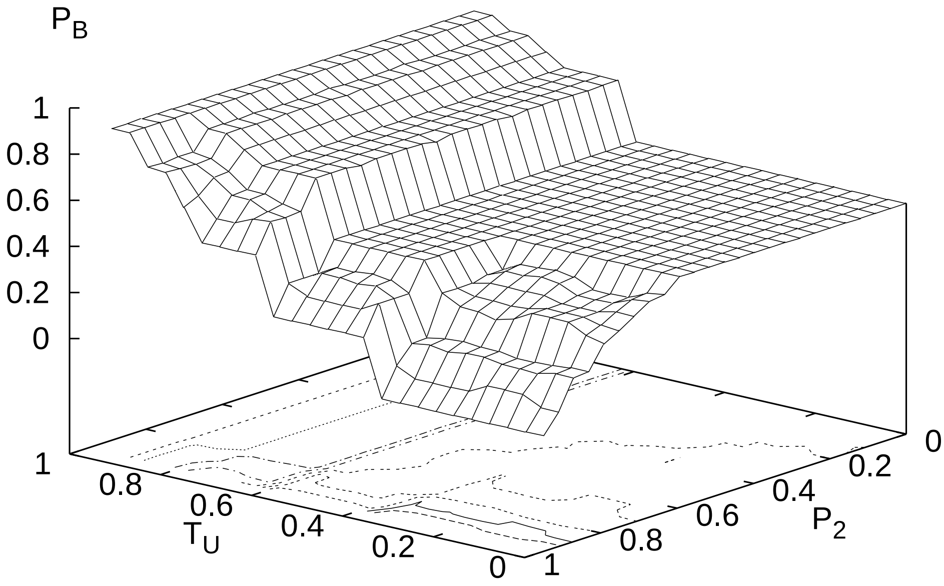



Abbildung 5.15: Abhängigkeit der temporären Bedienpriorität $P_B$ von der Priorität $P_2$ und der Auslastung des Kanals $T_U$

Fall, daß der Wert von $P_2$ größer 0.85 ist, zu einer Absenkung der Dringlichkeit.
Auch diese Darstellung zeigt ein großes ausgedehntes Plateau, das sich auf die große Einflußbreite des linguistischen Terms „Nicht Kritisch“ der Variablen $T_U$ zurückführen läßt.

**Allgemeines**

In alle drei Regelbasen wurden die linguistischen Variablen unter Anwendung des „UND-Operators“ miteinander verknüpft. Die Defuzzyfizierung erfolgte unter Anwendung der Schwerpunktmethode nach Gleichung 5.14.

## 5.4 Simulation des hierarchischen Fuzzy Controllers

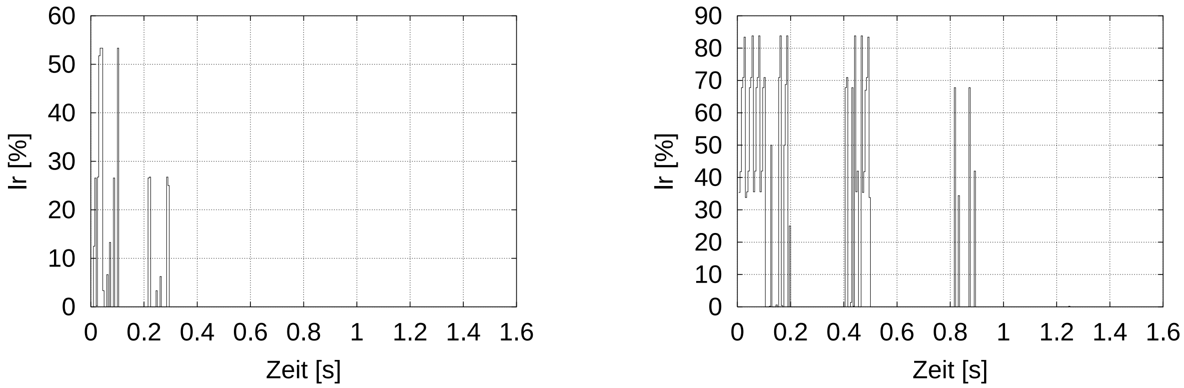
In diesem Abschnitt wird der entwickelte Policing Controller mit Hilfe des in Kapitel 3 beschriebenen Knotenmodells evaluiert. Als Eingangsdaten für den implementierten Controller wird die in Abb. 4.1 und durch Tabelle 4.1 spezifizierte Verkehrslast eingeprägt. Die Abschätzung der Güte des Controllers erfolgt an Hand der jeweiligen Verlustraten der Dienste und Auslastung der assoziierten Warteschlangen. Daneben werden die temporären Prioritäten $P_1$, $P_2$ sowie die Bedienpriorität analysiert, da deren Verlauf die Abhängigkeit und den Einfluß der verschiedenen Eingangsgrößen auf das Regelverhalten des Fuzzy Controllers reflektiert.
Die Konfidenzintervalle der in dem folgenden Kapitel gezeigten Ergebnisse betragen $\leq 10\%$ der Mittelwerte und sind wegen der Übersichtlichkeit nicht dargestellt.

### 5.4.1 Verlustraten

Die Abbildungen 5.16 und 5.17 zeigen die Verlustraten der Dienste 3 - 5. Die Dienste 1 und 2 haben über den gesamten Simulationszeitraum eine konstante Verlustrate von 0%. Es treten Verlustspitzen zwischen 50% und 95% auf. Die Verluste bei Dienst 3 belaufen sich, obwohl vertragskonform, wegen seiner geringen Priorität von 0.3 auf bis zu 50%. Diese niedrige Priorität bedeutet, daß nur eine geringe Bindung an die Vertragsparameter vorliegt bzw. daß in diesem Fall die Parameter mit einem größeren Toleranzintervall ausgestattet sind.
Weiterhin fällt auf, daß das Muster des Verlaufs aller Verlustkurven stark durch den Dienst 5 geprägt ist. Dieser Dienst arbeitet mit einer höheren Priorität und einer größeren Bandbreite als die Dienste 3 und 4, so daß die Auslastung der Warteschlange in Verbindung mit der großen Priorität dazu führt, daß dieser Dienst relativ häufig bedient wird. Die übrigen Warteschlangen werden in diesen Aktivitätsphasen stärker ausgelastet.
Aus diesem Zusammenhang läßt sich ableiten, daß die Zielsetzung des Controllers, die Dienste in Abhängigkeit von der adaptiv ermittelten Priorität zu bedienen, umgesetzt werden konnte.

### 5.4.2 Auslastung der Warteschlangen

Die Auslastung der unterschiedlichen Warteschlangen ist in den Abbildungen 5.18(a) bis 5.19(b) und 5.19(c) dargestellt. Es zeigt sich, daß diese Ressourcen zur Steuerung der Datenströme doch in erheblichem Umfang genutzt werden. Auch hier manifestiert sich mit Nachdruck, daß Dienst 5 einen maßgeblichen Einfluß auf die systemweite Auslastung hat.




(a) Dienst 3 (b) Dienst 4

Abbildung 5.16: Verlustraten

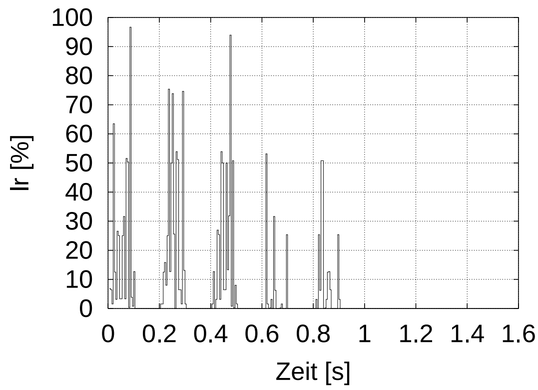
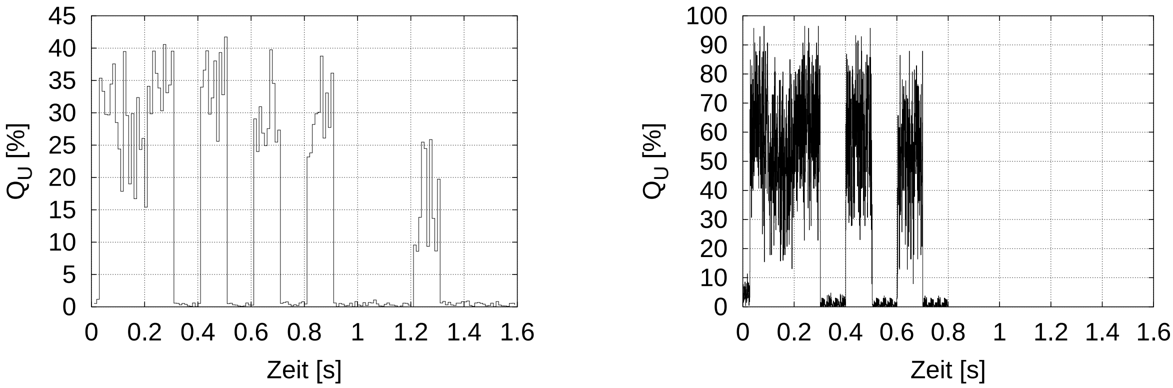



Abbildung 5.17: Verlustrate des Dienstes 5

Die Lastkurve aller Ressourcen weist dieselbe Periode auf wie die Verkehrscharakteristik von Dienst 5. Die maximale Belastung der Warteschlangen der Dienste 1 und 2 beläuft sich auf 42% bzw. 95%. Es treten keine Verluste auf. An den übrigen Warteschlangen sind transiente Überlastungen vorhanden, so daß Daten verworfen werden müssen und infolgedessen Verluste auftreten.

### 5.4.3 Temporäre Priorität $P_I$

Um das Verhalten des Fuzzy Controllers noch genauer analysieren zu können, ist die Betrachtung der temporären inneren Zustandsgrößen unerläßlich. In Abbildung 5.20 verweisen die Pfeile auf die Arbeitspunkte, die sich bei der vorgegebenen Priorität $P_S$ und der Übertragungsrate bzw. der Abweichung von der vereinbarten Übertragungsrate einstellen. Auf

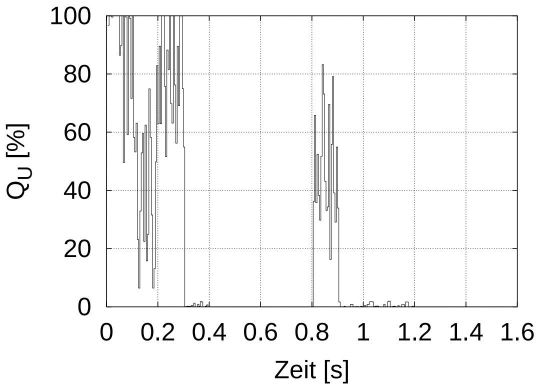



(a) Dienst 1 (b) Dienst 2

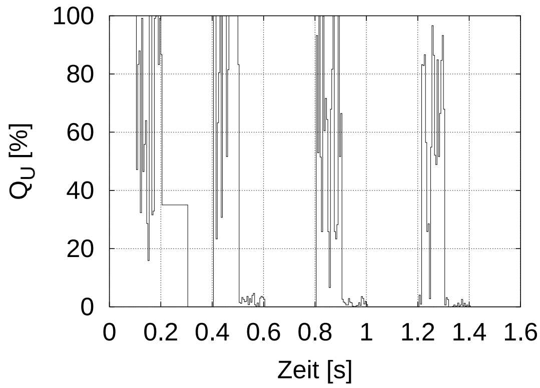
Abbildung 5.18: Auslastung der Warteschlangen

Grund des speziellen Verkehrsmusters (Abbildung 4.1 in Verbindung mit Tabelle 4.1) ergeben sich für alle Dienste während des gesamten Simulationsintervalls feste Arbeitspunkte[13], so daß ein Vergleich mit der spezifischen Priorität möglich ist. Die entsprechenden Ausgangswertewerte $P_1$ wurden dem Graphen entnommen und in Bild 5.21 den korrespondierenden Prioritäten $P_S$ der einzelnen Dienste gegenübergestellt. Die Dienste 1 und 3 verhalten sich vertragskonform, so daß in diesem Fall $P_1$ nur von der festen Priorität $P_S$ abhängt. Aus Abb. 5.20 kann entnommen werden, daß sich $P_{1,Dienst1}$ auf einen Wert von $\approx 0.9$ und $P_{1,Dienst3}$ sich auf einen Wert von 0.25 einstellt. Hier ergibt sich eine Absenkung der Priorität, was für das Auftreten der Datenverluste verantworlich ist. Bei den übrigen Diensten erfolgt eine Bewertung in Abhängigkeit von der Priorität und der Überschreitung der deklarierten Bandbreite. Die Priorität von Dienst 2 wird, obwohl eine Überschreitung der Bandbreite von $\approx 29\%$ vorliegt, nicht heruntergestuft. Mit dieser im Vergleich zu den anderen Diensten doch hohen resultierenden Beurteilung ($P_{1,Dienst2} = 0.5$), die maßgeblich in die weitere Bearbeitung einfließt, ist zu erklären, daß Dienst 2 keine Verluste erleidet. Die Prioritäten der Dienste 4 und 5 werden auf Grund massiver Überschreitungen der deklarierten Bandbreite von 60% ($S_4$) und $\approx 54\%$ ($S_5$) herabgestuft. Der resultierende Wert für $P_{1,Dienst4}$ beläuft sich auf Grund dessen auf 0, $P_{1,Dienst5}$ stellt sich wegen der hohen Dienstpriorität ($P_S = 0.6$) auf 0.25 ein.
Mit Hilfe dieser Priorität und der Auslastung der Warteschlangen $Q_U$ erfolgt die Berechnung der temporären Priorität $P_2$.

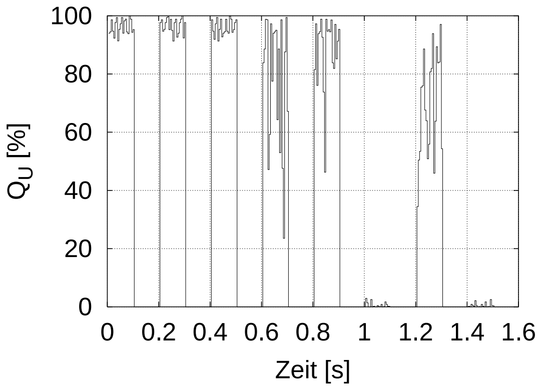
[13]Es kommen bei der Simulation nur Dienste mit *zwei* Zuständen zum Einsatz. Entweder ist die Quelle aktiv und sendet Daten mit der maximalen Rate oder sie ist in Ruhe. Im letzten Fall stehen keine Daten zur Übertragung an, so daß er für den Entscheidungsprozeß, welche Warteschlange bedient werden soll, keine Relevanz hat. In diesem Fall werden jedoch, weil unwirksam, keine neuen Kenngrößen berechnet. Der letzte berechnete Wert wird beibehalten und protokolliert, so daß sich ein fester Arbeitpunkt einstellt




(a) Dienst 3




(b) Dienst 4




(c) Dienst 5

Abbildung 5.19: Auslastung der Warteschlangen

### 5.4.4 Temporäre Priorität $P_2$

Die Abbildungen 5.22, 5.23 und 5.24 zeigen den Verlauf der temporären Priorität $P_2$. Die Bestimmung dieser internen Größe beruht auf der Auswertung der temporären Priorität $P_1$ und der Auslastung der Warteschlangen $Q_U$. Im Wesentlichen spiegelt sich hier der Verlauf der Auslastung der Warteplätze der einzelnen Dienste wider. Eine hohe Auslastung führt zu einer von der Priorität $P_1$ abhängigen Änderung der Bewertung. Abbildung 5.22(a) zeigt die Priorität $P_2$ für den Dienst 1. Wegen der Auslastung der Warteschlange von maximal 42% erfolgt eine Herabstufung der Wertigkeit auf $< 0.65$. Im Bereich $> 0.8s$, wenn die Verkehrslast von Dienst 2 wegfällt, ist $P_2$, bis auf einen kleinen Bereich bei $1.2s$, wenn die Dienste 4 und 5 gleichzeitig geschaltet werden, bei 0.25. Beim Dienst 2, der nur in der ersten Hälfte des Simulationszeitraums aktiv ist, kann man im Wesentlichen zwei Bereiche mit unterschiedlichen maximalen Prioritäten differenzieren. Im Bereich 1 ist der Wert von

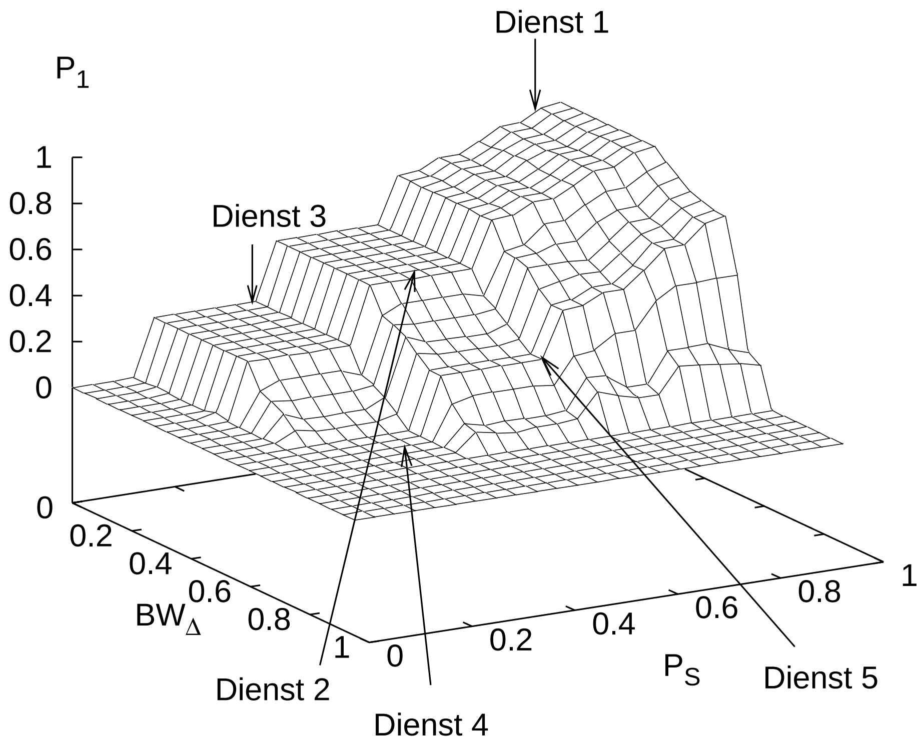



Abbildung 5.20: Arbeitspunkte der Dienste

$P_2 \approx 0.73$, während der anderen Phasen stellt sich ein Wert $< 0.25$ ein. Dieser Sprung wird durch das periodische Schalten des Dienst 5 hervorgerufen. Wegen der hohen Priorität von Dienst 5 und der dadurch bedingten vorrangigen Bedienung werden die Speicherkapazitäten des Systems genutzt. Das wiederum bewirkt, da die Auslastung im $FC_2$ ausgewertet wird, ein Ansteigen von $P_2$. Auch im Verlauf der Priorität $P_2$ der übrigen Dienste kann dieser Einfluß abgelesen werden.
Die Priorität der Dienste 3 und 4 ist während des gesamten Verlaufs wegen der geringen Priorität $P_1$ kleiner 0.5. Die Priorität $P_2$ von Dienst 5 nimmt im Bereich $< 0.8s$ Werte aus dem Intervall $[0.4, 0.75]$ an. Nach Abschalten der Verkehrslast von Dienst 2 verringert sich der Einfluß merklich. Hier wird die in Abschnitt 5.4.1 aufgestellte Hypothese über den Einfluß des Dienst 5 untermauert. Auf Grund der großen Abweichung von der deklarierten Bandbreite ist der Einfluß der Priorität $P_S$ von Dienst 4 bedeutungslos. Weiterhin ist der Arbeitspunkt für Dienst 2 nicht richtig optimal angepaßt.

### 5.4.5 Bedienpriorität $P_B$

Die Bedienpriorität $P_B$ hängt von der temporären Priorität $P_2$ und der Auslastung des Links $T_U$ ab. Die Verläufe sind in den Abbildungen 5.25, 5.26 und 5.27 dargestellt. Auch

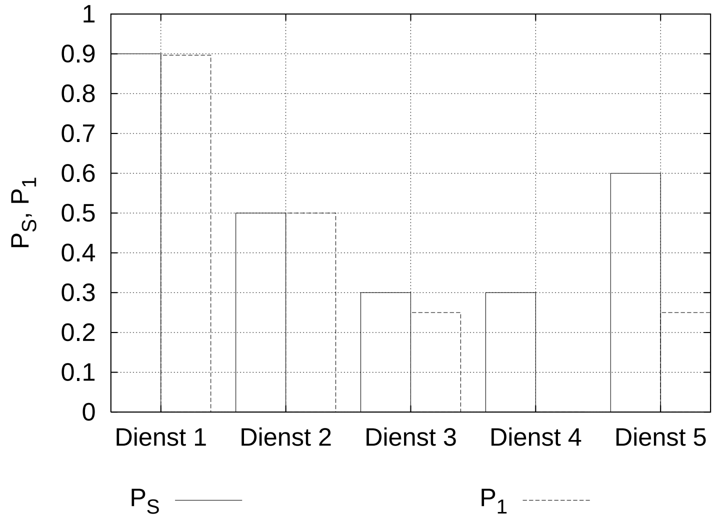



Abbildung 5.21: Vergleich der temporären Prioritäten $P_1$ mit den korrespondierenden Dienstprioritäten $P_S$

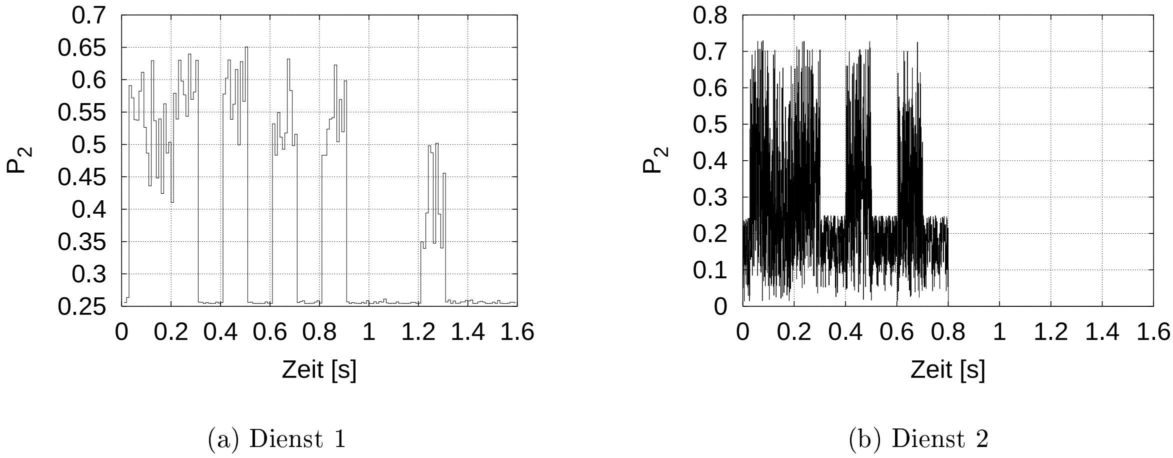
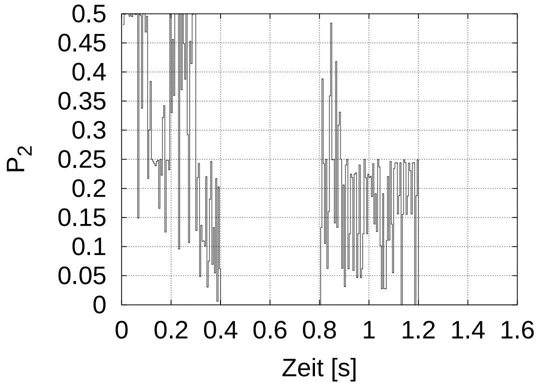
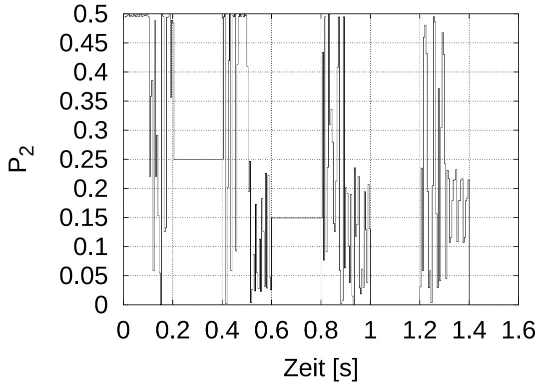



Abbildung 5.22: Temporäre Priorität $P_2$

hier zeigt sich, daß Dienst 5 einen großen Einfluß auf die Bearbeitung der Warteschlangen aller Dienste durch den Multiplexer hat. Weiterhin gilt für die höher prioren Dienste 1, 2 und 5, daß bei ihnen Wertigkeiten $< 0.55$ auftreten, während bei den übrigen Diensten (3 und 4) diese im Wesentlichen $< 0.25$ sind. Ausnahmen bilden die Intervalle ($[0.8s, 0.82s]$, $[1.1s, 1.2s]$ und $[1.3s, 1.4s]$), in denen die Kanalauslastung nicht kritisch ist (Abb. 5.28).




(a) Dienst 3




(b) Dienst 4

Abbildung 5.23: Temporäre Priorität $P_2$

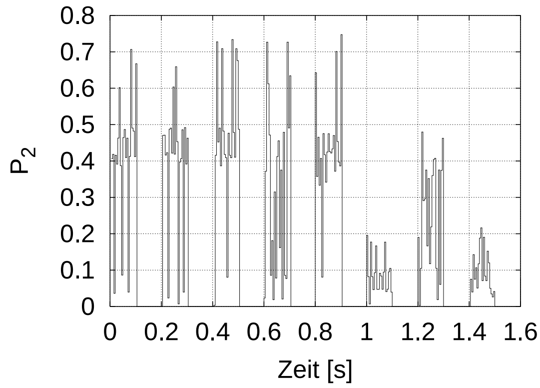


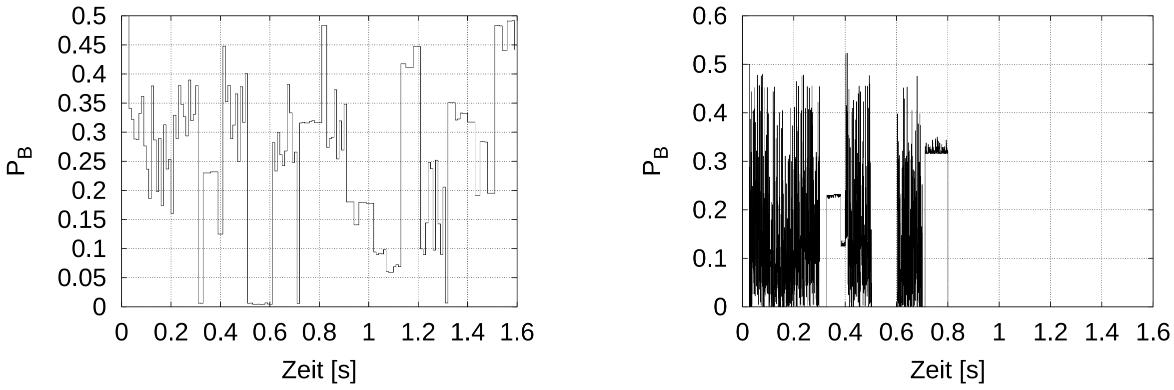

Abbildung 5.24: Temporäre Priorität $P_2$ des Dienstes 5

### 5.4.6 Auslastung des Links

Abbildung 5.28 zeigt die Auslastung des angeschlossenen Links. Der Kanal wird auf Grund des vorliegenden Angebots an seiner Grenze betrieben. In dem anderen Fall, d. h., wenn das Angebot unterhalb der maximalen Bandbreite liegt, existiert noch eine Transportreserve, die unter den Diensten verteilt wird. Ein Vergleich mit den Verlustraten (Abb. 5.16 und 5.17) zeigt, daß die Ressourcen in dem Bereich $[0.9, 1.6]$ so verteilt werden, daß keine Verluste mehr auftreten.

### 5.4.7 Einfluß der Dienstpriorität

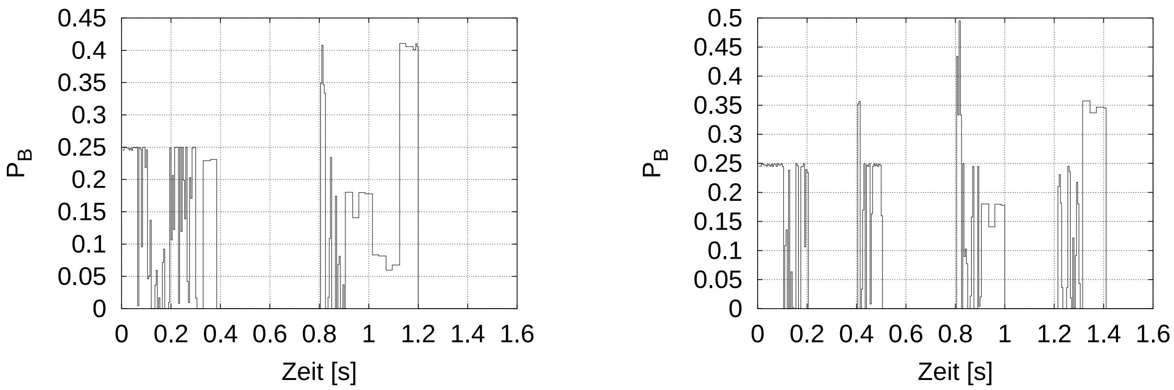
Um den Zusammenhang zwischen der Dienstpriorität und der Verlustrate bzw. der Auslastung zu untersuchen, wurde bei Simulationsläufen die Dienstpriorität $P_{S,Dienst3}$ im Inter-




(a) Dienst 1

(b) Dienst 2

Abbildung 5.25: Bedienpriorität $P_B$




(a) Dienst 3

(b) Dienst 4

Abbildung 5.26: Bedienpriorität $P_B$

vall $[0.1, 0.9]$ in Stufen von 0.1 inkrementiert. Die Abbildung 5.29 zeigt die Verlustrate der einzelnen Dienste in Abhängigkeit von der Priorität $P_{S,Dienst3}$. Es kann abgelesen werden, daß schon bei einer Erhöhung der Priorität $P_{S,Dienst3}$ auf 0.4 die Verlustrate des Dienstes auf 0 sinkt. Verbunden damit ist die Erhöhung der Verlustrate von Dienst 5 auf $\approx 6\%$ und von Dienst 2 auf $\approx 0.2\%$. Die Verlustrate von Dienst 1 bleibt konstant 0, die von Dienst 4 ist unverändert ungefähr 8.5%. Abbildung 5.30 stellt den Verlauf der Auslastung der Warteschlangen in Abhängigkeit von der Priorität $P_{S,Dienst3}$ dar. Bei einer Vergrößerung der Priorität $P_S$ auf 0.4 sinkt die Auslastung der Warteschlange von Dienst 3 auf $\approx 6\%$. Gleichzeitig steigt die Nutzung der Ressourcen der Dienste 1 und 2 um 1.5% bzw. 2% an. Der Ablauf dokumentiert, daß durch eine gezielte Änderung der Priorität $P_S$ Einfluß auf

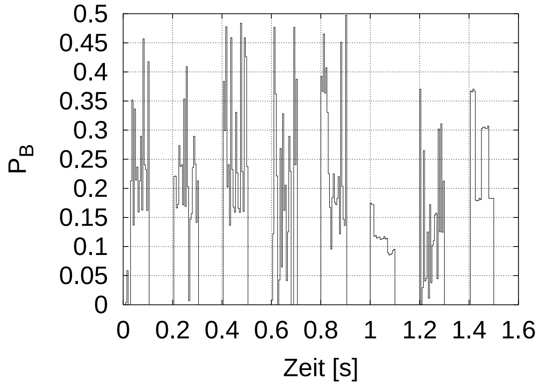



Abbildung 5.27: Bedienpriorität $P_B$ für Dienst 5

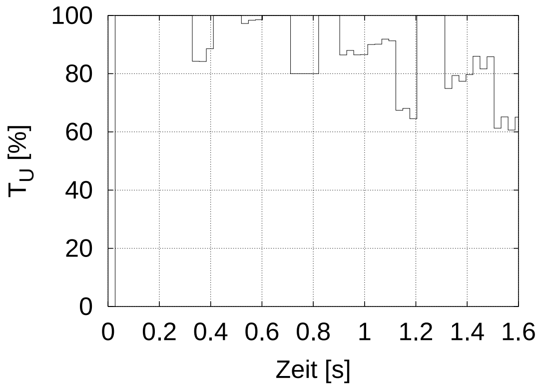



Abbildung 5.28: Auslastung der Bandbreite des angeschlossenen Links

die Behandlung eines Dienstes genommen werden kann.

### 5.4.8 Bewertung des Fuzzy Logic basierten Policing Controllers

In der Tabelle 5.4 sind die Kennzahlen (nach Abschnitt 4.2.2) zur Bewertung der Auslastung und der Verlustrate für die Dienste und den Kanal aufgelistet. Die Auslastung der Warteschlangen liegt für den Dienst 1 bei $\approx 12\%$ und steigt bis auf $\approx 18\%$ für den Dienst 5. Der Kanal ist mit $\approx 89\%$ voll ausgelastet. Eine weitere Steigerung ist nicht möglich, da durch die Wahl der Verkehrsparameter die eingeprägte Last nicht immer die vollständige Bandbreite ausnutzt.
Die Verlustraten für die Dienste 1 und 2 liegen bei 0.1% bzw. 3.3%. Die Raten der übrigen Dienste sind $< 8\%$. Trotz dieser in Bezug auf die Untersuchung guten Kennwerte konnte man bei der Analyse der internen Signale des Fuzzy Controllers Fehlanpassungen erkennen. Die temporäre Priorität $P_{1,Dienst3}$ war geringer als die korrespondierende Priorität

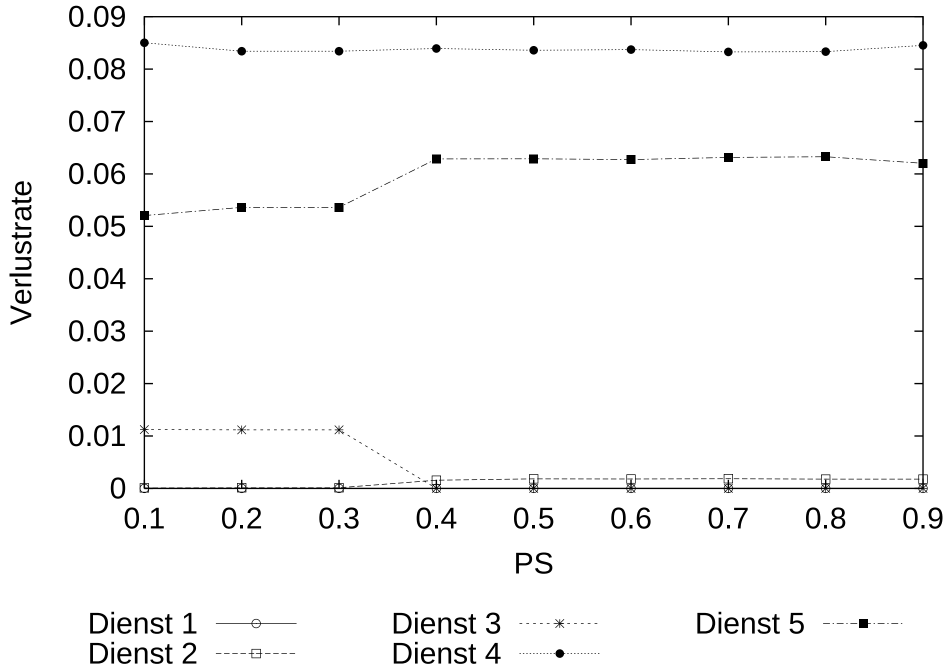



Abbildung 5.29: Verlustrate in Abhängigkeit von der Priorität $P_S$

| Dienst | Auslastung | Verlustrate |
|---|---|---|
| 1 | 0.024 | 0.001 |
| 2 | 0.120 | 0.033 |
| 3 | 0.084 | 0.047 |
| 4 | 0.158 | 0.069 |
| 5 | 0.186 | 0.086 |
| Mittelwert | 11.44% | 4.72% |
| Kanalauslastung | 0.894 | — |

Tabelle 5.4: Kennzahlen des Fuzzy Logic basierten Policing Controllers

$P_{S,Dienst3}$, obwohl dieser Dienst konform zum Verkehrsvertrag betrieben wurde. Es zeigt sich aber auch, daß die gewünschte Strategie, nämlich, daß mit Hilfe der Priorität $P_S$ die Bindung an die Vertragsparameter beschrieben werden kann, realisierbar ist. Dienst 1 ist mit einer hohen Wertigkeit von 0.9 ausgestattet. Es treten keine Verluste auf. Dienst 3 wird mit einer erheblich geringeren Priorität charakterisiert, was dann zum Auftreten von Verlusten führt.

In dem vorangehenden Abschnitt konnte gezeigt werden, daß durch den Einsatz der Fuzzy Logic ein Policing Controller aufgebaut werden konnte, der sich durch gute Kennwerte auszeichnet. Die Auslastung des Zugangsknotens beträgt ca. 90%. Die Verlustraten sind

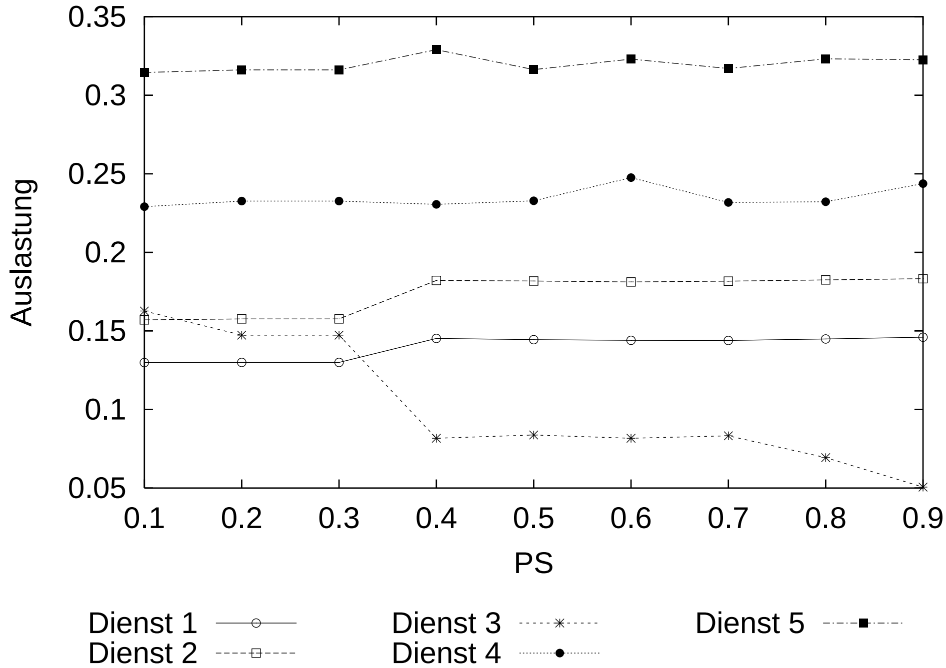



Abbildung 5.30: Auslastung der Warteschlangen in Abhängigkeit von der Priorität $P_S$

systemweit niedrig, aber bei der Bewertung der Dienste an Hand ihrer Priorität und der Abweichung von der deklarierten Bandbreite ergaben sich noch einige Fehlanpassungen. Weiterhin wurde bei diesem Entwurf die statische Dienstpriorität im Vergleich zu den Verkehrsparametern zu hoch bewertet. Durch eine angemessene Änderung konnte die Bearbeitung gezielt beeinflußt werden.

Aufbauend auf diesen Werten und unterstützt durch eine detaillierte Analyse des Übertragungsverhaltens des Reglers kann, wie schon in Abbildung 5.7 angedeutet, durch iteratives Anpassen der Regelbasen und Zugehörigkeitsfunktionen eine schrittweise Verbesserung des Regelverhaltens des Fuzzy Logic basierten Policing Controllers erreicht werden. Dieses Verfahren ist jedoch sehr zeitintensiv. Darüber hinaus kann es auch nicht in jedem Fall gewährleisten, daß der *gesamte* Lösungsraum nach einer Parameterkonstellation, die ein globales Optimum darstellt, durchsucht wird. Es besteht dann die Gefahr, daß das iterativ ermittelte Resultat nur eine beschränkte, *lokale* Lösung darstellt. Auf Grund des leider doch nur unvollständigen Wissens um die Zusammenhänge der unterschiedlichen Parameter besteht die Gefahr, daß Regeln, die im ersten Moment unlogisch erscheinen oder aber Schlußfolgerungen, die an „normalen“ Maßstäben gemessen, widersinnig sind, in Kombination mit den Membershipfunktionen doch zu einer guten Lösung führen können.

Auf der Suche nach einem *globalen* Optimum soll deshalb im Weiteren zur Anpassung der Regelbasen und Zugehörigkeitsfunktionen auf Methoden aus dem Bereich der Computational Intelligence zurückgegriffen werden.

# Kapitel 6

# Genetische Algorithmen

Als genetische Algorithmen (im Folgenden GA) werden Optimierungsstrategien bezeichnet, die in Anlehnung an die natürliche Genetik entstanden sind. Die Grundlage dieser engen Verwandtschaft ist auch für die Wahl des Namens dieser Algorithmen verantwortlich, die in einer Veröffentlichung durch John Holland im Jahre 1975 vorgestellt wurden. Bei ihrer Entwicklung stand zunächst nicht die Lösung konkreter technischer Probleme im Vordergrund. Sie stellten ein Modell der Natur dar und dienten dazu, die Evolutionsmechanismen zu simulieren. Man erhoffte sich davon, genauere Einblicke in die Entwicklung von Populationen und dem genetischen Informationsaustausch zu erlangen. In den folgenden Jahren seit dieser wegweisenden Arbeit wurde dieses Optimierungverfahren in den verschiedensten Gebieten erfolgreich eingesetzt. Es ist inzwischen theoretisch und praktisch bewiesen, daß GAs robust bei der Suche nach einer Lösung in komplexen Optimierungsräumen sind und effektiv und effizient arbeiten. Ein wesentlicher Vorteil dieses Verfahrens stellt die klare Trennung zwischen dem eigentlich zu optimierenden Problem auf der einen Seite und dem genetischen Algorithmus auf der anderen Seite dar. Der genetische Algorithmus arbeitet also ohne jegliches Wissen über den zu optimierenden Prozeß. An dieser Stelle soll kurz auf die Vorgehensweise und die Bestandteile eines Genetischen Algorithmus eingegangen werden, eine grundlegende und anschauliche Einführung zu Genetischen Algorithmen findet sich in [26].

## 6.1 Grundlagen genetischer Algorithmen

Das Verfahren beruht darauf, den Optimierungsvorgang auf der Grundlage des darwinistischen Prinzips (survival of the fittest) der natürlichen Auslese der Evolution anzupassen.

Ausgehend davon wird nicht nur *eine* Lösung eines Problems bearbeitet sondern eine Multimenge ($X(t) = x_1, \cdots, x_n$). Die Menge dieser Repräsentationen wird in enger Anlehnung an die biologische Begrifflichkeit *Population* genannt. Auf diese Populationen werden dann verschiedene Operationen angewandt, deren Funktionen und Nomenklatur ebenfalls der Biologie entnommen wurden. Die Lösungen $x_i$ werden mittels einer bijektiven Abbildung in *Bitstrings* sog. *Individuen* oder auch *Chromosome* transformiert. Die einzelnen Bits

der Strings heißen, in Anlehnung an die evolutionäre Abstammung dieses Optimierungsverfahrens, *Gene*.
Der grundlegende Gedanke dabei ist, daß sich durch die evolutionären Mechanismen immer die Individuen stärker entwickeln, die sich am besten an die vorliegenden Gegebenheiten anpassen können. Die Anwendbarkeit der Lösung auf das vorliegende Einsatzgebiet wird mit Hilfe einer *Fitnessfunktion* bewertet. Die Eigenschaften, die in den Genen codiert sind, werden dann in Abhängigkeit von ihrer Fitness an mehr oder weniger Nachkommen (*Selektion*) der folgenden Generation vererbt (*Crossover*). Damit der Optimierungsprozeß bei einer gefundenen Lösung nicht zum Stillstand kommt und darüber hinaus auch gewährleistet werden kann, daß auch der gesamte Lösungsraum untersucht wurde, so daß sichergestellt werden kann, daß nicht nur lokale sondern auch globale Optima entdeckt werden, werden die Bits der Individuen mit einer vorgegebenen Wahrscheinlichkeit, der sog. Mutationsrate (*Mutation*), gekippt.

Mit Hilfe dieser Operationen erzeugt der genetische Algorithmus fortlaufend aus der aktuellen Population neue Generationen, wobei sich die besseren Individuen im Laufe der Zeit immer stärker gegenüber den schlechteren durchsetzen werden.

### 6.1.1 Funktionsweise von genetischen Algorithmen

**Codierung**
Die Codierung ist das Interface zwischen dem GA und der eigentlichen Optimierungsaufgabe. Der Anwender legt hier eine Transformation fest, mit der die Lösungen auf einen Bitstring abgebildet werden bzw. aus einem Individuum wieder eine reale Lösung abgeleitet werden kann. Bei den in dieser Arbeit benutzten einfachen GAs (simple GAs) werden alle Strings ausschließlich aus Nullen und Einsen gebildet und heißen daher auch *binäre* Strings. Die Codierung stellt eine wesentliche Voraussetzung für den Erfolg der Optimierung dar. So können z. B. die Konvergenzgeschwindigkeit eines GA und die Unterscheidungsfähigkeit zwischen lokalen und globalen Extrema sehr wesentlich von dieser Umsetzung abhängen. Desweiteren ist diese Abbildung ein Grund für die universelle Einsatzmöglichkeit der GAs. Diese Codierung bewirkt eine klare Trennung zwischen dem eigentlichen Problem auf der einen Seite und dem Optimierungsverfahren auf der anderen Seite.

**Selektion**
Die Selektion dient dazu, aus einer Population $P(t)$ diejenigen Individuen auszuwählen, die mittels Reproduktion ihre Anlagen an die nächste Population $P(t+1)$ weitergeben dürfen. Dazu bedarf es zunächst einer Bewertung der Individuen der Population $P(t)$. Jedes Mitglied aus $P(t)$ bekommt eine Fitness zugeordnet. Dies ist in der Regel ein reeller Zahlenwert, der auf der Basis einer Bewertungsfunktion errechnet wird. Die Ableitung dieser Fitnessfunktion ist nicht trivial und setzt Information und Expertenwissen über die eigentliche Problemstellung voraus. Im Folgenden wird gemäß der Fitness der einzelnen Individuen festgelegt, in welchem Verhältnis die Anlagen an die nachfolgende Generation weitergegeben werden. Individuen mit einer hohen Fitness geben ihre Veranlagung stärker an die nächste Generation weiter als Individuen mit einer geringeren Güte. Die durch

die Methode der Selektion ausgewählten Strings werden dann als Mitglieder der folgenden Generation reproduziert. Der Mechanismus der Selektion der Strings mit höheren Fitnesswerten läßt sich durch einen Vergleich mit einem Rouletterad erklären. Die Summe aller Fitnesswerte einer Generation entspricht bei diesem Vergleich dem gesamten Rouletterad (Abbildung 6.1). Jeder String enthält nun soviele Nummern auf dem Rouletterad zugeteilt wie dem prozentualen Anteil seiner Fitness an der Gesamtfitness der Generation entspricht. Nun wird das Rouletterad gedreht und der String, dem die ermittelte Nummer zugeteilt wurde, wird für die folgende Generation reproduziert. Die Art der bevorzugten

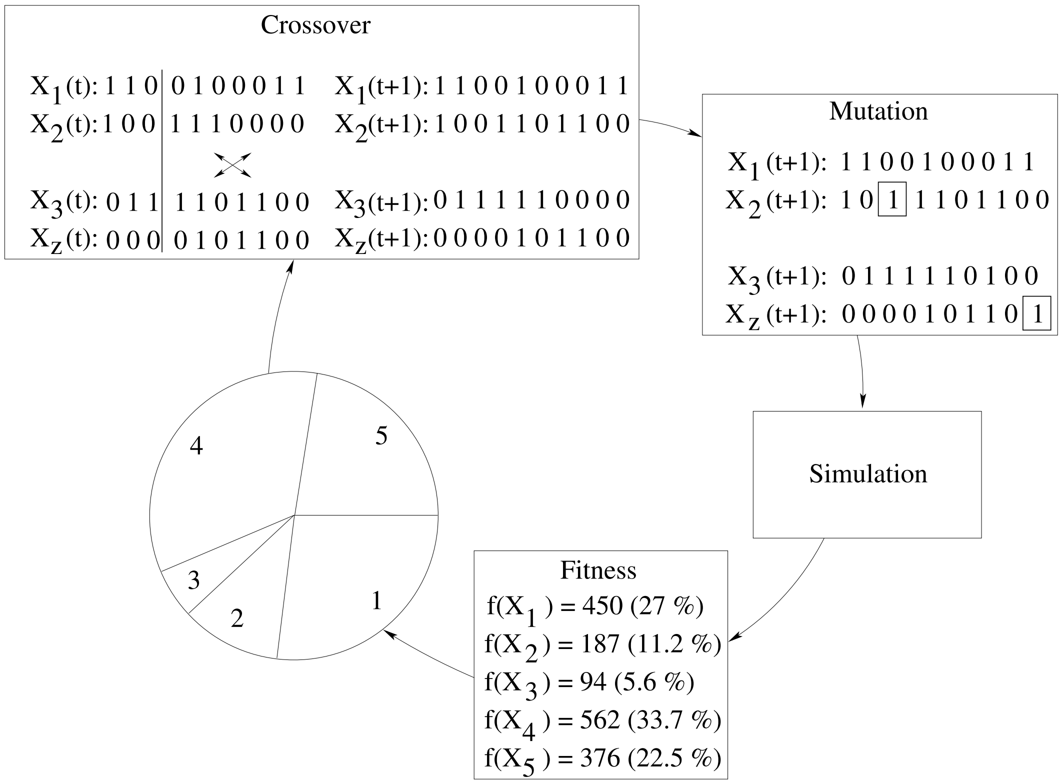



Abbildung 6.1: Prinzipieller Ablauf des genetischen Algorithmus

Berücksichtigung entsprechend ihrer Fitness wird auch als Reproduktion bezeichnet. Nachdem durch diesen Mechanismus die Mitglieder der kommenden Generation erzeugt wurden, kommen anschließend die Methoden Crossover und Mutation zum Einsatz.

**Crossover**

Mit Hilfe des Crossovers werden, um die Anlagen, die in den Genen codiert sind, weiterzugeben, Teile der Chromosomen in Abhängigkeit von einer Zufallsvariablen miteinander getauscht. Beim sog. One-Point-Crossover wird zufällig ein Punkt innerhalb der Strings bestimmt, an dem die beiden Strings geteilt und die jeweils korrespondierenden Teile getauscht werden (vgl. Abb. 6.1). Beim Two-Point-Crossover wird ein Segment aus beiden

Strings herausgetrennt und in den jeweils anderen eingesetzt. Bei Uniform-Crossover wird für jede Stelle der beiden Strings per Zufall entschieden, ob die in ihr enthaltene Information getauscht werden soll. Das Crossover wird jedoch nicht auf alle Strings einer Generation angewandt sondern immer nur auf einen bestimmten Prozentsatz dieser Strings. Dieser Prozentsatz, auch Crossover- Wahrscheinlichkeit genannt, wird zu Beginn des GAs festgelegt und kann einen starken Einfluß auf die Qualität des Ergebnisses haben.
In diesem Austausch der Informationen manifestiert sich der eigentliche evolutionäre Mechanismus eines GA. Da bei der Selektion nur die Originalindividuen kopiert werden, ist kein genetischer Fortschritt zu erwarten. Tauscht man jedoch die Veranlagungen der stärksten Individuen einer Generation miteinander aus, ist die Wahrscheinlichkeit eines evolutionären Fortschritts, sprich eine Steigerung der Fitness, relativ hoch. Problematisch wird das Verfahren, wenn es sich an einem lokalen Optimum festläuft bzw. zu schnell konvergiert. Dies ist vor allem dann der Fall, wenn die lokalen Optima im Lösungsraum sehr steile Peaks bilden und somit alle Individuen in der näheren Umgebung bedeutend kleinere Fitnesswerte aufweisen. Um dieser Problematik entgegenzuwirken, werden die einzelnen Individuen noch abschließend einer Mutation unterzogen.

**Mutation**
Die Mutation sorgt durch die Modifizierung einzelner Bits dafür, daß auch Zustände eines Suchraums berücksichtigt werden können, die in der Ausgangsgeneration nicht vorhanden sind. Dies kann beispielsweise passieren, wenn sämtliche Individuen der Startgeneration zufällig erzeugt werden und das letzte Bit bei allen Strings eine 1 ist. In diesem Fall sorgt die Mutation dafür, daß irgendwann im Laufe der Optimierung dieses letzte Bit auch 0 werden kann. Die Modifizierung des Bits bedeutet also, daß aus einer 1 eine 0 wird bzw. aus einer 0 eine 1 (vgl. Abb. 6.1). Auch die Mutation eines Bits wird nur auf einen vor dem Start des genetischen Algorithmus festgelegten Prozentsatz aller Bits angewandt, der auch Mutationswahrscheinlichkeit genannt wird. Typische Werte für diese beiden Wahrscheinlichkeiten sind 0.6 für ein Crossover und 0.0333 für eine Mutation. Diese beiden Werte wurden von De Jong, dem Erfinder der genetischen Algorithmen, als Vorschläge angegeben [26, 39].

**Fitness Funktion**
Primäres Ziel der genetischen Algorithmen ist es, die Fitness einer Funktion bzw. eines Systems zu maximieren. Um die mit Hilfe dieser Fitnessfunktionen ermittelten Fitnesswerte zur Reproduktion im genetischen Algorithmus einsetzen zu können, muß beachtet werden, daß bei allen Funktionen ein hoher Wert bei den gewählten Parametern wie z. B. der Verlustrate einer schlechten Fitness entspricht. Daher ist zum Einsatz der ausgewählten Fitnessfunktionen als Entscheidungskriterium für die Berücksichtigung eines Strings bei der Reproduktion zuvor erst eine Umrechung nötig. Hierzu wird die mit einer der Gleichungen berechnete Gesamtfitness von dem Wert subtrahiert, der sich als Summe bei der Anwendung der ausgewählten Funktion für eine Verlustrate von 100 % für alle Quellen ergibt ($Fitness_{max}$). So wird der tatsächliche Fitnesswert im Folgenden $Fitness_{GA}$ eines Controllers ermittelt, der dann schließlich bei den verschiedenen Ansätzen der Optimierung

mit Hilfe genetischer Algorithmen zur Anwendung kommt :

$$Fitness_{GA} = Fitness_{max} - Fitness_{Gesamt} \tag{6.1}$$

## 6.2 Optimierung der Controller mit Hilfe genetischer Algorithmen

### 6.2.1 Grundlegende Überlegungen

Wie im Abschnitt 6.1.1 detailliert beschrieben wurde, arbeitet jeder genetische Algorithmus auf der Basis einer Fitnessfunktion. Mit Hilfe dieser Fitnessfunktion wird objektiv die Anwendbarkeit eines Bitstrings aus einer Population zur Lösung des vorliegenden Problems bestimmt. Gleichzeitig wird damit aber auch ermittelt, in welchem Umfang dieses Mitglied bei der Reproduktion zur Erzeugung der nächsten Generation berücksichtigt wird. Daher ist die Festlegung einer geeigneten Fitnessfunktion essentiell für den Einsatz der genetischen Algorithmen [7].

Zum erfolgversprechenden Einsatz eines genetischen Algorithmus zur Optimierung des im vorigen Kapitel beschriebenen Fuzzy Controllers muß daher eine Fitnessfunktion abgeleitet werden, die alle relevanten Faktoren, von denen das Verhalten des Controllers abhängig ist, entsprechend ihrem Einfluß berücksichtigt.

#### Die Regelstrategie

Bei der dafür notwendigen Analyse der verschiedenen Einflußgrößen des Controllers im Vorfeld der Untersuchung ergab sich, daß zwei wesentliche Faktoren dafür verantwortlich zeichnen, wie die gesamte verfügbare Übertragungskapazität zwischen den einzelnen Quellen aufgeteilt wird:

- die Eingangs- oder Dienstpriorität
- die Konformität zum Verkehrsvertrag, die sich in den Untersuchungen durch die Überschreitung der vereinbarten Verkehrsparameter manifestiert.

Im einzelnen bedeutet das, wenn bei zwei Quellen mit gleicher Eingangspriorität $P_S$ die erste Quelle einen größeren Wert für $BW_\Delta$ aufweist als die zweite, die zweite Quelle bei der Bandbreitenzuteilung bevorzugt zu behandeln ist. Diese bevorzugte Behandlung gilt ebenfalls für den Fall, daß von zwei Quellen mit einem gleichen Wert $BW_\Delta$ die erste Quelle eine höhere Eingangspriorität $P_S$ aufweist als die zweite. Bei einer Optimierung des im vorigen Kapitel vorgestellten Fuzzy Controllers sind diese beiden Punkte als wichtigste Einflußfaktoren zu berücksichtigen. Wenn wieder von dem in Kap. 3 beschriebenen Zugangsknoten sowie den fünf in Abb. 4.1 und durch Tabelle 4.1 spezifizierte Eingangslasten ausgegangen wird, sollten von einem optimierten Knoten im Vergleich zu den Verlustraten des im vorigen Kapitel vorgestellten Controllers (vgl. Abb. 5.16 bis 5.30) folgende Änderungen erwartet werden:

- Die Verlustrate des Dienstes 1 soll weiterhin, auf Grund der hohen Priorität $P_S$ und der Konformität zum Verkehrsvertrag, konstant 0% aufweisen.
- Die Verlustraten von Dienst 2 und 4 liegen höher als die von Dienst 5. Die Priorität $P_{S,Dienst5}$ ist hoch und auf Grund der großen Transferrate ist die Warteschlange immer gut ausgelastet. Beide Faktoren resultieren dann in einer hohen Bedienpriorität $P_B$, so daß Dienst 5 bevorzugt behandelt wird.
- Der Dienst 3 ist während der gesamten Simulation vertragskonform. Auf Grund der kleinen Dienstpriorität ist von einem mittleren Verlust auszugehen.
- Wegen der großen Abweichung von den vereinbarten Verkehrsparametern werden bei Dienst 4 große Verluste erwartet.
- Durch die Berücksichtigung der Auslastungskennzahlen ($Q_U$, $S_U$ und $T_U$) bei der Bestimmung der Bedienpriorität ist eine hohe Auslastung der Ressourcen und systemweit eine geringe Verlustrate absehbar.
- Die Verlustraten und die Auslastung sind durch die Priorität $P_S$ beeinflußbar.

### 6.2.2 Codierung der Regelbasen

Bei dem gewählten Ansatz werden die drei Regelbasen separat codiert. Die einzelnen Strings werden dann für die Verarbeitung miteinander verkettet.

#### Die Fuzzy Controller $FC_1$ und $FC_2$

Bei diesen Regelbasen gestaltet sich die Umsetzung in einen handhabbaren Bitstring schon recht kompliziert. Da beide Regelbasen Ausgangsvariablen mit je *fünf* verschiedenen linguistischen Termen (SK,K, M, G und SG) aufweisen, werden für eine Darstellung als Bitstring in diesem Fall mindestens drei Bits benötigt, mit denen dann insgesamt acht verschiedene Werte codiert werden können. Tabelle 6.1 zeigt eine einfache Umsetzung der Fuzzy-Sets in einen Bitstring. Da die drei zusätzlichen Strings durch Crossover oder Mutation entstehen

| Bitstring | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|---|---|---|---|---|---|---|---|---|
| Fuzzy-Set | SK | K | M | G | SG | xxx | xxx | xxx |

Tabelle 6.1: Einfache, unvollständige Codierung der Regelbasis

können, müssen diese Kombinationen ebenfalls belegt werden. Tabelle 6.2 zeigt einen möglichen Ansatz zur vollständigen Codierung der beiden Regelbasen $FC_1$ und $FC_2$. Mit Hilfe dieser Aufstellung können die acht binären Werte dann einfach in die fünf korrespondierenden linguistischen Terme der Ausgangsvariablen umgesetzt werden. Bei einer redundanten Codierung auf der Grundlage dieser Tabelle können verschiedene Bitstrings einen identischen Zustand der Regelbasis darstellen. So kann z. B. der linguistische Term SK sowohl

| Bitstring | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|---|---|---|---|---|---|---|---|---|
| Fuzzy-Set | SK | K | M | G | SG | SK | K | M |

Tabelle 6.2: Vollständige Codierung der Regelbasis

als *000* als auch als *101* dargestellt werden. Dieses Transformationsschema wird von Kropp und Baitinger gewählt [48]. Der Vorteil dieser Codierung ist die einfache Umrechnung der binären Codierung in den korrespondierenden Wert der Regelbasis mit Hilfe einer Division des zugehörigen dezimalen Wertes Modulo 5. Der Nachteil dieser Transformation besteht darin, daß benachbarte Fuzzy-Sets nur durch die Änderung mehrerer Bits erreicht werden können, so daß keine geschlossene Überprüfung des Lösungsraumes gegeben ist. Durch eine Gray-Codierung (Tabelle 6.3) der linguistischen Terme kann dieser gravierende Nachteil allerdings umgangen werden. Der große Nachteil dieser beiden Transformationsschemata

| Bitstring | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
|---|---|---|---|---|---|---|---|---|
| Fuzzy-Set | SK | K | M | G | SG | xxx | xxx | xxx |

Tabelle 6.3: Gray-Codierung der Regelbasis

liegt aber in der Belegung der drei redundanten Bitstrings. Ausgehend von den Tabellen 6.2 und 6.3 zeigt sich, daß nur durch die Veränderung eines Bits entweder bei der Mutation oder dem Crossover sich der linguistische Term vom kleinsten Wert (101 = SK) in den größten Wert (100 = SG) bzw. umgekehrt wandeln kann. Diese Sprünge führen zu einem suboptimalen Verhalten des genetischen Algorithmus.

Bei dem in den Untersuchungen verwendeten Transformationsschema wurde deshalb eine Codierung verwendet, bei der die fünf erforderlichen linguistischen Terme durch die drei zusätzlich zur Verfügung stehenden Bitstrings so geschickt ergänzt wurden, daß zum einen durch die Mutation eines einzelnen Bits nur möglichst geringe Veränderungen der linguistischen Terme bewirkt werden. Außerdem sollte zusätzlich angestrebt werden, daß sich bei einer Mutation eines Bits der korrespondierende linguistische Term mit großer Wahrscheinlichkeit ändert.

Die Optimierung des Umsetzungsschemas unter den beschriebenen Randbedingungen wurde daher ebenfalls mit Hilfe eines genetischen Algorithmus durchgeführt. [1] Bei einer Co-

| Bitstring | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
|---|---|---|---|---|---|---|---|---|
| Fuzzy-Set | SK | K | M | K | G | SG | G | M |

Tabelle 6.4: GA basierte Codierung der Regelbasis

dierung der linguistischen Terme durch Verwendung der Tabelle 6.4 ist durch die Mutation

[1] Die Beschreibung des Verfahrens erfolgt im Anhang C.2.

eines einzelnen Bits maximal eine Veränderung zum Ąübernächsten Term möglich. Die Mutation eines einzelnen Bits ohne eine daraus resultierende Veränderung des linguistischen Terms ist - wie bei einer Codierung mit Hilfe der Tabelle 6.1 - nicht möglich.

#### Der Fuzzy Controller $FC_3$

Die Codierung der Regelbasis des Fuzzy Controllers $FC_3$ verknüpft die linguistischen Variablen $P_2$ sowie die Kanalauslastung $T_U$, die durch 5 bzw. 3 Membership-Funktionen definiert sind. Auf Grund dieser vorgegebenen Randbedingungen umfaßt die gesamte Regelbasis 15 Zustände.

#### Darstellung der Regelbasen als Bitstring

Bei Verwendung der Tabelle 6.4 zur Transformation der Regelbasen der zwei Controller $FC_1$ und $FC_2$ kann jede dieser Regelbasen, die jeweils 25 Felder aufweisen, mit je 75 Bits codiert werden. Für die Umsetzung der Regelbasis von $FC_3$ werden 45 Bits benötigt. Für die Bearbeitung der Regelbasis mit Hilfe der genetischen Algorithmen werden die einzelnen Bitstrings dann miteinander verkettet (Abb. 6.2.2).

$$\underbrace{001011\cdots111101}_{FC_1}\underbrace{011010\cdots011001}_{FC_2}\underbrace{000001\cdots111111}_{FC_3}$$

Als Gesamtlänge des Bitstrings, durch den die Regelbasen aller drei Teilcontroller beschrieben werden können, ergibt sich also

$$75 + 75 + 45 = 195 Bits.$$

### 6.2.3 Darstellung der Zugehörigkeitsfunktionen als Bitstring

Um auch die Zugehörigkeitsfunktionen der Fuzzy Controller mit Hilfe von genetischen Algorithmen verändern und optimieren zu können, müssen diese geeignet codiert werden. Wie bei der Umsetzung der Regelbasis, die bereits im vorigen Abschnitt beschrieben wurde, werden auch hier Bitstrings konstanter Länge eingesetzt. Daher scheidet der von Herrera ([31]) gewählte Ansatz, bei dem die genetischen Algorithmen direkt mit den Parametern der Zugehörigkeitsfunktionen (als reelle Zahlen) sowie dafür angepaßten genetischen Operationen arbeiten, ebenso aus wie der Ansatz von Kinzel ([45]), der Zugehörigkeitsgrade verwendet. Da Leitch und Probert für ihren Ansatz ([50]) Strings variabler Länge einsetzen, ist auch deren Codierung in diesem Zusammenhang nicht einsetzbar. In dieser Arbeit wird im Folgenden eine Codierung verwendet, die auf einer von Karr ([44]) vorgeschlagenen Codierung basiert. Karr codiert bzw. decodiert jeden Parameter einer Zugehörigkeitsfunktion mit Hilfe der folgenden Formeln.
Codierung:

$$b = \frac{P - P_{min}}{P_{max} - P_{min}}(2^m - 1) \tag{6.2}$$

Decodierung:

$$P = P_{min} + \frac{b}{2^m - 1}(P_{max} - P_{min}) \tag{6.3}$$

Dabei ist P der decodierte Wert, $P_{min}$ der minimale x-Wert und $P_{max}$ der maximale x-Wert. b gibt den dezimalen Wert eines Bitstrings der Länge m an. Jeder Parameter einer Zugehörigkeitsfunktion kann also einen von $2^m$ Werten aus dem Intervall zwischen $P_{min}$ und $P_{max}$ annehmen, so daß eine Genauigkeit von $\frac{P_{max}-P_{min}}{2^m}$ erreicht werden kann. Die Genauigkeit ist von der Größe des Wertes m abhängig. Allgemein gilt, je größer der Wert für m gewählt wird, desto genauer kann die Codierung erfolgen. Nachteilig ist aber, daß mit der vergrößerten Auflösung eine Verlängerung des resultierenden Bitstrings einhergeht. Der eingesetzte Fuzzy Controller verwendet ausschließlich dreieck- und trapezförmige Zugehörigkeitsfunktionen (vgl. 5). Aus den Abbildungen 5.6(a) und 5.6(b) wird deutlich, daß zur vollständigen Beschreibung einer trapezförmigen Zugehörigkeitsfunktion vier Parameter benötigt werden, während für eine dreieckförmige Funktion drei Parameter ausreichend sind. Um nun auch bei der Optimierung der Zugehörigkeitsfunktionen mit Strings konstanter Länge arbeiten zu können, wurden alle dreieckförmigen durch trapezförmige Zugehörigkeitsfunktionen ersetzt. Dies ist ohne Auswirkungen auf die Arbeitsweise des Controllers möglich, da die dreieckförmigen Zugehörigkeitsfunktionen ein Spezialfall der trapezförmigen sind. In diesem Fall gilt:

$$\gamma_{Trapez} = \delta_{Trapez} = \gamma_{Dreieck} \tag{6.4}$$

Auch Castro ([14]) arbeitet mit dieser Kodierung nach Karr. Weil der von ihm entworfene Controller ausschließlich symmetrische, dreieckförmige Zugehörigkeitsfunktionen verwendet, kommt er mit jeweils $2 \cdot m$ Bits pro Funktion aus. Da der hier eingesetzte Controller jedoch nicht nur mit symmetrischen Funktionen arbeitet, müssen alle vier Parameter eines Trapezes für die Optimierung des Controllers codiert werden, so daß für die Codierung einer Zugehörigkeitsfunktion $4 \cdot m$ Bits nötig sind, wenn jeder Parameter durch einen Bitstring der Länge $m$ codiert wird. Für diese Arbeit wurde $m = 8$ gewählt, so daß das Intervall zwischen $P_{min}$ und$P_{max}$ in $2^m = 256$ Abschnitte unterteilt ist. Bei dem zu codierenden Controller gilt - wie den Abb. 5.10 bis 5.12 entnommen werden kann - für alle linguistischen Variablen:

$$P_{min} = 0 \text{ und } P_{max} = 1 \tag{6.5}$$

Die Parameter der Zugehörigkeitsfunktionen können also alle mit einer Genauigkeit von $\frac{1}{2^m} = 0.0039$ codiert werden. Im Laufe der Optimierung hat sich diese Genauigkeit als ausreichend dargestellt. Als Beispiel für eine solche Codierung soll hier die Zugehörigkeitsfunktion des linguistischen Terms *Mittel* der linguistischen Variable *Queueauslastung* ($Q_U$) codiert werden. Aus Abb. 3.5 können für die drei Parameter der dreieckförmigen Zugehörigkeitsfunktion die Werte $\alpha = 0.24, \gamma = 0.49$ und $\beta = 0.74$ entnommen werden. Eine Umsetzung in eine trapezförmige Funktion, die zur Codierung benötigt wird, ergibt für die vier Paramter die Werte $a = 0.24, \gamma = \delta = 0.49$ und $\beta = 0.74$. Diese vier Werte werden nun unter Benutzung der Gleichung 6.2 wie folgt codiert :

| Parameter | Wert | $\cdot$ 255 | Wert | String |
|---|---|---|---|---|
| $\alpha$ | 0.24 | 61.20 | 61 | 00111101 |
| $\gamma$ | 0.49 | 124.95 | 125 | 01111101 |
| $\delta$ | 0.49 | 124.95 | 125 | 01111101 |
| $\beta$ | 0.74 | 188.70 | 189 | 10111101 |

Tabelle 6.5: Codierung der Parameter einer Zugehörigkeitsfunktion

Die Zugehörigkeitsfunktion des linguistischen Terms *Mittel* der linguistischen Variablen Queuelänge wird also durch den Bitstring

$$\underbrace{00111101}_{\alpha}\underbrace{01111101}_{\gamma}\underbrace{01111101}_{\delta}\underbrace{10111101}_{\beta}$$

codiert. Wie bei der Codierung der Regelbasis (vgl. Kapitel 6.2.2) entstehen auch bei der Codierung der Zugehörigkeitsfunktionen die Bitstrings einer Population durch eine Aneinanderkettung aller Parameter. Da der Controller mit sechs linguistischen Variablen ($BW_\Delta$, $P_S$, $P_1$, $P_2$, $P_B$ und $Q_U$), die durch je fünf linguistische Terme definiert werden, und der linguistischen Variablen $T_U$, die mit drei linguistischen Termen arbeitet (vgl. Kapitel 5.3.3), haben die zur Verwendung des genetischen Algorithmus zu erzeugenden Bitstrings folgende Länge (mit m = 8):

$$Stringlänge = 4 \cdot m \cdot (6 \cdot 5 + 3) = 1056$$

Beim Einsatz eines genetischen Algorithmus zur Optimierung der durch Bitstrings der Länge 1236 dargestellten Zugehörigkeitsfunktionen sind einige Unterschiede zur Optimierung der Regelbasis (vgl. Kapitel 6.2.2) zu beachten. Durch die vom GA angewandten Operationen Crossover und Mutation kann es passieren, daß die dadurch entstandenen Strings nicht mehr die aus Abb. 5.6 erkennbare Bedingung

$$\alpha \leq \gamma \leq \delta \leq \beta$$

erfüllen. Um dieses Problem zu umgehen, werden zur Bestimmung der Zugehörigkeitsfunktion zuerst alle vier korrespondierenden Parameter decodiert. Anschließend werden diese vier Werte ihrer Größe nach den vier Parametern $\alpha$, $\beta$, $\gamma$ und $\delta$ zugeordnet.
Neben diesem Effekt kann es außerdem durch Mutation sehr schnell dazu kommen, daß völlig unbrauchbare Controller entstehen, weil die durch Decodierung des Bitstrings entstehende Menge von Zugehörigkeitsfunktionen größere Teilbereiche einer oder mehrerer linguistischer Variablen nicht mehr abdeckt.
Ein weiterer Nachteil der gewählten Codierung mit Bitstrings konstanter Länge ist die fehlende Möglichkeit, die Anzahl der linguistischen Terme einer Variablen während der Simulation bei Bedarf zu erhöhen. Dieser Freiheitsgrad läßt sich nur durch eine Codierung mit variabler Stringlänge erreichen. Es zeigt sich aber auch, daß hierfür eine Anpassung

der Regelbasis an die Anzahl der linguistischen Terme erfolgen muß.
Weil der Aufwand für eine Codierung mit variabler Stringlänge und die damit verbundenen Veränderungen der Regelbasis zu einem weiteren Anstieg der Rechenzeit führen, wurde daher im Rahmen dieser Arbeit auf diese Erweiterungsmöglichkeit zugunsten einer Beschleunigung der Optimierung verzichtet.

### 6.2.4 Parallele Optimierung der Regelbasen und Zugehörigkeitsfunktionen

Bei der separaten Optimierungen der Regelbasen bzw. Zugehörigkeitsfunktionen ergab sich, daß durch diese getrennten Ansätze nur marginale Verbesserungen erzielt werden konnten. Aus diesem Grunde werden im Folgenden, um das Reglerverhalten gemäß der in Absatz 6.2.1 dargelegten Strategie zu beeinflussen, sowohl Regelbasen als auch die Zugehörigkeitsfunktionen gleichzeitig bearbeitet.
Zu diesem Zweck muß eine Codierung eingesetzt werden, die beide Einflußfaktoren berücksichtigt. Hierzu werden die Strings der binär codierten Regelbasen und die entsprechend Abschnitt 6.2.3 transformierten Zugehörigkeitsfunktionen verkettet.

$$\underbrace{001\cdots101}_{Regelbasis_i}\ \cdots\ \underbrace{111\cdots010}_{Regelbasis_n}\underbrace{0111001\cdots1101001}_{Zugehörigkeitsfunktion_1}\ \cdots\ \underbrace{1001101\cdots0101111}_{Zugehörigkeitsfunktion_n}$$

Die so entstandenen Bitketten hatten deshalb eine Länge von *1251* Bits und wurden in dieser Form zur gleichzeitigen Optimierung der beiden Einflußfaktoren verwendet.
Bei verschiedenen Simulationen im Vorfeld wurde mit einer Mutationsrate zwischen 0.005 und 0.03 sowie eine Crossover-Wahrscheinlichkeit von 0.6 gearbeitet. Es stellte sich heraus, daß eine Mutationsrate von 0.01 zu den besten Ergebnissen führte. Mit dieser Mutationswahrscheinlichkeit werden pro String im Durchschnitt 12 Bits mutiert, so daß der Einfluß einigermaßen begrenzt ist. Bei einer Erhöhung der Rate wurden soviele Parameter verändert, daß oft schon nach wenigen Generationen sehr viele unbrauchbare Controller entstanden. Bei einer Rate kleiner 0.01 war die Laufzeit des genetischen Algorithmus bis sich eine verbesserte Lösung ergab zu groß. Die Programmlaufzeit wurde unakzeptabel lang.
In den folgenden Abschnitten werden die Ergebnisse der Fuzzy Controller dargestellt, die bei Einsatz der unterschiedlichen Fitnessfunktionen die höchsten Fitnesswerte aufwiesen.

### 6.2.5 Simulation

Bei den in Abschnitt 6.2.6 dargestellten Ergebnisse, handelt es sich um Mittelwerte, die sich aus einer repräsentativen Anzahl von Simulationen ergeben haben. Bei einer Aussagesicherheit von 95% ist der relative Fehler stets $\leq 10\%$.

### 6.2.6 Bestimmung der Fitness Funktion

Nachdem die Codierung der Fuzzy Controller erfolgt ist, muß nun eine geeignete Fitness Funktion ermittelt werden, die eine geeignete Optimierung des Policing Controllers, unter

Berücksichtigung der beschriebenen Randbedingungen, zuläßt.

### Fitness Funktion 1

Ein erster, naheliegender Ansatz ist die Ermittlung der Fitness des Bedien-Controllers als Summe aller Verlustraten $LR_i$ an den Warteschlangen der einzelnen Dienste über den gesamten Simulationszeitraum:

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \tag{6.6}$$

Da hierbei jedoch der Verlust von Datenpaketen aller Quellen, unabhängig von den vorgegebenen Verkehrsparametern $BW_\Delta$ und der dienstabhängigen Priorität $P_S$, in gleichem Maße zu einem Fitnesswert beiträgt, ist dieser erste Ansatz ungeeignet. Auf Grund dieses offenkundigen Mängels kommt dieses Bewertungsschema nicht zum Einsatz.

### Fitness Funktion 2

Eine Verbesserung des ersten Ansatzes wird durch die Multiplikation der Verlustrate $LR_i$ vor der Aufsummierung mit der dem Dienst assoziierten Priorität $P_S$ erreicht.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \cdot P_{S,i} \tag{6.7}$$

Dieser Ansatz wurde für die ersten Untersuchungen zur separaten Optimierung der Regelbasis verwendet. Bei der anschließenden Optimierung von Regelbasis und Zugehörigkeitsfunktionen zeigte sich schon nach wenigen Generationen, daß der Einfluß der Priorität $P_S$ auf den resultierenden Fitnesswert noch zu gering war.

### Fitness Funktion 3

Die Vergrößerung des Einflusses der Priorität $P_S$ auf die Fitness des Controllers wurde durch die Berücksichtigung der quadratisch bewerteten Dienstpriorität erreicht.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \cdot P_{S,i}^2 \tag{6.8}$$

Diese Funktion lieferte dann bei den Untersuchungen deutlich verbesserte Ergebnisse. Bei der Simulation zeigten sich im Wesentlichen vier verschiedene Optima. In der Tabelle 6.6 sind die Auslastung der Warteschlangen und des Kanals sowie die Verlustraten der einzelnen Dienste ($S_i$) dargestellt.

Die Auslastung des Kanals beträgt gemäß Tabelle 6.7 bei allen Ansätzen ca. 89.4%. Unterschiedlich sind die Belegung der Warteschlangen sowie die Verlustraten. Ansatz drei (Generation 127) zeigt eine nahezu den Vorstellungen entsprechende Lösung. Die Dienste

| Auslastung | | | | |
|---|---|---|---|---|
| Dienst | Generation | | | |
| | 5 | 52 | **127** | 164 |
| 1 | 0.125 | 0.043 | **0.186** | 0.091 |
| 2 | 0.012 | 0.153 | **0.006** | 0.206 |
| 3 | 0.120 | 0.000 | **0.001** | 0.021 |
| 4 | 0.079 | 0.151 | **0.018** | 0.207 |
| 5 | 0.208 | 0.183 | **0.258** | 0.006 |
| Mittelwert | 10.88% | 10.6% | **9.38%** | 10.62% |
| **Verluste** | | | | |
| Dienst | Generation | | | |
| | 5 | 52 | **127** | 164 |
| 1 | 0.011 | 0.003 | **0.034** | 0.008 |
| 2 | 0.005 | 0.060 | **0.000** | 0.082 |
| 3 | 0.087 | 0.000 | **0.000** | 0.018 |
| 4 | 0.033 | 0.063 | **0.000** | 0.120 |
| 5 | 0.095 | 0.078 | **0.068** | 0.000 |
| Mittelwert | 4.62% | 4.08% | **2.04%** | 4.56% |

Tabelle 6.6: Kennzahlen der Dienste bei Verwendung des mit Hilfe von Fitness Funktion 3 ermittelten Reglers

| Generation | | | |
|---|---|---|---|
| 5 | 52 | **127** | 164 |
| 0.894 | 0.894 | **0.894** | 0.894 |

Tabelle 6.7: Auslastung des Kanals

2, 3 und 4 haben keine Verluste. Bei den Diensten 1 und 5 treten Verluste in Höhe 3.4% und ca. 6.8% auf.
Um das Verhalten des Controllers genauer analysieren zu können, ist eine Interpretation der beiden internen Zustandsgrößen hilfreich.

### Der Fuzzy Controller 1

Abbildung 6.2 zeigt den Verlauf der temporären Priorität $P_1$ in Abhängigkeit von $BW_\Delta$ und der Priorität $P_S$. Dieses Kennlinienfeld ist stark zerklüftet und weist keine ausgeprägten Plateaus auf. Auffällig ist aber, daß die Prioritäten für die Serivces nahezu den gleichen Wert aufweisen. Die Dienste 1, 2 und 3 haben eine Priorität von ca. 0.39, während sich die Dienste 4 und 5 durch eine leicht erhöhte Priorität von 0.43 bzw. 0.41 ausweisen.

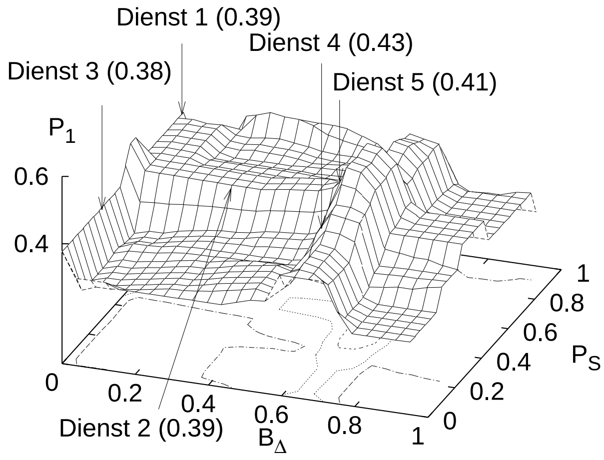



Abbildung 6.2: Kennlinienfeld des Fuzzy Controllers $FC_1$

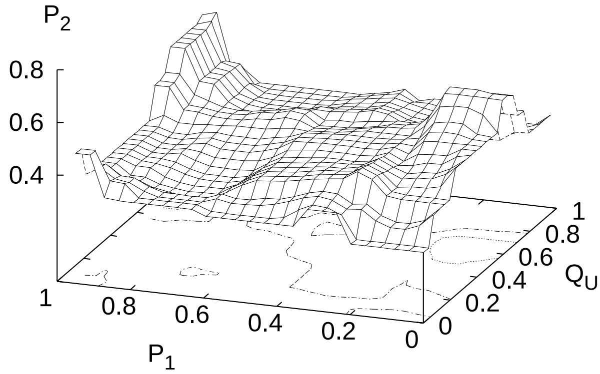



Abbildung 6.3: Kennlinienfeld des Fuzzy Controllers $FC_2$

**Der Fuzzy Controller 2**
Im Fuzzy Controller 2 werden $P_1$ und die Auslastung der Warteschlangen miteinander verknüpft. Da wie Tabelle 6.6 zu entnehmen ist, die Auslastung der Warteplätze der Dienste 2, 3 und 4 relativ gering ist. Die detaillierte Auswertung hat gezeigt, daß die temporären Spitzenwerte der Auslastung für die drei Dienste $< 2\%$ sind. $P_2$ liegt für diese Dienste im Bereich zwischen $[0.5, 0.58]$. Das Intervall für die Dienste 1 und 5 hat wegen der Schwankungen der Auslastung eine größere Streuung. Während sich der Wert bei Dienst 1 im

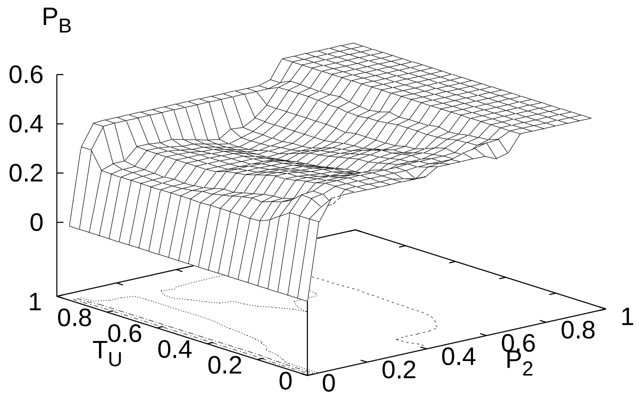



Abbildung 6.4: Kennlinienfeld des Fuzzy Controllers $FC_3$

Bereich von 0.37 bis 0.5 bewegt, steigt der maximale Wert für Dienst 5 auf 0.58 an.

**Fuzzy Controller 3**
Gerade in diesem Bereich ist der Regler stark von der genutzten Kanalkapazität abhängig. Bei einer Auslastung $< 92\%$ können schon geringe Änderungen zu einer Beeinflussung der Bedienpriorität führen. Das Kennlinienfeld für diesen Regler ist in Abbildung 6.4 dargestellt.

**Einfluß der Dienstpriorität**
Einen weiteren relevantern Punkt stellt die Abhängigkeit der Reglers von der Priorität $P_S$ dar. Die Ergebnisse dokumentieren, daß der Regler nicht wesentlich auf Änderungen von $P_S$ reagiert.

Die mit Hilfe dieser Fitnessfunktion berechneten Controller zeigen noch nicht in allen Belangen das erwarteten Verhalten, so daß im Weiteren auch noch der Einfluß von $BW_\Delta$, berücksichtigt wurde.

### Fitness Funktion 4

Um den Einfluß des Verhalten der Dienste mit in den Regler einfließen zu lassen, wurde die Summe (Gl. 6.8) um eine Multiplikation mit dem Faktor $(1 - BW_{\Delta,i})$ erweitert.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \cdot P_i^2 \cdot (1 - BW_{\Delta,i}) \tag{6.9}$$

Der Einsatz dieser Funktion zur Berechnung der Fitness eines Fuzzy Controllers ergab

| Dienst | Auslastung | Verluste |
|---|---|---|
| 1 | 0.392 | 0.050 |
| 2 | 0.073 | 0.000 |
| 3 | 0.013 | 0.001 |
| 4 | 0.056 | 0.010 |
| 5 | 0.112 | 0.006 |
| Mittelwert | 12.92% | 1.34% |
| Kanalauslastung | 0.894 | — |

Tabelle 6.8: Kennzahlen der Dienste bei Verwendung des mit Hilfe von Fitness Funktion 4 ermittelten Reglers

keine Verbesserungen im Vergleich zu der vorhergehenden Optimierung ohne Berücksichtigung der Abweichung der gemessenen von der deklarierten Bandbreite. Die Auslastung der Speicherkapazitäten belief sich auf ca. 13%, die durchschnittliche Fehlerrate ist mit einem Betrag von 1.34% sehr niedrig. Die Verluste der Dienste 1 und 3 konnten aber auch durch eine Änderung der Priorität $P_S$ nicht aufgefangen werden.

### Fitness Funktion 5

Daher wurde anschließend auch hier der quadratische Einfluß der Bandbreiten-Abweichung durch eine Multiplikation mit $(1 - BW_{\Delta,i})^2$ als weiterer Ansatz gewählt.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \cdot P_i^2 \cdot (1 - BW_{\Delta,i})^2 \tag{6.10}$$

Die Kennlinienfelder der resultierenden Fuzzy Controller sind in den Abbildungen im Anhang D.1 auf den Seiten 190ff. dargestellt. Bei der Bewertung der Fitness eines Controllers

| Dienst | Auslastung | Verluste |
|---|---|---|
| 1 | 0.078 | 0.007 |
| 2 | 0.021 | 0.003 |
| 3 | 0.010 | 0.000 |
| 4 | 0.182 | 0.033 |
| 5 | 0.306 | 0.094 |
| Mittelwert | 11.94% | 2.74% |
| Kanalauslastung | 0.894 | — |

Tabelle 6.9: Leistungskennzahlen der Dienste bei Verwendung des mit Hilfe der Fitness Funktion 5 ermittelten Reglers

mit Hilfe der Formel 6.10 ergaben sich die in Tabelle 6.9 gezeigten Leistungskennzahlen.

Die Dienste 1 und 2 weisen Verluste, gemittelt über den gesamten Simulationsverlauf, von 0.7% bzw. ≈ 0.4% auf. Dienst 3 hat keine Verluste. Die Verlustraten der anderen Dienste sind erwartungsgemäß höher, überschreiten aber die 10% Grenze nicht. Der durchschnittliche Wert ist mit 2.74% auch relativ klein. Weiterhin ist eine Auslastung der Warteplätze mit 11.94% gegeben. Das skizzierte Verhalten entspricht daher nur teilweise den in Abschnitt 6.2.1 festgelegten Zielen.

**Einfluß der Dienstpriorität**

Die Abbildung D.4(b) zeigt den Verlauf der mittleren Auslastung der Warteschlangen der einzelnen Dienste als Funktion von $P_S$. Die Belegung der Speicherplätze für Dienst 3 hat den maximalen Wert von 18% bei einer Priorität $P_S$ von 0.2. Im Bereich von 0.3 bis 0.7 ist der Betrag konstant 0.8%. Bei größeren Prioritäten $P_S$ sinkt sie auf 0% ab.
Stark korreliert sind die Dienste 3 und 4. Die auf Grund der größeren Auslastung der Ressourcen freien Übertragungskapazitäten werden durch Dienst 4 genutzt. Die Auslastung sinkt für $P_S = 0.2$ auf einen durchschnittlichen Wert von 6% ab. Dieses Verhalten zeigt sich auch im Verlauf der Verlustrate (Abbildung D.4(a)). Eine vergrößerte mittlere Auslastung ist dafür verantwortlich, daß die Verlustrate ansteigt. Geht die mittlere Belegung der Warteplätze zurück, sinkt auch die Verlustrate. Der genaue zeitliche Verlauf der Auslastung kann den Abbildungen D.6 für $P_S = 0.2$ und D.6 für $P_S = 0.3$ entnommen werden. Es zeigt sich, daß bei diesem Controller die Auslastung und auch die Verlustrate durch die Priorität $P_S$ beeinflußbar ist.

**Fitness Funktion 6**

Um den Einfluß der beiden Faktoren Priorität $P_S$ und die Abweichung der gemessenen Bandbreite $BW_\Delta$ von der deklarierten Bandbreite zu erhöhen, wurden sie anschließend in höheren Potenzen 6.11 zur Berechnung der Fitness eingesetzt.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \cdot P_i^3 \cdot (1 - BW_{\Delta,i})^3 \tag{6.11}$$

Bei diesem Ansatz wurde implizit auch die Gewichtung dieser beiden Faktoren zueinander weiter variiert. Wie der Tabelle 6.10 zu entnehmen ist, kam es offensichtlich zu einer Verbesserung. Bei dem Dienst 1 traten Verluste in Höhe von 1.7%. Die Verlustrate von Dienst 3 ist 0%. Die durchschnittliche Verlustrate beläuft sich auf 2.48%. Die Auslastung des System konnte auf 18.46% angehoben werden.

**Einfluß der Dienstpriorität**

Die Abbildungen D.10(a) und D.10(b) zeigen die Abhängigkeiten der Verlustrate und der Auslastung aller Dienste von der Priorität $P_{Dienst,3}$. Die Verläufe dokumentieren, daß durch die Veränderung der Priorität $P_{Dienst,3}$ gezielt Einfluß auf die Qualitätsparameter genommen werden kann. Durch Erhöhung der Priorität kann die Auslastung der Warteschlange

| Dienst | Auslastung | Verluste |
|---|---|---|
| 1 | 0.277 | 0.017 |
| 2 | 0.140 | 0.003 |
| 3 | 0.076 | 0.000 |
| 4 | 0.104 | 0.002 |
| 5 | 0.326 | 0.102 |
| Mittelwert | 18.46% | 2.48% |
| Kanalauslastung | 0.894 | — |

Tabelle 6.10: Leistungskennzahlen der Dienste bei Verwendung des mit Hilfe der Fitness Funktion 6 ermittelten Reglers

von Dienst 3 gezielt gesteuert werden. Ausreißer ergeben sich bei einer Priorität von 0.2 und 0.8. Möglich wird dieses Verhalten durch die feine Abstufung der temporären Priorität $P_1$ in Abhängigkeit von $P_S$ (Abbildung D.7). Das Kennlinienfeld zeigt bei $P_S = 0.2$ und $P_S = 0.8$ Einbrüche, die mit einer Verringerung der temporären Priorität $P_1$ einhergehen. Dieser Effekt trägt im Folgenden dazu bei, daß sich die Verlustrate aller Dienste nachhaltig ändert. Korreliert mit der Nutzung der Warteplätze ist die Verlustrate. Auch hier konnte durch die Erhöhung der Dienstpriorität systematisch Einfluß auf den Verlauf genommen werden.
Unter Anwendung dieser Fitnessfunktion konnte die Regelstrategie fast vollständig umgesetzt werden. Die systemweite Verlustrate ist niedrig, die Auslastung der Ressourcen ist mit ca. 20% zufriedenstellend. Allein die Forderung, daß der Dienst 1 nicht verlustbehaftet sein darf, konnte nicht realisiert werden.
Um die Abhängigkeit des System von der Verlustwahrscheinlichkeit noch weiter zu steigern, wurde dieser Term im folgenden Ansatz quadratisch berücksichtigt.

**Fitness Funktion 7**

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i^2 \cdot P_i^3 \cdot (1 - BW_{\Delta,i})^3 \tag{6.12}$$

Ausgehend von der Tabelle 6.11 zeigt der Controller, der mit Hilfe dieser Fitnessfunktion erzeugt wurde, daß die verfolgte Regelstrategie hier Berücksichtigung gefunden hat. Die Verlustraten der Dienste 1 und 3 betragen 0%. Die systemweite Verlustrate ist mit 2.96% akzeptabel.
Die Nutzung der Warteschlangen ist gegeben. Die Ressourcen im Zugangsknoten werden zu 10.58% ausgelastet. Problematisch ist, wie die Abbildungen D.14 und D.15 zeigen, die Abhängigkeit der Verlustrate und Auslastung von der Dienstpriorität. Den Diagrammen ist zu entnehmen, daß sich der Einfluß auf einen minimalen Bereich beschränkt.
Fitness Funktion 8 stellt einen weiteren Versuch dar, die Regelstrategie umzusetzen.

| Dienst | Auslastung | Verluste |
|---|---|---|
| 1 | 0.004 | 0.000 |
| 2 | 0.020 | 0.002 |
| 3 | 0.012 | 0.000 |
| 4 | 0.126 | 0.017 |
| 5 | 0.367 | 0.129 |
| Mittelwert | 10.58% | 2.96% |
| Kanalauslastung | 0.894 | — |

Tabelle 6.11: Leistungskennzahlen der Dienste bei Verwendung des mit Hilfe der Fitness Funktion 7 ermittelten Reglers

**Fitness Funktion 8**

$$Fitness_{Gesamt} = \sum_{i=1}^{n} \frac{LR_i \cdot P_i^2}{(1 - BW_{\Delta,i})} \tag{6.13}$$

Unter Verwendung dieser Fitnessfunktion kristallisierten sich gleich drei Lösungen heraus. Die Optima ergaben sich nach 20, 96 und 143 Generationen.

**Fallbeispiel 1: 20.Generation**
Der Controller, der sich nach 20 Generationen ergab, hat eine durchschnittliche Verlustrate von $\leq 4\%$. Der Auslastungsgrad ist mit 10.4% gering. Da die Regelstrategie, auch wegen der Verluste bei Dienst 1, in entscheidenden Punkten nicht umgesetzt werden konnte, erfolgt keine weitere Beschreibung des Verhaltens.

**Fallbeispiel 2: 96 Generation**
Der zweite Controller, der sich nach 96 Generationen ergab, zeichnet sich durch eine mittlere Auslastung der Ressourcen von 17.6% aus. Die Nutzung der Warteplätze ist, wie aus der Tabelle entnommen werden kann, über alle Applikationen verteilt und beschränkt sich nicht nur auf einen Teil der Dienste. Die durchschnittliche Verlustrate ist mit 2.9% akzeptabel. Dienst 1 und 3 haben keine Verluste.
Die Abbildungen D.16 und D.17 zeigen den Einfluß der Dienstpriorität $P_{S,3}$ auf die Verlustrate und Auslastung aller Dienste. Im Bereich $P_{S,3} < 0.7$ ist das Systemverhalten nahezu konstant. Für $P_{S,3} > 0.7$ sinkt die Auslastung der Warteschlange von Dienst 3 geringfügig, was das Verhalten der Dienste 2 und 4 nur peripher beeinflußt.
Mit diesem Controller kann die Regelstrategie in vielen Belangen realisiert werden. Die vertragskonformen Dienste 1 und 3 haben keine oder nur vernachlässigbar kleine Verluste. Die durchschnittliche Verlustrate - gemittelt über alle Dienste - ist gering. Die Auslastung der Warteschlangen ist global gegeben. Die Übertragungsressourcen des Kanals sind optimal genutzt.

| Auslastung | | | |
|---|---|---|---|
| Dienst | Generation | | |
| | 20 | 96 | 143 |
| 1 | 0.157 | 0.123 | 0.423 |
| 2 | 0.146 | 0.209 | 0.003 |
| 3 | 0.030 | 0.070 | 0.000 |
| 4 | 0.117 | 0.185 | 0.002 |
| 5 | 0.067 | 0.293 | 0.003 |
| Mittelwert | 10.34% | 17.6% | 8.62% |

| Verluste | | | |
|---|---|---|---|
| Dienst | Generation | | |
| | 20 | 96 | 143 |
| 1 | 0.015 | 0.000 | 0.065 |
| 2 | 0.087 | 0.033 | 0.000 |
| 3 | 0.019 | 0.000 | 0.000 |
| 4 | 0.051 | 0.027 | 0.000 |
| 5 | 0.026 | 0.087 | 0.000 |
| Mittelwert | 3.96% | 2.94% | 1.3% |

Tabelle 6.12: Leistungskennzahlen der Dienste bei Verwendung des mit Hilfe der Fitness Funktion 8 ermittelten Reglers

| Generation | | |
|---|---|---|
| 20 | 96 | 143 |
| 0.894606 | 0.894 | 0.895 |

Tabelle 6.13: Auslastung des Kanals

**Fallbeispiel3: 143.Generation**
Der dritte Controller entwickelte sich nach 143 Generationen. Die mittlere Auslastung beträgt nur 8.6%. Die Nutzung der Warteplätze beschränkt sich allerdings, wie aus der Tabelle 6.12 entnommen werden kann, im Wesentlichen auf Dienst 1. Der Ausnutzungsgrad hat einen beachtlichen Betrag von ca. 42.3%. Die durchschnittliche Verlustrate - gemittelt über alle Dienste - beträgt nur 1.3%. Verluste treten nur bei Dienst 1 in Höhe von 6.5% auf.
Die Abbildungen D.18 und D.19 dokumentieren die Abhängigkeit der Verlustrate und Auslastung von der Dienstpriorität $P_{S,3}$. Für $P_{S,3} < 0.8$ ist das Systemverhalten konstant. Eine Abhängigkeit von der Priorität ist nicht nachweisbar. Erst für $P_{S,3} > 0.8$ sinken die Verlustrate und Auslastung des Dienstes 1. Dieses Verhalten wird, wie die Kennlinien zeigen, durch Dienst 3 kompensiert. Die Verläufe zeigen, daß in diesem Fall eine Erhöhung der Dienstpriorität nicht zu einer Verbesserung der Übertragungsqualität führt. Vielmehr

wird die Wertigkeit des Dienstes herabgesetzt, wodurch dann vermehrt Verluste auftreten. Der Regler zeigt ein *inverses* Verhalten.
Mit diesem Controller konnte die Regelstrategie nicht realisiert werden. Die Verluste sind zwar sehr gering, beschränken sich jedoch, wie auch die Auslastung, nur auf Dienst 1. Die Abhängigkeit von $P_{S,3}$ ist minimal und invers zu dem dem gewünschten Verhalten.
Bei den Fitness Funktionen 9 und 10 werden die Dienstpriorität $P_S$ und die Abweichung von der deklarierten Bandbreite mit höheren Potenzen berücksichtigt.

### Fitness Funktion 9

In Gl. 6.14 werden die Dienstpriorität und die Abweichung von der deklarierten Bandbreite quadratisch berücksichtigt.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} \frac{LR_i \cdot P_i^2}{(1 - BW_{\Delta,i})^2} \tag{6.14}$$

Die Kennzahlen des Controllers, der sich nach 94 Generationen ergab, sind in Tabelle 6.14

| Dienst | Auslastung | Verluste |
|---|---|---|
| 1 | 0.421 | 0.065 |
| 2 | 0.008 | 0.000 |
| 3 | 0.002 | 0.000 |
| 4 | 0.010 | 0.000 |
| 5 | 0.005 | 0.000 |
| Mittelwert | 8.92% | 1.3% |
| Kanalauslastung | 0.895225 | — |

Tabelle 6.14: Leistungskennzahlen der Dienste bei Verwendung des mit Hilfe der von Fitness Funktion 9 ermittelten Reglers

wiedergegeben. Der Controller zeichnet sich durch eine asymmetrische Verteilung der Last aus. Der Auslastungsgrad wird im Wesentlichen durch die Belegung der Warteschlange von Dienst 1 bestimmt. Ebenso wie bei dem in Abschnitt 6.2.6 Controller treten die Verluste nur bei Dienst 1 auf.
Die Abbildungen D.20(a) und D.20(b) zeigen, daß $P_{S,3}$ keinen Einfluß auf auf das Verhalten die Auslastung und Verlustrate der anderen Dienste hat.
Das beschriebene Verhalten demonstriert, daß der Regler sich unter Verwendung der Fitness Funktion (Gl. 6.14) ergeben hat, nicht geeignet ist, die Regelstrategie umzusetzen.

**Fitness Funktion 10**

In einem weiteren Versuch werden die Dienstpriorität und die Abweichung von der deklarierten Bandbreite kubisch berücksichtigt.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} \frac{LR_i \cdot P_i^3}{(1 - BW_{\Delta,i})^3} \tag{6.15}$$

Nach 64 Generationen hate sich ein Controller entwickelt, der durch die Kennzahlen in Tabelle 6.15 charakterisiert wird. Die mittlere Auslastung liegt nach Tabelle 6.15 bei ca.

| Dienst | Auslastung | Verluste |
|---|---|---|
| 1 | 0.140 | 0.016 |
| 2 | 0.069 | 0.005 |
| 3 | 0.038 | 0.000 |
| 4 | 0.138 | 0.032 |
| 5 | 0.247 | 0.068 |
| Mittelwert | 12.64% | 2.42% |
| Kanalauslastung | 0.895 | — |

Tabelle 6.15: Leistungskennzahlen der Dienste bei Verwendung des mit Hilfe der Fitness Funktion 10 ermittelten Reglers

12.64%, die Verluste betragen durchschnittlich 2.42%. Die Verlustrate beläuft sich, entgegen der Regelstrategie, bei Dienst 1 auf 1.6%.
Die Abhängigkeit der Verlustraten und die Auslastung der Warteschlangen für unterschiedliche Dienstprioritäten $P_{S,3}$ ist in den Abbildungen D.24(a) und D.24(b) dargestellt. Die Verlustrate von Dienst 3 ist invariant gegenüber Änderungen der Dienstpriorität. Sie beträgt durchgängig 0%. Die Auslastung von Dienst 3 weist beim Übergang von $P_{S,3} = 0.1$ nach $P_{S,3} = 0.2$ eine Veränderung auf. Dies kann mit Hilfe der Kennlinienfelder, die das Übertragungsverhalten des Reglers reflektieren, schlüssig erklärt werden. Das Übertragungsverhalten des Fuzzy Controllers 1 ist in Abbildung D.21, im Anhang auf der Seite 203, darstellt. Es ist verhältnismäßig eben und weist nur für kleine Dienstprioritäten und bei sehr kleinen Abweichungen von der deklarierten Bandbreite einen Gipfel auf. Da Dienst 3 vertragskonform betrieben wird, gilt $BW_{\Delta} = 0$, so daß $P_1$ den Wert 0.7 annimmt. Dieser Wert wird auf Grund der geringen Auslastung der Warteschlange durch den Fuzzy Controller 2 unverändert weitergegeben und führt am Ausgang von $FC_3$ zu einer hohen Bedienpriorität, die dafür verantwortlich ist, daß keine Verluste auftreten. Bei Erhöhung der Dienstpriorität ($P_{S,3} = 0.2$) liegt der Arbeitspunkt nicht mehr auf dem Gipfel im Kennlinienfeld D.21, was zu einer Absenkung der temporären Priorität $P_1$ führt. Diese Verringerung zieht eine Verkleinerung der Bedienpriorität nach sich. Dienst 3 konkurriert stärker mit den anderen Diensten um die Übertragungskapazität. Als Folge davon steigt die Auslastung der Ressourcen an. Auch in diesem Fall ist das Verhalten des Controllers

wiederum invers.
Der weitere Verlauf der Kennlinien scheint unkorreliert. Er resultiert daraus, daß die Übertragungscharakteristik D.22 steile Kanten gerade im Bereich des Arbeitspunktes (für $Q_U \approx 20\%$) aufweist. An diesen Stellen wird die kontinuierliche Übertragungscharakteristik unterbrochen. Kleine Änderungen von Eingangssignalen bewirken so Sprünge der Ausgangsgrößen. Eine gezielte Beeinflussung des Verhaltens des Controllers ist daher nicht gegeben.
Auf Grund des beschriebenen Zusammenhänge ist dieser Ansatz nicht dazu geeignet, die Regelstrategie umzusetzen.

### 6.2.7 Fuzzy Logic basierte Fitness-Funktion

Die in Abschnitt 6.2.6 ausgeführten Beschreibungen zur Herleitung einer geeigneten Fitnessfunktion zeigen die Komplexität und Schwierigkeiten bei diesem Verfahren. In vielen Fällen setzen diese Ansätze ein tiefgründiges Verständnis des Systems sowie der Parameter und deren Abhängigkeiten voneinander voraus. Durch viele Iterationen kann dann eventuell eine angemessene Lösung angenähert werden.
Um diese Probleme zu umgehen, wird im Folgenden mit einem **neuen** Verfahren die Fitness einer Population durch einen weiteren Fuzzy Controller bestimmt. Abbildung 6.5 skizziert die Struktur dieser Methode. Der Fuzzy Controller 1 dient wie bisher zur Überwachung

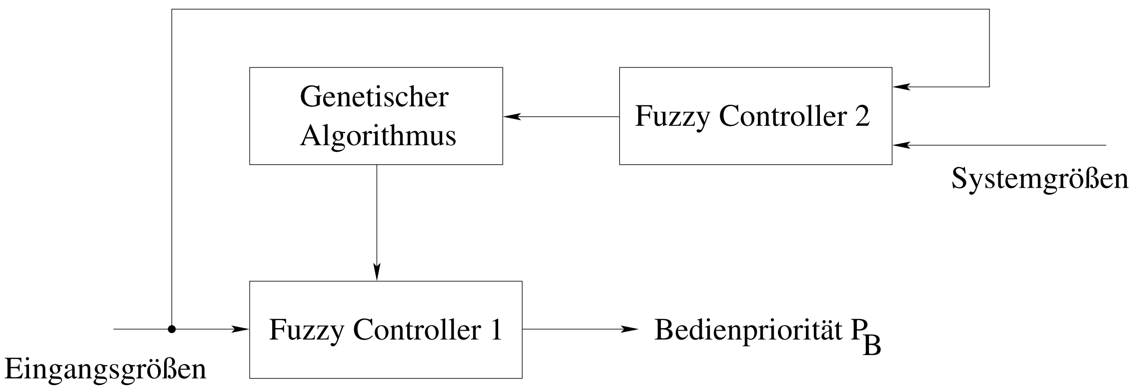



Abbildung 6.5: Optimierung mit Hilfe einer Fuyy Logic basierten Fitness Funktion

und Regelung der Datenströme, indem die Bedienpriorität bei einer Übertragung für jeden Dienst bestimmt wird. $P_B$ wird dabei aus unterschiedlichen Systemgrößen abgeleitet.
Zur Optimierung wird auch wieder ein genetischer Algorithmus angewandt. Im Gegensatz zu den bisherigen Untersuchungen wird jedoch ein weiterer Fuzzy Controller zur Bestimmung der Fitness einer Population eingesetzt. Wie bei den anderen Simulationen ist die Populationsgröße 20.

### Aufbau des Fitness Controllers

Abbildung 6.6 zeigt den schematischen Aufbau des Fuzzy Logic basierten Fitness Controllers (FLFC). Die Struktur ist aus den in Abschnitt 5.3.2 genannten Gründen wieder hierarchisch aufgebaut. Die linguistischen Variablen (Abbildungen D.25(a) bis D.27 auf

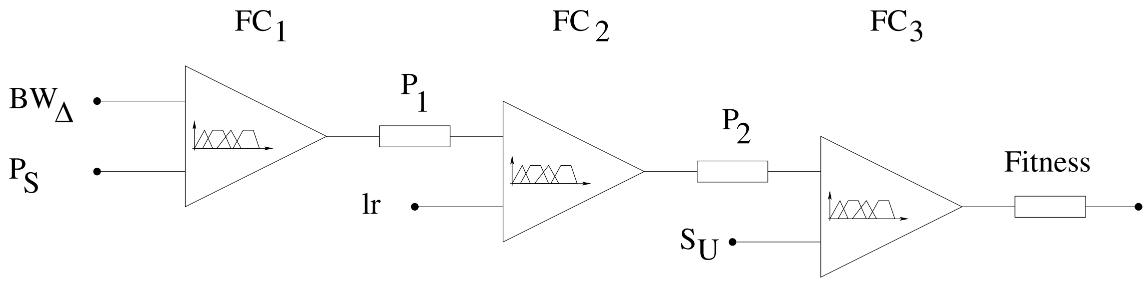



Abbildung 6.6: Hierarchischer Aufbau des Fitness Controllers

den Seiten 206ff) werden wieder durch fünf Sets beschrieben. Die Termbasis setzt sich in allen Fällen aus den folgenden Termen zusammen.

Sehr Klein (SK) - Klein (K) - Mittel (M) - Groß (G) - Sehr Groß (SG)

Das Übertragungsverhalten der einzelnen Teilcontroller ist in den Abbildungen 6.7 bis 6.9 dargestellt.

**Der Fuzzy Controller** $FC_1$
In der Einheit $FC_1$ (Abb. 6.7) werden die Dienstpriorität $P_S$ und die Abweichung von der deklarierten Bandbreite verarbeitet. Das Ergebnis resultiert in einer effektiven Dienstpriorität $P_1$. Leitgedanke beim Entwurf des Regler war, daß diese wirksame Priorität direkt von der Dienstpriorität $P_S$ abhängig ist. Steigt $P_S$, so wird auch $P_1$ größer. Durch $BW_\Delta$ wird die Priorität relativiert. Abweichungen von der deklarierten Bandbreite führen zu einer Verminderung der Priorität.
Der Controller weist einige Plateaus und zwei Kanten bei $P_S \approx 0.4$ und $P_S \approx 0.6$ auf, die das Übertragungsverhalten nachhaltig beeinträchtigen können.

**Der Fuzzy Controller** $FC_2$
Der Fuzzy Controller 2 verknüpft die effektive Priorität $P_1$ mit der Verlustrate $LR$ (Abb. D.26). Die Ausgangsgröße ist die temporäre Priorität $P_2$. Richtungweisend bei der Entwicklung der Regelbasis dieses Teilcontroller war, einen *direkten* Zusammenhang zwischen $P_2$ und der Verlustrate $LR$ sowie der temporären Priorität $P_1$ zu konstruieren[2]. D. h., daß der Betrag von $P_2$ ansteigt, wenn entweder $P_1$ oder $LR$ größer werden.

[2]Ausgangspunkt ist der in Abschnitt 6.1.1 entwickelte Optimierungsansatz.

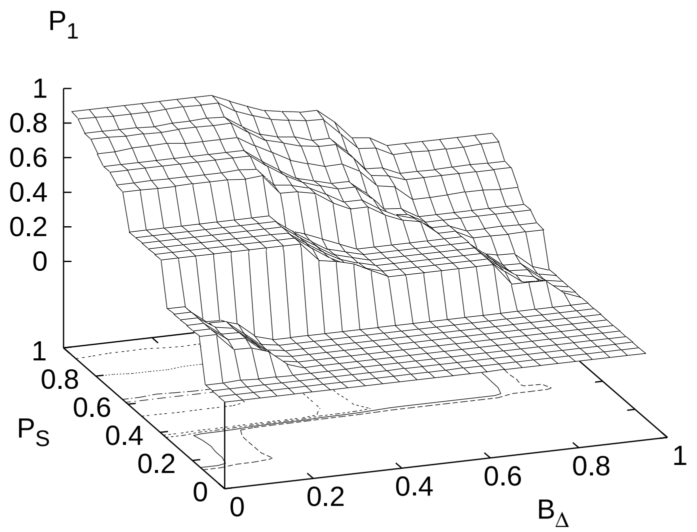



Abbildung 6.7: Übertragungsverhalten des Fitness Controllers $FC_1$

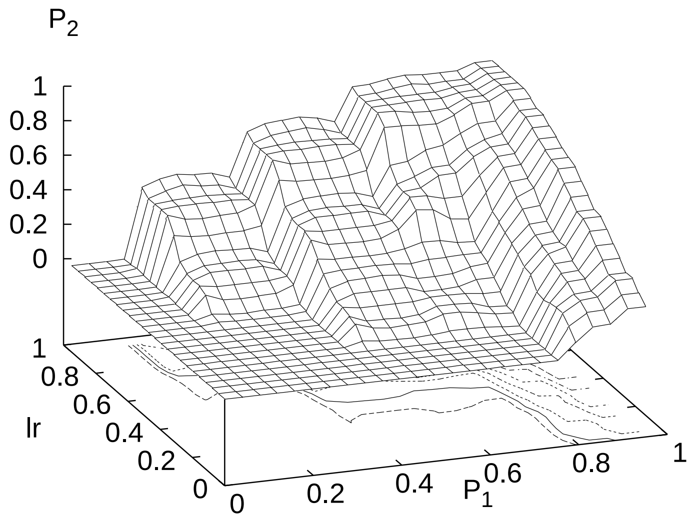



Abbildung 6.8: Übertragungsverhalten des Fitness Controllers $FC_2$

**Der Fuzzy Controller** $FC_3$
Ein weiterer Punkt der Regelstrategie betraf die Systemauslastung. Deshalb werden in dem

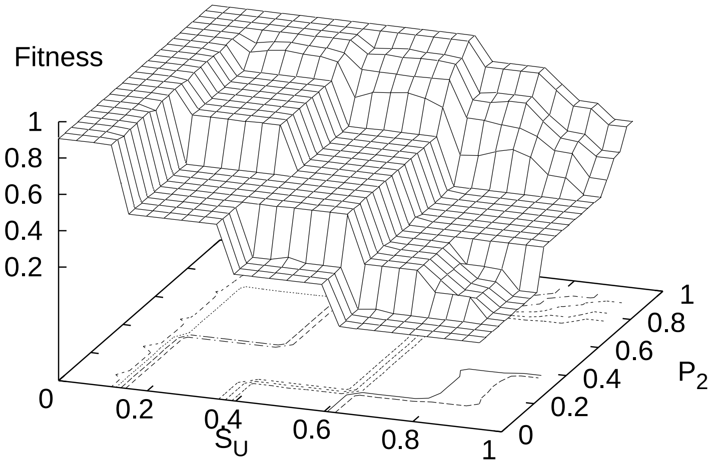



Abbildung 6.9: Übertragungsverhalten des Fitness Controllers $FC_3$

Teilcontroller $FC_3$ die temporäre Priorität $P_2$ und $S_U$ herangezogen, um die Ausgangsgröße, die Fitness der entwickelten Population abzuschätzen. Das Übertragungsverhalten ist in Abbildung 6.9 dargestellt. Auch hier war die Grundlage bei Entwurf der Regelbasis, daß ein großer Ausgangswert einen minderwertigen Controller indiziert[3]. So führt eine geringe Auslastung der Ressourcen ebenso wie eine große temporäre Priorität $P_2$ zu einer gesteigerten Fitness. Ausgehend von dem ursprünglichen Fuzzy Controller zur Verwaltung der Datenströme ergaben sich bei Verwendung des FLFC-Verfahrens vier Optima (Tab. 6.16).

### 6.2.8 Auswertung der Controller

**Fallbeispiel 1: 8.Generation**

Die mittlere Auslastung der Ressourcen des Controllers ist mit 18.24% sehr hoch. Die Verlustrate entspricht mit 2.94% ebenso den Erwartungen, die in der Regelstrategie fixiert wurden. Bei Dienst 1 treten keine Verluste auf, die Verlustrate von Dienst 3 ist vernachlässigbar klein. Die Auslastung des Kanals ist mit $\approx$ 89% optimal.
Abbildung D.28 zeigt die Abhängigkeit der Verlustraten und der Auslastung von der Dienstpriorität $P_{S,3}$. Die Verlustraten der einzelnen Dienste sind invariant bezüglich einer Änderung von $P_{S,3}$. Die Auslastung der Warteschlange von Dienst 3 kann allerdings gezielt beeinflußt werden. Wenn $P_{S,3} > 0.5$, sinkt die Auslastung der dienstspezifischen Ressourcen von 12% auf $\approx$ 2%. Bei einem Wert von $P_{S,3} > 0.7$ beträgt die Auslastung nur

[3]Die Optimierung erfolgt konform zu Abschnitt 6.1.1.

| Auslastung | | | | |
|---|---|---|---|---|
| Dienst | Generation | | | |
| | 8 | 29 | 63 | 127 |
| 1 | 0.006 | 0.203 | **0.135** | 0.017 |
| 2 | 0.152 | 0.172 | **0.200** | 0.214 |
| 3 | 0.163 | 0.179 | **0.083** | 0.058 |
| 4 | 0.247 | 0.265 | **0.103** | 0.178 |
| 5 | 0.344 | 0.183 | **0.261** | 0.429 |
| Mittelwert | 18.24% | 20.04% | 15.64% | 17.92% |
| Verluste | | | | |
| Dienst | Generation | | | |
| | 8 | 29 | 63 | 127 |
| 1 | 0.000 | 0.006 | **0.000** | 0.000 |
| 2 | 0.004 | 0.023 | **0.039** | 0.011 |
| 3 | 0.000 | 0.025 | **0.000** | 0.002 |
| 4 | 0.030 | 0.062 | **0.000** | 0.018 |
| 5 | 0.113 | 0.020 | **0.109** | 0.116 |
| Mittelwert | 2.94% | 2.72% | 2.96% | 2.94% |

Tabelle 6.16: Leistungskennzahlen der Dienste bei Verwendung des mit Hilfe eines Fuzzy Controllers ermittelten Reglers

| Generation | | | |
|---|---|---|---|
| 8 | 29 | 63 | 127 |
| 0.894748 | 0.895 | 0.894 | 0.894 |

Tabelle 6.17: Auslastung des Kanals

noch $\approx$ 0.1%. Kompensiert wird dieser Effekt durch eine erhöhte Nutzung der Speicherplätze der Dienste 2, 4 und 5.
Die Analyse der umfangreichen Kenndaten zeigt, daß der mit Hilfe des FLFC-Verfahrens ermittelte Controller in der Lage ist die Regelstrategie zu realisieren.

**Fallbeispiel 2: 29. Generation**

Ein weiteres Optimum hat sich bei der Generation 29 ergeben. Die Auslastung der Ressourcen bei Simulation des ermittelten Controllers ist mit ca. 20% sehr groß. Sie ist, wie Tabelle 6.16 zeigt, gleichmäßig über alle Dienste verteilt. Die systemweite Verlustrate ist mit ca. 2.8% akzeptabel. Nachteilig wirken sich die Verluste von Dienst 1 in Höhe von 0.6% aus.
Die Abhängigkeit der Systemgrößen von $P_{S,3}$ wird durch die Kennlinien in Abbildung D.29

illustriert. Es zeigt sich, daß durch $P_{S,3}$ das Systemverhalten nachhaltig beeinflußt werden

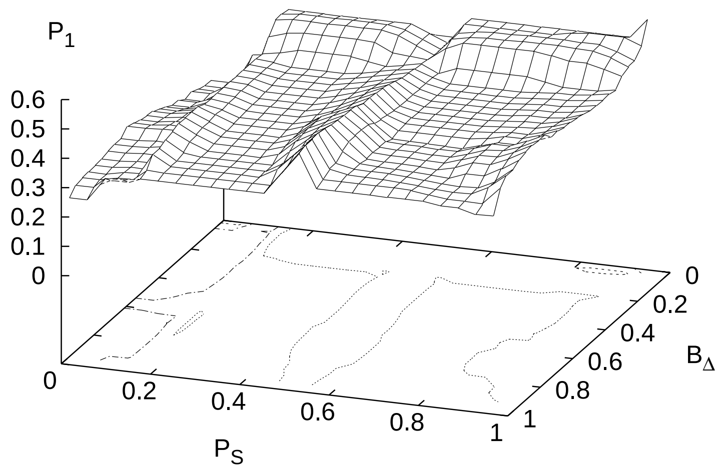



Abbildung 6.10: Übertragungskennlinien des $FC_1$

kann. Für $P_{S,3} \leq 0.4$ ist das System invariant gegenüber Änderungen. Ist $P_{S,3} = 0.5$ steigen sowohl die Auslastung als auch die Verluste von Dienst 3 an. Was zu einer Entlastung der übrigen Dienste führt. Die Übertragungscharakteristik 6.10 zeigt, daß die temporäre Priorität bei $P_S = 0.5$ ein lokales Extremum aufweist, das für ein Ansteigen der Auslastung und der Verluste von Dienst 3 verantwortlich ist. Für $P_{S,3} > 0.5$ ist die Verlustrate dann konstant 0%, die Auslastung sinkt auf $\approx 5\%$. Im Wesentlichen werden die Effekte durch die Dienste 2, 4 und 5 aufgefangen. Dienst 1 wird nicht beeinflußt.
Das Verhalten des Controllers entspricht, trotz seiner guten Kennwerte, wegen der Verluste von Dienst 1 nicht der Regelstrategie.

### Fallbeispiel 3: 63. Generation

Der Regler, der sich nach 63 Generationen ergab, ist durch eine mittlere Auslastung von 15.6% und eine durchschnittliche Verlustrate von $\leq 3\%$ gekennzeichnet. Die Belegung der Ressourcen erstreckt sich gleichmäßig über alle Dienste. Die Verluste der Dienste 1, 3 und 4 belaufen sich auf 0%.
Die Abhängigkeit der Verlustrate und der Auslastung von der Dienstpriorität $P_{S,3}$ ist in Abbildung D.30 dargestellt. Es zeigt sich, daß nach der Erhöhung der Priorität $P_{S,3}$ auf 0.3 für diesen Dienst keine Verluste mehr auftreten. Im Weiteren Verlauf erfolgt dann aber keine sichtbare Beeinflussung der Verlustraten.
Die Auslastung der Ressourcen dagegen ist an die Priorität gebunden. Durch die Erhöhung

von $P_{S,3}$ nimmt die korrespondierende Auslastung stetig ab. Für $P_{S,3} = 0.1$ und $P_{S,3} = 0.3$ ist der Abfall des Auslastungsgrades stärker als im übrigen Verlauf der Kennlinie. Die zunehmende Priorität bewirkt ein Ansteigen der Bedienpriorität, so daß Dienst 3 häufiger bearbeitet wird. Diese Übertragungskapazität geht dann anderen Diensten verloren. In dem speziellen Fall erfolgt eine Beschneidung der Bandbreite von Dienst 4. Ausgeglichen wird die fehlende Transferkapazität durch eine stärkere Auslastung der Warteplätze.
Die Auswertung der vorliegenden Kenndaten hat gezeigt, daß auch dieser mit dem FLFC-Verfahren entworfene Controller geeignet ist, die Regelstrategie zu realisieren.

**Fallbeispiel 4: 127. Generation**

Der nach der 127sten Generation ermittelte Controller hat eine mittlere Auslastung von 17.9% und eine Verlustrate von 2.94%. Die Belegung der Ressourcen ist im Wesentlichen auf die nicht vertragskonformen Dienste beschränkt. Bei den Dienste 1 und 3 werden die Warteplätze nur zu 1.7% bzw. 5.8% ausgelastet. Bei Dienst 1 treten keine Verluste auf, die Verlustrate von Dienst 3 ist mit 0.2% tolerierbar.
Die Abhängigkeit der Auslastung und der Verlustrate von $P_{S,3}$ ist in Abbildung D.31 dargestellt. Durch eine Änderung der Priorität werden die anderen Dienste nur peripher beeinflußt. Im Wesentlichen wird nur das eigene Verhalten gesteuert. Eine Erhöhung von $P_{S,3}$ auf 0.3 bewirkt ein Ansteigen der Auslastung und das Auftreten von Verlusten. Erst wenn $P_{S,3} > 0.7$, nehmen Auslastung und Verlustrate wieder ab.
Auf Grund der Kenndaten, weist sich auch dieser Controller als eine anwendbare Lösung aus. Nachteilig wirkt sich jedoch die nur begrenzte Abhängigkeit der Verlustrate und der Auslastung von der Priorität $P_{S,3}$ aus.

### 6.2.9 Beurteilung der Controller

In diesem Kapitel wurden die Fuzzy Controller zur Regelung der Datenströme mit Hilfe genetischer Algorithmen entworfen. Die Optimierung setzt eine sog. Fitness Funktion voraus, mit deren Hilfe die Bewertung der erzeugten Regler erfolgt. Es wurden verschiedene arithmetische Funktionen entwickelt, die dann zur Beurteilung der konstruierten Regler, herangezogen wurden. Eine nachfolgende Simulation der Controller, die sich durch eine hohe Fitness ausweisen konnten, mit einem vorgegebenen Lastmuster, lieferte dann die Möglichkeit, die Leistungsfähigkeit der Ansätze zu bewerten.
Es zeigte sich, daß mit diesem Verfahren eine Vielzahl von potentiellen, qualitativ hochwertigen Lösungen hervorgebracht werden konnte. Die Anzahl „guter“ Controller, die die genetischen Algorithmen bei den unterschiedlichen Fitness Funktionen entwickelten, war unterschiedlich und beschränkte sich nicht auf nur ein Ergebnis.
Durchgängig konnte bei allen Reglern festgestellt werden, daß die Kanalbandbreite optimal genutzt wurde. Die durchschnittliche Auslastung der Speicherkapazität lag zwischen 10% und 20%. Die Verteilung auf die einzelnen Dienste war jedoch unterschiedlich. Während bei einigen Ansätzen die Last auf alle Dienste verteilt wurde, existieren auch Regler (z. B. Tab. 6.14) bei denen sich die Nutzung der Ressourcen auf einen Dienst beschränkt. Wei-

terhin konnte durch das Verfahren in allen untersuchten Fällen sichergestellt werden, daß die systemweite Verlustrate gering war. Auch hier konnten zwei Tendenzen festgestellt werden. Bei einigen Ansätzen entwickelten sich die Verlustraten entsprechend der in Abschnitt 6.2.1 gegebenen Regelstrategie. Bei anderen Lösungen, die sich bei Anwendung der Fitness Funktionen 8 und 9 ergaben, konnte ein inverses Verhalten eruiert werden. Der Dienst mit der höchsten Priorität wies als einziger Verluste auf. Durch diese Konstellation konnte die durchschnittliche Verlustrate sehr klein gehalten werden. Dieses inverse Verhalten zeigte sich dann auch bei der Auswertung der Kennlinien, die die Abhängigkeit der Verlustraten und die Auslastung der Warteplätze von der Dienstpriorität $P_{S,3}$ dokumentieren. Hier führte eine Vergrößerung von $P_{S,3}$ dann zu einer Vergrößerung der Verluste.
Zusammenfassend kann jedoch festgestellt werden, daß der Entwurf von Reglern mit Hilfe der genetischen Algorithmen im Allgemeinen qualitativ hochwertige Controller lieferte, mit denen es meist möglich war, die vorgegebene Regelstrategie zu realisieren. Schwierigkeiten traten jedoch bei der Entwicklung der Fitness Funktion auf. Die Einflüsse der einzelnen Parameter auf das Systemverhalten sowie die Zusammenhänge und Abhängigkeiten der Signale untereinander sind auch/gerade für die automatische Entwicklung von Reglern essentiell.
Aus diesem Grund wurde ein *neues* Verfahren, bei dem die Fitness einer Population durch einen unscharfen Regler bestimmt wird, eingesetzt. Die Analyse der Ergebnisse in Abschnitt 6.2.6 hat gezeigt, daß mit Hilfe der Fuzzy Logic relativ einfach und ohne detaillierte Kenntnisse um die Zusammenhänge zwischen den einzelnen Parametern, allein auf der Basis von umgangssprachlichen Konstrukten, Regler entwickelt werden können, mit denen die Regelstrategie umgesetzt werden kann.

Bei vielen automatisch konstruierten Systemen konnte festgestellt werden, daß die Membershipfunktionen stark verändert und auch innerhalb des Wertebereichs verschoben waren. Die Abbildungen 6.11 bis 6.13 zeigen die Termbasen der linguistischen Variablen $BW_{\Delta}$, $P_S$ und $P_1$, die sich nach der 52sten Generation unter Verwendung der Fitness Funktion 3 (Gl. 6.8) eingestellt haben. In der Abbildung 6.11 überlappen sich die beiden Terme „Klein“ und „Mittel“ . Der Einflußbereich der Membershipfunktion, die den Term „Mittel“ beschreibt erstreckt sich von 0 bis $\approx 0.85$. Der Bereich des Terms „Klein“ reicht von $\approx 0.17$ bis $\approx 0.9$. Demnach setzt der Einfluß des Ausdrucks „Mittel“ früher ein als der des Terms „Klein“
;. Dieser hat aber dann einen Einfluß, der sich fast über den gesamten Wertebereich erstreckt. Den beiden Graphen 6.12 und 6.13 können noch weitere Beispiele entnommen werden, bei denen die Bezeichnung sowie Lage und Form der Membershipfunktion nicht mehr konsistent sind. Eine einfache Vertauschung der Terme ist aber nicht möglich, da die Regelbasis (Tab. 6.18) genau auf diese Sets abgestimmt ist. Darüber hinaus kann auch nicht immer, wie in dem obigen Beispiel gezeigt, eine *eindeutige* Zuordnung zwischen den Sets erfolgen.
Dieser Sachverhalt unterstreicht aber nochmals eindeutig die in Abschnitt 5.4.8 beschriebene Motivation zur Begründung des Einsatzes computergestützter Verfahren zur Anpassung der Regelbasen und Zugehörigkeitsfunktionen. Eine weitere Analyse dieser Problematik geht über den Umfang dieser Arbeit hinaus und soll an anderer Stelle erfolgen.

| $P_1$ | | $P_S$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $BW_\Delta$ | SK | K | G | M | K | M |
| | K | M | K | SG | G | M |
| | M | M | M | SK | K | SG |
| | G | SK | G | K | G | SK |
| | SG | M | G | K | K | K |

Tabelle 6.18: Darstellung der Regelbasis des $FC_1$, die sich nach 52 Generationen unter Verwendung von Fitness Funktion 3 ergab.

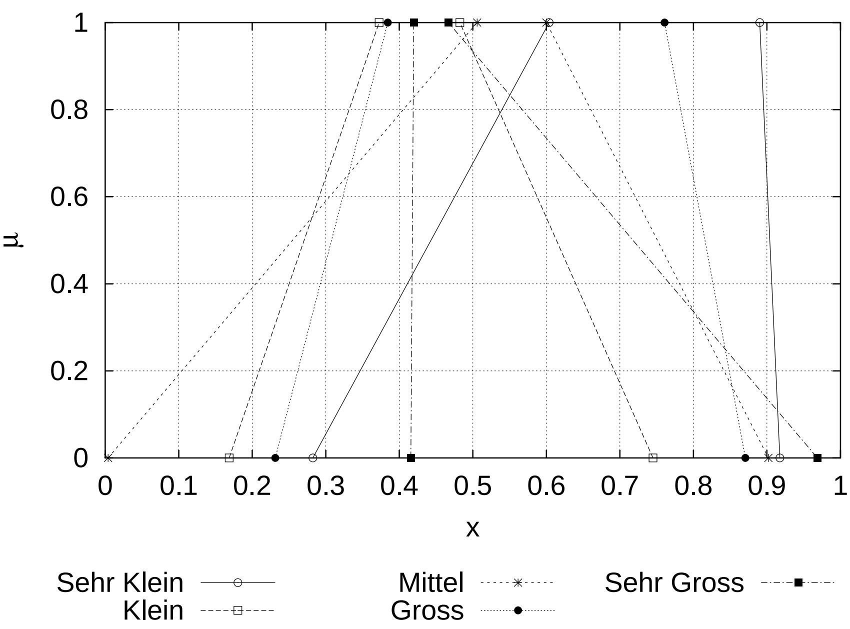



Abbildung 6.11: Darstellung des linguistischen Terms $BW_\Delta$ nach 52 Generationen unter Verwendung von Fitness Funktion 3

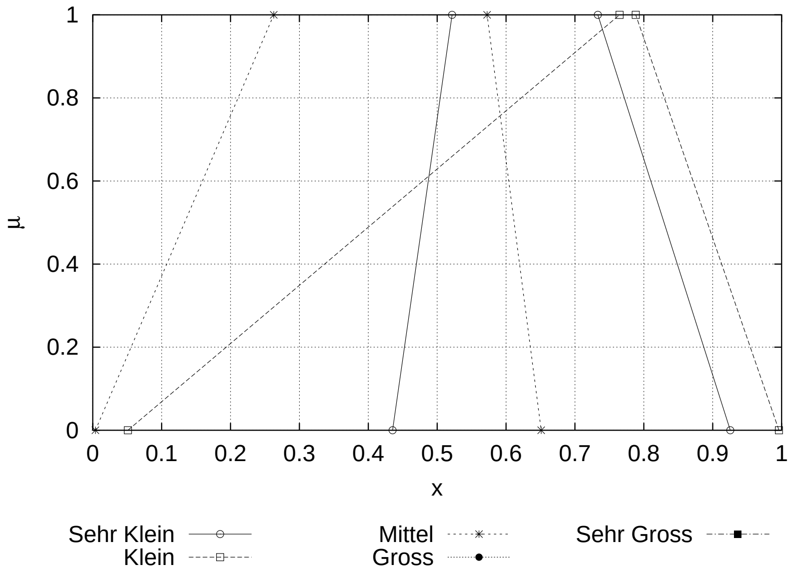



Abbildung 6.12: Darstellung des linguistischen Terms $P_S$ nach 52 Generationen unter Verwendung von Fitness Funktion 3

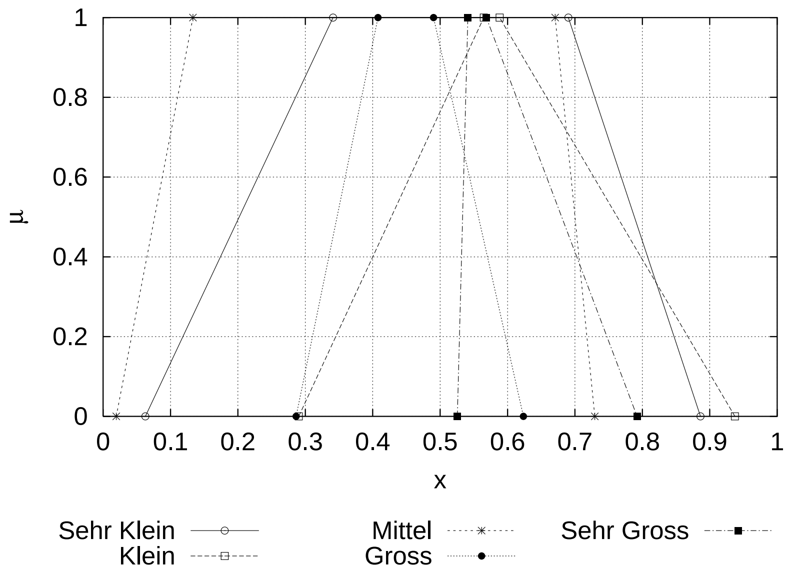



Abbildung 6.13: Darstellung des linguistischen Terms $P_1$ nach 52 Generationen unter Verwendung von Fitness Funktion 3

# Kapitel 7

# Der Call Admission Controller

Wie in der Einleitung beschrieben, ist das Umfeld, in dem die Call Admission Controller eingebettet sind, heterogen. Die Dienste unterscheiden sich deutlich in ihren Verkehrscharakteristiken. Das Verhalten der Benutzer ist indifferent und die weitere Entwicklung der Applikationen sowie deren Anforderungen an das Übertragungssystem sind noch ungewiß. Demgegenüber stehen die Interessen der Betreiber, die einen hohen Überschuß erwirtschaften wollen. Die CAC stellt ein Instrument dar, mit dem diese unterschiedlichen Anforderungen berücksichtigt und umgesetzt werden können.
Wie in Abschnitt 2.2.3 erläutert, kann die Zugangskontrolle funktional in zwei unterschiedliche Einheiten aufgeteilt werden. Auf der einen Seite muß festgestellt werden, wie groß der Bedarf an Übertragungsbandbreite ist, um die geforderte Dienstqualität abzudecken. Im einfachsten Fall dienen bei den konventionell Verfahren die maximale oder durchschnittliche Übertragungsrate zur Charakterisierung der Anforderungen. Bei komplizierteren Methoden fließen noch Varianzen der Bandbreite und tolerierbare Verlustraten in die Bedarfsermittlung mit ein. Das Ergebnis stellt dann eine *effektive* Bandbreite dar, mit der der Dienst arbeitet und die durchgängig von dem Managementsystem garantiert werden muß.
Auf der anderen Seite stehen die verfügbaren Übertragungsreserven, die sich im Wesentlichen durch die Ermittlung des Unterschiedes zwischen der bereits belegten Bandbreite und der gesamten verfügbaren Übertragungskapazität (Gl. 2.2) ergibt. Steht noch genügend Bandbreite zur Verfügung, kann die Verbindung geschaltet werden. Im anderen Fall wird ein Aufbau abgelehnt.
Konventionelle Verfahren basieren auf einem deterministischen Ansatz. Viele Parameter müssen dazu im Vorfeld durch Messungen an den Datenströmen der unterschiedlichen Dienste ermittelt werden. Die so bestimmten Kenngrößen werden dann, entsprechend der unterlagerten Kontrollstrategie, in Gleichungen zueinander in Beziehung gesetzt. Die Korrelation der einzelnen Größen sowie deren Einfluß auf die Qualität der Ergebnisse müssen empirisch ermittelt werden. Dieser beschriebene Ablauf ist *zeitintensiv* und führt in vielen Fällen zu relativ komplizierten Zusammenhängen. Daneben treten Schwierigkeiten, auf den Verkehr optimal zu managen, wenn andere, durch neue Applikationen oder durch ein geändertes Benutzerverhalten nicht analysierte Verkehrsmuster eingeprägt werden.
In dieser Arbeit werden *neue* Ansätze vorgestellt, um ein effizientes Verkehrsmanagement

aufzubauen. Diese Verfahren basieren im Wesentlichen auf der in Kapitel 5 beschriebenen Fuzzy Logic. Sie sind anscheinend in diesem vielschichtigen und stetig veränderndem Umfeld die beste Lösung, um ein effizientes Verkehrsmanagement aufzubauen.
Zur Untersuchung der Anwendbarkeit intelligenter Technologien in dem Bereich der Zugangskontrolle werden die in den Kapiteln 5 und 6 beschriebenen Verfahren und Techniken zum Aufbau eines effizienten und adaptiven Controllers angewandt.

## 7.1 Die Regelstrategie

Die Systembeschreibung nach Abschnitt 3.2 ist der Ausgangspunkt für die Entwicklung einer Regelstrategie, die durch den CA-Controller umgesetzt werden soll. Auf Grund der beschriebenen Zusammenhänge, können die Anforderungen wie folgt umrissen werden.

- Die Call Loss Rate soll für jeden Dienst minimiert werden.
- Weiterhin ist anzustreben, daß die Paketverluste der einzelnen Dienste minimal sind und den im Verkehrsvertrag vereinbarten Grenzwerten entsprechen.
- Daneben muß die systemweite Paketverlustrate auf ein tolerierbares Minimum begrenzt werden.
- Diese Anforderungen sind durch eine angemessene Auslastung der Pufferspeicher des Zugangsknotens zu erfüllen.
- Neben der Nutzung der Warteplätze sollte auch die Auslastung des Kanal effizient sein. Die Übertragungsreserven sollen auf ein Minimum reduziert werden.

Die beschriebene Strategie ermöglicht es, den Fuzzy Logic basierten Regler mit den in Abschnitt 4.4 beschriebenen Verfahren zu vergleichen. In vielen Fällen kann dann die Leistungsfähigkeit der unterschiedlichen Ansätze durch einen Vergleich der Kennzahlen abgeschätzt werden.
Neben der Verbesserung der konventionellen Controller können aber auch leicht Systeme entwickelt und implementiert werden, *die neue, bisher noch nicht berücksichtigte Regelziele* verfolgen. Ein möglicher Ansatz ist, daß alle Dienste *fair* behandelt werden. Dieses Verfahren kann darin resultieren, daß in der Regelstrategie berücksichtigt wird, daß alle Dienste eine *nahezu identische* Call Loss Rate aufweisen.
Um diesen Anforderungen gerecht zu werden, muß ein Regler aufgebaut werden, dessen Stellgröße sowohl von dem lokalen Knotenzustand als auch von den *vorgegebenen* Verkehrsparametern der Dienste abhängig ist.

## 7.2 Der Aufbau des Fuzzy Logic basierten Admission Controllers

Grundlage für die Realisierung bildet der in Abbildung 2.4 gezeigte schematische Aufbau eines Admission Controllers. Es werden dort zwei funktionale Einheiten unterschieden, die die Anforderungen des Dienstes auf der einen Seite berücksichtigen und die Zustandsgrößen des Knotens auf der anderen Seite bewerten. Die Umsetzung dieser Einheiten erfolgt durch die in Abbildung 7.1 dargestellte Konfiguration. Sie besteht aus einem Controller, der eine Kennzahl, die die *effektive* Anforderung, die der Dienst an das Übertragungssystem stellt, wiedergibt und einer zweiten Einheit, die die unterschiedlichen systeminternen Signale auswertet. Durch diese Aufspaltung ist eine getrennte Einstufung der Anforderungen und des Systemzustandes möglich. Der Ausgangswert $P_U$ ist die *effektive* Auslastungskennzahl. Der Wertebereich dieser Größe ist das Intervall $[0, 1]$. 0 zeigt an, daß die vorhandenen Übertragungsressourcen, wie die Warteplätze oder die Kanalkapazität, nicht benutzt werden. Eine 1 hingegen impliziert, daß alle Ressourcen ausgeschöpft sind, und keine Übertragungsreserven mehr bereitstehen.
Die Berechnung der Auslastungszahl kann sich dabei auf die Zustandsparameter $S_U$, $Q_U$ und $T_U$, der Paketverlustrate $LR$ oder der Call Loss Rate stützen. Bei diesem Ansatz bietet sich darüber hinaus noch an, zwischen der $CLR_{System}$, die die allgemeine bzw. systemweite Verlustrate beschreibt, und der dienstspezifischen CLR, der $CLR_{Dienst}$, zu differenzieren. Diese Unterteilung wird, wie im Folgenden noch ersichtlich wird, notwendig, um spezielle Regelstrategien durchzusetzen. In einem letzten Schritt werden beide Ausgangsgrößen in der *Decision Unit* miteinander verglichen. In Abhängigkeit von der dort implementierten Entscheidungslogik wird festgestellt, ob die noch zur Verfügung stehende Bandbreite ausreichend ist, um den Verkehr, der durch die Anforderungskennzahl $P_R$ beschrieben ist, zu vermitteln. Diese Logik kann im einfachsten Fall durch einen Vergleich der Übertragungsreserve mit der Anforderungskennzahl realisiert werden. In diesem Fall kommt Gleichung 7.1 zum Tragen.

$$P_{ATR} = (1 - P_U) \geq P_R \tag{7.1}$$

Aus der Auslastungskennzahl wird durch Negation, die Übertragungsreserve $P_{ATR}$[1] ermittelt. Ist diese Reserve größer oder gleich der Anforderung, die der spezifische Dienst an die Übertragung stellt, kann die Verbindung geschaltet werden. In dem anderen Fall wird der Verbindungswunsch abgelehnt.
Natürlich können auch andere, komplexere und umfassendere Entscheidungsstrategien umgesetzt werden. In den folgenden Abschnitten wird auch hier zur Auswertung der Kennzahlen ein Fuzzy Controller zum Einsatz kommen.

### 7.2.1 Der Traffic Qualifier

Wird ein Verbindungsaufbau eingeleitet, müssen die charakteristischen Kenngrößen des Dienstes bewertet und die Verkehrsquelle eingestuft werden, was dann zu einer Bedienan-

[1] Available Transfer Ressources

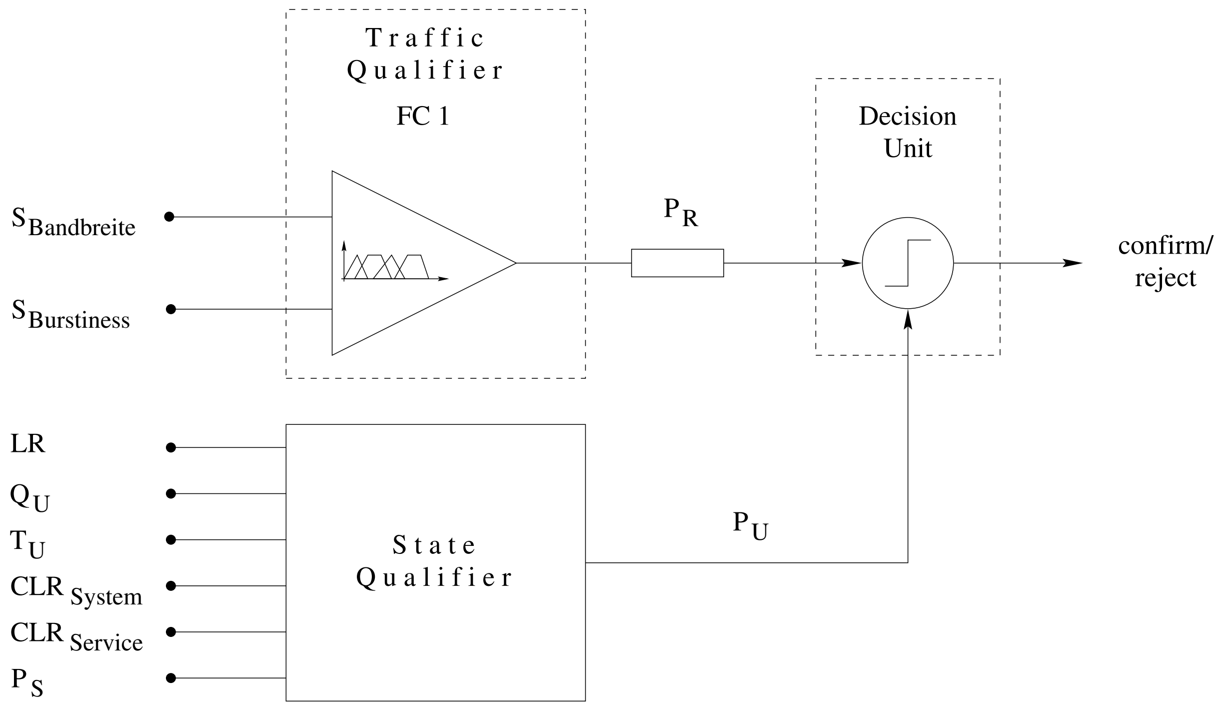



Abbildung 7.1: Funktionaler Aufbau eines Fuzzy Logic basierten Admission Controllers

forderung $P_R$ führt. Diese Bewertung wird durch den *Traffic Qualifier*, der durch den Fuzzy Controller $FC_1$ realisiert wird, vorgenommen. Die Ermittlung der Kennzahlen erfolgt in dem vorliegenden Ansatz anhand der deklarierten Bandbreite bezogen auf die maximale Übertragungsrate des abgehenden Kanals.

$$S_{Bandbreite} = \frac{Deklarierte\ Bandbreite}{\ddot{U}bertragungsbandbreite\ des\ Ausgangslinks} \tag{7.2}$$

Die zweite Eingangsgröße ist die normierte Burstiness $S_{Burstiness}$. Sie wird wie folgt berechnet.

$$S_{Burstiness} = \frac{dienstspezifische\ Burstiness}{maximale\ Burstiness} \tag{7.3}$$

In der Tabelle 7.1 ist die Regelbasis des Fuzzy Controllers dargestellt. Nachvollziehbar - durch Auswertung der Zeilen - ist, daß die Anforderungen, die an das Kommunikationssystem gestellt werden, mit zunehmender Bandbreite $S_{Bandbreite}$, wenn die Burstiness konstant ist, steigen. Eine vergrößerte Burstiness $S_{Burstiness}$ - Auswertung der Spalten - führt hingegen in gewissen Grenzen zu einer Verminderung der Anforderung, da die durchschnittliche Übertragungsrate reziprok zum Burstfaktor ist. Mit Hilfe der im Anhang in Abschnitt E.1, Seite 213 ff. dargestellten Membershipfunktionen für die linguistischen Variablen $S_{Bandbreite}$,

| $P_R$ | | $S_{Bandbreite}$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| | SK | SK | K | M | G | SG |
| | K | SK | K | M | G | SG |
| $S_{Burstiness}$ | M | SK | SK | K | M | G |
| | G | SK | SK | K | M | G |
| | SG | SK | SK | SK | K | M |

Tabelle 7.1: Regelbasis des *Traffic Qualifiers.*

$S_{Burstiness}$ und $P_R$ ergibt sich das in Abbildung 7.2 dargestellte Übertragungsverhalten des Traffic Qualifiers. Die Darstellung des Übertragungsverhaltens skizziert noch einmal deut-

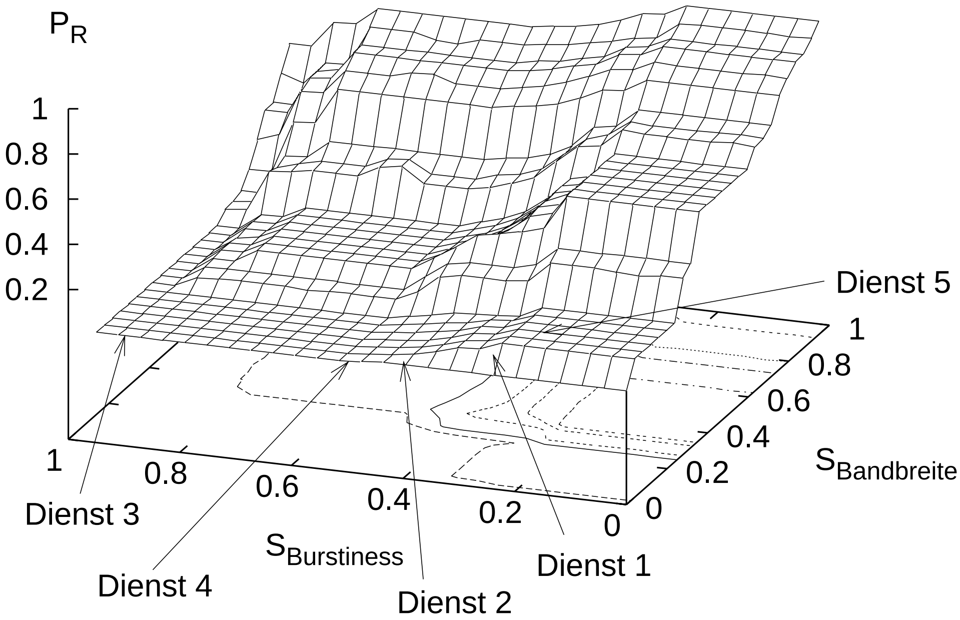



Abbildung 7.2: Übertragungsverhalten des Traffic Qualifiers

lich das Konzept, das durch den Traffic Qualifier verfolgt werden soll. Eine Erhöhung der Bandbreite führt in jedem Fall zu einer Steigerung der Anforderungskennzahl $P_R$. Durch die Burstiness kann allerdings die effektive Anforderung, die an das Übertragungssystem gestellt wird, nachhaltig beeinflußt werden. Eine große Burstiness impliziert, gemäß des Zusammenhangs nach Gleichung 2.1, eine geringe durchschnittliche Übertragungsbandbreite $BW_{\oslash}$. Auf Grund dieses Zusammenhangs kann die Verminderung der Übertragungsanforderung begründet werden. Mit Hilfe der Kenngrößen, die in der Tabelle 4.2 abgebildet sind, ergeben sich für die einzelnen Dienste folgende, in Tabelle 7.2 wiedergegebenen, Arbeitspunkte. Diese Arbeitspunkte sind in Abbildung 7.2 durch Pfeile gekennzeichnet. Die Dienste 1 und 5 weisen eine Anforderungszahl von 0.1 bzw. 0.25 auf.

| Dienst | $S_{Bandbreite}$ | $S_{Burstiness}$ | $P_R$ |
|---|---|---|---|
| 1 | 0.032 | 0.25 | 0.1 |
| 2 | 0.003 | 0.4 | 0.05 |
| 3 | 0.003 | 0.9 | 0.02 |
| 4 | 0.003 | 0.5 | 0.02 |
| 5 | 0.006 | 0.15 | 0.25 |

Tabelle 7.2: Arbeitspunkte

### 7.2.2 Der State Qualifier

Der Status des Zugangsknotens wird mit Hilfe des *State Qualifiers* bestimmt[2]. In dieser Einheit wird die Auslastung des Zugangsknotens in Abhängigkeit von den unterschiedlichen Systemgrößen des Zugangsknotens ermittelt. Zu der Bewertung können lokale - dienstspezifische - Parameter ($Q_U$ und $LR$) sowie globale - knotenspezifische Kenngrößen ($T_U$, $S_U$, $CLR_{System}$ und $CLR_{Dienst}$) herangezogen werden. Der Ausgangswert wird dann durch $P_U$ charakterisiert.

## 7.3 Simulation des unscharfen Admission Controllers

Bei dieser Untersuchung werden unterschiedliche Ansätze zur Bestimmung der Kenngröße $P_U$ verfolgt, um die Auslastungsfunktion des Systems zu optimieren. Die in den folgenden Abschnitten abgebildeten Ergebnisse stellen die Mittelwerte dar. Die ermittelten Vertrauensintervalle sind $\leq 10\%$ der entsprechenden Mittelwerte. In den Diagrammen sind sie aus Gründen der Übersichtlichkeit nicht dargestellt.

#### Auslastungsfunktion 1

Im einfachsten Fall wird die Systemauslastung durch einen konstanten, von dem aktuellen Zustand des Zugangsknotens unabhängigen Wert beschrieben.

$$P_U = C = konst. \text{ mit } C \in ]0,1[ \tag{7.4}$$

[2]Für die nachfolgenden Untersuchungen wird vorausgesetzt, daß das Netz blockierungsfrei ist, also stets eine Route durch das Netzwerk verfügbar ist.

$P_U$ stellt definitionsgemäß die Auslastungskennzahl des Zugangssystems dar. Die Übertragungsreserve, d. h. die Kennzahl, die die effektiv verfügbare Bandbreite beschreibt, wird in der Decision Unit durch Bildung des Komplements ermittelt.

$$effektive\ \ddot{U}bertragungsreserve = P_{ATR} = 1 - P_U \tag{7.5}$$

$P_{ATR}$ stellt die Übertragungsreserve dar, d. h. die Kennzahl, die die effektiv freie Bandbreite beschreibt. Diese Kennzahl wird mit der Anforderungskennzahl $P_R$ verglichen. Ist $P_R$ kleiner als die Übertragungsreserve, kommt eine Verbindung zustande. D. h., wenn die Anforderungen, die an das Übertragungssystem gestellt werden, geringer als die durch $P_{ATR}$ beschriebenen freien verfügbaren Ressourcen sind, kann die gewünschte Verbindung geschaltet werden. Im anderen Fall wird der Verbindungsaufbau abgelehnt.
Bei den Diensten 2, 3 und 4 beträgt die minimale Bandbreite, entsprechend zur Tabelle 4.2, $0.5 Mbit/s$. Der normierte Wert $S_{Bandbreite}$ beläuft sich dann auf 0.0032. In diesem Fall befindet sich der Arbeitspunkt auf dem untersten Plateau. Für diese Konfiguration ist die Anforderungskennziffer $P_R$ konstant und unabhängig von der Burstiness. Erst für große Auslastungskennzahlen bzw. für kleine $P_{ATR}$ sind dann abweichende Ergebnisse zu erwarten. In diesem Fall können die Dienste mit hohen Anforderungen nicht mehr bedient werden. Der Übergang in den neuen stabilen Zustand erfolgt abrupt. Er ist dadurch gekennzeichnet, daß $P_R$ entweder kleiner als die Auslastungskennziffer ist, dann beläuft sich die CLR auf 100% oder aber $P_R > P_{ATR}$, so daß die CLR 0% beträgt.

- Fallbeispiel mit $P_U = 0.4$
  Stellvertretend für den Bereich von $P_U = [0.1 - 0.8]$ sind in den Abbildungen E.4 und E.5 (215ff.) die Kennlinien für $P_U = 0.4$ dargestellt.
  Die Call Loss Rate beträgt für alle Dienste konstant 0%. Die Anforderungskennzahl ist in jedem Fall, wie der Tabelle 7.2 entnommen werden kann, kleiner als die effektive Übertragungsreserve $P_{ATR} = 1 - P_U = 0.6$. Infolgedessen werden alle Verbindungen, unabhängig von der Auslastung der Ressourcen, aufgebaut. Die Auslastung des Kanals (Abb. E.8) beläuft sich deshalb bei 50 Erlang schon auf 100%. Die Überkapazitäten müssen folglich durch die Warteplätze aufgefangen werden. Auch diese werden dann schon zu 97% genutzt, so daß es zu erheblichen Verlusten (Abb. E.4) kommt.

- Fallbeispiel mit $P_U = 0.9$
  Die Abbildungen E.6 bis E.8 (216ff.) zeigen die Ergebnisse für den Fall, daß $P_U = 0.9$ ist. Schon ein grober Vergleich mit den Kennlinien, die im letzten Abschnitt beschrieben wurden, zeigt, daß bei der Behandlung der Dienste, in Abhängigkeit von dem Grenzwert, drastische Änderungen auftreten.
  Die CLR der Dienste 2, 3 und 4 (Abb. E.6) beträgt 0%. Es werden stets alle Verbindungen aufgebaut. Auf Grund des speziellen Lastmusters des Dienstes 4, kann die Paketwarteschlange häufiger abgearbeitet werden als bei den Diensten 2 und 3, so daß im Gegensatz zu diesen, die mittlere Auslastung der Warteplätze, wie in Abbildung E.8 gezeigt, etwas geringer ausfällt. Diese verringerte Nutzung manifestiert sich dann

aber auch im Verlauf der Verlustkurven (Abb. E.7). Die Verlustrate des Dienstes 4 ist während der gesamten Simulation geringer als die der Dienste 2 und 3.
Die CLR der Dienste 1 und 5 beträgt konstant 0%, so daß, weil keine Verbindungen aufgebaut werden, die Warteplätze nicht belegt werden und auch keine Datenverluste auftreten können.

**Auslastungsfunktion 2**

Bei diesem Ansatz wird die Auslastung des Kanals zur Bestimmung der Übertragungsreserve $P_{ATR}$ herangezogen. Die freie Bandbreite kann durch Gleichung 7.6 beschrieben werden.

$$\ddot{U}bertragungsreserve\ P_{ATR} = 1 - P_U = 1 - T_U \tag{7.6}$$

Die Ergebnisse, die sich bei der Simulation unter Verwendung der Gl. 7.6 zur Bewertung des Status des Zugangsknotens ergaben, sind im Anhang in den Abbildung E.9 bis E.11 auf den Seiten 218ff. dargestellt. Abbildung E.9 zeigt die Call Loss Rate der Dienste. Die CLR der Dienste 2, 3 und 4 fallen in gesamten Verlauf geringer aus als die der Dienste 1 und 5. Diese Dienste weisen im allgemeinen auf Grund ihrer - im Vergleich zu den anderen Diensten - großen Übertragungsbandbreite, eine relativ hohe Anforderungskennzahl auf. Werden die Ressourcen knapper, können diese Anforderungen nicht mehr abgedeckt werden, so daß Verbindungen nicht aufgebaut werden können. Infolge der erhöhten CLR treten dann im Weiteren aber weniger Datenverluste auf (Abb. E.10). Im Gegensatz dazu, können Verbindungen, die sich durch einen geringeren Bedarf an Bandbreite auszeichnen, noch vermittelt werden. Es werden deshalb weniger Verbindungswünsche abgewiesen, aber die Paketverluste steigen. Die maximalen Verlustraten für Dienst 2 belaufen sich bei einem Angebot von 250 Erlang auf $\approx 24\%$.
Abbildung 7.3 zeigt den Verlauf der CLR bei einem Angebot von 50 Erlang. Die Verluste streuen über einen Bereich von 13.8%. Deutlich zu erkennen ist hier die differenzierte Behandlung der einzelnen Anwendungen. Die Dienste 3 und 4 weisen die kleinste Anforderungskennzahl auf, so daß die CLR mit 71.8% und 71.5% am geringsten ist. Die Call Loss Rate von Dienst 2 beträgt 76.7%. Die Verlustraten für die Anwendungen 1 und 5 beziffern sich auf 85.3% und 83.3%. Die Staffelung entspricht im Wesentlichen der ermittelten Anforderungskennzahl. Der Dienst mit den geringsten Anforderungen hat demnach die kleinsten, der mit dem größten Bedarf an Übertragungsressourcen die höchsten Verluste beim Aufbau von Verbindungen. Die CLR hängt bei diesem Ansatz direkt von $P_R$ ab. Im Weiteren kann mit Hilfe des Graphen, der in Abbildung E.11 dargestellt ist, auch ein Zusammenhang zwischen der Call Loss Rate und der Auslastung der Warteplätze hergeleitet werden. Die Belegung der Paketspeicher der Dienste 1 und 5 ist nämlich in weiten Teilen geringer als die Auslastung der Warteplätze der übrigen Dienste.
Auf der Basis dieser geschilderten Zusammenhänge kann formuliert werden, daß sich die Staffelung der CLR an die Klassifikation der Anforderungen durch $P_R$ anlehnt. Die Auslastung der Ressourcen sowie der Verlauf der Verlustraten zeigen eine reziproke Abhängigkeit von der Auslastungskennziffer.
Die Übertragungsbandbreite des Kanals wurde mit maximal 95% bei 250 Erlang optimal

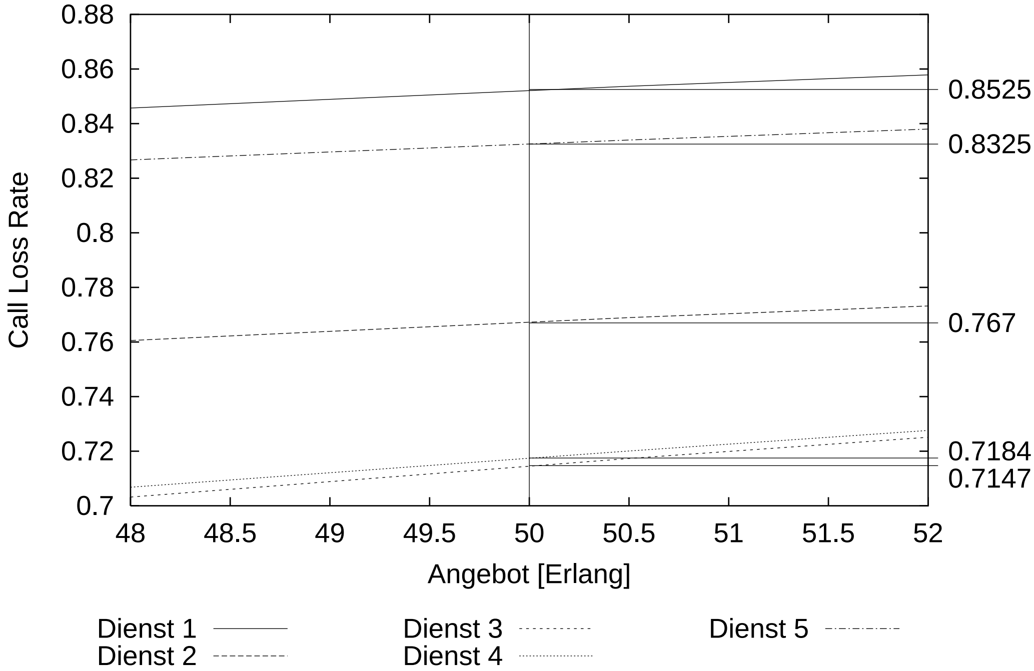



Abbildung 7.3: Verlauf der CLR bei einem Angebot von 50 Erlang unter Verwendung von Gl. 7.6 zur Bestimmung der verfügbaren Übertragungsreserven

genutzt. Sie bietet darüber hinaus noch die Möglichkeit, statistische Schwankungen der Verkehrslast auszugleichen.

**Auslastungsfunktion 3**

Neben dem globalen Ansatz, bei dem die Auslastung des Kanals als Entscheidungskriterium herangezogen wurde, um festzustellen, ob eine Verbindung aufgebaut werden kann, dient im Folgenden die Auslastung der dienstspezifischen Warteplätze als Bewertungsgrundlage. Beurteilungen erfolgen so an Hand von lokalen Merkmalen. Die freien Übertragungsressourcen lassen sich mit Gleichung 7.7 beschreiben.

$$P_{ATR} = 1 - Q_U \tag{7.7}$$

Die Ergebnisse sind im Anhang in den Abbildungen E.12 bis E.14 auf den Seiten E.2.3ff. dargestellt. Die CLR steigt für alle Dienste gleichmäßig an. Bei 250 Erlang beträgt sie $\approx$ 95%. Eine detaillierte Untersuchung zeigt eine Staffelung der Call Loss Rate (Abb. 7.4). Die Streuung der Verluste beträgt bei 50 Erlang 9.55%. Dienst 1 weist mit 77.15% die höchste Verlustrate auf. Die CLR der Dienste 2 und 3 beziffern sich auf 74.15% und 72.85%, die der Anwendungen 4 und 5 auf 67.6% und 70.95%. Der abgebildete Verlauf der Kennlinien entspricht nicht der Einstufung der Dienste durch die Anforderungskennzahl.

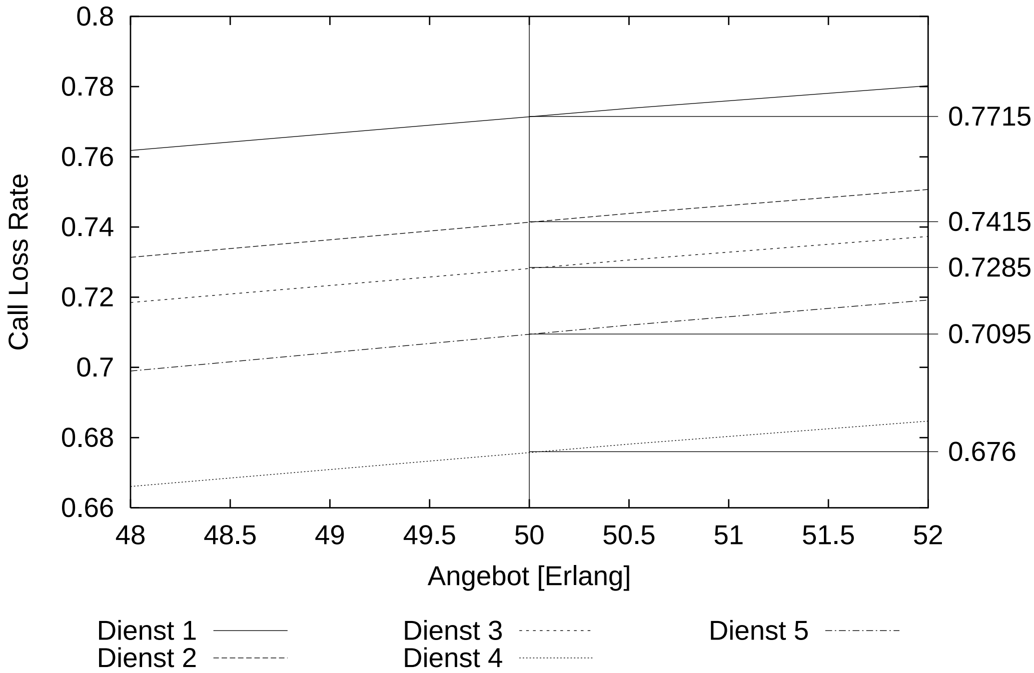



Abbildung 7.4: Verlauf der CLR bei einem Angebot von 50 Erlang unter Verwendung von Gl. 7.7 zur Bestimmung der verfügbaren Übertragungsreserven

Er kann aber durch die Kombination von $P_R$ zum einen und durch das eingeprägte Verkehrsmuster zum anderen, schlüssig erklärt werden. Auf der einen Seite zeigt eine hohe Anforderungskennzahl wie bisher an, daß die Anwendung umfangreiche Anforderungen an das Übertragungssystem stellt. Auf der anderen Seite wird die Transferreserve bei diesem Ansatz mit Hilfe der mittleren Auslastung der dienstspezifischen Warteplätze ermittelt. Bei den Diensten 4 und 5 wechseln sich die Phasen der unterschiedlichen Aktivitäten (Tabelle 4.2 in Kombination mit Abb. 4.1) relativ häufig ab. Die Warteschlangen werden infolgedessen öfter abgearbeitet als die der übrigen Dienste, so daß die Auslastung relativ gering ist. Die unterschiedlichen aktiven Phasen alternieren bei Dienst 5 periodisch mit einer Frequenz von 10 Hz. Bei einem Übergang von einer Phase in der der Dienst Daten mit einer hohen Bandbreite erzeugt, die zu einer gesteigerten Auslastung der Speicher führt, folgt eine Periode geringerer Aktivität. In dieser Phase wird die Warteschlange geleert, so daß wieder Ressourcen für weitere Verbindungen zur Verfügung stehen. Dienst 1 dagegen sendet kontinuierlich Daten, so daß der Speicher, wie der Abbildung E.14 entnommen werden kann, stets gut ausgelastet ist. Dieses Verhalten zieht dann eine erhöhte CLR nach sich.
Beide Effekte überlagern sich, so daß Dienst 4 die geringste und die Anwendung 1 die höchste Call Loss Rate aufweist.
Der Zusammenhang zwischen der Paketverlustrate und der CLR ist reziprok. Eine hohe Call Loss Rate bedeutet in jedem Fall eine geringe Paketverlustrate.

### Auslastungsfunktion 4

Um in die Berechnung der Auslastungkennziffer sowohl globale als auch lokale Merkmale einfließen zu lassen, wurde zur Bestimmung der Übertragungsreserve $P_{ATR}$, Gleichung 7.8 herangezogen.

$$P_{ATR} = (1 - Q_U) \cdot (1 - T_U) \tag{7.8}$$

In dieser Gleichung ist die Belegung der Warteschlangen mit der Auslastung des abgehenden Links verknüpft. Die Ergebnisse sind im Anhang in den Abbildungen E.15 bis E.17 auf den Seiten 222ff. dargestellt. Es zeigt sich, daß der Einfluß der Kanalauslastung das Verhalten der Decision Unit maßgeblich bestimmt. Die Auslastung des Links steigt, wie Abbildung E.17 zeigt, schnell mit zunehmenden Angebot an. Bei einem Angebot von 50 Erlang beträgt die Auslastung bereits 95%, so daß die Übertragungsreserve klein wird. Dienste mit einer hohen Übertragungsanforderung können nicht mehr oder nur noch in geringem Maße vermittelt werden. Die Warteplätze werden deshalb nur in einem verringerten Umfang genutzt.
Die Streuung der Call Loss Rate beträgt bei einem Angebot von 50 Erlang (Abb. 7.5) 14.21%. Diese Kennlinien zeigen, daß auch dieser Ansatz die festgelegten Regelziele nicht

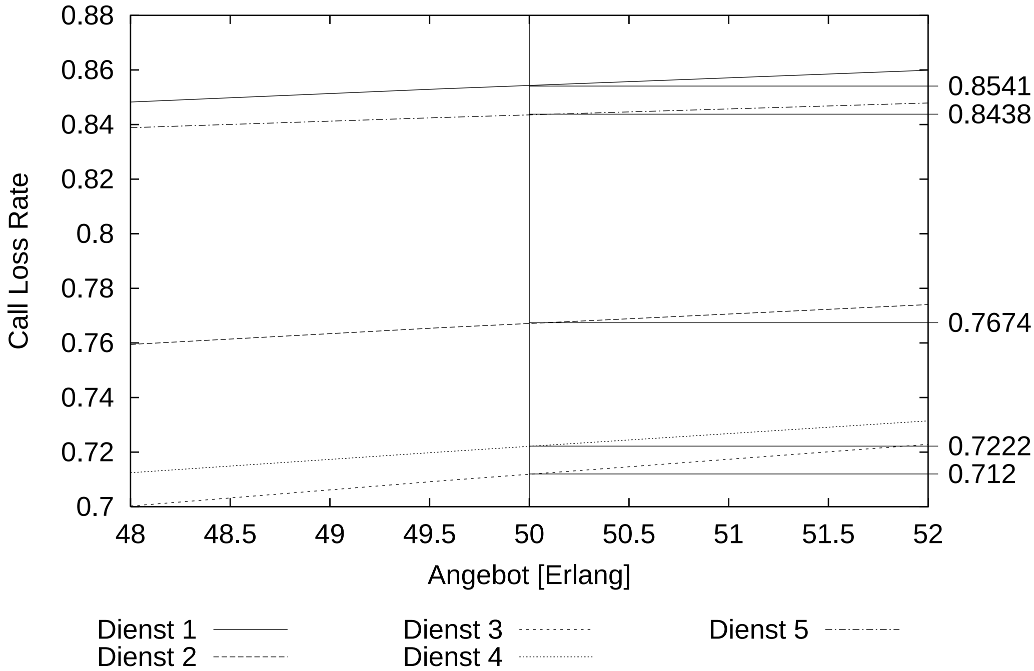



Abbildung 7.5: Verlauf der CLR bei einem Angebot von 50 Erlang unter Verwendung von Gl. 7.8 zur Bestimmung der verfügbaren Übertragungsreserven

in vollem Umfang erfüllt. Die Dienste werden nicht fair behandelt, was sich durch die Call

Loss Rate, die bei dem Vergleichswert von 50 Erlang relativ stark streut, belegen läßt. Weiterhin ist die Auslastung der Warteplätze der einzelnen Dienste nur suboptimal.

### Auslastungsfunktion 5

Eine weitere Möglichkeit, den Zustand des Zugangsknotens zu qualifizieren, besteht in der Bewertung der allgemeinen Auslastung der Warteplätze. Die relative Belegung der Pufferspeicher ergibt sich aus der Anzahl der reservierten zur Gesamtzahl aller Paketspeicher des Zugangsknotens.

$$S_U = \frac{\sum belegte\ Warteplätze}{\sum Warteplätze} \tag{7.9}$$

Die freien, für die Übertragung verfügbaren Ressourcen lassen sich dann mit Gl. 7.10 ermitteln.

$$P_{ATR} = 1 - S_U \tag{7.10}$$

Die Ergebnisse sind im Anhang in den Abbildungen E.18 bis E.20 dargestellt. Die Dienste 1 und 5 weisen mit Abstand die höchste Call Loss Rate auf. Die CLR des Dienstes 2 hat bei 250 Erlang einen Betrag von 60%, die der Anwendungen 3 und 4 beläuft sich auf 50%. Hier ist wieder eine Abhängigkeit von der Anforderungskennzahl erkennbar. Ein hoher Bedarf an Übertragungsressourcen führt zwangsläufig zu einer erhöhten Call Loss Rate. Problematisch an diesem Ansatz ist allerdings, daß der für die Dienste 1 und 5 zur Verfügung stehende lokale Speicher nicht erschöpfend genutzt wird. Die bei diesen auftretenden Verläufe der Auslastungskennlinien lassen sich jedoch wie folgt erklären. Dienst 1 weist mit 0.1 (Tabelle 7.2) eine hohe Anforderungskennziffer auf. Da die allgemeine Auslastung der Warteplätze allerdings groß ist, was sich nach Gl. 7.10 in einer geringen freien Übertragungskapazität niederschlägt, können nur Verbindungen mit geringen Anforderungen aufgebaut werden. Die Paketspeicher werden infolgedessen im Mittel zu 90% genutzt. Anders bei Dienst 5. Die Auslastung des Paketspeichers liegt im Mittel bei 80%. Wie im Abschnitt 7.3 beschrieben, ist die Auslastung des Wartespeichers unter anderem abhängig von der eingeprägten Verkehrslast. Wechseln Phasen unterschiedlicher Aktivitäten relativ schnell, können Warteschlangen eher abgebaut werden als bei Systemen, die Datenströme mit einer konstanten Rate liefern. Dies führt trotz der hohen Anforderungen zu einer gegenüber Dienst 1 verringerten Auslastung der verfügbaren Speicherplätze. Abbildung 7.6 zeigt den Verlauf der CLR bei einem Angebot von 50 Erlang. Die Streuung der CAll Loss Rate beträgt 58%. Weiterhin muß festgestellt werden, daß wie bei den anderen Ansätzen die CLR den Verlauf der Paketverlustrate beeinflußt. Kommen nur wenige Verbindungen zustande, ist LR klein.
Die dargestellten Ergebnisse zeigen, daß dieser Ansatz nicht den in der Regelstrategie formulierten Zielen des Call Admission Controllers entspricht.

### Auslastungsfunktion 6

Bei diesem Ansatz wird der State Qualifier durch einen Fuzzy Controller (Abb. 7.7) realisiert. Die Eingangsgrößen sind die Auslastung der Warteschlange $Q_U$ und des Kanals $T_U$.

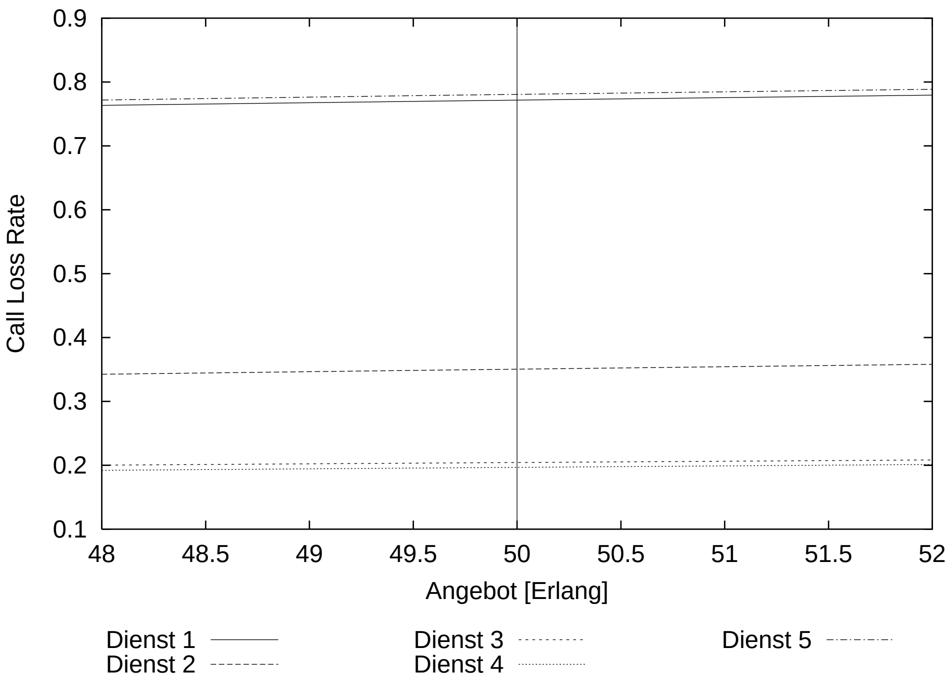



Abbildung 7.6: Verlauf der CLR bei einem Angebot von 50 Erlang unter Verwendung von Gl. 7.10 zur Bestimmung der verfügbaren Übertragungsreserven

Das Ausgangssignal $P_{ATR}$ qualifiziert die verfügbaren Übertragungsressourcen. Die Re-

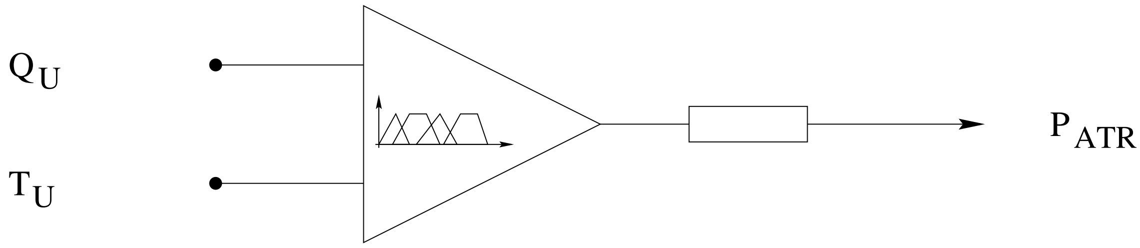



Abbildung 7.7: Fuzzy Logic basierter State Qualifier

gelbasis des Fuzzy Controllers ist in Tabelle 7.3 beschrieben. Das Grundgedanke bei der Entwicklung des Reglers war, daß eine intensive Auslastung des abgehenden Links dazu führt, daß die Ausgangsgröße heraufgesetzt wird.
Eine geringe Auslastung der Warteschlange kann diesen Zusammenhang teilweise kompensieren. Es stehen dann genügend Speicherplätze zur Verfügung, um die fehlenden Übertragungskapazitäten vorübergehend auszugleichen. Kann der Füllstand mit „Groß" und „Sehr Groß" bewertet werden, sind die Reserven so gering, daß der Zugang beschränkt werden

| $P_{ATR}$ | | $T_U$ | | |
|---|---|---|---|---|
| | | NK | N | K |
| $Q_U$ | SK | SG | SG | G |
| | K | SG | SG | G |
| | M | G | G | M |
| | G | K | K | K |
| | SG | SK | SK | SK |

Tabelle 7.3: Regelbasis des Fuzzy Logic basierten State Qualifiers

muß. Der Betrag von $P_{ATR}$ wird dann entsprechend verkleinert. Dieser Zustand kann auftreten, wenn die Warteschlange trotz freier Übertragungskapazitäten nicht abgearbeitet wird, weil die Priorität der Datenpakete zu gering ist oder aber, wenn der Kanal zu stark überbucht ist. Im ersten Beispiel ist die Überlastsituation lokal, sie beschränkt sich dann nur auf eine Warteschlange. Im zweiten Fall sind die Ressourcen dann systemweit zu stark belegt.

Die im Anhang in den Abbildungen E.21 bis E.23 dargestellten Kennlinien zeigen den Verlauf der Call Loss Rate, der Verlustrate und der Auslastung. Die CLR der Dienste 2, 3 und 4 ist innerhalb kürzester Zeit auf 100% angewachsen. Der Verlauf der CLR der Dienste 1 und 5 steigt in dem Bereich bis 100 Erlang auf ca. 70% an. Im Folgenden flacht die Steigung ab. Bei 250 Erlang beträgt die Verlustrate dann 80%. Ursache für diesen Verlauf ist die übermäßige Gewichtung der deklarierten Bandbreite nach Tabelle 7.1. Dienste, die sich durch sehr große Bandbreiten in Kombination mit einem kleinen bis mittleren Burstfaktor auszeichnen (Dienst 1 und 5) werden in ihren Anforderungen stärker bewertet als andere Dienste. Dieses Regelverhalten bedingt, daß sich die Paketverlustraten für die Dienste 2, 3 und 4 auf 0% belaufen. Während die übrigen Verlustraten bezogen auf die gesamte Simulationszeit immer kleiner als 2.5% sind. Die Auslastung der Ressourcen für die Dienste 1 und 5 beträgt maximal 50%, die Ressourcen des Kanals werden nur zu 80% genutzt.

Der Verlauf der Kennlinien belegt eindeutig, daß der Controller nicht richtig angepaßt ist. Die Dienste werden nicht fair behandelt, d. h., daß die Unterschiede der dienstspezifischen CLR gravierend sind. Bandbreite wird hier nur für Verbindungen mit hohen Anforderungen reserviert. Neben diesem Verhalten ist auch die Auslastung der Warteplätze minimal. Das Verhalten der Reglers ist in allen Belangen nicht konform zur festgelegten Regelstrategie, so daß in einem weiteren Ansatz diese grundlegenden Schwachstellen behoben werden.

### Auslastungsfunktion 7

Bei den bisherigen Ansätzen wurden für die Zugangskontrolle nur die Auslastung der vorhandenen Ressourcen berücksichtigt. Als Folge davon ergaben sich Regler, die das Sy-

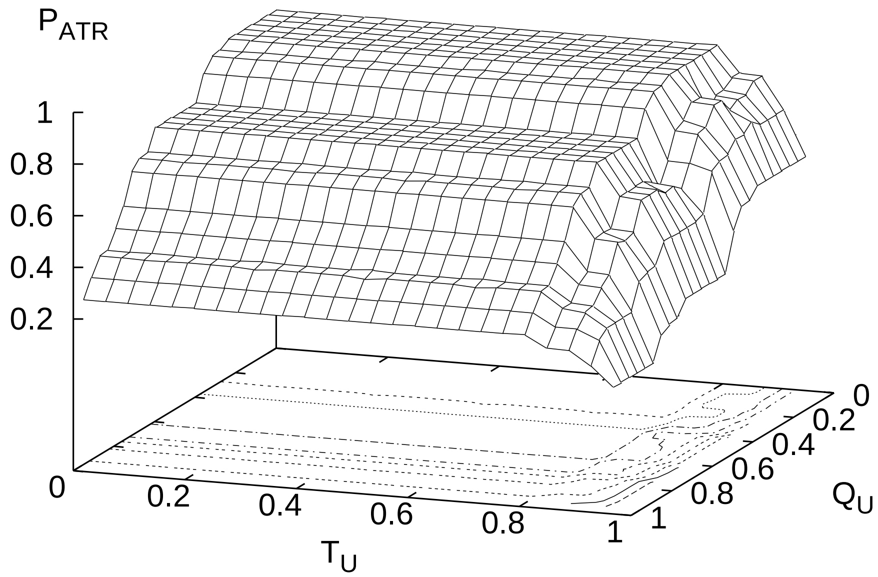



Abbildung 7.8: Übertragungsverhalten des State Qualifiers

stemverhalten so beeinflußten, daß die *effektive* Nutzung der Warteplätze und der Übertragungsbandbreite im Vordergrund stand. Die Aufgaben konnten unter Berücksichtigung dieser Randbedingungen optimal abgewickelt werden.

Nachteilig im Sinne der vorgegebenen Regelstrategie wirkte sich die teilweise sehr unterschiedliche Behandlung der einzelnen Dienste aus. Die genaue Untersuchung des Verhaltens der Regelverfahren bei einem Angebot von 50 Erlang zeigte eine große Streuung der CLR bei den Diensten. Die implementierten Verfahren zeichnen sich durch ein mehr oder weniger faires bzw. unfaires Verhalten aus. Im Folgenden soll deshalb versucht werden, ein Regelverhalten zu implementieren, bei dem sowohl eine effiziente Auslastung der Ressourcen als auch eine faire Behandlung der Dienste Berücksichtigung finden.

Zur Realisierung dieser Anforderungen wird die in der Abbildung 7.10 dargestellte Topologie eines Fuzzy Logic basierten State Qualifiers gewählt. Der Controller ist hierarchisch aufgebaut und besteht aus drei Teilcontrollern.

Im Fuzzy Controller $FC_1$ werden die dienstspezifische und systemweite Call Loss Rate ($CLR_{Dienst}$, $CLR_{System}$) miteinander verknüpft. Das Ausgangssignal stellt dann eine effektive Bedienwahrscheinlichkeit $P_{CLR}$) auf der Basis der Call Loss Rate dar. Die mittlere CLR des Zugangsknotens oder auch die globale $CLR_{System}$ läßt sich aus den lokalen - dienstspezifischen - Call Loss Rates durch die Bildung des arithmetischen Mittels (Gl.

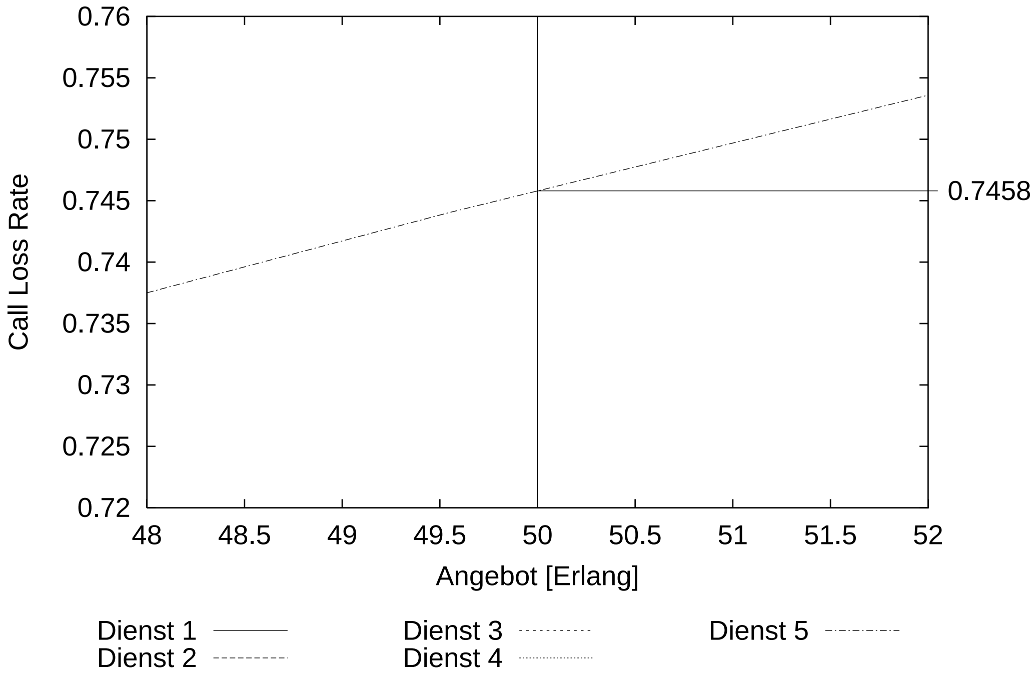



Abbildung 7.9: Verlauf der CLR bei einem Angebot von 50 Erlang unter Verwendung des Fuzzy Controllers nach Abbildung 7.7 zur Bestimmung der verfügbaren Übertragungsreserven

7.11) bestimmen.

$$CLR_{System} = \frac{1}{n} \sum_{i}^{n} CLR_{Dienst,i} \tag{7.11}$$

Die Regelbasis dieses Controllers ist in Tabelle E.1, das Übertragungsverhalten in Abbildung E.24 dargestellt. Die Strategie, die bei der Implementierung dieses Controllers verfolgt wurde, ist, die Differenz zwischen der lokalen und mittleren Call Loss Rate zu egalisieren. Im Wesentlichen lassen sich in der Regelbasis (Tab. E.1) zwei unterschiedliche Bereiche lokalisieren. Für den Fall, daß die linguistische Variable $CLR_{System}$ die Werte „Sehr Klein" und „Klein"annimmt, ist die Ausgangsgröße $P_{CLR}$ „Sehr Groß" bzw. „Groß" . Da die mittlere CLR relativ klein ist, können alle Verbindungen, unabhängig von dem Unterschied zwischen $CLR_{System}$ und $CLR_{Dienst}$ geschaltet werden. Dies schließt auch den Sachverhalt mit ein, daß $CLR_{System} > CLR_{Dienst}$.
Nimmt $CLR_{System}$ die Werte „Groß " und „Sehr Groß " an, wird unabhängig von dem Unterschied zwischen $CLR_{System}$ und $CLR_{Dienst}$ die Bedienwahrscheinlichkeit auf „Klein" und „Sehr Klein" reduziert. Das führt dann zwangsläufig zu einer Erhöhung der dienstspezifischen CLR. Auf Grund dieses Zusammenhangs und der speziellen Form der Membershipfunktionen ergibt sich ein treppenförmiger Verlauf des Übertragungsverhaltens. Die Breite der einzelnen Stufen wird durch den Einflußbereich der unterschiedlichen Sets bestimmt.

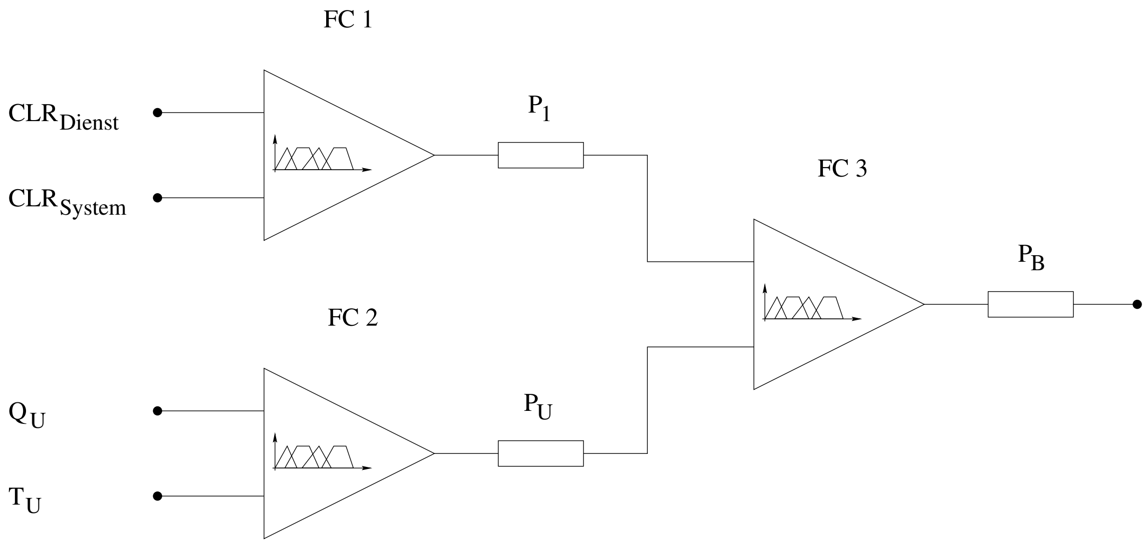



Abbildung 7.10: Hierarchisch aufgebauter State Controller zur Bestimmung der Auslastung des Zugangssystems

In der Einheit $FC_2$ werden die *verfügbaren* Ressourcen des Zugangsknotens qualifiziert. Die Regelbasis dieses Teilcontrollers ist in Tabelle E.2, Seite 229, wiedergegeben. Das Übertragungsverhalten ist in Abbildung E.25 dargestellt. Kann die Auslastung des Links mit „Nicht Kritisch“ oder „Normal“ bewertet werden und ist die Belegung der Warteplätze maximal als „Mittel“ zu beschreiben, stehen genügend Reserven zur Verfügung, um die Verkehrslast zu bewältigen. Die Übertragungscharakteristik, Seite 229, zeigt in diesem Fall ein ausgedehntes Plateau[3]. Erst wenn die Auslastung der Pufferspeicher mit „Groß“ und „Sehr groß“ bewertet werden kann, was einer Auslastung der Warteschlangen größer 60% bedeutet, muß die Übertragungsreserve $P_{ATR}$ reduziert werden. Weiterhin ergibt sich eine drastische Verkleinerung von $P_{ATR}$, wenn der Link in einem kritischen Zustand, d. h., wenn die Auslastung der Bandbreite größer 90% ist.

Zusammengefaßt werden die linguistischen Ausgangswerte $P_{CLR}$ und $P_{ATR}$ mit Hilfe des Fuzzy Controllers $FC_3$. Die Regelbasis ist in Tabelle E.3, die Übertragungscharakteristik in Abbildung E.26 dargestellt. Die Ausgangsgröße $P_B$ hängt direkt von den Eingangsgrößen ab. Sowohl eine Vergrößerung der Auslastungskennzahl $P_{ATR}$ als auch ein hohe effektive $P_{CLR}$ heben die Bedienwahrscheinlichkeit an. Als Folge davon ergibt sich der pyramidenförmige Verlauf der Übertragungscharakteristik.
Dieser so strukturierte State Qualifier liefert am Ausgang kein Äquivalent zu den freien

[3]Auch hier muß nochmals angemerkt werden, daß diese Plateaus auf ein indifferentes Verhalten des Systems hinweisen. Änderungen der Eingangsgrößen führen nicht zwangsläufig zu einer Veränderung der Ausgangsgröße. Die Untersuchung des Einflusses, den diese Ebenen auf die Stabilität des Übertragungsverhaltens haben, geht weit über den Rahmen dieser Arbeit hinaus und muß an anderer Stelle fortgesetzt werden.

Übertragungsressourcen. Vielmehr bedeutet ein großer Ausgangswert, daß die Bedienwahrscheinlichkeitgroß ist.
Ist $P_R < P_B$ wird die Verbindung geschaltet. Eine geringe Entscheidungsschwelle hingegen führt dazu, daß Verbindungswünsche vermehrt abgelehnt werden. Der Verlauf der CLR ist

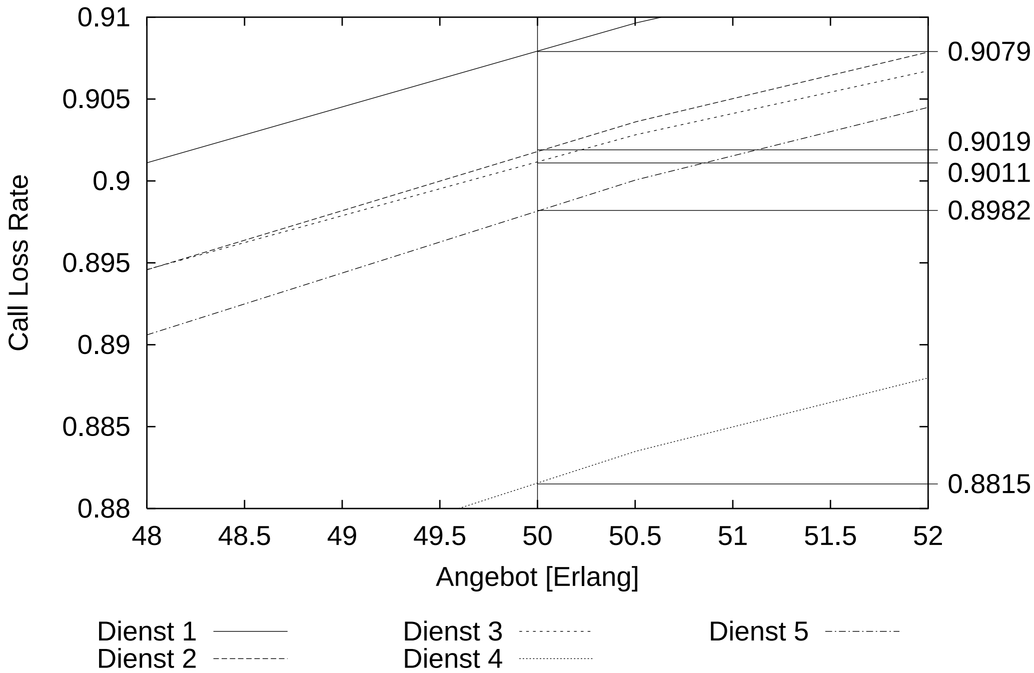



Abbildung 7.11: Verlauf der CLR bei einem Angebot von 50 Erlang unter Verwendung des Fuzzy Controllers nach Abbildung 7.7 zur Bestimmung der verfügbaren Übertragungsreserven

in Abbildung E.27, Seite 231, dargestellt. Die Kennlinien liegen im gesamten Bereich sehr eng beieinander. Die detaillierte Darstellung des Verlaufs bei 50 Erlang (Abb. 7.11) weist eine Streuung der CLR von nur 2.64% aus. Dieses Ergebnis zeigt, daß durch den Einsatz eines Fuzzy Logic basierten Controllers ein bisher nicht berücksichtigtes Regelziel, die *faire* Behandlung der unterschiedlichen Dienste, die sich in einer nahezu identischen Call Loss Rate für alle Dienste ausdrückt, leicht zu implementieren ist. Durch dieses Verfahren entstehen auf der anderen Seite aber bezüglich der Auslastung kleinere Einschränkungen. Die Abbildung E.29 zeigt, daß die Nutzung sowohl der Warteplätze als auch der Linkbandbreite sehr gering ist. Die Auslastung der Kanalressourcen hat ein Maximum mit 80% bei 18 Erlang und sinkt dann bis auf 20% bei 250 Erlang ab. Die Warteplätze werden bis zu 40% belegt.
Die Verlustraten (Abb. E.28) sind gering. Sie sind für alle Dienste im gesamten Bereich kleiner 4%.

## 7.4 Aufbau eines parallelen Admission Controllers

Bei dem in Abbildung 7.1 skizzierten Aufbau eines Zugangscontrollers, wurden die Anforderungen, die ein Dienst an das Übertragungssystem stellt mit Hilfe des Traffic Qualifiers, ermittelt. Der interne Zustand, der durch die Auslastung der Ressourcen oder durch die verfügbaren Reserven beschrieben werden kann, wurde mit dem State Qualifier abgeleitet. Beide Werte wurden dann in der Decision Unit miteinander verglichen. Waren die Anforderungen geringer als die verfügbaren Ressourcen konnte eine Verbindung aufgebaut werden. Im anderen Fall mußte der Verbindungswunsch abgelehnt werden.
Bei diesem neuen, wiederum hierarchisch aufgebauten Controller (Abb. 7.12) werden die Ausgangswerte $P_R$ und $P_U$ nicht mehr miteinander verglichen, sondern in einer weiteren Einheit benutzt, um eine *Bedienpriorität* abzuleiten. Es wird hier eine Kennzahl ermittelt, die unscharf aussagt, mit welcher Wahrscheinlichkeit eine Verbindung zustande kommt. Dieser Wert wird dann aber mit einem Schwellwert verglichen. Ist die Bedienwahrscheinlichkeit $P_B$ größer als die Entscheidungsschwelle, kann eine Verbindung aufgebaut werden. $FC_1$ arbeitet wie der in Abschnitt 7.2.1 beschriebene Traffic Qualifier. Er verknüpft die ge-

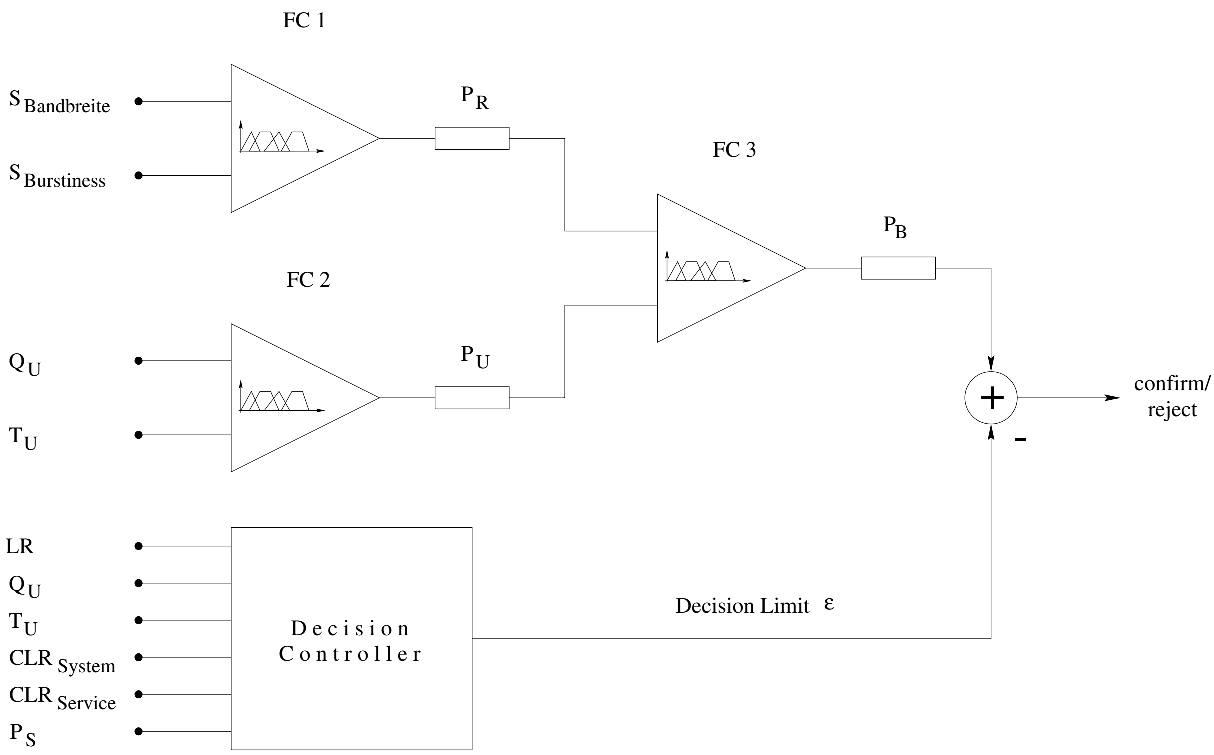



Abbildung 7.12: Struktur eines parallel aufgebauten Admission Controllers

wünschte normierte Übertragungsbandbreite mit der normierten Burstiness und ermittelt so einen Kennwert, der die benötigte Bandbreite reflektiert. Der Ausgang des Controllers $P_R$ beschreibt so das *effektive* Anforderungsprofil des Dienstes. Das Übertragungsverhalten

| $P_B$ | | $P_U$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_R$ | SK | SK | K | G | SG | SG |
| | K | SK | SK | M | G | SG |
| | M | SK | SK | M | G | SG |
| | G | SK | SK | SK | G | SG |
| | SG | SK | SK | SK | K | SG |

Tabelle 7.4: Regelbasis des Fuzzy Controllers $FC_3$ nach Abbildung 7.12

kann der Abbildung 7.8, die assoziierte Regelbasis der Tabelle 7.1 entnommen werden. $FC_2$ dagegen wertet den Zustand des Knotens aus und ist so ein Maß für die *freie*, für weitere Verbindungen noch verfügbare Übertragungsreserve. $P_U$, die Ausgangsgröße, klassifiziert den Dienst in Abhängigkeit von dem Zustand des Zugangsknotens. Die Beschreibung des Übertragungsverhaltens ist in Abschnitt 7.3 erfolgt. Die Regelbasis dieses Teilcontrollers ist in Tabelle E.2, Seite 229, wiedergegeben. Aus dieser Tabelle ist leicht abzulesen, daß die verfügbaren Ressourcen als *Groß* bzw. *Sehr Groß* bewertet werden können, wenn die Auslastung des Kanals und der Warteschlange gering ist. Werden mehr Speicherplätze belegt oder/und steigt die Verkehrslast des Kanals an, reduzieren sich die verfügbaren Reserven. Die Verknüpfung der Hilfsgrößen $P_R$ und $P_U$ erfolgt mit dem Controller $FC_3$. Die Regelbasis ist in Tabelle 7.4 dargestellt. Die Ausgangsgröße $P_B$ beschreibt dann subjektiv, mit welcher Wahrscheinlichkeit eine Verbindung aufgebaut werden kann.
In einem letzten Schritt wird diese Ausgangsgröße mit einem Schwellwert $\varepsilon$, der sich aus verschiedenen Systemgrößen ableiten kann, verglichen. Ist $P_B$ größer als das ermittelte *Decision Limit*, wird die Verbindung aufgesetzt. In dem anderen Fall wird der Verbindungswunsch abgelehnt.

Um die Qualität dieses Controllers nachweisen zu können, wurden auch hier unterschiedliche Funktionen benutzt, um eine Entscheidungsschwelle $\varepsilon$ zu bestimmen. Es wurden feste und dynamische Grenzen für die Bestimmung von $\varepsilon$ gewählt.

### 7.4.1 Fallbeispiel 1: $\varepsilon = const.$

$\varepsilon$ wurde aus dem Intervall $[0.1, 0.9]$ gewählt. Die Ergebnisse zeigen eine deutliche Verbesserung zu dem in Abschnitt 7.3 beschriebenen Verhalten des Controllers. Bei diesem neuen Ansatz erfolgt der Vergleich einer variablen, von der Auslastung der Ressourcen abhängigen Größe $P_B$, mit der konstanten Entscheidungsschwelle. Infolgedessen existiert hier kein Grenzwert bei dem sich das Verhalten abrupt ändert. Die CLR liegt für alle Schwellwerte unterhalb von 90%. Die Paketverlustrate hat bei allen Entscheidungsschwellen einen Wert kleiner 12%. Problematisch (Abb. 7.14) stellt sich allerdings die Auslastung der Warteplätze der unterschiedlichen Dienste dar. Ab einem $\varepsilon > 0.6$ ist die Auslastung

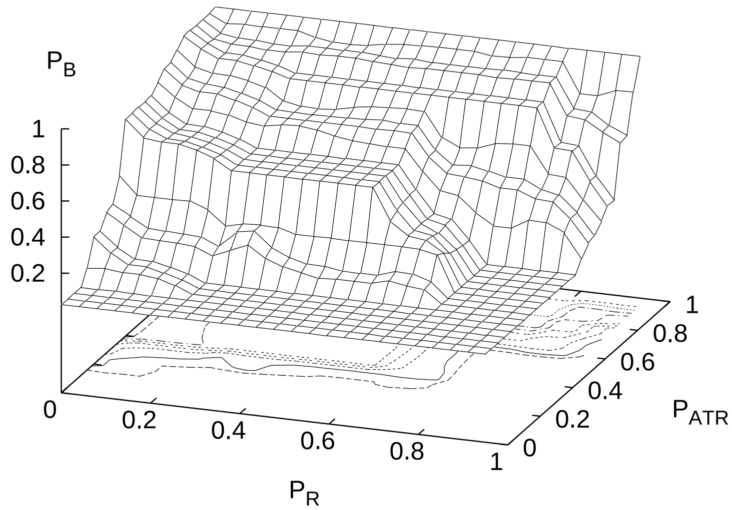



Abbildung 7.13: Übertragungsverhalten des $FC_3$

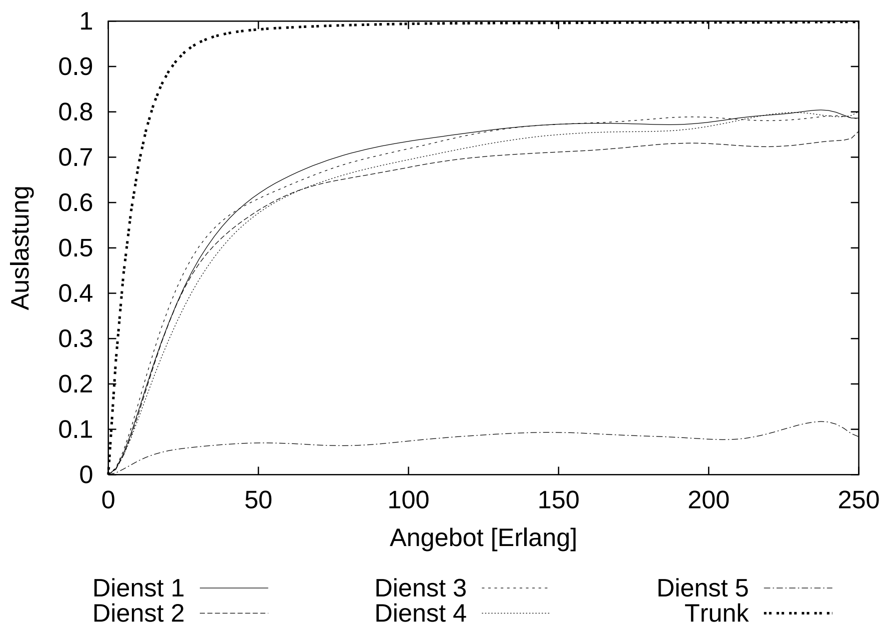



Abbildung 7.14: Auslastung der Übertragungsressourcen für $\varepsilon = 0.9$

der Pufferspeicher von Dienst 5 viel geringer als die der übrigen Dienste. Das Verhalten resultiert zum einen aus der großen Anforderungszahl nach Tabelle 7.2 und zum anderen

aus der hohen Auslastung der Linkbandbreite. Die Regelbasis des Fuzzy Controllers $FC_2$, für die Auswertung der verfügbaren Ressourcen verantwortlich ist, ist im Anhang in der Tabelle E.2 dargestellt. Auf Grund der eindeutigen Überlastsituation, wie in Abbildung 7.14 dargestellt, dominiert die Auslastung des Links das Übertragungsverhalten. Die Bedienwahrscheinlichkeit $P_B$ nimmt einen minimalen Wert an.
Die Paketverlustrate für alle Dienste streut im Bereich zwischen 4 und 12%.

### 7.4.2 Fallbeisiel 2: $\varepsilon = f(Q_U)$

Bei diesem Verfahren entsprach $\varepsilon$ der Auslastung der dienstspezifischen Warteschlange ($\varepsilon = Q_U$). Durch diese Wahl konnten die Streuung der Call Loss Rate sowie die unterschiedlich starke Belegung der Pufferspeicher egalisiert werden. Eine Nutzung der Warteplätze lag für alle Dienste bei durchschnittlich 85%.
Die maximale Paketverlustrate betrug 25%, die CLR hatte eine Wert von 90% bei 250 Erlang.

### 7.4.3 Fallbeispiel 3: $\varepsilon = f(S_U)$

Für den Fall, daß $\varepsilon$ aus der Systemauslastung abgeleitet wurde ($\varepsilon = S_U$), ergaben sich nur marginale Änderungen gegenüber den in Abschnitt 7.4.2 beschriebenen Ergebnissen. Die Auslastung der Warteschlangen streut bei 250 Erlang zwischen 80% und 85%. Die Paketverlustrate belief sich auf maximal 15%.

### 7.4.4 Fallbeispiel 4: $\varepsilon = f(T_U)$

Bei diesem Ansatz ist, wie die Ergebnisse zeigen (Abb. 7.15), die maximale CLR bei 250 Erlang 95%. Wiederum werden auch hier die Dienste in Bezug auf ihre CLR unterschiedlich behandelt. Der Verlauf der Call Loss Rate von Dienst 5 unterscheidet sich erheblich von dem der übrigen Dienste. Das Verhalten resultiert, wie schon im Abschnitt 7.4.1 erörtert, zum einen aus der großen Anforderungszahl nach Tabelle 7.2 und zum anderen aus der restriktiven Bewertung der hohen Auslastung des Links durch den Fuzzy Controller $FC_2$.

## 7.5 Aufbau eines seriellen Admission Controllers

Neben der parallelen Topologie wurde ein seriell strukturierter Admission Controller (Abb. 7.16) entwickelt. Die Regelbasen der einzelnen Teilcontroller sind in den Tabellen 7.5 bis 7.8 dargestellt. Der Fuzzy Controller $FC_1$ verknüpft die Auslastung des Kanals $T_U$ mit der Belegung der Warteplätze $Q_U$. Die Beschreibung des Übertragungsverhaltens erfolgte bereits in Abschnitt 7.3. Die Tabelle 7.6 dokumentiert die Regelbasis des Fuzzy Controllers $FC_2$. Die Auswertung der Spalten ergibt, daß $P_2$ direkt abhängig ist von der temporären Priorität $P_1$. Das heißt, daß bei konstanter, normierter Bandbreite $S_{Bandbreite}$, die temporäre Priorität $P_2$ steigt, wenn $P_1$ größer wird. Übertragen bedeutet das, daß die Wahr-

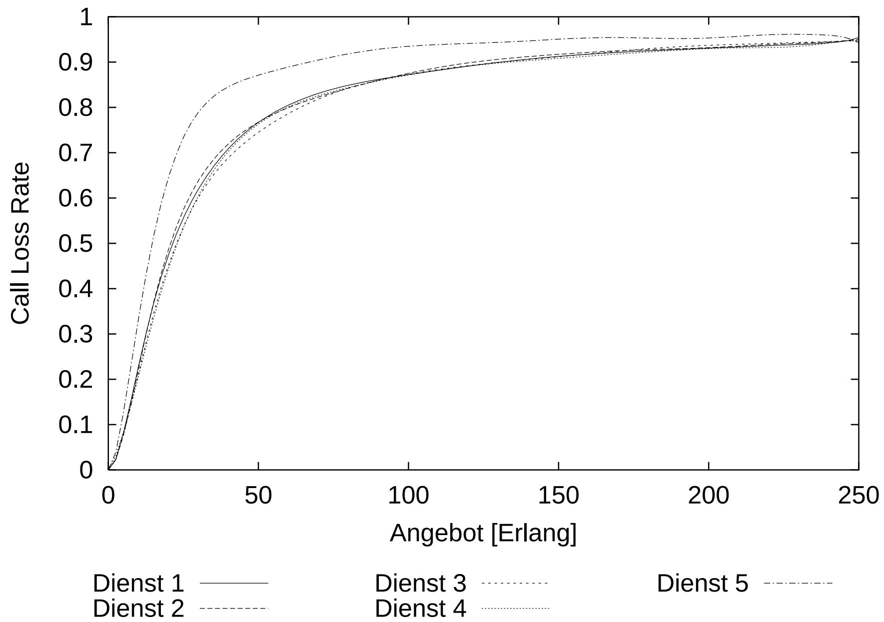



Abbildung 7.15: Verlauf der Call Loss Rate für $\varepsilon = T_U$

| $P_1$ | | $T_U$ | | |
|---|---|---|---|---|
| | | NK | N | K |
| $Q_U$ | SK | SG | SG | G |
| | K | SG | SG | G |
| | M | G | G | M |
| | G | K | K | K |
| | SG | SK | SK | SK |

Tabelle 7.5: Reglbasis des Fuzzy Controllers $FC_1$

scheinlichkeit, daß eine Verbindung geschaltet werden kann, vom Grad der Auslastung der Übertragungsressourcen abhängig ist. Stehen genügend Reserven zur Verfügung, nimmt die temporäre Priorität einen großen Wert an. Bei einer starken Auslastung des Systems wird der Betrag von $P_2$ verringert.
Die Auswertung der Zeilen ergibt eine reziproke Abhängigkeit der Priorität $P_2$ von der normierten Bandbreite $S_{Bandbreite}$. Je größer der Bedarf an Übertragungsbandbreite ist, desto kleiner ist die Wahrscheinlichkeit, daß eine Verbindung aufgebaut werden kann. Die Ausnahme stellt der Fall dar, daß $P_1$ den Wert „Sehr Groß“ annimmt. $P_2$ ist dann konstant „Sehr Groß“ .
Durch die Burstiness kann dieses Reglerverhalten teilweise relativiert werden. Nach Tabel-

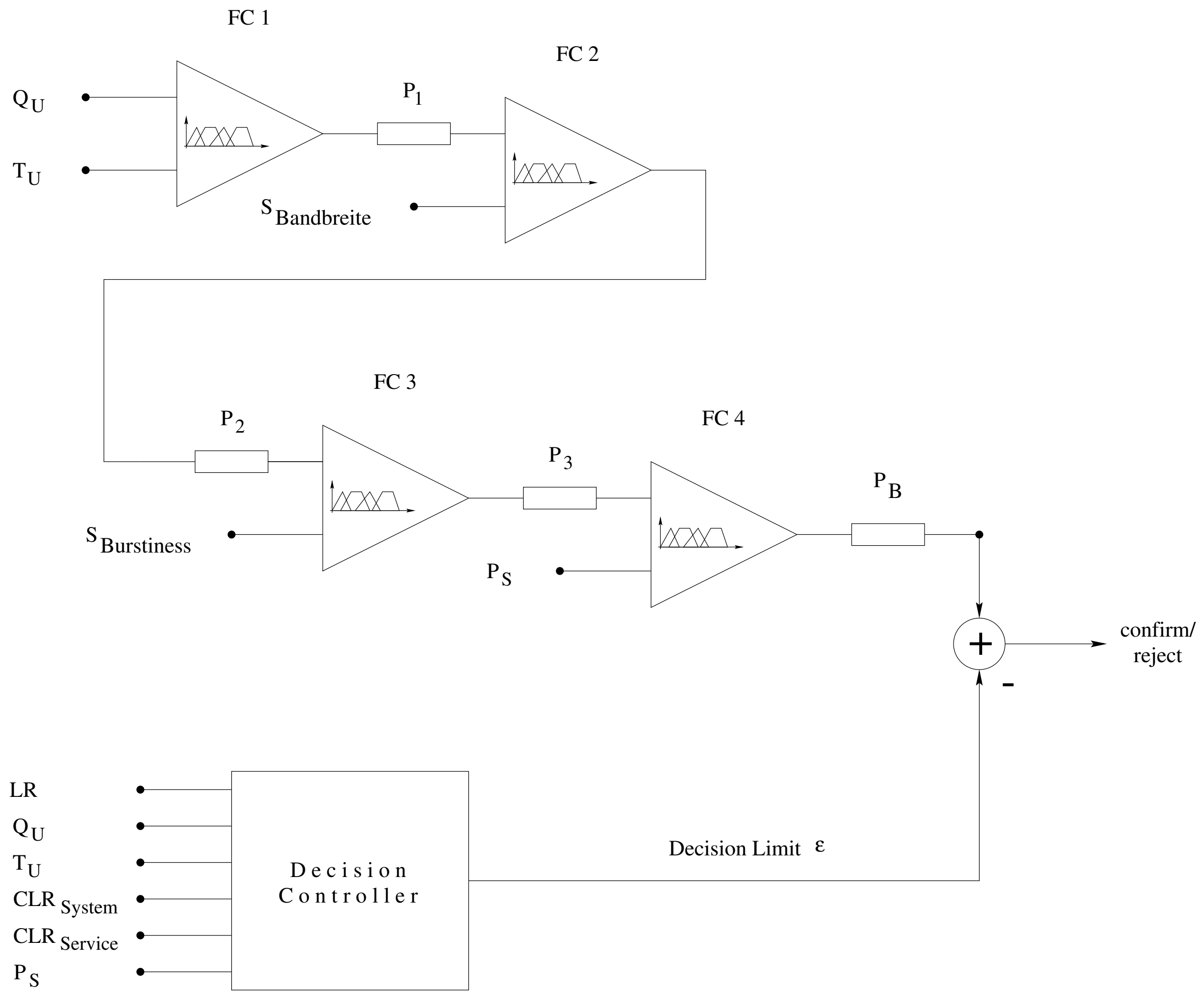



Abbildung 7.16: Struktur eines seriell aufgebauten Admission Controllers

| $P_2$ | | $S_{Bandbreite}$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_1$ | SK | K | K | SK | SK | SK |
| | K | M | M | K | K | SK |
| | M | G | G | M | K | K |
| | G | SG | SG | G | G | M |
| | SG | SG | SG | SG | SG | SG |

Tabelle 7.6: Reglbasis des Fuzzy Controllers $FC_2$

le 7.7, die die Regelbasis des Fuzzy Controllers $FC_3$ beschreibt, erfolgt in Abhängigkeit von der normierten Burstiness $S_{Burstiness}$ eine unterschiedliche Bewertung des Systemzustandes, der durch die temporäre Priorität $P_2$ beschrieben wird. Eine „Große" bzw. „Sehr

| $P_3$ | | $S_{Burstiness}$ SK | K | M | G | SG |
|---|---|---|---|---|---|---|
| $P_2$ | SK | SK | K | K | K | M |
| | K | K | K | K | K | M |
| | M | K | M | M | M | G |
| | G | M | M | M | G | SG |
| | SG | M | G | G | SG | SG |

Tabelle 7.7: Reglbasis des Fuzzy Controllers $FC_3$

Große" Burstiness $S_{Burstiness}$ impliziert, daß der Dienst ein Lastmuster mit einem sehr differenzierten Verkehrsaufkommen erzeugt. Es wechseln sich Phasen, in denen sehr viele Daten übermittelt werden müssen, mit Zeiträumen in denen das Verkehrsaufkommen sehr gering ist, ab. Das hat zur Folge, daß die dienstspezifischen Warteschlangen abgearbeitet werden können. Eine hohe Burstiness führt deshalb zu einer Vergrößerung der temporären Priorität $P_3$ im Vergleich zu $P_2$.
Auf der anderen Seite vermindert eine „Sehr Kleine" bzw. „Kleine" Burstiness $S_{Burstiness}$ die Priorität $P_3$ gegenüber $P_2$ für den Fall, daß die linguistische Variable $P_2$ die Werte „Mittel" , „Groß" oder „Sehr Groß" annimmt.
Die Bedienpriorität $P_B$ wird mit dem Fuzzy Controller $FC_4$ ermittelt. Die Regelbasis ist in Tabelle 7.8 dargestellt. Sie wird aus der temporären Priorität $P_3$ und der dienstspezifischen Priorität $P_S$ abgeleitet. Bei dem implementierten Ansatz beschreibt $P_S$ die Anforderungen,

| $P_B$ | | $P_S$ SK | K | M | G | SG |
|---|---|---|---|---|---|---|
| $P_3$ | SK | SK | SK | SK | SK | SK |
| | K | K | K | SK | SK | SK |
| | M | M | M | K | K | K |
| | G | G | G | G | G | G |
| | SG | SG | SG | SG | SG | SG |

Tabelle 7.8: Regelbasis des Fuzzy Controllers $FC_4$

die ein Dienst an das Übertragungssystem stellt. Eine große Dienstpriorität bedeutet in diesem Fall, daß eine strikte Bindung an die im Verkehrsvertrag vereinbarten Qualitätsparameter erfolgen muß. Eine kleine Priorität sagt aus, daß die Toleranzen weiter sind und

in dem vorgegebenen Maß ausgenutzt werden können. Diese Interpretation hat zur Folge, daß eine geringe temporäre Priorität $P_3$ zusammen mit einer großen Dienstpriorität dazu führt, daß eine Verbindung nicht aufgebaut werden darf. Dies zeigt sich in einer kleinen Bedienpriorität $P_B$. Ist $P_3$ auf Grund freier Ressourcen auf der anderen Seite „Groß“ oder „Sehr Groß“ führt dies zu einer Anhebung der Bedienpriorität $P_B$ im Vergleich zu $P_3$.
Um die Qualität dieses Controllers nachweisen zu können und um ihn mit den anderen Ansätzen vergleichen zu können, wurden auch hier unterschiedliche Funktionen benutzt, um eine Entscheidungsschwelle $\varepsilon$ zu bestimmen. Es wurden feste und dynamische Grenzen für die Bestimmung von $\varepsilon$ gewählt.

### 7.5.1 Fallbeispiel: $\varepsilon = const.$

Der Wertebereich von $\varepsilon$ ist das Intervall $[0.1, 0.9]$. Die Auswertung der Simulationsergebnisse zeigt, daß zwei unterschiedliche Bereiche separiert werden können. Für $\varepsilon < 0.6$ ist die Call Loss Rate für den Dienst 3 stets 0%. Dieses Verhalten ist durch die geringe dienstspezifische Priorität (Tabelle 4.2) zusammen mit der hohen Burstiness dieses Dienstes zu erklären. Infolgedessen sind die Paketverlustrate sowie die Auslastung der Warteplätze überdurchschnittlich hoch. Die Paketverluste der übrigen Dienste ist stets kleiner 10%. Die Auslastung der Pufferspeicher liegt bei durchschnittlich 50%.
Für $\varepsilon \geq 0.6$ ergibt sich eine gravierende Veränderung des Verhaltens des Controllers. Dadurch, daß die CLR für Dienst 3 jetzt $> 0\%$ ist, sich aber immer noch von der Call Loss Rate der anderen Dienste unterscheidet (Abb. 7.17), wird die Auslastung der Warteplätze

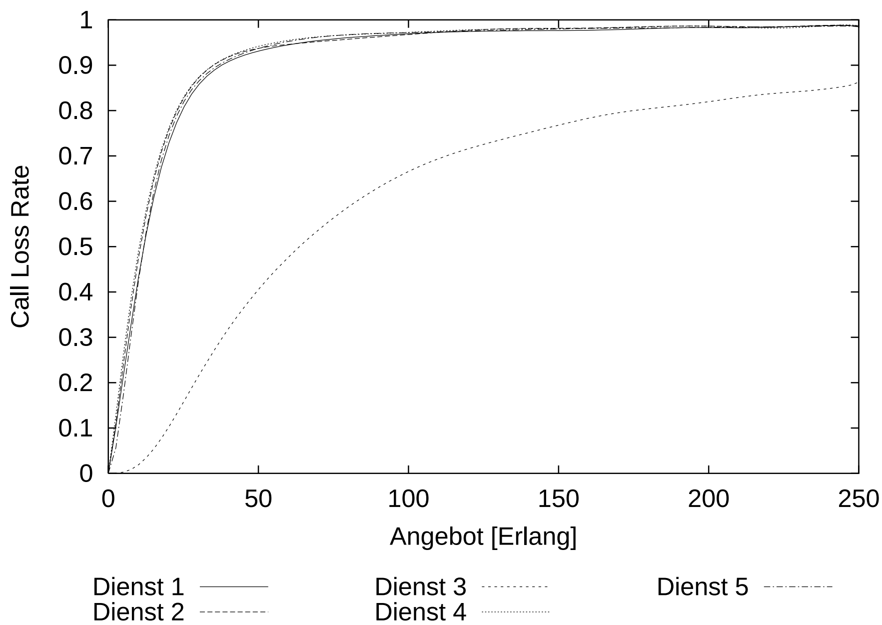



Abbildung 7.17: Verlauf der Call Loss Rate für $\varepsilon = 0.6$

geringer. Abbildung 7.18 zeigt den Verlauf der Auslastungskennlinien. Während sich Warteplätze der Dienste 1, 2 4 und 5 maximal 10% beträgt, wird die Warteschlange des Dienstes 3 erheblich stärker ausgelastet. Dieser differenzierte Verlauf schlägt sich dann auch in der

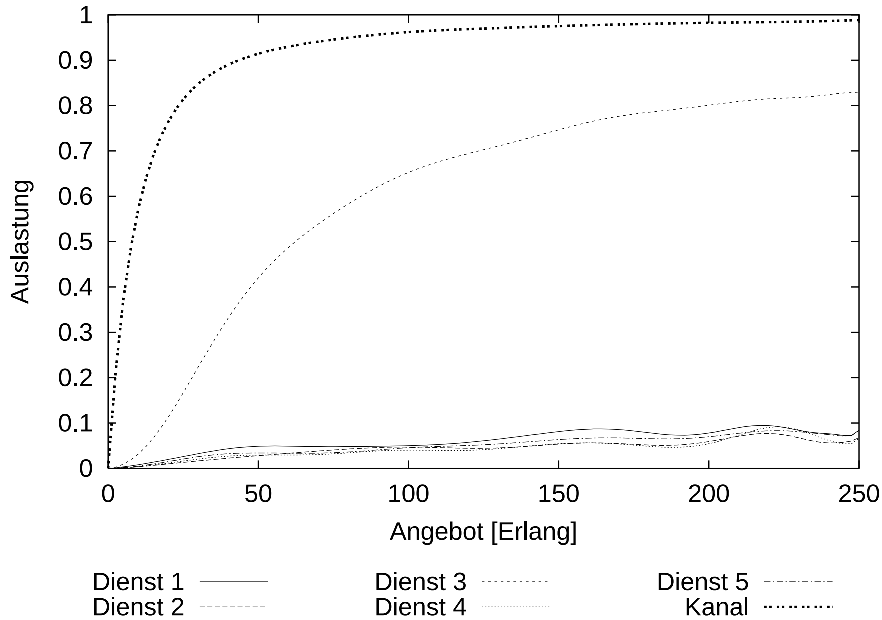



Abbildung 7.18: Auslastung der Übertragungsressourcen für $\varepsilon = 0.6$

Paketverlustrate ($< 18\%$ *bei* 250 *Erlang*) nieder. Die Paketverluste der übrigen Dienste ist stets kleiner 0.01%.

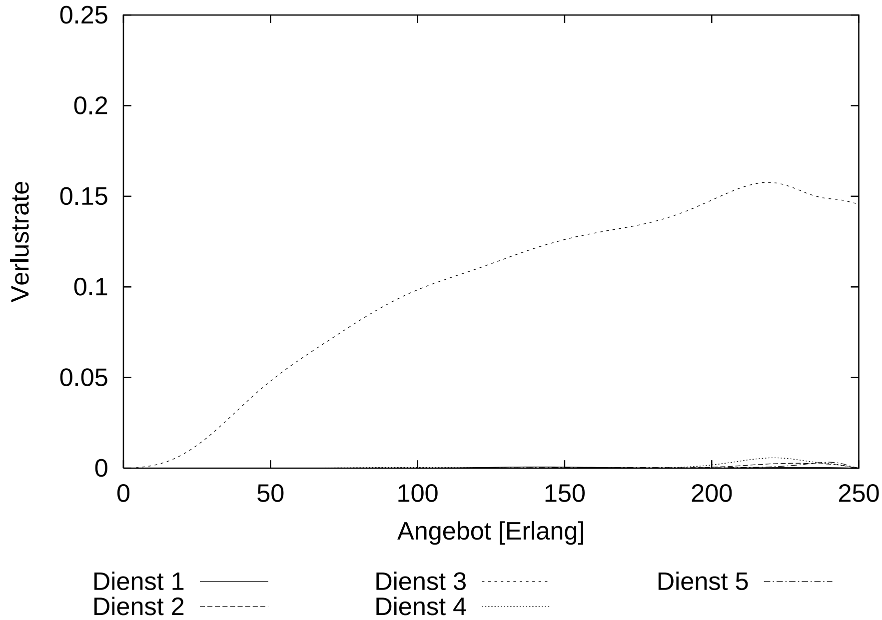



Abbildung 7.19: Verlustraten für $\varepsilon = 0.6$

### 7.5.2 Fallbeispiel: $\varepsilon = Q_U \cdot T_U$

Wie in den vorangegangenen Abschnitten beschrieben, wurde auch bei diesem Controller der Schwellwert $\varepsilon$ durch weitere dynamische Funktionen realisiert. Die Untersuchung umfaßte dabei die Fälle, daß $\varepsilon$ aus der lokalen sowie globalen Belegung der Warteplätze, der Auslastung des Links sowie unterschiedlichen Verknüpfungen dieser Parameter abgeleitet wurde. Stellvertretend für diese vielen Ansätze sollen im Folgenden die Ergebnisse vorgestellt werden, die sich bei der Berücksichtigung der lokalen Belegung der einzelnen Warteschlangen und der Auslastung des Kanals ergaben.

$$\varepsilon = Q_U \cdot T_U \tag{7.12}$$

Abbildung 7.20 zeigt den Verlauf der Call Loss Rate. Im Gegensatz zu dem in Abschnitt 7.5.1 beschriebenen Verfahren (Abbildung 7.17) zeichnet sich hier eine weitgehend einheitliche Behandlung der unterschiedlichen Dienste ab. Der Bereich in dem die Kennlinien bei

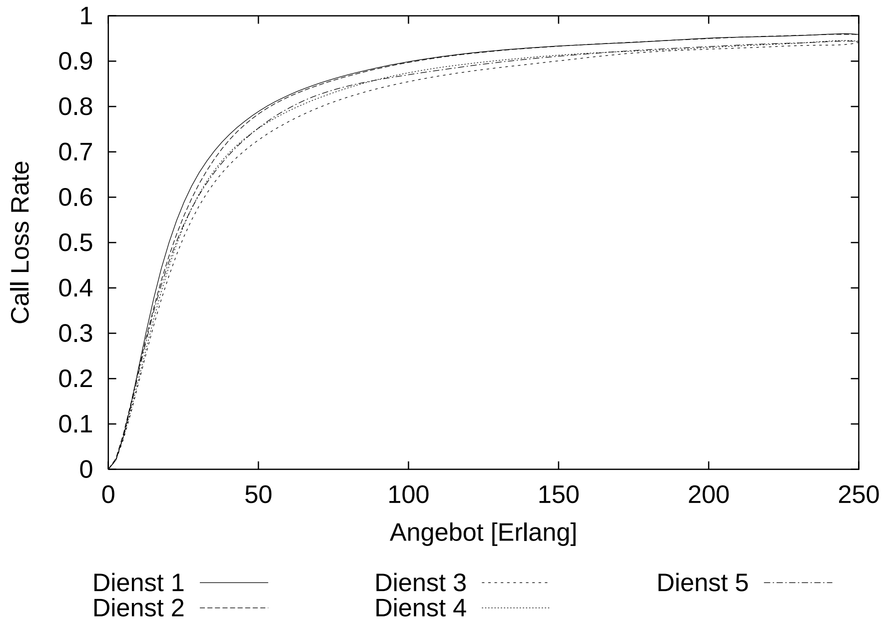



Abbildung 7.20: Verlauf der Call Loss Rate für $\varepsilon = Q_U \cdot T_U$

50 Erlang streuen, beläuft sich, wie der Abbildung 7.21 zu entnehmen ist, auf 6.3%. Dienst 1 hat mit 78.9% die größte Call Loss Rate. Die CLR von Dienst 3 beträgt 72.6%. Wegen seiner geringen Anforderungen in Kombination mit der hohen Burstiness kann dieser Dienst auch vermittelt werden, wenn die Übertragungsreserven gering sind. Das Verhalten spiegelt sich dann auch im Verlauf der Kennlinien der Paketverlustraten wider. Über den gesamten Verlauf ist die Verlustrate für alle Dienste kleiner 2%. Die höchste Verlustrate hat der Dienst 3. Die Verluste der Dienste 1 und 2 belaufen sich auf Werte kleiner 0.5%. Die Auslastung der Warteplätze liegt bei den Diensten 1, 2, 4 und 5 für ein Angebot größer

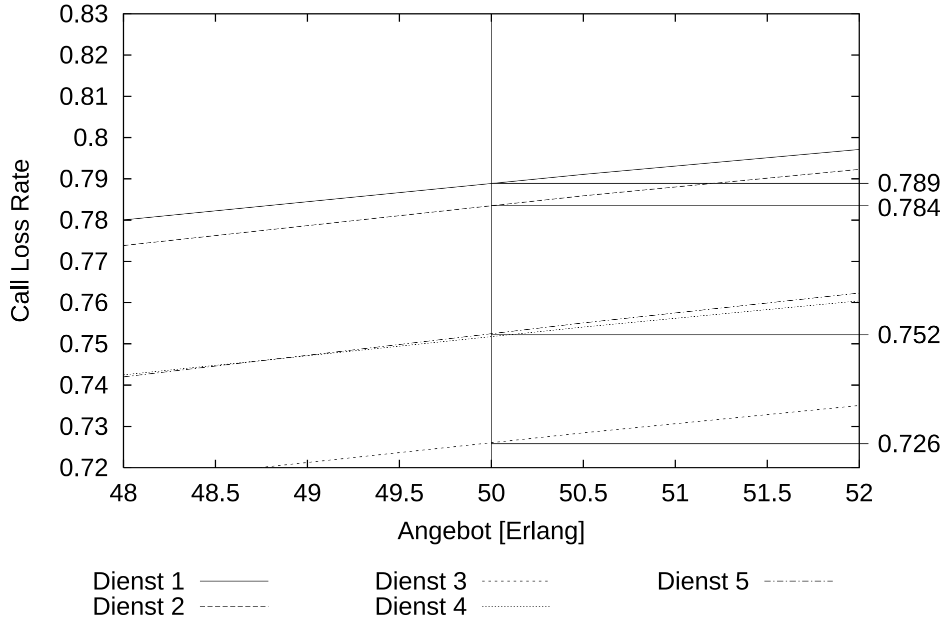



Abbildung 7.21: Verlauf der Call Loss Rate für $\varepsilon = Q_U \cdot T_U$

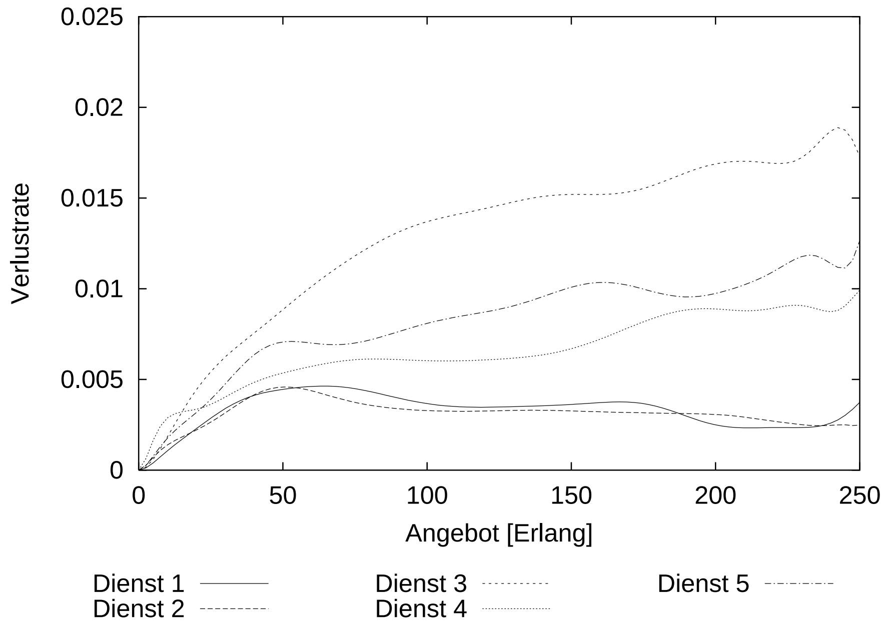



Abbildung 7.22: Verlustraten für $\varepsilon = Q_U \cdot T_U$

50 Erlang zwischen 56% und 78%. Die Belegung der Warteplätze von Dienst 3 ist jedoch noch größer. Bei 250 Erlang beläuft sie sich auf 90%. Das Verhalten dieses Connection

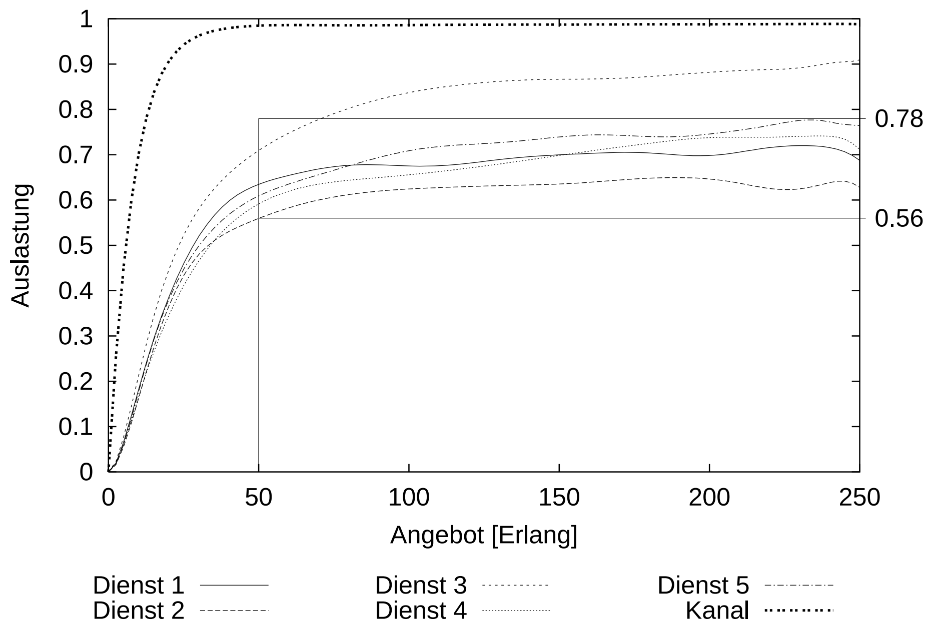



Abbildung 7.23: Auslastung für $\varepsilon = Q_U \cdot T_U$

Admission Controllers entspricht nahezu den Zielvorgaben, die in Abschnitt 7.1 definiert wurden. Die Auslastung der Warteplätze ist überdurchschnittlich. Die Paketverluste sind systemweit gering. Darüber hinaus ist die Verteilung der Call Loss Rate nahezu fair.

### 7.5.3 Weitergehende Untersuchungen

Neben den beschriebenen Verfahren wurde $\varepsilon$ mit dem in Abschnitt 7.3 dargestellten State Qualifier, der die $CLR_{Dienst}$ und die $CLR_{System}$ verknüpft, bestimmt. Die Ergebnisse zeigen, daß das Verhalten des Admission Controllers als „fair“ beschrieben werden kann. Die Streuung der CLR beläuft sich wiederum auf Werte kleiner 1% bei einem Angebot von 50 Erlang. Die Paketverlustrate ist in dem gesamten Bereich für alle Dienste kleiner 0.5%. Allerdings ist die Belegung der Warteplätze, wie auch die Ergebnisse in Abschnitt 7.3 gezeigt haben, mit maximal 10% sehr gering. Dieses Verhalten beeinflußt die Auslastung der Kanalressourcen. Die durchschnittliche Nutzung beträgt nur 30%.

## 7.6 Optimierung des Admission-Controllers

Wie bei den Policing Verfahren wurde auch hier versucht, mit dem Einsatz genetischer Algorithmen die in den Abschnitten 7.4 und 7.5 dargestellten Admission Controller zu optimieren.

### 7.6.1 Codierung der Regelbasen und Zugehörigkeitsfunktionen

In einem ersten Schritt erfolgte analog zu den in den Abschnitten 6.2.2 und 6.2.3 beschriebenen Vorgehensweisen, die Codierung der Regelbasen und Zugehörigkeitsfunktionen sowie der Entscheidungsschwelle $\varepsilon$.
Zur Bearbeitung der Parameter mit Hilfe des Genetischen Algorithmus wurden die Bitstrings entsprechend der Abbildung 7.13 miteinander verkettet werden.

$$\underbrace{001\cdots101}_{Regelbasis_i}\cdots\underbrace{111\cdots010}_{Regelbasis_n}\underbrace{0111001\cdots1101001}_{Zugehörigkeitsfunktion_1}\cdots\underbrace{1001101\cdots0101111}_{Zugehörigkeitsfunktion_n}\underbrace{0100101}_{Entscheidungsschwelle} \tag{7.13}$$

### 7.6.2 Fitnessfunktion

Wie das Kapitel 6.2.6 gezeigt hat, ist die Bestimmung der Fitness Funktion sehr zeitaufwendig und setzt ein detailliertes Verständnis des Systems sowie der Parameter und deren Abhängigkeiten voneinander voraus. Auch bei der Simulation des Admission Controllers ist es natürlich möglich, viele unterschiedliche Funktionen herzuleiten. Da aber der Ansatz mit Hilfe einer Fuzzy Logic basierten Fitness Funktion einfach war und darüber hinaus sehr gute Ergebnisse lieferte, wurde auch hier die Fitness mit Hilfe eines Fuzzy Controllers ermittelt.

### 7.6.3 Simulation

Wie in Kapitel 6.2.6 wurde mit einer Mutationsrate von 0.01 sowie einer Crossover-Wahrscheinlichkeit von 0.6 gearbeitet.
Die Untersuchungen lieferten aber trotz einer parallelen Simulation auf mehreren Rechnern und eines mehrwöchigen Beobachtungszeitraums bei beiden Topologien keine brauchbaren Ergebnisse. Es zeigte sich, daß Änderungen der Entscheidungsschwelle einen gravierenden Einfluß auf die Entwicklung brauchbarer Lösungen hatten. Im Folgenden wurde dann auf die Codierung der Entscheidungsschwelle verzichtet. Aber auch nach diesem Eingriff wurden keine verwertbaren Controller entwickelt.

## 7.7 Bewertung der Admission Controller

In den in den Kapitel 7.2, 7.4 und 7.5 beschriebenen Untersuchungen über die Leistungsfähigkeit der Fuzzy Logic basierten Call Admission Controller wurden im Wesentlichen zwei unterschiedliche Strategien verfolgt.

1. Bei dem ersten Ansatz wurde der CAC, wie in Abbildung 7.1 durch zwei funktionale Einheiten realisiert. Es war eine klare Trennung zwischen einer Instanz zur Bestimmung der Anforderungen und einer weiteren zur Klassifikation der Auslastung des Zugangsknotens gegeben. Mit Hilfe des Traffic Qualifiers wurden die Anforderungen

der neuen Verbindung charakterisiert. Der State Qualifier diente zur Abschätzung des Zustandes des Zugangsknotens. Durch einen einfachen Vergleich der Anforderungen mit den zur Verfügung stehenden Ressourcen konnte dann ermittelt werden, ob eine Verbindung geschaltet werden konnte.
Bei der Untersuchung erfolgte die Bestimmung der Übertragungsanforderungen mit Hilfe einer Fuzzy Logic basierten Einheit. Die Auslastung der Ressourcen dagegen wurde mit unterschiedlichen Hilfsgrößen angenähert. Zur Auswahl standen unterschiedliche Systemgrößen wie die Auslastung der Warteplätze und Bandbreite, die Paketverlustrate sowie die Call Loss Rate zur Verfügung. Darüber hinaus können diese Parameter in beliebigen Kombinationen miteinander verknüpft werden.
Die Simulationsergebnisse zeigen, daß das Leistungsspektrum in Abhängigkeit von den Kenngrößen stark variierte. Bei Anwendung einer konstanten Funktion zur Beschreibung des Zustandes der Zugangseinheit ergab sich, daß die Paketverluste schon bei einem geringen Angebot über 60% lagen. Die Warteplätze und der Link wurden vollständig ausgelastet. Die CLR wechselte bei einem Schwellwert abrupt zwischen 0% und 100%.
Dieses Verhalten konnte durch die Wahl einer dynamischen, lastabhängigen Systemgröße zur Beschreibung der Auslastung der Ressourcen verbessert werden. Die Paketverlustraten sowie die CLR konnten bei Verwendung der auf den Seiten 128 bis 132 beschriebenen Funktionen drastisch reduziert werden. Die Ressourcen des Knotens wurden überdurchschnittlich genutzt, aber nicht in Überlast betrieben.
Neben dieser erheblichen Steigerung der Leistungsfähigkeit, konnte im Weiteren durch den Einsatz einer Fuzzy Logic basierten Einheit zu Abschätzung der Auslastung des Zugangsknotens die Umsetzung der Fairness Funktion auf einfache und effektive Weise, wie im Abschnitt Funktion 7 auf Seite 134 beschrieben, erreicht werden.

2. Bei dem zweiten Ansatz wurde eine abweichende Strategie verfolgt. Mit Hilfe eines Fuzzy Controllers, der sich aus mehreren Teilcontrollern zusammensetzte, sollte durch eine unscharfe Beurteilung der verschiedensten System- und Lastgrößen eine Bedienpriorität bestimmt werden. Ein Vergleich mit einem Schwellwert $\varepsilon$ entschied schließlich darüber, ob eine Verbindung aufgebaut werden konnte.

   (a) Bei dem mehrstufigen parallelen Admission Controller nach Abschnitt 7.4 wurde der Schwellenwert mit Hilfe verschiedener Systemgrößen und Funktionen gestaltet. Die Wahl eines konstanten $\varepsilon$ zeigte über den gesamten Wertebereich, gegenüber den in Abschnitt 7.3 beschriebenen Kennlinien, ein erheblich verbessertes Verhalten. Bei diesem neuen Ansatz erfolgt der Vergleich einer variablen, von der Auslastung der Ressourcen abhängigen Bedienpriorität, mit der konstanten Entscheidungsschwelle. Infolgedessen existiert hier kein Grenzwert bei dem sich das Verhalten abrupt ändert. Die CLR liegt für alle Schwellwerte unterhalb von 90%. Die Paketverlustrate hat bei allen Entscheidungsschwellen einen Wert kleiner 12%. Durch die Wahl von dynamischen Entscheidungsschwellen ist eine weitere Verbesserung erreicht worden. Die Auslastung der Ressourcen des

Zugangsknotens ist überdurchschnittlich. Die Paketverlustraten streuen zwar dienstabhängig und ändern sich auch mit der Schwellwertfunktion, können aber in angemessenen Lastsituationen bis $\approx 100$ *Erlang* auf akzeptable Werte gesenkt werden.

(b) Der seriell strukturierte Admission Controller, dessen Simulationsergebnisse in Abschnitt 7.5 präsentiert wurden, zeigt ein ähnliches Verhalten. Auch hier erfolgt der Vergleich einer dynamischen, auslastungsabhängigen Bedienpriorität mit unterschiedlichen Schwellwerten.
Die Paketverlustraten sind sowohl bei Einsatz eines konstanten als auch dynamischen Schwellwertes sehr gering und erfüllen die Anforderungen der Dienste. Die Belegung der Warteplätze ist überdurchschnittlich. Es zeigt, daß der Einfluß der Burstiness das Verhalten des Controllers maßgeblich beeinflußt. Weist ein Dienst eine hohe Burstiness in Kombination mit einer geringen Dienstpriorität auf, führt das zu einer Steigerung der Auslastung und einer Vergrößerung der Paketverlustrate.

Die Untersuchung hat gezeigt, daß mit Fuzzy Logic basierten Call Admission Controllern die in Abschnitt 7.1 beschriebenen Randbedingungen leicht umgesetzt werden konnten. Die Auslastung der Warteplätze war überdurchschnittlich. Die dienstspezifischen sowie die systemweiten Paketverlusteraten konnten auf einen tolerierbaren Wert begrenzt werden.
Neben diesen für den Betrieb des Netzes essentiellen Randbedingungen konnte auch gezeigt werden, daß es möglich ist neue, bisher noch nicht berücksichtigte Ziele zu verfolgen. In dem Abschnitt 7.3 wurde ein Controller entwickelt der eine faire Behandlung der Dienste, die alle eine nahezu identische CLR aufweisen, erlaubt.
Festzuhalten bleibt, daß es mit Hilfe der Fuzzy Logic basierten Call Admission Controller leicht möglich war, die definierte Zielsetzung umzusetzen.

# Kapitel 8

# Neuronale Netze

Die Praxis zeigt bei konventionellen Systemen wie sie in den vorhergehenden Kapiteln untersucht wurden, daß sich menschliche Erfahrung schwer in einer formalen Art und Weise wiedergeben und auf Regler projizieren läßt. Diese Probleme verstärken sich natürlich dann auch noch mit der Komplexität und der Dynamik des Systems. Die Fuzzy Logik stellte einen ersten Ansatz dar, um die Erfahrung und Anschauung in den Entwurf einer Steuerung einfließen zu lassen. So wurde die Fuzzy Logik erfolgreich eingesetzt, um das Verhalten der Controller zu optimieren. Ein gutes Ergebnis zeichnet sich aber ab, wenn bestimmte Randbedingungen in das Verhalten des Reglers übernommen werden sollen.
Bei der Bestimmung von Fuzzy Parametern, d. h. Regelbasen, Mitgliedsfunktionen und Verbindungsgewichten traten jedoch vielfach Schwierigkeiten auf: nämlich logische Konzepte mit präzisen Aktionen zu verbinden. Die Folge davon waren dann Iterationsschritte, bei denen die Parameter auf Grund von Simulationsergebnissen angepaßt wurden. Der Optimierungsprozeß war deshalb langwierig und setzte fundierte Kenntnisse über die Systemabläufe und die Abhängigkeiten der Parameter untereinander voraus. Infolge dessen sollen hier Verfahren zum Einsatz kommen, mit denen es möglich ist, eine objektive Anpassung des Regelverhaltens zu erreichen. Mit solch einem Ansatz kann dann das Reglerverhalten adaptiert werden, wenn keine genaue Zuordnung zwischen Eingangs- und Ausgangsgößen gegeben oder erkennbar oder aber, wenn das Systemverhalten dynamisch ist. Auf Grund dieser Tatsache soll im Weiteren untersucht werden, ob diese Systeme mit Hilfe neuronaler Netze entworfen werden können. Die Künstlichen Neuronalen Netze (KNN) sind vielen deterministischen und statistischen Anaylsemethoden überlegen, weil sie

- nichtlineare Systeme steuern können
- verrauschte und unregelmäßige Daten aus der Umwelt verarbeiten können
- sehr schnell auf komplexe Aufgaben reagieren können
- leicht und schnell trainiert werden können
- Informationen von vielen Variablen und Parameter verarbeiten können
- verallgemeinern können

## 8.1 Grundlagen künstlicher Neuronaler Netze

Es existieren in der Literatur unzählige verschiedene Möglichkeiten, neuronale Netze zu beschreiben. Für diese Arbeit wurde folgende Definition[1] des Begriffs *künstliches neuronales Netz* zugrunde gelegt:

> Ein neuronales Netz besteht aus einer Menge einfacher, untereinander verbundener Verarbeitungseinheiten, deren Arbeitsweise natürlichen Nervenzellen nachempfunden ist. Die Funktionalität des Netzes ist in den *gewichteten* Verbindungen, die zwischen den einzelnen Knoten bestehen, abgelegt. Die Anpassung dieser Verbindungsstärken ist induktiv und erfolgt auf Grund von Trainingsmustern.

Künstliche Neuronale Netze sind den biologischen neuronalen Netzen nachgebildet. Sie bestehen aus vielen einfach aufgebauten Verarbeitungseinheiten, die untereinander über gewichtete Verbindungen miteinander gekoppelt sind. Sie weisen eine massiv parallele Struktur auf. In Analogie zu den biologischen Netzen nennt man diese Verarbeitungseinheiten dann Neuronen. Abbildung 8.2 zeigt das Modell eines Neurons. Es beschreibt nicht exakt alle Aspekte eines natürlichen Neurons, sondern stellt eine grobe Verallgemeinerung dar. In Anlehnung an das von McCulloch und Pitts im Jahre 1943 vorgeschlagene Modell kann das Neuron als Addierer mit Schwellwert aufgefaßt werden. Diese Neuronen werden in den sogenannten Feed-Forward Netzen schichtweise zusammengefaßt und über gerichtete Verbindungen zwischen den Schichten zu einem hierarchisch gestaffelten Netz zusammengeschaltet. Neuronen innerhalb einer Schicht sind nicht assoziiert.

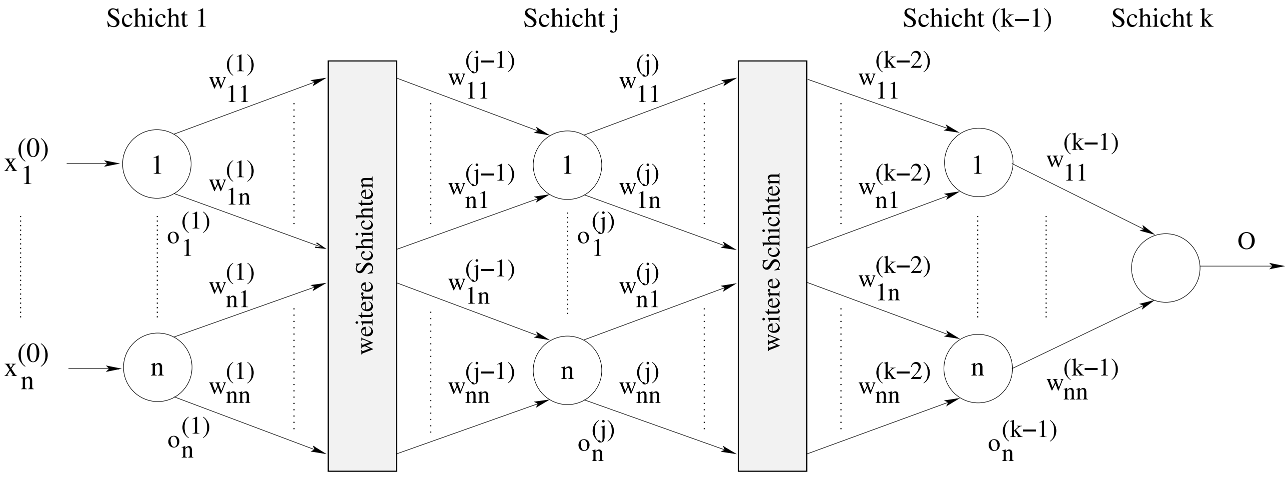



Abbildung 8.1: Prinzipieller Aufbau eines Neuronalen Netzes

Das in Abbildung 8.1 dargestellte KNN besteht aus k Schichten für die im Folgenden

[1]nach Dr. Kevin Gurney; Department of Psychology; University of Sheffield

auch der synonyme Ausdruck Layer benutzt wird. Die Schicht 1 ist der sogenannte Input-Layer, die Schicht k entspricht dem Output-Layer. Die dazwischenliegenden Ebenen stellen die sog. Hidden-Layers dar. Je nach Aufgabe, die das KNN übernehmen soll, kann die Anzahl der Hidden-Layer $N_{Schicht} = j, j = 1...(k-2)$ sowie die Zahl der Neuronen $N^{(j)}_{Neuronen} = i, i = 1...n$ in den unterschiedlichen Schichten variieren. Am Eingang des Netzes wird den Input Neuronen der Eingabevektor $x = (x^{(0)}_1, x^{(0)}_2, ..., x^{(0)}_i, ..., x^{(0)}_n)$ zur Bearbeitung übergeben. Die Eingangssignale werden normiert und zur weiteren Bearbeitung an die nachfolgenden verdeckten Schichten weitergeleitet. Die Aktivierungen der einzelnen Neuronen werden dann durch das Netz zum Ausgang propagiert, wo sie zusammengefaßt und ausgegeben werden. Die einzelnen Schichten sind über gewichtete und gerichtete Verbindungen miteinander verknüpft. Die Verbindungsgewichte $w^{(j)}_{in}$ sind so zu interpretieren, daß der $hochgestellte^{Index}$ den Layer in dem das Ausgangsneuron verankert ist, angibt. Die $tiefgestellten_{Indizes}$ bezeichnen die Nummern der Start- und Endneuronen innerhalb der unterschiedlichen Schichten j und (j+1). Der Output Layer besteht bei dem eingesetzten Netzwerk aus nur einem Neuron und liefert das Verarbeitungsergebnis in Abhängigkeit von den gegebenen Gewichtungen und der Topologie des Netzes. Der Aufbau einer solchen Verarbeitungseinheit oder Neuron ist in der Abbildung 8.2 schematisch dargestellt. Im All-

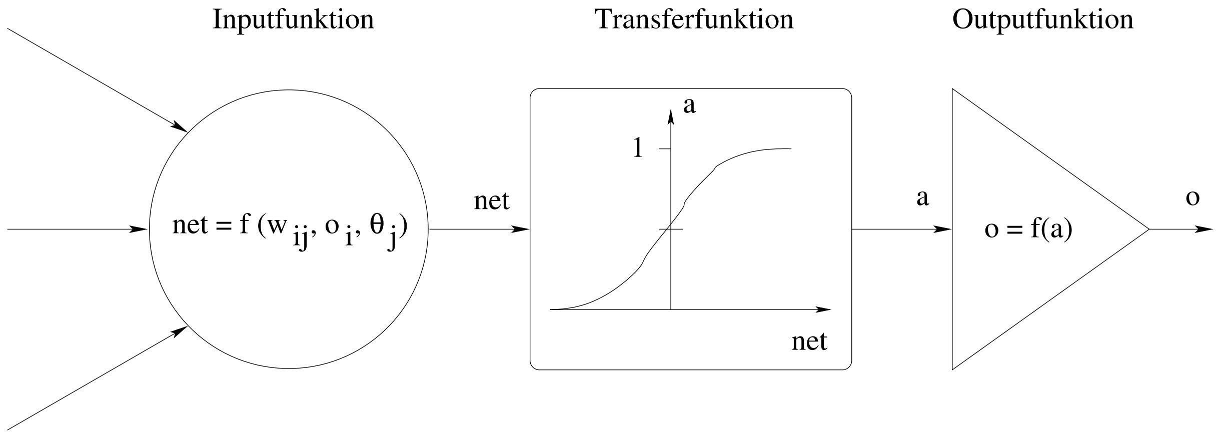



Abbildung 8.2: Elementare Struktur eines Neurons

gemeinen wandelt jeder Knoten die Aktivitäten aller vorgeschalteten Neuronen in einen einzigen Ausgabewert um, der dann als Eingabeaktivität für die nachfolgenden Verarbeitungseinheiten dient. Diese Transformation geschieht im Wesentlichen in drei Stufen.

### 8.1.1 Die Inputfunktion

Die Inputfunktion wichtet die eingehenden Signale $o_i$ mit dem Faktor $w_{ij}$ und faßt sie zu einer Gesamteingabe, dem sogenannten Nettoinput $net$, zusammen. Bei den folgenden Gleichungen stellt $w^{k-1}_{ij}$ das Verbindungsgewicht zwischen dem i-ten Neuron der Schicht (k-1) und dem Neuron j der Schicht k dar. $o^{k-1}_i$ ist der Ausgabewert der Verarbeitungseinheit

$i$ der Ebene $k-1$, während $\Theta$ den sogenannten Biaswert des Neurons verkörpert. Die Gleichungen 8.1 und 8.2 zeigen zwei in vielen Systemen bewährte, eingesetzte eingesetzte Inputfunktionen.

$$net_j^{(k)} = \sum_{i=1}^{n} w_{ij}^{k-1} o_i^{k-1} + \Theta \tag{8.1}$$

In Gl. 8.1 werden die Eingangswerte $o_i^{k-1}$ mit den assoziierten Gewichtungen multipliziert und dann zu der gewichteten Gesamteingabe aufsummiert.

$$net_j^{(k)} = \prod_{i=1}^{n} w_{ij}^{k-1} o_i^{k} + \Theta \tag{8.2}$$

Bei einer Realisierung der Inputfunktion nach Gl. 8.2 werden die gewichteten Eingaben aufmultipliziert.

### 8.1.2 Die Transferfunktion

Die Transferfunktion gibt für jedes Neuron den sog. Aktivierungszustand $a$ in Abhängigkeit von dem Nettoinput $net_j$ an. Das Verhalten des Netzes wird gerade durch diese Aktivierungsfunktion nachhaltig geprägt. Es kommen hier vielfältige Funktionen, wie lineare, Schwellwert-, stochastische und semilineare Transformationen zum Einsatz.

- Lineare Verarbeitungseinheiten einschließlich der Identität erzeugen eine Aktivierung, die proportional zur Gesamteingabe ist. Bei dieser Klasse von Transferfunktionen werden die Netzaktivitäten durch eine lineare Abbildung oder direkt in den Aktivierungszustand überführt. Trotz der einfachen Transformation wird dieser Abbildungstyp selten eingesetzt, da die Signale vielfach begrenzt werden müssen.

- Bei Schwellwertsystemen wird der Nettoinput mit einem konstanten Grenzwert verglichen. Entsprechend der daraus entwickelten Klassifikation der Netzeingangs wird die Aktivierung der Verarbeitungseinheit bestimmt. Diese Klasse von Aktivierungsfunktionen wird in vielen Fällen eingesetzt, weil sie eine Möglichkeit darstellt, mit diskreten Aktivierungszuständen zu arbeiten.

- Eine weitere Klasse bilden die semilinearen Aktivierungsfunktionen, die in der sog. sigmoiden Transferfunktion (Gl. 8.3) den bedeutendsten Vertreter hat.

  $$a(net_j) = \frac{1}{1+e^{-net_j}} \tag{8.3}$$

  Sie erfüllt die Eigenschaften linearer Funktionen bezüglich ihrer Stetigkeit und Differenzierbarkeit; denn anders als bei den Schwellwertsystemen sind hier keine Sprünge von Ruhe zu voller Aktivität möglich. Ein Übergang vollzieht sich kontinuierlich. Allerdings bietet sie genau wie die Schwellwertsysteme gute Eigenschaften bezüglich der Separationsfähigkeit. Neben diesen Charakteristika erfolgt eine Begrenzung der Aktivierung, so daß auch der Einfluß sehr großer Signale nicht über alle Maßen wachsen kann. Kleine Signaländerungen, um den Arbeitspunkt, auf die das Netz empfindlicher reagieren sollte, werden verstärkt, da die Funktion ihre höchste Sensibilität (ihre größte Steigung) im Bereich um den Wendepunkt hat.

### 8.1.3 Die Outputfunktion

Die Weitervermittlung des ermittelten Aktivierungszustandes als Größe $o_i(t)$ an die Verbindungsstruktur des Netzverbundes bzw. an den zu steuernden Prozeß obliegt der Ausgabefunktion. Die meisten Netzmodelle wählen als Ausgabefunktion die Identität (Gl. 8.4), so daß der Aktivierungszustand einer Verarbeitungseinheit ohne erneute Transformation weitergegeben wird.

$$o = f(a) = a \tag{8.4}$$

In einigen Fällen wird diese Funktion allerdings benutzt, um eine Normierung der Aktivierung durchzuführen. Bei Herleitungen und Rechnungen kann durch die Identitätsfunktion die Ausgabe $o_i$ der einzelnen Neuronen durch ihre Aktivierung substituiert werden.

## 8.2 Lernverfahren

Da die Anzahl der Parameter, die das Verhalten des KNN beeinflussen, sowie die Struktur (Anzahl Neuronen, Verknüpfungen, Transferfunktion, etc. ), die das Verhalten des KNN im Wesentlichen beeinflussen, von der Komplexität des zu steuernden Systems abhängt, ist es fast unmöglich, a priori die Gewichtungen so einzustellen, daß das Neuronale Netz eine akzeptable Beziehung zwischen den Eingangs- und Ausgangsgrößen liefert. Für die Abstimmung des Netzes werden dann Verfahren benötigt, die ein Adaptieren der Parameter ermöglichen. Im Folgenden sollen einige dieser Verfahren, die auch später in der Simulation auf ihre Verwendbarkeit geprüft werden, vorgestellt werden. Die für neuronale Netzwerke existierenden Lernverfahren können im Allgemeinen in zwei Gruppen unterteilt werden.

### 8.2.1 Lernen mit Unterweisung (Supervised Learning)

Bei diesem Verfahren muß eine Transformation $T$ entwickelt werden, die die Eingangswerte x auf einen bestimmten vorgegebenen Ausgangsvektor abbildet. Da diese Abbildungsvorschrift beliebig komplex sein kann, soll sie von vorgegebenen Trainingsmustern $(x_1, y_1), (x_2, y_2), ..., (x_i, y_i), ..., (x_n, y_n)$ abgeleitet werden. Zu diesem Zweck werden die Eingabewerte wiederholt dem Netz übergeben und der Ausgang wird mit gewünschten Werten verglichen. Abweichungen zwischen den gewünschten und tatsächlichen Werten führen dazu, daß die Gewichte an Hand eines Algorithmus so verändert werden, daß der Fehler bis auf eine akzeptablen Wert verkleinert wird. Am Ende der Adaptionsphase sollte die Transformation $T$ dann folgende Bedingungen erfüllen.

$$\forall i \in 1, ..., n : T(x_i) = y_i \tag{8.5}$$

$$\forall x \notin x_1, x_2, ..., x_n : T(x) = y \tag{8.6}$$

Die Gleichung 8.5 gibt an, daß das Neuronale Netz die Transformation ausreichend gut entwickelt hat, um die Eingangswerte $x_i$ auf die gewünschten Ausgangswerte abzubilden. Gleichung 8.6 zeigt an, daß das Netz in der Lage ist, auch auf nicht trainierte Eingangs-Ausgangsbeziehungen adäquat zu reagieren. Das heißt, daß das System ausgehend von den Trainingsmustern soweit verallgemeinern kann, daß der vollständige Zustandsraum erschlossen wird.
Neben diesem Verfahren existieren noch Algorithmen, die Netze ohne Unterweisung adaptieren.

### 8.2.2 Lernen ohne Unterweisung (Unsupervised Learning)

Für das Lernen ohne Unterweisung wird lediglich der Eingabevektor benötigt. Das neuronale Netz muß iterativ und ohne Unterweisung Muster, Korrelationen und Klassifizierungen finden und abspeichern. Die Bewertung der der Netzausgabe erfolgt dann nicht durch vorgegebene Muster sondern wird z. B. aus der Reaktion des zu steuernden Prozesses abgeleitet.

## 8.3 Lernalgorithmen

Die oben beschriebenen Verfahren ermöglichen es, Unterschiede zwischen dem gewünschten und dem tatsächlichen Aktivierungszustand des Neuronalen Netzes festzustellen. Lernregeln werden dann im Weiteren benötigt, um die Verbindungsgewichte zwischen den einzelnen Neuronen so zu adaptieren, daß das Neuronale Netz die Eingangswerte beliebig genau auf die gewünschten Ausgangswerte abbildet. Bei mehrschichtigen Systemen mit einer rückkopplungsfreien Netzwerkstruktur stellt der sog. *Backpropagations-Algorithmus* ein robustes Verfahren zur Anpassung der Gewichtungen zur Verfügung.

Das Ziel bei diesem Lernverfahren ist es, die Verbindungsgewichte des Netzes so zu verändern, daß der *quadratische Fehler* zwischen den an den Ausgängen des Netzes gewünschten Sollwerten $o_{Soll,i}$ und den tatsächlichen Werten $o_{Ist,i}$, die das Netz liefert, minimiert wird. Die Verringerung des Fehlers soll alle Trainingsmuster umfassen. Der absolute Fehler läßt sich bei den gegebenen Voraussetzungen dann wie folgt mathematisch beschreiben:

$$E_{ges} = \sum_{m \in Muster} \underbrace{\left( \frac{1}{2} \sum_{k \in Ausgänge} \left( o_{Soll,k} - o_{Ist,k} \right)^2 \right)}_{E_m} \tag{8.7}$$

$E_m$ ist der Fehler für ein beliebiges Muster m. Dieser so ermittelte Fehler wird dann *rückwärts*, d. h. vom Ausgang des Netzes zum Eingang *zurück (back)* propagiert. Die Anpassung der Gewichtungen erfolgt dann sukzessive unter Anwendung von Gl. 8.8.

$$w_{ij}(t+1) = w_{ij}(t) + \sigma \cdot \delta_j \cdot o_i \tag{8.8}$$

$\sigma$ stellt die sog. Lernrate dar, $\delta_i$ ist das Fehlermaß, das sowohl die Größe der Gewichtsänderung als auch die Richtung in der diese Anpassung erfolgen soll, vorgibt. Abgeleitet wird dieses Maß mit Hilfe der sog. *Delta - Regel* für die Ausgangsneuronen bzw. der *generalisierten Delta-Regel* für Gewichtungen in den verdeckten Schichten. Wichtig ist, daß diese Regeln die Gewichte immer in Richtung des steilsten Gradienten in Bezug zur Fehleränderung adaptieren. Ausgehend von der Gleichung 8.9

$$\Delta w_{ij} = -\sigma \frac{\partial E_m}{\partial w_{ij}} \tag{8.9}$$

ergibt sich für das Fehlermaß $\delta_j$ der Ausgabeschicht folgender Zusammenhang [2].

$$\delta_j = (o_{soll,j} - o_{ist,j}) \cdot o_{ist,j} \cdot (1 - o_{ist,j}) \tag{8.10}$$

Dieser Fehler wird zurückpropagiert und zur Berechnung des Fehlersignals der vorgelagerten Schicht benutzt. Da für die Neuronen dieser Schichten kein Sollwert $o_{soll}$ vorgegeben werden kann, muß diese Größe durch einen äquivalenten Faktor ersetzt werden. Durch Anwendung der Kettenregel kann der Lernfehler dann rekursiv aus dem Produkt des Fehlersignals der übergeordneten Neuronen und den assoziierten Gewichtungen gewonnen werden (Gl. 8.11).

$$\delta_j = (\sum_j w_{ij} \cdot \delta_i) \cdot o_j \cdot (1 - o_j) \tag{8.11}$$

Der Vorgang wird so lange wiederholt, bis der Input Layer erreicht ist. Die Fehler $\delta_j$ sind dann bekannt, so daß die Gewichte der einzelnen Verbindungen angepaßt werden können (Gl. 8.8).
Das Backpropagationsverfahren ist robust und seit vielen Jahren in den unterschiedlichsten Applikationen im Einsatz. Das eigentliche Problem bei der Verwendung des oben beschriebenen Algorithmus besteht darin, daß er auf einem überwachten Lernverfahren mit einer festen vorgegebenen Beziehung zwischen den Eingängen und Ausgängen basiert. Diese Methoden setzen also voraus, daß zu gegebenen Eingabevektoren korrespondierende Ausgangswerte existieren, so daß ein Zusammenhang zwischen diesen Werten abgeleitet werden kann.

Neben dieser Einschränkung ist ein weiteres Problem, daß die konventionellen Verfahren in vielen Fällen nur bei einem sog. *Offline-Training* erfolgversprechend agieren können.

Bei den hier untersuchten Systemen ist der Zusammenhang zwischen den unterschiedlichen Parametern so komplex und dynamisch, teilweise sind die erwarteten Reaktionen konträr, so daß keine eindeutige Abbildung der Eingangswerte auf den Ausgang, die die Entwicklung einer Übertragungsfunktion zuließe, vorgegeben werden kann. Vielmehr muß sich das Antwortverhalten des Controllers *Online* aus dem vorliegenden Systemzustand und den gegebenen Eingangsmustern ableiten lassen und wenn notwendig, muß eine Anpassung des Reglers vorgenommen werden können.

[2] Die Ableitung der Formeln für die Anpassung der Gewichtungen ist im Anhang beschrieben.

## 8.4 Neuronale Netze mit Reinforcement

Zur Lösung des Problems bieten sich Neuronale Netze an, die mit Hilfe des *Reinforcement-Verfahrens* trainiert werden. Grundlage dieser Methoden ist, wie in vielen Bereichen der Künstlichen Intelligenz, die Interaktion mit dem Zielsystem. Das Ablauf dieser Verfahren kann mit einem Zitat von Barto [6] beschrieben werden.

> The basic concept behind Reinforcement Learning techniques is that if an action is followed by a satisfactory response, then the tendency to produce that action is strengthened, i. e. *reinforced*.

Die Abbildung 8.3 zeigt den schematischen Aufbau eines solchen Systems. Das Aktionsnetzwerk, ein künstliches neuronales Netz, leitet aus den aktuellen Eingaben die Stellgröße $y(t)$ ab. Innerhalb des physikalischen Systems erfolgt dann, in Abhängigkeit von dieser Steuergröße, ein interner Zustandswechsel. Der neue Folgezustand kann an Hand der internen

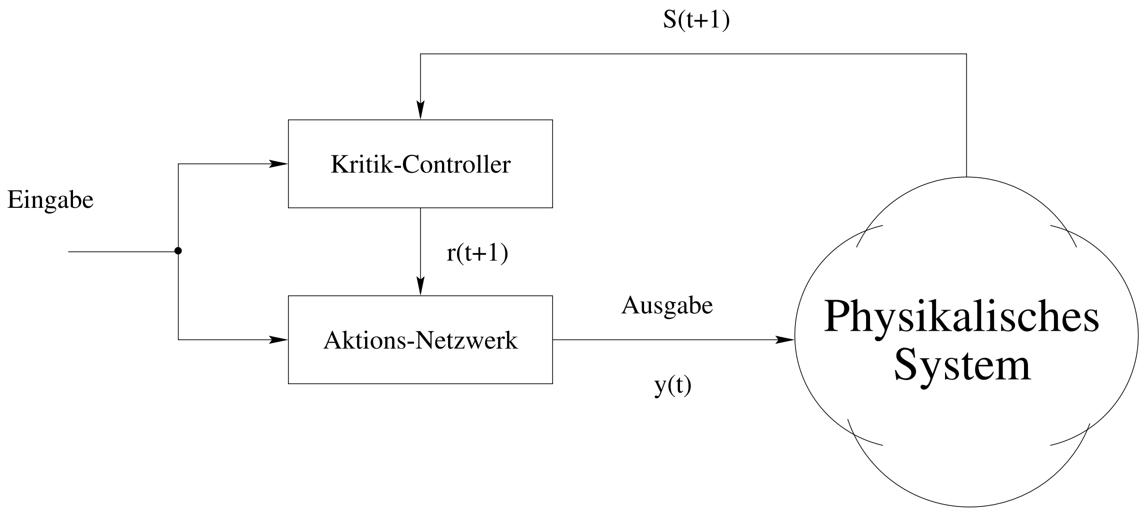



Abbildung 8.3: Struktur des Reinforcement-Verfahrens

Statusgrößen $S(t+1)$ erfaßt werden. Aus diesen Parametern wird in dem Kritik-Controller das sog. *Reinforcement-Signal* $r(t+1)$ ermittelt. $r(t+1)$ ist eine skalare Größe, die die letzte Entscheidung[3] des Aktionsnetzwerkes evaluiert. Es erfolgt also nur eine Aussage darüber, ob die letzte Aktion gut oder schlecht war.
Der Wertebereich und die Auflösung dieses Signal ist abhängig von der Applikation. Im einfachsten Fall setzt er sich bei binären Systemen aus zwei Werten $r(t) \in \{0, 1\}$ zusammen. Falls $r(t) = 0$, bedeutet dies, daß $y(t)$ eine schlechte Entscheidung war, $r(t) = 1$ hingegen, verweist auf eine gute Aktion. Natürlich ist es aber auch möglich, mit einer feineren

[3]Die Laufvariable $(t + 1)$ verdeutlicht, daß das externe Kritiksignal zeitlich zu der korrespondierenden Stellgröße $y(t)$ verschoben ist. $r(t)$ hängt also von Eingaben ab, die aus *einem* früheren Zeitschritt resultieren.

Unterteilung zu arbeiten. In einem weiteren Schritt ist es vorstellbar, daß man mit einer begrenzten Anzahl von Werten unterschiedliche diskrete Einstufungen des Erfolges oder Mißerfolgs zuläßt ($r(t) \in \{0, 0.25, 0.5, 0.75, 1\}$). Um eine noch differenziertere Bewertung der Aktion $y(t)$ zuzulassen, kann $r(t)$ auch durch eine reelle Zahl ($r(t) \in [0,1]$) dargestellt werden, so daß eine kontinuierliche Einstufung der Qualität der Entscheidung $y(t)$ möglich ist.
Dieses Signal macht jedoch weder Aussagen über den Betrag der Fehlanpassung des Ausgangssignals $y(t)$ noch über die Richtung, in der die Adaption der Gewichte des Aktionsnetzes erfolgen muß, um das Ausgabeverhalten zu optimieren bzw. das Kritiksignal zu maximieren.
Bei den in Abschnitt 8.2.1 dargestellten überwachten Lernverfahren wurden diese Informationen leicht, durch einen Vergleich zwischen Soll- und Ist-Werten, ermittelt. Da $r(t)$ aber nur ein Skalar aus dem Intervall $[0, 1]$ ist, müssen die für die Maximierung des Kritiksignals notwendigen Größen indirekt ermittelt werden.
Williams hat 1992 ein Verfahren veröffentlicht [77], mit dem es möglich ist, die für eine Adaption des Aktions-Netzes notwendigen Parameter näherungsweise zu bestimmen. Das als *stochastically hillclimbing* bekannte Verfahren basiert auf dem Vergleich zwischen einem *prognostizierten* Kritiksignal $p(t+1)$ und der durch die Kritikeinheit bestimmten *tatsächlichen* Beurteilung der letzten Aktion $r(t+1)$. Bei diesem Verfahren handelt es sich um einen Online-Algorithmus, bei dem durch direkte Interaktion mit dem Zielsystem der gesamte Zustandsraum nach einer möglichen Lösung untersucht werden kann. Zur Erläuterung dieses Verfahrens wird die Abbildung 8.4 herangezogen.

### 8.4.1 Das Aktionsnetzwerk

Das Aktionsnetzwerk wird durch ein neuronales Netzwerk mit vier Ebenen repräsentiert. Im Wesentlichen erfolgt hier die Umsetzung der Eingaben, unter Berücksichtigung der bereits adaptierten Verhaltensweise, in Aktionen oder Entscheidungen ($\hat{y}(t)$). Die Ausgangsgröße $y(t)$ dient dann als Basis für die Ableitung - unter Berücksichtigung der Kritiksignale $r$ und $p$ - einer angepaßten Aktion.
Der Eingabevektor setzt sich aus den aktuellen und früheren Daten, die der Benutzer zur Verfügung stellt bzw. gestellt hat sowie internen Größen des zu steuernden Systems zusammen.

### 8.4.2 Die Kritikeinheit

In dieser Einheit werden gemäß des von Williams beschriebenen Verfahrens zwei Signale generiert. $r(t+1)$ ist die aus den Statusinformationen des Prozesses abgeleitete Kennzahl, die die letzte Stellgröße $y(t)$ evaluiert. Darüber hinaus wird zur Bestimmung eines Fehlersignals versucht dieses Kritiksignal im Vorfeld durch eine Abschätzung $p(t+1)$ anzunähern. Die Prognose basiert auf den aktuellen Eingabewerten $X_i(t)$ sowie einigen Systemparametern $S_i(t)$.

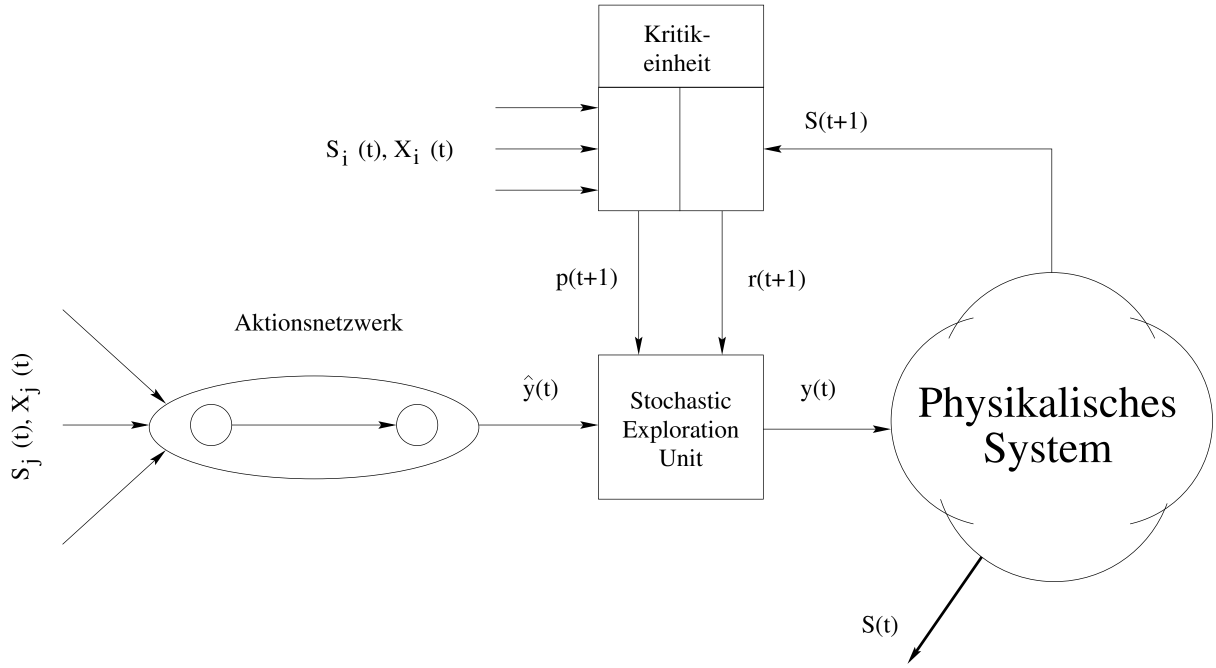



Abbildung 8.4: Reinforcement Artificial Neural Network

### 8.4.3 Stochastic Exploration Unit

Wie bei den konventionellen Lernverfahren müssen auch bei dem Reinforcement-Verfahren die Parameter des Netzes adaptiert werden. Da der Fehler nicht durch den Vergleich von Soll- und Ist-Werten bestimmt werden kann, erfolgt hier eine Änderung der Verbindungsgewichte in Abhängigkeit von dem Verlauf der Kritikfunktion. Ausgangspunkt des Lernverfahrens bildet die Beziehung 8.12.

$$\Delta w_{ij} \sim \frac{\partial r}{\partial w_{ij}} \tag{8.12}$$

Für die exakte Berechnung des Gradienten wird allerdings der genaue Zusammenhang zwischen dem Kritiksignals $r(t)$ von der Ausgangsgröße $y(t)$ des Aktionsnetzwerkes in Form einer Funktion benötigt. Mit Hilfe der Kettenregel ergibt sich der Zusammenhang 8.13.

$$\frac{\partial r}{\partial w_{ij}} = \frac{\partial r}{\partial y}\frac{\partial y}{\partial w_{ij}} \tag{8.13}$$

In den meisten Fällen ist diese Abhängigkeit nicht bekannt, so daß der Gradient $\frac{\partial r}{\partial y}$ nur abgeschätzt werden kann. Wie oben ausgeführt, gibt $r(t)$ nur Auskunft über die Qualität der letzten Entscheidung, die Richtung und der Betrag der Korrektur zur Adaption der Verbindungsgewichte kann nicht direkt aus diesem Signal abgeleitet werden. Eine Möglichkeit, um die Informationen indirekt abzuleiten, bietet die sog. *Stochastic Exploratory Method.*

Bei diesem Verfahren geht man davon aus, daß die optimale Stellgröße in dem vorliegenden Systemzustand und bei den gegebenen Eingangswerten mit großer Wahrscheinlichkeit in einem unscharfen Bereich um $\hat{y}(t)$ liegt. Mittelpunkt dieses Bereichs ist der durch das Aktionsnetzwerk bestimmte Ausgangswert $\hat{y}(t)$. Die tatsächliche Aktion $y(t)$ wird dann mit Hilfe von stochastischen Methoden aus dieser Umgebung ausgewählt.
In Abhängigkeit von der Größe des Intervalls kann der gesamte Aktionsraum erforscht werden. Weiterhin macht die Ausdehnung des Aktionsintervalls eine Aussage über die Qualität des durch das Aktionsnetzwerk bestimmten Ausgangswerts $\hat{y}$. Ein kleines Aktionsintervall impliziert bei diesem Verfahren, daß $\hat{y}$ der geeigneten Lösung sehr nahe kommt, so daß nur in einem sehr beschränkten Bereich nach einer optimaleren Lösung gesucht wird. Stellt $\hat{y}(t)$ dagegen nur eine suboptimale Reaktion auf den Eingangsvektor dar, wird das Aktionsintervall vergrößert. Im nächsten Schritt besteht dann die Möglichkeit eine Aktion aus einem umfassenderen Lösungsraum auszuwählen. Diese Lösung kann sich dann essentiell von $\hat{y}(t)$ unterscheiden und bietet dann so die Möglichkeit, einen erweiterten Lösungsbereich abzuschreiten.
Ein weiterer Vorteil des Verfahrens ist darin begründet, daß die Aktion $y(t)$ mit Hilfe von Zufallszahlen ausgewählt wird. Durch den Einsatz dieser Methode wird gewährleistet, daß das System nicht in einem lokalen Optimum verharrt.

Bei dem hier gewählten Ansatz wird die Größe des unscharfen Intervalls $I(t)$, in dem nach einem optimaleren Ausgangssignal $y(t)$ als dem durch das Aktionsnetzwerk vorgegebenen Wert $\hat{y}(t)$ gesucht wird, mit Hilfe der Gleichung 8.14 ermittelt.

$$I(t) = (\underbrace{r(t+1) - p(t+1)}_{D_K})^2,\ 0 \leq r(t+1) \leq 1,\ 0 \leq p(t+1) \leq 1 \tag{8.14}$$

Die Differenz zwischen dem Kritiksignal $r(t+1)$ und dem prognostizierten Signal $p(t+1)$ wird quadratisch berücksichtigt, um keine Fallunterscheidung in Abhängigkeit von dem Betrag der Kennzahlen vornehmen zu müssen. Ist $D_K$ klein, heißt das, daß die erwartete mit der tatsächlichen Reaktion nahezu deckungsgleich ist. Die Suche nach einem besseren Ausgabewert beschränkt sich dann auf ein relativ kleines Intervall um $\hat{y}(t)$. In diesem Fall wird angenommen, daß $\hat{y}(t)$ mit einer großen Wahrscheinlichkeit die beste Aktion für den anliegenden Eingabevektor darstellt.
Im Gegensatz dazu wird der Lösungsraum erweitert, wenn die Differenz $D_K$ größer wird. Da sich in diesem Fall $r(t+1)$ und $p(t+1)$ erheblich unterscheiden, bedeutet dies, daß $\hat{y}(t)$ eine suboptimale Reaktion auf den Eingangsvektor darstellt. Ein erweitertes Intervall eröffnet dann die Möglichkeit eine Aktion, die sich grundlegend von $\hat{y}(t)$ signifikant unterscheiden kann, zu selektieren.
Wenn das Lösungsintervall dann bestimmt ist, muß eine Aktion $y(t)$ zufällig aus diesem Bereich ausgewählt werden. Zu diesem Zweck werden gleichverteilte Zufallszahlen $X_i$ mit Hilfe einer dreieckförmigen Verteilungsfunktion (Abb. 8.5) auf das Intervall $[\hat{y} - I_L, \hat{y} + I_R]$ abgebildet. Diese Transformation erfolgt durch Bildung der inversen Funktion unter Berücksichtigung der Randbedingung, daß Zufallszahlen $X_i \leq 0.5$ auf den linken, Zahlen

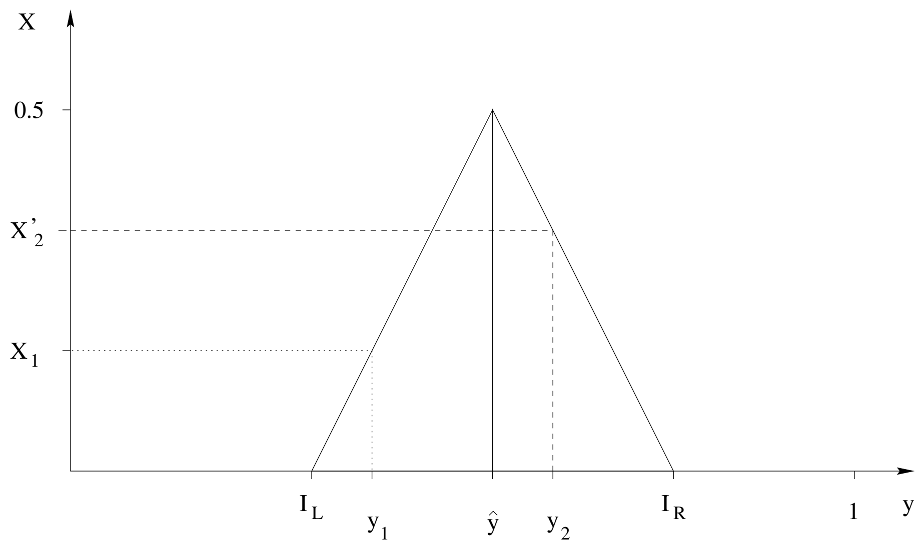



Abbildung 8.5: Trapezoide Transformation der gleichverteilten Zufallszahlen $X_i$

für die gilt $X_i > 0.5$ auf den rechten Bereich der triangulären Ausgangsfunktion umgesetzt werden (Abb. 8.5). Die Transformation erfolgt dann mit Hilfe der Gleichung 8.15.

$$y^* = \begin{cases} L\left(\frac{0.5}{I_L} \cdot (X - \hat{y} + I_L)\right) & : \; X \in [0, 0.5] \\ R\left(-\frac{0.5}{I_R} \cdot (X^T - \hat{y} - I_R)\right) & : \; X \in \,]0.5, 1] \quad und \quad X^T = X - 0.5 \end{cases} \tag{8.15}$$

X ist der Platzhalter für die gleichverteilte Zufallszahl, $\hat{y}$ ist der von dem Aktionsnetzwerk ermittelte Ausgangswert, $I_L$ und $I_R$ sind die Randwerte des Suchintervalls I. Für diese Randwerte $I_L$ und $I_R$ des Lösungsraumes gelten außerdem folgende Beziehungen:

$$\begin{aligned} I_L &= \begin{cases} 0 & : \; \hat{y} - \frac{I}{2} < 0 \\ \hat{y} - \frac{I}{2} & : \; sonst \end{cases} \\ I_R &= \begin{cases} 1 & : \; \hat{y} + \frac{I}{2} > 1 \\ \hat{y} + \frac{I}{2} & : \; sonst \end{cases} \end{aligned} \tag{8.16}$$

Diese Einschränkungen werden notwendig, damit nur Aktionen aus dem Intervall $[0, 1]$ Berücksichtigung finden.

### Bestimmung des Gradienten

Mit den in den vorhergehenden Abschnitten beschriebenen Einheiten und Verfahren kann der Gradient $\frac{\partial r}{\partial y}$ wie folgt abgeschätzt werden:

$$\frac{\partial r}{\partial y} \approx \underbrace{[r(t+1) - p(t+1)]}_{I} \underbrace{\left[\frac{y(t) - \hat{y}(t)}{I_R - I_L}\right]}_{II} \tag{8.17}$$

Die zeitliche Verschiebung zwischen den einzelnen Parametern wird durch die unabhängigen Variablen $(t+1)$ und $t$ verdeutlicht. Das unterstreicht nochmals die Tatsache, daß das die Reinforcementsignale zum Zeitpunkt $(t+1)$ auf Aktionen beruhen, die zu einem früheren Zeitpunkt $t$ eingeleitet wurden.
$r(t+1)$ evaluiert die Aktion $y(t)$, während $p(t+1)$ die Aktion $\hat{y}(t)$ qualifiziert. Term II der Gleichung 8.17 ist die normalisierte Differenz zwischen der tatsächlichen und der erwarteten, d. h. durch das Aktionsnetzwerk bestimmten, Aktion.
Um zu prüfen, ob die die Verarbeitungsvorschrift konsistent ist, werden folgende Fälle eingehender untersucht.

**Fallunterscheidung:** $r(t+1) > p(t+1)$

Da das tatsächliche aus den Systemparametern abgeleitete Kritiksignal größer als die prognostizierte Kennzahl ist, stellt $y(t)$ eine optimalere Lösung als die durch das Aktionsnetzwerk ermittelte Aktion $\hat{y}(t)$ dar. Für diesen Fall muß das Verhaltens des Aktionsnetzwerkes so adaptiert werden, daß $\hat{y}(t)$ in Richtung von $y(t)$ verschoben wird.

**Fallunterscheidung:** $r(t+1) < p(t+1)$

Stellt die Aktion $y(t)$ auf der anderen Seite eine schlechtere Lösung als $\hat{y}(t)$ dar, muß eine Anpassung der Gewichte des Aktionsnetzes in der Art erfolgen, daß $y(t)$ weiter von $\hat{y}(t)$ weg verschoben wird.

Mit Hilfe dieses Verfahrens kann nun der Gradient $\frac{\partial r}{\partial y}$ näherungsweise bestimmt werden. Das ursprüngliche Reinforcement-Verfahren reduziert sich mit diesem Ansatz auf ein überwachtes Lernverfahren. Die Adaption der Gewichtungen können somit nach dem in Abschnitt 8.3 beschriebenen Verfahren erfolgen.

### Anpassung der Gewichte

Für jeden Eingabevektor kann wie bisher durch die Verarbeitung durch das neuronale Netz ein Ausgangswert abgeleitet werden. Mit Hilfe der Methode der sog. Stochastic Exploration ist es dann allerdings auch möglich den Gradienten $\frac{\partial r}{\partial y}$ zu bestimmen, so daß eine Anpassung der Gewichte des Aktionsnetzwerkes vorgenommen werden kann. Allgemein gilt:

$$\frac{\partial r}{\partial w_{ij}} = \frac{\partial r}{\partial a} \cdot \frac{\partial a}{\partial net} \cdot \frac{\partial net}{\partial w_{ij}} \tag{8.18}$$

Auf Grund der Identität am Ausgang des Aktionsnetzes gilt für :

$$\frac{\partial r}{\partial a} = \frac{\partial r}{\partial y} = [r(t+1) - p(t+1)] \left[\frac{y(t) - \hat{y}(t)}{I_R - I_L}\right] \tag{8.19}$$

Unter Berücksichtigung der sigmoiden Transferfunktion gilt dann folgender Zusammenhang

$$\frac{\partial a}{\partial net} = o_j \cdot (1 - o_j) \tag{8.20}$$

$$\frac{\partial net}{\partial w_{ij}} = o_j \tag{8.21}$$

Mit diesen Beziehungen lautet der Zusammenhang zur Berechnung des Gradienten

$$\frac{\partial r}{\partial w_{ij}} = [r(t+1) - p(t+1)] \left[\frac{y(t) - \hat{y}(t)}{I_R - I_L}\right] \cdot o_j \cdot (1 - o_j) \cdot o_j \tag{8.22}$$

Die Gewichtsänderung ergibt sich somit zu:

$$\Delta w_{ij} = \eta \cdot [r(t+1) - p(t+1)] \left[\frac{y(t) - \hat{y}(t)}{I_R - I_L}\right] \cdot o_j \cdot (1 - o_j) \cdot o_j \tag{8.23}$$

## 8.5 Simulation

Das dargestellte Verfahren zur Realisierung eines adaptiven Controllers soll zur Optimierung der Policing Verfahren und Zugangsalgorithmen eingesetzt werden. Die Freiheitsgrade, die dieses Verfahren bietet sind sehr umfangreich. Die Untersuchungen sind infolgedessen sehr komplex und überaus zeitaufwendig. Die Arbeit mit diesem Verfahren ist deshalb noch nicht abgeschlossen. Eine vollständige Analyse ist vorgesehen, erfolgt aber zu einem späteren Zeitpunkt.
Ein vorläufiges Ergebnis, das den Einsatz eines neuronalen Netzes, das mit Hilfe des beschriebenen Reinforcement Verfahrens adaptiert wurde, ist in Tabelle 8.1 wiedergegeben.

| | Adaptionszyklus | | | |
|---|---|---|---|---|
| | 1 | | 2 | |
| Dienst | Auslastung | Verlustrate | Auslastung | Verlustrate |
| 1 | 0.227 | 0.025 | 0.036 | 0.007 |
| 2 | 0.028 | 0.004 | 0.017 | 0.002 |
| 3 | 0.006 | 0.002 | 0.003 | 0.000 |
| 4 | 0.121 | 0.028 | 0.112 | 0.024 |
| 5 | 0.172 | 0.086 | 0.350 | 0.140 |

| Kanal | 0.894 | — | 0.894 | — |
|---|---|---|---|---|

| | Adaptionszyklus | | | |
|---|---|---|---|---|
| | 3 | | 20 | |
| Dienst | Auslastung | Verlustrate | Auslastung | Verlustrate |
| 1 | 0.003 | 0.000 | 0.003 | 0.000 |
| 2 | 0.006 | 0.000 | 0.006 | 0.000 |
| 3 | 0.002 | 0.000 | 0.002 | 0.000 |
| 4 | 0.127 | 0.025 | 0.127 | 0.025 |
| 5 | 0.369 | 0.162 | 0.369 | 0.163 |
| Kanal | 0.894 | — | 0.894 | — |

| | Adaptionszyklus | | | |
|---|---|---|---|---|
| | 50 | | 100 | |
| Dienst | Auslastung | Verlustrate | Auslastung | Verlustrate |
| 1 | 0.003 | 0.000 | 0.003 | 0.000 |
| 2 | 0.006 | 0.000 | 0.006 | 0.000 |
| 3 | 0.002 | 0.000 | 0.002 | 0.000 |
| 4 | 0.127 | 0.025 | 0.127 | 0.025 |
| 5 | 0.370 | 0.163 | 0.369 | 0.163 |
| Kanal | 0.894 | — | 0.894 | — |

Tabelle 8.1: Entwicklung der Auslastung der Warteschlangen und der Verlustraten der einzelnen Dienste in Abhängigkeit von dem Adaptionszyklus

Die Ergebnisse dargestellten Mittelwerte, die sich bei der Simulation eingestellt haben, zeigen, daß schon nach *drei* Adaptionszyklen[4], d. h. nachdem dreimal dasselbe Verkehrsmuster eingeprägt worden ist, die Verlustraten für die Dienste 1, 2 und 3 nur noch 0% betrugen. Während des weiteren Simulationsverlaufes waren dann nur noch marginale Änderungen bei den Diensten 4 und 5 zu beobachten. Die Auslastung der Warteschlangen betrug nach drei Zyklen für Dienst 1 0.3%, für Dienst 2 0.6% und für Dienst 3 2%. Die Auslastung der Warteplätze bei den Dienste 4 und 5 lag bei 12.7% und 36.9%.
Die Ergebnisse zeigen, daß schon nach einer kurzen Anpassungsphase (3 Zyklen !), Kennwerte vorliegen, die demonstrieren, daß der Regler scheinbar in der Lage ist, die gewünschte Regelstrategie effizient, innerhalb kürzester Zeit umzusetzen.

[4]Unter einem Adaptionszyklus ist *eine* Simulation mit einer fest vorgegebenen Verkehrslast zu verstehen.

# Kapitel 9

# Vergleich der Controller

In den vorangegangenen Kapiteln wurden unterschiedliche Policing und Call Admission Controller untersucht. Ausgangspunkte waren stets deterministische Verfahren, um die Leistungsfähigkeit der auf Computational Intelligence basierten Verfahren einordnen zu können. Im Folgenden sollen die Qualität der unterschiedlichen Systeme verglichen werden.

## 9.1 Policing Controller

In den Tabellen 9.1 und 9.2 sind die Kennwerte dargestellt, die sich bei der Simulation von vier unterschiedlichen Policing Controllern ergaben. In der ersten Spalte sind die

| Dienst | Auslastung der Warteschlangen | | | |
|---|---|---|---|---|
| | GCRA | Fuzzy | GA-Fuzzy | RFNN |
| 1 | 0.004 | 0.024 | 0.135 | 0.003 |
| 2 | 0.002 | 0.120 | 0.200 | 0.006 |
| 3 | 0.001 | 0.084 | 0.083 | 0.002 |
| 4 | 0.002 | 0.158 | 0.103 | 0.127 |
| 5 | 0.001 | 0.186 | 0.261 | 0.369 |
| | Durchschnittliche Auslastung | | | |
| Mittelwert | 0.2% | 11.4% | 15.6% | 10.1% |

Tabelle 9.1: Gegenüberstellung der Auslastung der Systemressourcen bei Einsatz unterschiedlicher Policing Controller

Ergebnisse, die sich unter Verwendung des GRCA[1] einstellten, aufgeführt. Bei den anderen Verfahren handelt es sich um den „manuell" entworfenen Fuzzy Controller[2] (*Fuzzy*). Der mit Hilfe der genetischen Algorithmen abgeleitete Controller wird in den Tabellen mit *GA-Fuzzy* bezeichnet. Es handelt sich dabei um den in Abschnitt 6.2.8 auf Seite 116

[1]Die Untersuchung des Generic Cell Rate Algorithm erfolgte in Abschnitt 4.3 auf den Seiten 46ff

[2]Die Beschreibung des Controllers erfolgt in Kapitel 5.3 auf den Seiten 67ff.

beschriebenen Fuzzy Controller, der sich unter Verwendung einer Fuzzy Logic basierten Fitnessfunktion nach 63 Generationen ergab. In der Spalte *RFNN* sind die Ergebnisse dargestellt, die sich bei der Anwendung von Neuronalen Netzen entwickelten. Die Adaption dieser Netze erfolgte auf der Basis des in Abschnitt 8.4 auf den Seiten 162ff. beschriebenen Reinforcement Verfahrens.

Der GRCA weist mit durchschnittlich 0.25% die geringste Auslastung der Warteplätze

| | Verlustrate | | | |
|---|---|---|---|---|
| Dienst | GCRA | Fuzzy | GA-Fuzzy | RFNN |
| 1 | 0.00 | 0.000 | 0.000 | 0.000 |
| 2 | 0.25 | 0.000 | 0.039 | 0.000 |
| 3 | 0.00 | 0.000 | 0.000 | 0.000 |
| 4 | 0.25 | 0.025 | 0.000 | 0.025 |
| 5 | 0.25 | 0.162 | 0.109 | 0.163 |
| Mittelwert | 15% | 3.74% | 2.96% | 3.76% |

Tabelle 9.2: Gegenüberstellung der Verlustraten bei Einsatz unterschiedlicher Policing Controller

auf. Die mittlere Verlustrate beträgt 15%. Die Verluste beschränken sich allerdings nur auf die Dienste, die nicht vertragskonform betrieben werden. Die Verlustrate beträgt in diesem Fall 25%. Bei Einhaltung der Vereinbarungen ist die Verlustrate 0%. Die Auslastung der Kanalbandreite beträgt auf Grund der Reservierung der maximalen Bandbreite nach Tabelle 4.3 auf Seite 51 nur 79%.

Bei den auf Fuzzy Logic basierten Controllern (*Fuzzy* und *GA-Fuzzy*) konnte festgestellt werden, daß die Kanalbandbreite optimal genutzt wurde. Die durchschnittliche Verlustrate konnte gegenüber dem deterministischen Ansatz entschieden gesenkt werden. Sie weist für den manuell entwickelten Fuzzy Controller nur 3.74% und für den mit Hilfe von genetischen Algorithmen entwickelten Controller nur noch 2.96% auf. Auch bei diesen Ansätzen werden die vertragskonformen Dienste in ihrer Übertragungsqualität nicht beeinträchtigt. Die doch erheblich verbesserten Kennwerte resultieren aus einer verstärkten Nutzung der Warteplätze. Die durchschnittliche Auslastung der Speicherkapazität lag bei $\approx 11.4\%$ und $\approx 15.6\%$. Darüber hinaus konnten bei diesen Controllern die Kenngrößen über die Dienstpriorität beeinflußt werden.

Die Auslastung der Warteplätze lag bei $\approx 10\%$. Auch hier wurden die vertragskonformen Dienste in Ihrer Übertragungsqualität nicht beeinträchtigt.

Aus diesen Ergebnissen kann abgeleitet werden, daß die Dienste ihre ausgehandelten Vertragsparameter verletzen können, ohne daß die Übertragungsqualität der vertragskonformen Verbindungen beeinträchtigt wird. Dieser Sachverhalt kann aber auch so interpretiert werden, daß durch den Einsatz von Fuzzy Logic oder neuronaler Netze, die effektive Bandbreite, die für die Beschreibung der Lastverhaltens bei Abschluß des Verkehrsvertrages notwendig ist, reduziert werden kann. Als Folge davon können dann weitere Verbindungen

aufgebaut werden.
Diese Ergebnisse zeigen noch einmal nachdrücklich, daß durch den Einsatz von biologisch nahen Optimierungsverfahren erhebliche Verbesserungen bei der Überwachung von Datenströmen in Zugangsknoten gegenüber den konventionellen, deterministischen Methoden erzielt werden können.
Abschließend kann noch festgestellt werden, daß der automatische Entwurf von Controllern mit Hilfe der genetischen Algorithmen im Allgemeinen qualitativ hochwertige Fuzzy Systeme lieferte, mit denen es möglich war, die vorgegebene Regelstrategie zu realisieren.

## 9.2 Der Call Admission Controller

Die Beurteilung der Fuzzy Logic basierten Call Admission Controller erfolgt durch einen Vergleich mit den beiden in den Abschnitten 2.2.4 und 2.2.4 präsentierten konträren Strategien. Das Peak Reservation Verfahren zeichnete sich durch eine geringe Auslastung der Warteplätze und einer hohen und in weiten Teilen dienstabhängigen Call Loss Rate aus. Die Paketverlustrate betrug, da für jede Verbindung soviele Ressourcen reserviert wurden, um jederzeit die maximalen Anforderungen der Dienste zu erfüllen, stets 0%. Dieser Ansatz führte aber auf der anderen Seite dazu, daß die Auslastung der Linkbandbreite mit 79% minimal war.
Die Reservierung der Ressourcen auf der Basis der minimalen Bandbreite wies im Gegensatz zu dem Peak Reservation Verfahren einen flachen Verlauf der CLR auf. Weiterhin zeigte sich deutich, daß die Dienste, in Abhängigkeit von ihren minimalen Bandbreiten, unterschiedlich behandelt wurden. Darüber hinaus wurden die Ressourcen stets in Überlast betrieben, so daß die Paketverlustrate schon bei einem Angebot von 50 Erlang größer 68% war.
Bei den untersuchten Call Admission Controllern wurden viele gute Lösungen entwickelt. Für einen Vergleich soll exemplarisch der in Abschnitt 7.5.2 beschriebene Controller herangezogen werden. Es zeigt sich, daß die Werte der Call Loss Rate des Fuzzy Logic basierten Controllers in dem gesamten Bereich kleiner waren als die des Peak Reservation Verfahrens, aber erwartungsgemäß größer als die mit Hilfe des MR-Verfahrens ermittelten Werte. Die Behandlung der Dienste ist im Gegensatz zu dem MR-Verfahren aber nahezu einheitlich. Die Streuung der Werte betrug hier nur 6.3%.
Bei der Auslastung der Warteplätze ergaben sich große Unterschiede. Während beim PR-Verfahren die Belegung stets kleiner 0.1% war, wurden bei der MR-Methode die Ressourcen in Überlast betrieben. Bei dem Fuzzy Logic basierten CAC Verfahren war die Auslastung der Warteplätze überdurchschnittlich. Die Belegung der Warteplätze belief sich bei den Diensten 1, 2, 4 und 5 für ein Angebot größer 50 Erlang auf Werte zwischen 56% und 78%. Die Auslastung der Warteplätze von Dienst 3 war sogar noch etwas größer. Die Übertragungsbandbreite wurde bis auf eine geringe Reserve ausgenutzt.
Da bei dem MR-Verfahren das gesamte System überlastet war, traten hohe Paketverlustraten auf. Durch Einsatz des Fuzzy Logic basierten Controllers, konnten diese auf maximal 2% gesenkt werden. Bei einem kleineren Angebot nehmen die Paketverluste dann Werte

an, die konform zu ihren Dienstanforderungen sind.
Neben der Abschätzung der Qualität der unterschiedlichen Controller an Hand von Kenngrößen wie die Verlustraten oder die Auslastung muß noch erwähnt werden, daß neue Zielsetzungen mit Hilfe der Fuzzy Logic einfach und schnell umgesetzt werden konnten. So konnte ein Call Admission Controller entwickelt werden, bei dem die CLR für alle Dienste nahezu identisch war. Die Streuung betrug lediglich 2.64%.
Auch hier zeigte sich, daß durch den Einsatz von Fuzzy Logic Ergebnisse erzielt werden, die eine Verbesserungen gegenüber den konventionellen deterministischen Methoden darstellen.

# Kapitel 10

# Zusammenfassung und Ausblick

## 10.1 Zusammenfassung

Die Konvergenz der unterschiedlichen Übertragungssysteme sowie die vielen neuen multimedialen Applikationen stellen erhebliche, weitgestreute Anforderungen an das Verkehrsmanagement der unterlagerten Kommunikationssysteme. Die bislang eingesetzten konventionellen Verfahren stellen in vielen Fällen auf Grund des sich ständig ändernden Verkehrsprofils eine suboptimale Lösung dar. In dieser Arbeit sollte die Einsetzbarkeit intelligenter Verfahren zur Steuerung des Zugangs zu den Kommunikationssystemen und die Überwachung der Datenströme überprüft werden.
Durch Einsatz einer ereignisorientierten Simulation wurde nachgewiesen, daß ein Fuzzy Controller das notwendige Regelverhalten aufweist, um die in einem Zugangsknoten zur Verfügung stehenden Ressourcen unter Berücksichtigung der spezifischen Kenndaten effizient zu verwalten. Aufbauend auf den Ergebnissen sollte das Regelverhalten durch die Anwendung von genetischen Algorithmen verbessert werden. Zu diesem Zweck wurden die internen Parameter der Fuzzy Controller - also die Regelbasen und die Zugehörigkeitsfunktionen - in geeigneter Form codiert und zu Bitstrings konstanter Länge zusammengefaßt. Erste Untersuchungen, die eine separate Anpassung der Regelbasen auf der einen Seite und der Zugehörigkeitsfunktionen auf der anderen Seite zum Gegenstand hatten, zeigten nur marginale Veränderungen des Regelverhaltens des Fuzzy-Controllers. Im Wesentlichen wurden nur lokale Optima gefunden.
Bei weiteren Untersuchung wurden beide Parametermengen simultan durch den genetischen Algorithmus geändert. Zur Bestimmung der Fitness eines Bitstrings fanden zwei verschiedene Verfahren Anwendung. Bei dem konventionellen Ansatz wurden die Abhängigkeiten zwischen den Systemgrößen durch algebraische Gleichungen konstruiert. Der Einfluß der einzelnen Variablen konnte durch die Potenzierung der Parameter oder durch die Bildung von Quotienten angepaßt werden. Die Ergebnisse, die sich nach umfangreichen Simulationen ergaben, brachten Controller mit einem optimierten Verhalten hervor. Die Fitnessfunktionen gestalteten sich allerdings relativ komplex, so daß um diese teilweise rechenintensiven Operationen zu umgehen und um die Zusammenhänge zwischen den Ein-

gangsgrößen klarer zu gestalten, im Folgenden der Einsatz eines unscharfen Reglers zur Bestimmung der Fitness der Bitstrings geprüft. Mit diesen Verfahren konnten Fuzzy Controller erzeugt werden, die ein erheblich verbessertes Verhalten gegenüber den konventionellen deterministischen Kontrollverfahren aufweisen. Aber auch der ursprüngliche unscharfe Regler konnte noch verbessert werden.

Neben der Optimierung der Netzzugangskontrolle und dem Policing Verfahren mit Hilfe der Fuzzy Logic wurde die Einsatzfähigkeit von Reinforcement Systemen in diesem Bereich geprüft. Es zeigte sich, daß auch dieser Ansatz schnell zu einer Verbesserung des Reglerverhaltens führt.
Abschließend kann festgehalten werden, daß der Einsatz intelligenter Verfahren im Bereich des Verkehrsmanagements einen großen Beitrag dazu liefert, die Leistungsfähigkeit von Kommunikationssystemen erheblich zu steigern.

## 10.2 Ausblick

Diese Untersuchung ist der Anfang eines stetig wachsenden Interesses an Methoden aus dem Bereich der Computational Intelligence. Während der Arbeit und der Dokumentation der unterschiedlichen Verfahren und Ergebnisse ergaben sich Fakten und Zusammenhänge, die als Ausgangspunkte für weitere Arbeiten dienen.

- In der vorliegenden Untersuchung wurde das Verhalten der Policing und der Call Admission Controller mit einer statischen Verkehrslast, um die Ergebnisse vergleichen zu können, durchgeführt. In nachfolgenden Untersuchungen muß nun das Verhalten der Controller mit zufälligen Verkehrscharakteristiken eruiert werden.

- Es wurde gezeigt, daß die Ergebnisse, die die separaten Fuzzy Logic basierten Controller lieferten, eine Verbesserung der korrespondierenden konventionellen Methoden darstellten. In einem weiteren Schritt muß der kombinierte Einsatz der Fuzzy Logic basierten Policing und Call Admission Controller untersucht werden.

- Das Verhalten des entworfenen CA-Controllers war nahezu optimal. Zur Berücksichtigung weiterer Parameter und zur Optimierung der Topologie des Fuzzy Controllers sollte der Einsatz automatischer Entwurfsverfahren untersucht werden.

- Es konnte erfolgreich gezeigt werden, daß mit Hilfe des Fuzzy Logic basierten Call Admission Controllers Strategien umgesetzt werden konnten, die nicht nur die technischen Randbedingungen in den Vordergrund stellen. Nachfolgend sollte noch untersucht werden, ob die Umsetzung sogenannter Mehrwertdienste problemlos möglich ist. Bei diesem Ansatz werden die wirtschaftlichen Aspekt in den Vordergrund gerückt. So werden weniger profitable Dienste in einem bestimmten Rahmen abgewiesen, um Ressourcen für Verbindungen, die eine höhere Wirtschaftlichkeit versprechen, vorzuhalten.

- Die Stabilität der entwickelten Controller war nicht Gegenstand der vorliegenden Untersuchung. Da die Kennlinienfelder der Controller aber teilweise ausgeprägte Plateaus aufwiesen und bei den Call Admission Controllern Rückkopplungen auftreten, ist eine Stabilitätsanalyse angebracht.

- Die Untersuchungen des Einsatzes von Reinforcement Verfahren sowohl bei der Überwachung der Datenströme als auch bei der Zugangskontrolle von Netzwerken sind komplex. Die Simulationen sind sehr zeitintensiv. Detaillierte Ergebnisse sind deshalb erst nach einem längeren Zeitraum zu erwarten.

# Anhang A

# Grundlagen

## A.1 Auslastung der reservierten Bandbreite bei Anwendung der PR-Methode

Basis für die Bestimmung der Auslastung der reservierten Bandbreite bei Anwendung der Peak Reservation Methode bildet die Abbildung A.1. Dargestellt sind die Phasen in denen

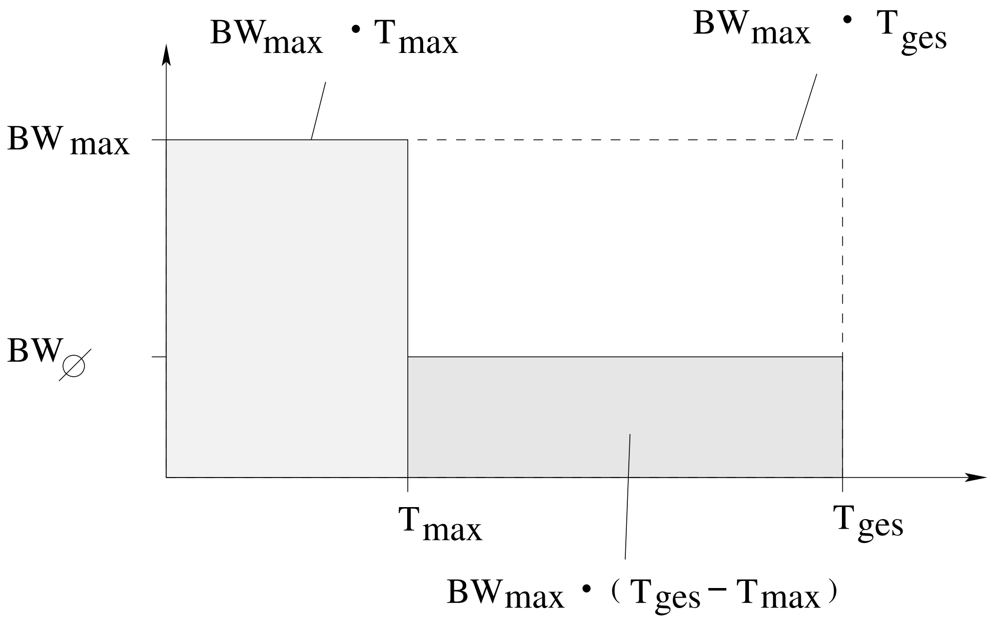



Abbildung A.1: Berechnung der Auslastung

der Dienst mit der maximalen und der durchschnittlichen Bandbreite ($BW_{max}, BW_{\oslash}$), die sich als Mittelwert aus allen Übertragungsraten ohne $BW_{max}$ berechnen läßt, arbeitet. Der Verbindungszeitraum ist $T_{ges}$. Die Nutzung der reservierten Bandbreite kann dann wie folgt bestimmt werden.

$$BW_{Utilization} = \frac{BW_{max} \cdot T_{max} + BW_{\oslash} \cdot (T_{ges} - T_{max})}{BW_{max} \cdot T_{ges}} \tag{A.1}$$

$$= \frac{\frac{BW_{max} \cdot T_{max}}{BW_{\oslash}} + T_{ges} - T_{max}}{\frac{BW_{max} \cdot T_{ges}}{BW_{\oslash}}} \text{ mit Burstiness } B = \frac{BW_{max}}{BW_{\oslash}} \tag{A.2}$$

$$= \frac{B \cdot T_{max} + T_{ges} - T_{max}}{B \cdot T_{ges}} \tag{A.3}$$

$$= \frac{T_{max} \cdot (B-1) + T_{ges}}{B \cdot T_{ges}} \tag{A.4}$$

$$= \frac{\frac{T_{max}}{T_{ges}} \cdot (B-1) + 1}{B} \tag{A.5}$$

Das Verhältnis $\frac{T_{max}}{T_{ges}}$ stellt das sog. *Peak Activity Ratio* dar und beschreibt das Zeitintervall, indem der Dienst mit der maximalen Bandbreite arbeitet, bezogen auf den gesamten Beobachtungszeitraum.

## A.2 Auslastung der reservierten Bandbreite bei Anwendung der MR-Methode

Basis für die Bestimmung der Auslastung der reservierten Bandbreite bei Anwendung der Peak Reservation Methode bildet die Abbildung A.2. Dargestellt sind die Phasen in denen der Dienst mit der minimalen und der durchschnittlichen Bandbreite ($BW_{min}, BW_{\oslash}$), die sich als Mittelwert aus allen Übertragungsraten ohne $BW_{min}$ berechnen läßt, arbeitet. Der Verbindungszeitraum ist $T_{ges}$. Die Nutzung der reservierten Bandbreite kann dann wie folgt erfaßt werden.

$$BW_{Bedarf} = \frac{BW_{min} \cdot T_{min} + BW_{\oslash} \cdot (T_{ges} - T_{min})}{BW_{min} \cdot T_{ges}} \tag{A.6}$$

$$= \frac{T_{min}}{T_{ges}} + \frac{BW_{\oslash} \cdot (T_{ges} - T_{min})}{BW_{min} \cdot T_{ges}} \text{ mit Burstiness } B = \frac{BW_{max}}{BW_{\oslash}} \tag{A.7}$$

$$= \frac{T_{min}}{T_{ges}} + \frac{1}{B} \cdot \frac{BW_{max}}{BW_{min}} \left(1 - \frac{T_{min}}{T_{ges}}\right) \tag{A.8}$$

Das Verhältnis $\frac{T_{max}}{T_{ges}}$ stellt das sog. *Minimal Activity Ratio* dar und beschreibt das Zeitintervall indem der Dienst mit der minimalen Bandbreite arbeitet, bezogen auf den gesamten Beobachtungszeitraum.

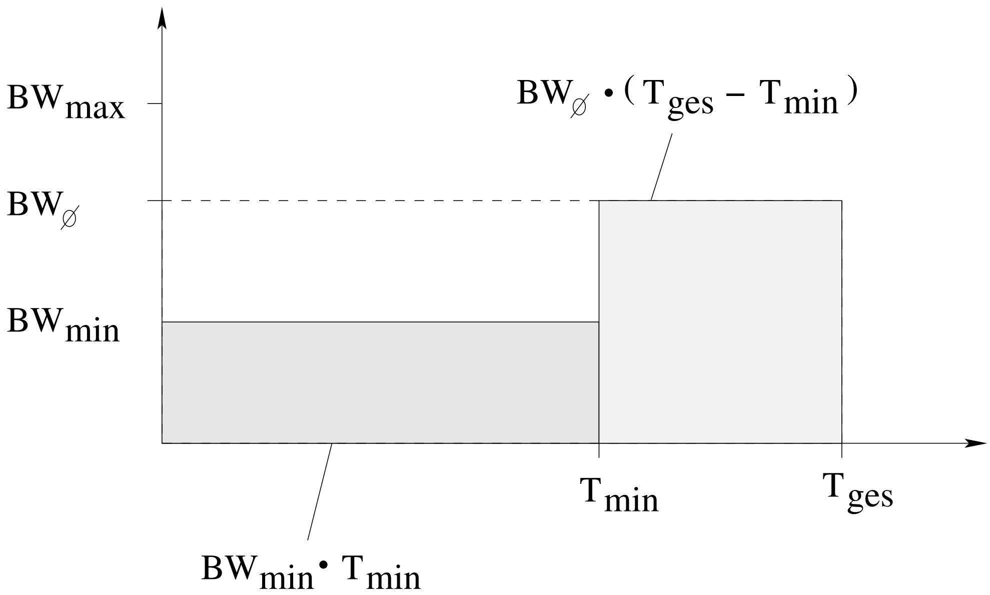



Abbildung A.2: Berechnung der Bandbreitenbedarfs

# Anhang B

# Simulation

## B.1 Der Ereigniskalender

Um die chronologische Abarbeitung der vielen unterschiedlichen Ereignisse zu gewährleisten, werden die Zeitpunkte an denen Zustandsänderungen im System erfolgen sowie die korrespondierenden Aktionen, in einen Ereigniskalender einsortiert. Die Realisierung kann durch eine lineare Liste, bei der der Primärschlüssel die Zeit ist, erfolgen. Ein neues Element wird einsortiert, indem die Liste solange durchlaufen wird, bis ein Eintrag gefunden wird, dessen Ereigniszeit größer ist als die des neuen Elements. Diese Struktur ist einfach zu realisieren. Der Nachteil des Verfahrens ist, daß der Aufwand zum Einsortieren neuer Elemente erheblich von der Anzahl der auftretenden Ereignisse abhängt. Viele Zustandswechsel führen dann zu einer Verlängerung der Simulationszeit.
Bei dem dieser Untersuchung zugrunde liegenden Simulationsprogramm wird diese Liste durch einen sogenannten *Heap* realisiert. Die Verwendung des *Heapsort-Verfahrens* gewährleistet, daß bei N Einträgen im Kalender immer nur höchstens *lg N* Elemente beim Einsortieren durchlaufen werden müssen. Nachteilig kann sich allerdings die Tatsache auswirken, daß bei einem Heap nicht garantiert werden kann, daß beim Einsortieren von zeitgleichen Ereignissen auch tatsächlich das Ereignis zuerst bearbeitet wird, welches zuerst in die Liste eingefügt wurde.

### B.1.1 Datenstruktur des Heaps

Der Heap basiert auf einer *Baumstruktur*, dessen *root* immer das Ereignis mit der kleinsten Zeitangabe enthält. Für jeden weiteren Knoten des Baumes gilt, daß die nächsten Verbindungen zu zwei Knoten führen[1], deren Zeit größer ist als die des übergeordneten Knotens. Diese Knoten werden als Nachfolger des oberen Knotens bezeichnet. Analog dazu wird dieser auch Vorgänger genannt. Das Ordnungsprinzip des hier verwendeten Heaps kann dann durch folgende Regel beschrieben werden:

[1] Binärer Baum

> Die Ereigniszeit eines jeden Knotens soll immer kleiner sein, als die seiner Nachfolger

Da der Aufbau eines hierarchischen Gefüges wie in Abb. B.1 gezeigt, mit Zeigern einen großen Aufwand bedeutet, wird die Baumstruktur auf ein eindimensionales Array abgebildet. Die erste Position des Feldes nimmt die Wurzel ein, ihre Nachfolger die Positionen

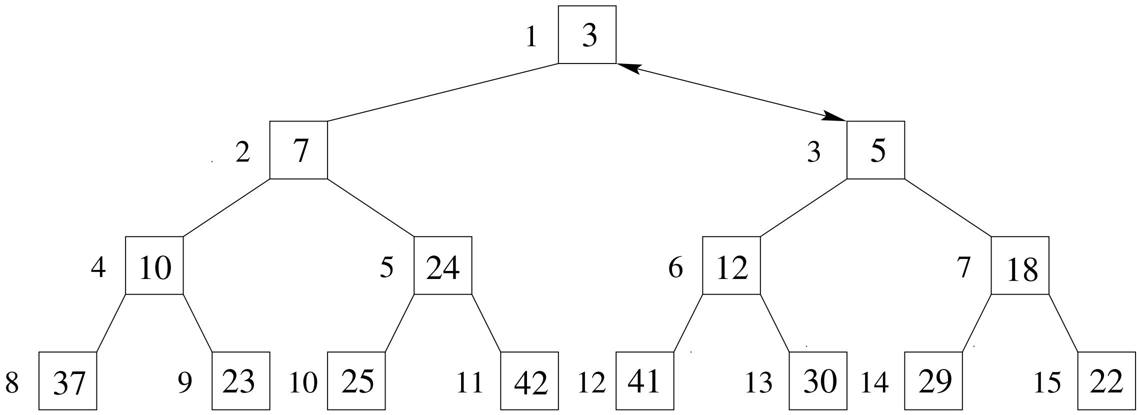



Abbildung B.1: Binäre Baumstruktur

2 und 3. Im Weiteren werden sukzessive alle Knoten des Baumes, eine Ebene nach der anderen von oben nach unten, in das Feld übernommen. Durch diese Transformation ist es dann möglich die Zeiger durch Indizes zu ersetzen. Der Vorgänger eines Knotens $i$ befindet sich im Feld auf der Position $i$ *div* 2 und die beiden Nachfolger auf den Positionen $2i$ und $2i + 1$. Zugriffe erfolgen nicht mehr über Zeiger sondern über Indizes, die aus der Position abgeleitet werden können.

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 7 | 5 | 10 | 24 | 12 | 18 | 37 | 23 | 25 | 42 | 41 | 30 | 29 | 22 | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | Feldindex |
| 1 | 2 | | 3 | | | | 4 | | | | | | | | Ebene |

Abbildung B.2: Projektion der Baumstruktur auf ein lineares Array

### B.1.2 Einfügen eines neuen Elementes

Der prinzipielle Ablauf der Vorgänge beim Einfügen eines neuen Elementes, ist in Abbildung B.3 dargestellt. Soll ein Ereignis in den Heap importiert werden, wird dieses immer an die letzte Stelle des Feldes gesetzt. Wird dadurch die Heap-Bedingung verletzt, muß der Knoten mit seinem Vorgänger ausgetauscht werden. Dieser Vorgang wird solange wiederholt, bis der Heap, entsprechend der Bedingung rekonstruiert worden ist. Diese Operation läuft in der Baumstruktur von untern nach oben entlang eines Pfades.

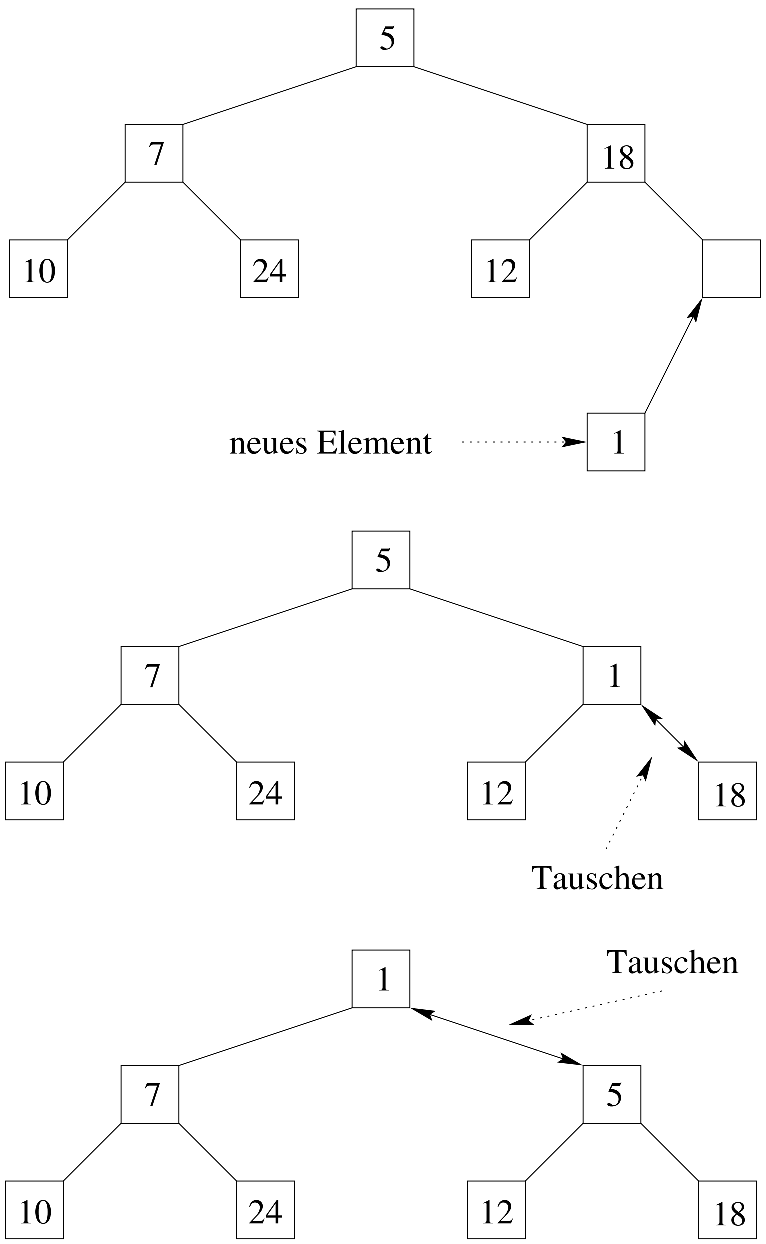



Abbildung B.3: Einfügen eines neuen Elementes in den Kalender

### B.1.3 Herausnehmen eines Elementes

Das Auslesen eines Ereignisses (Abb. B.4 bis B.6) führt dazu, daß die Wurzel des Baumes entfernt wird. An diese Position wird dann das letzte Element des Feldes gesetzt und somit eine Verletzung der Heap-Bedingung herbeigeführt. Durch diese Maßnahme wird dann eine Umstrukturierung des Baumes erzwungen. Das oben eingefügte Element wird mit den Nachfolgern in der unterlagerten Ebene verglichen und mit der Komponente ausgetauscht, die die kleinste Zeitangabe enthält. Das Element wandert solange durch den Baum, bis es an zwei Nachfolger gelangt, deren Zeitangaben größer sind als die eigene oder aber die unterste Ebene des Baums erreicht wurde.

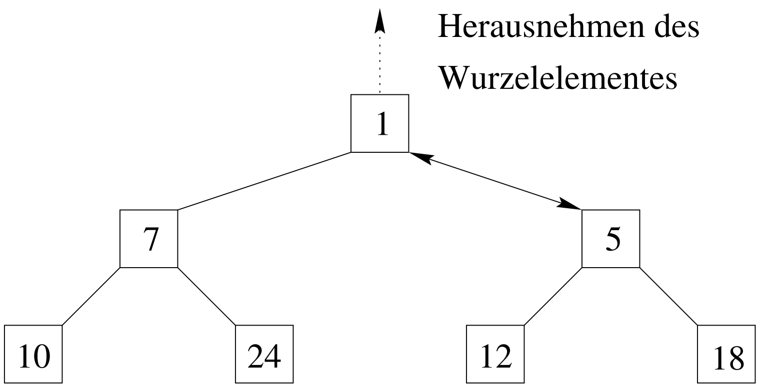



Abbildung B.4: Entfernung der Wurzel des Baumes

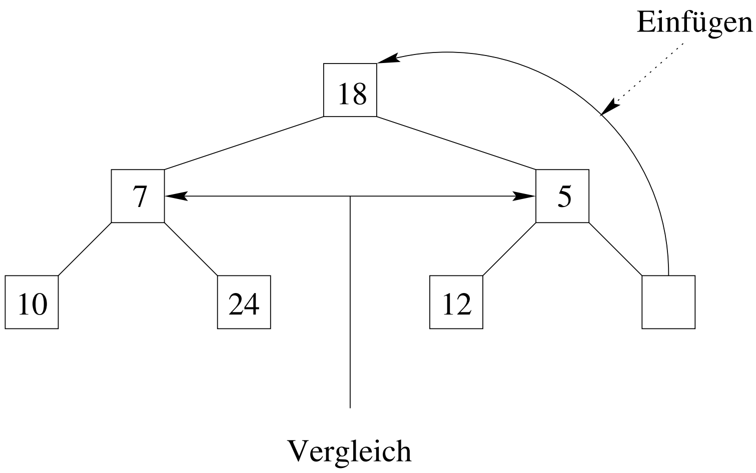



Abbildung B.5: Verletzung der Heap-Bedingung

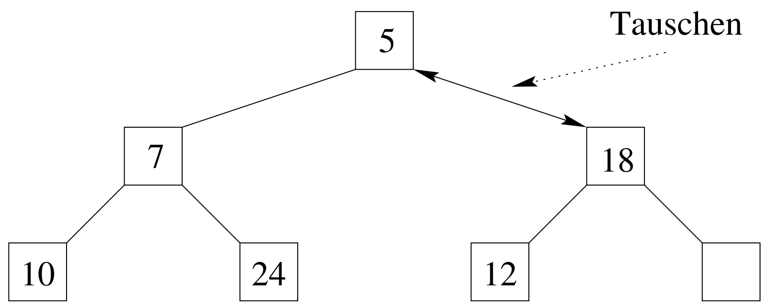



Abbildung B.6: Neue Strukturierung des Baumes

# Anhang C

# Genetischer Algorithmus

## C.1 Anwendungsbeispiel

Gegeben sei eine Population von vier Individuen mit einer jeweiligen Stringlänge von zehn Stellen. Die Ausgangsgeneration wird entweder vorgegeben oder zufällig generiert.

| String Nr. | Bitstring |
|---|---|
| 1 | 1 1 0 0 1 0 0 0 1 1 |
| 2 | 1 0 0 1 1 1 0 0 0 0 |
| 3 | 0 1 1 1 1 0 1 1 0 0 |
| 4 | 0 0 0 0 1 0 1 1 0 0 |

Tabelle C.1: Ausgangsgeneration

Diese Initialgeneration wird im Folgenden der Selektion unterzogen. Dazu werden die einzelnen Individuen mit einer Fitnessfunktion bewertet. In diesem Beispiel wird die Fitnessfunktion durch die Quadratwurzel realisiert.

| String Nr. | Bitstring | Wert | $f_i(x) = \sqrt{x}$ | $p_{select} = \frac{f_i(x)}{\sum_i f_i(x)}$ |
|---|---|---|---|---|
| 1 | 1100100011 | 803 | 28.34 | 0.35 |
| 2 | 1001110000 | 624 | 24.98 | 0.30 |
| 3 | 0111101100 | 492 | 22.18 | 0.27 |
| 4 | 0000101100 | 44 | 6.63 | 0.08 |
| Summe | | | 82.13 | 1.00 |
| Durchschnitt | | | 20.53 | 0.25 |
| Maximum | | | 28.34 | 0.35 |

Tabelle C.2: Selektion

Die Fitness-Werte der einzelnen Strings führen zu folgender Belegung des Roulette-Rades.

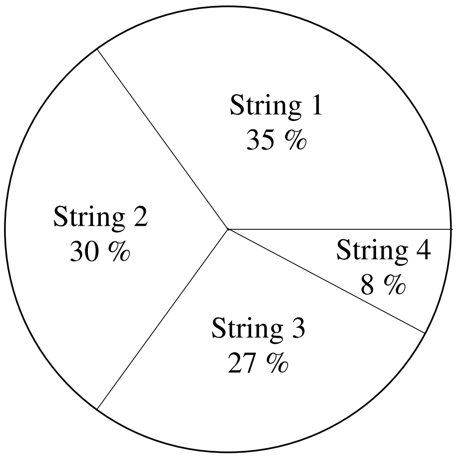



Abbildung C.1: Aufteilung des Roulette-Rades an Hand der prozentualen Anteile der Fitness an der Gesamtfitness

Die Größe der Tortenstücke beeinflußt die Fortpflanzungswahrscheinlichkeit. Basierend auf dieser Verteilung wird die nachkommende Generation gebildet. Dabei kommt in diesem Beispiel heraus, daß der zweite und dritte String sich jeweils einmal fortpflanzen, während der erste String zwei Individuen der nächsten Generation bildet. Die Population gestaltet sich wie folgt. Diese Population wird im Folgenden der Rekombination unterzogen. In

| String Nr. | Bitstring |
|---|---|
| 1 | 1 1 0 0 1 0 0 0 1 1 |
| 2 | 1 0 0 1 1 1 0 0 0 0 |
| 3 | 0 1 1 1 1 0 1 1 0 0 |
| 4 | 1 1 0 0 1 0 0 0 1 1 |

Tabelle C.3: Zusammensetzung der nächsten Generation

diesem Schritt wird in Abhängigkeit von der Verteilung ermittelt, welche Individuen Informationen in welchem Umfang austauschen. In diesem Fall soll ein One-Point-Crossover zwischen dem ersten und dem zweiten nach der ersten Stelle und dem dritten und vierten nach der dritten Stelle erfolgen. Der Vergleich der beiden Generationen zeigt, daß schon die erste Nachkommengeneration eine Steigerung der Fitness erfahren hat. Sowohl die Durchschnittsfitness, als auch die maximale Fitness liegen in der zweiten Generation höher.

| String Nr. | Bitstring | Wert | $f_i(x) = \sqrt{x}$ |
|---|---|---|---|
| 1 | 1001110000 | 624 | 24.98 |
| 2 | 1100100011 | 803 | 28.34 |
| 3 | 0110100011 | 419 | 20.47 |
| 4 | 1101101100 | 876 | 29.6 |
| Summe | | | 103.39 |
| Durchschnitt | | | 25.85 |
| Maximum | | | 29.6 |

Tabelle C.4: Ermittlung der Fitness

## C.2 Beschreibung des GA zur Optimierung der Codierung der Regelbasen

Der für diese Aufgabe eingesetzte genetische Algorithmus arbeitete mit der Populationsgröße 60, einer Crossover-Wahrscheinlichkeit von 0.6 sowie einer Mutationswahrscheinlichkeit von 0.03. Die Startgeneration wurde ausschließlich mit zufällig gebildeten Bitstrings der Länge 24 initialisiert. Die bei dieser Optimierung erzeugten Strings wurden mit Hilfe der Tabelle 6.2 umgesetzt. Hierbei werden den acht möglichen dreistelligen Binärstrings von links beginnend jeweils drei Bits des codierten Individuums zugeordnet. Der Bitstring 010 000 110 101 001 001 011 100 codiert z. B. die folgende Umsetzung:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| zu decodierender String | 000 | 001 | 011 | 010 | 110 | 111 | 101 | 100 |
| zugeordnete Teilstrings des GAs | 010 | 000 | 110 | 101 | 001 | 001 | 011 | 100 |
| Fuzzy-Set | M | SK | K | SK | K | K | G | SG |

Tabelle C.5: GA basierte Codierung der Regelbasis

Als Fitnessfunktion eines Strings wurde dazu die Summe aller zwölf durch Mutation eines einzelnen Bits möglichen Veränderungen ausgewählt. Dabei wurden diese Veränderungen, die einen Wert zwischen 0 (keine Veränderung) und 4 (Übergang von NN nach SG bzw. umgekehrt) annehmen konnten, in quadratischer Form berücksichtigt. Damit wurde der Einfluß einer starken Veränderung im Vergleich zu einer geringen erhöht. Zusätzlich wurde jede mögliche Mutation eines Bits, die zu keiner Veränderung des korrespondierenden linguistischen Terms führte, durch die Addition des maximal möglichen Wertes von 16 zur Fitness eines Strings berücksichtigt. Damit sollte sichergestellt werden, daß die Mutation eines einzelnen Bits auch zu einer Veränderung des linguistischen Terms führt. Wenn durch Mutation bzw. Crossover ein String erzeugt wurde, dessen Decodierung ergab, daß nicht alle fünf linguistischen Terme mindestens einmal in der Codierungstabelle enthalten sind, erhielt dieser Strings den maximal möglichen Fitnesswert von 192 ($12x4^2$). Weil auch bei dieser Aufsummierung der berechneten Differenzen ein guter String durch eine niedrigere Summe charakterisiert wird als ein String mit einer geringeren Fitness, wurde die zur Reproduktion eingesetzte Fitnessfunktion durch Subtraktion der berechneten Summe vom maximal möglichen Wert 192 ermittelt. Da bei dieser Optimierung mehrere verschiedene Strings mit einem gleich guten Fitnesswert ermittelt wurden, wurde zuletzt eine Codierung ausgewählt, bei der die linguistischen Terme aufsteigend nach der Reihenfolge ihrer Schwerpunkte sortiert waren. Diese durch Einsatz des genetischen Algorithmus ermittelte Codierung ist in Tabelle C.5 dargestellt.

# Anhang D

# Policing Controller

## D.1 Fitness Funktion 5

Im Folgenden sind die Übertragungskennlinien der Fuzzy Controller $FC_1$, $FC_2$ und $FC_3$, die mit Hilfe der Fitness Funktion 5 ermittelt wurden, abgebildet.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \cdot P_i^2 \cdot (1 - BW_{\Delta,i})^2 \tag{D.1}$$

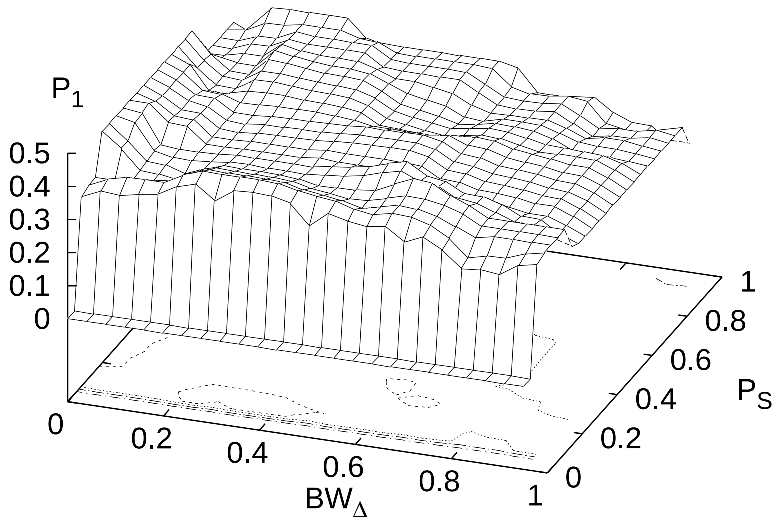



Abbildung D.1: Kennlinienfeld des Fuzzy Controllers $FC_1$

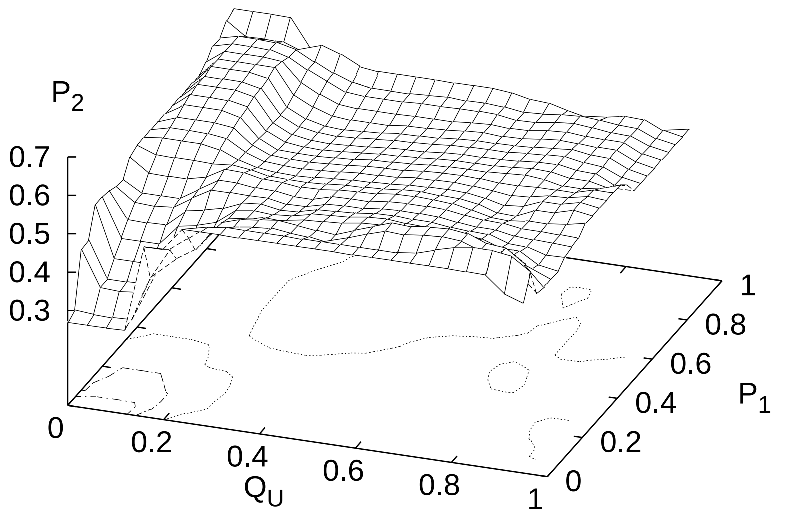



Abbildung D.2: Kennlinienfeld des Fuzzy Controllers $FC_2$

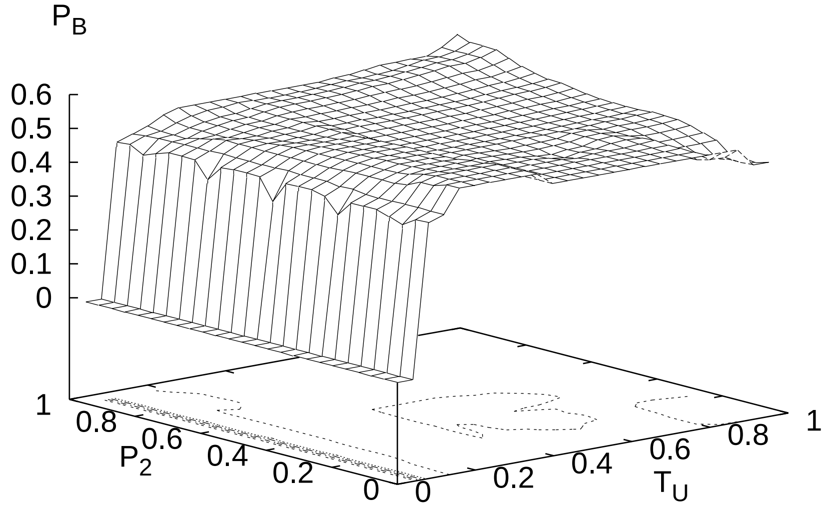
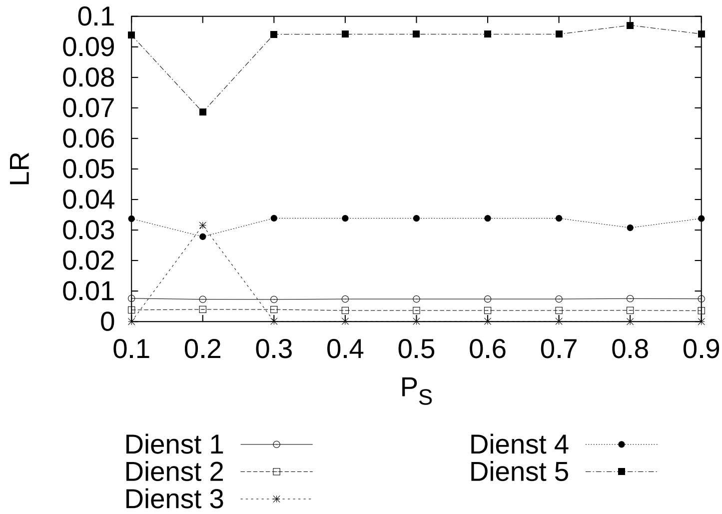
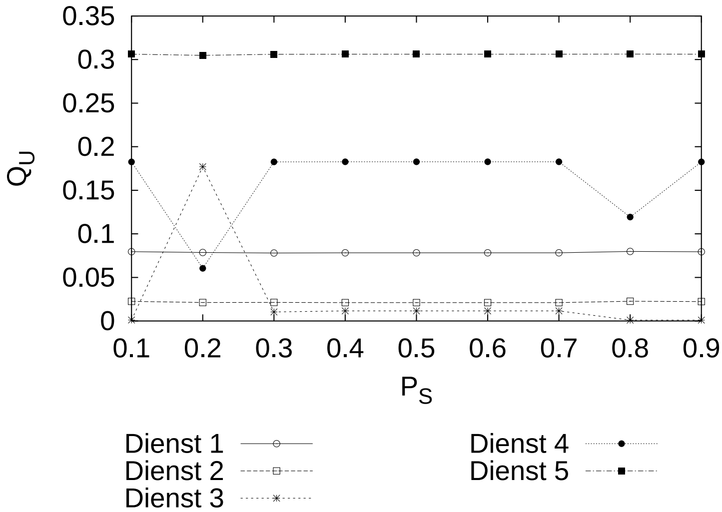



Abbildung D.3: Kennlinienfeld des Fuzzy Controllers $FC_3$

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controller dessen Regelbasis und Zugehörigkeitsfunktionen mit einem genetischen Algorithmus erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.




(a) Verlustrate




(b) Auslastung

Abbildung D.4: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

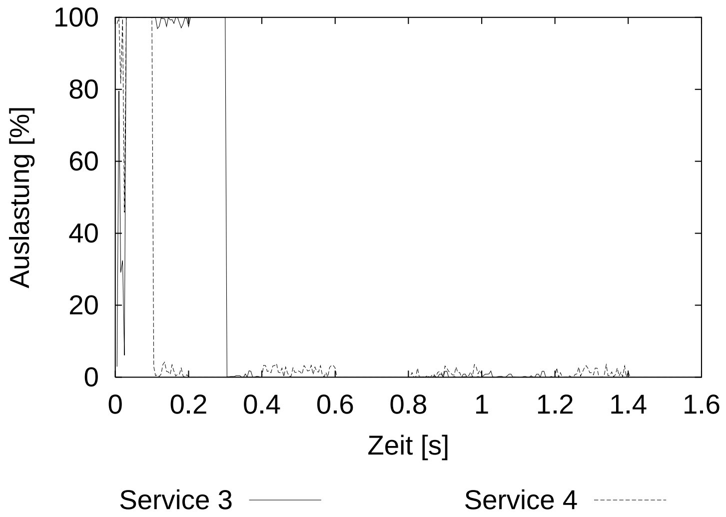



Abbildung D.5: Verlauf der Auslastung für $P_S = 0.2$

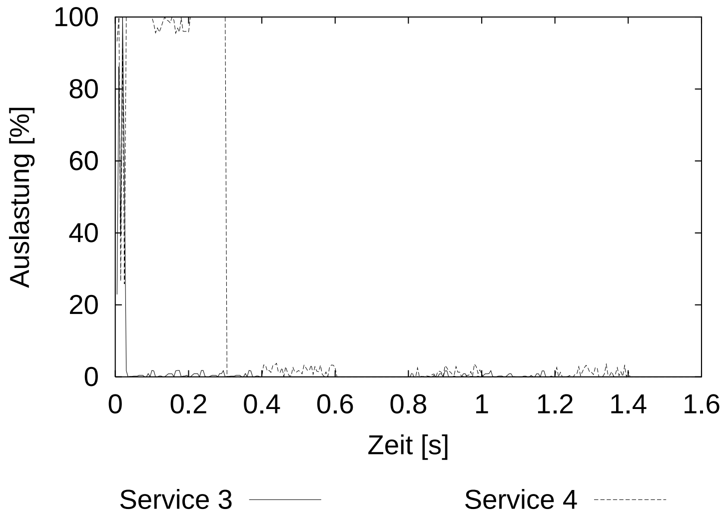



Abbildung D.6: Verlauf der Auslastung für $P_S = 0.3$

## D.2 Fitness Funktion 6

Im Folgenden sind die Übertragungskennlinien der Fuzzy Controller $FC_1$, $FC_2$ und $FC_3$, die mit Hilfe der Fitness Funktion 6 ermittelt wurden, abgebildet.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i \cdot P_i^3 \cdot (1 - BW_{\Delta,i})^3 \tag{D.2}$$

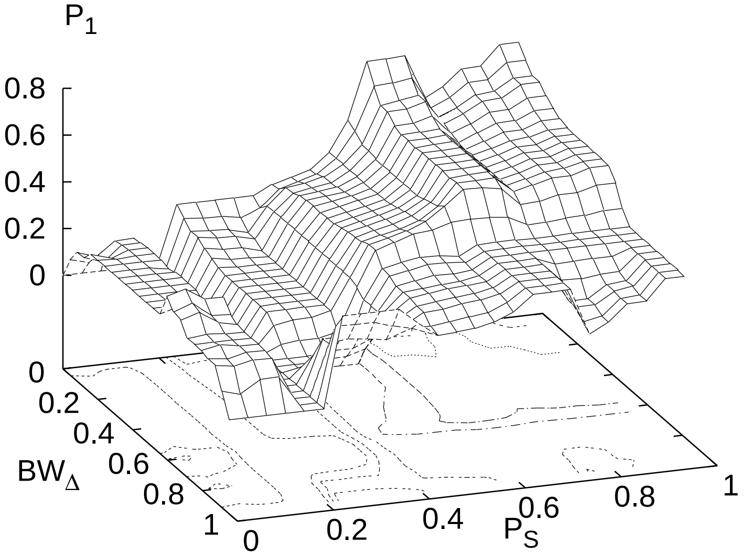



Abbildung D.7: Kennlinienfeld des Fuzzy Controllers $FC_1$

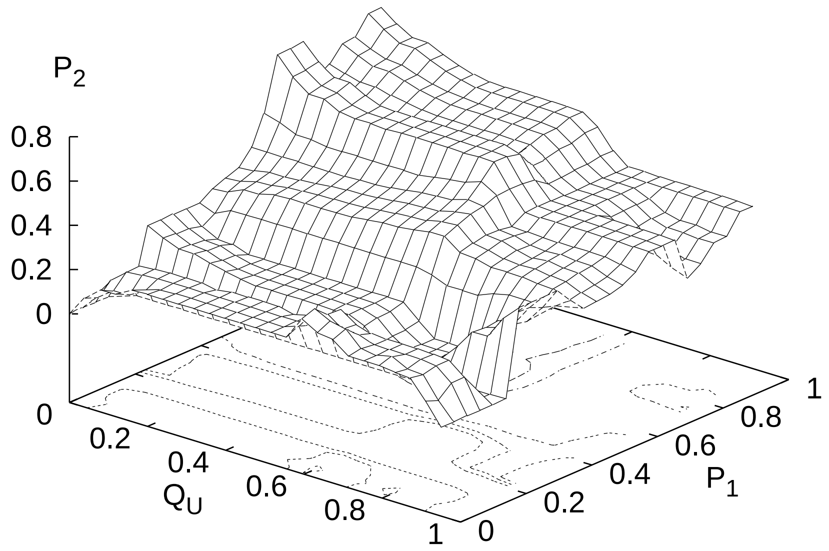



Abbildung D.8: Kennlinienfeld des Fuzzy Controllers $FC_2$

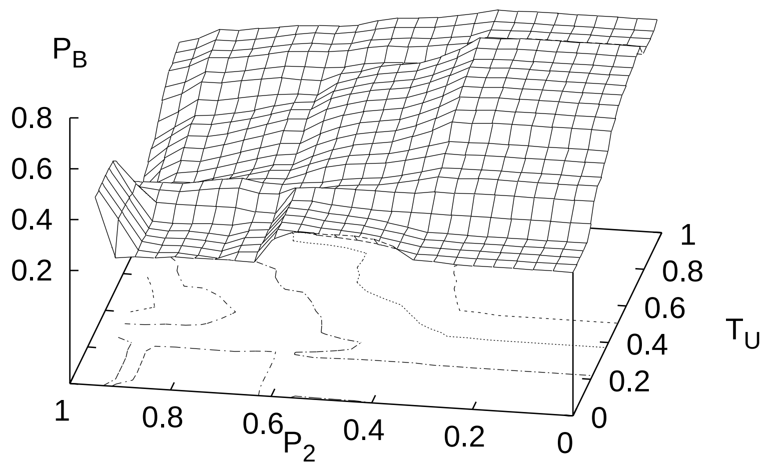
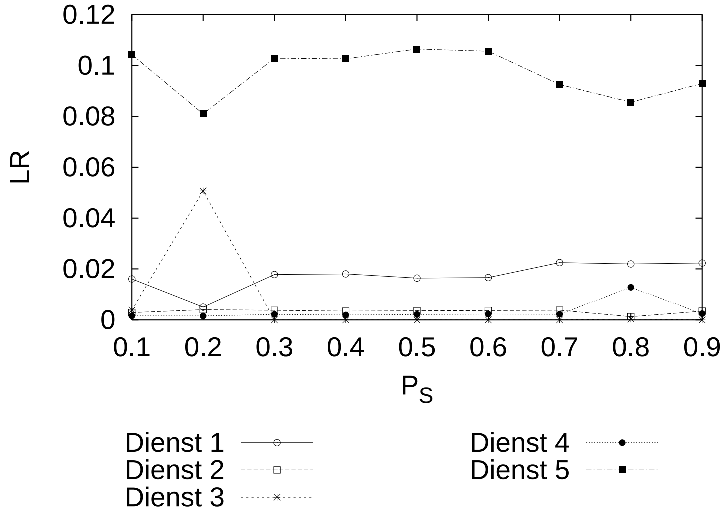
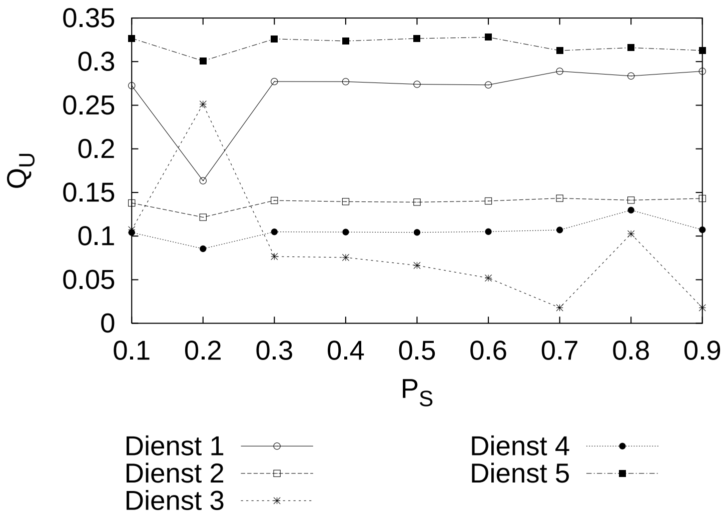



Abbildung D.9: Kennlinienfeld des Fuzzy Controllers $FC_3$

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem genetischen Algorithmus erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.




(a) Verlustrate




(b) Auslastung

Abbildung D.10: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

## D.3 Fitness Funktion 7

Im Folgenden sind die Übertragungskennlinien der Fuzzy Controller $FC_1$, $FC_2$ und $FC_3$, die mit Hilfe der Fitness Funktion 7 ermittelt wurden, abgebildet.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} LR_i^2 \cdot P_i^3 \cdot (1 - BW_{\Delta,i})^3 \tag{D.3}$$

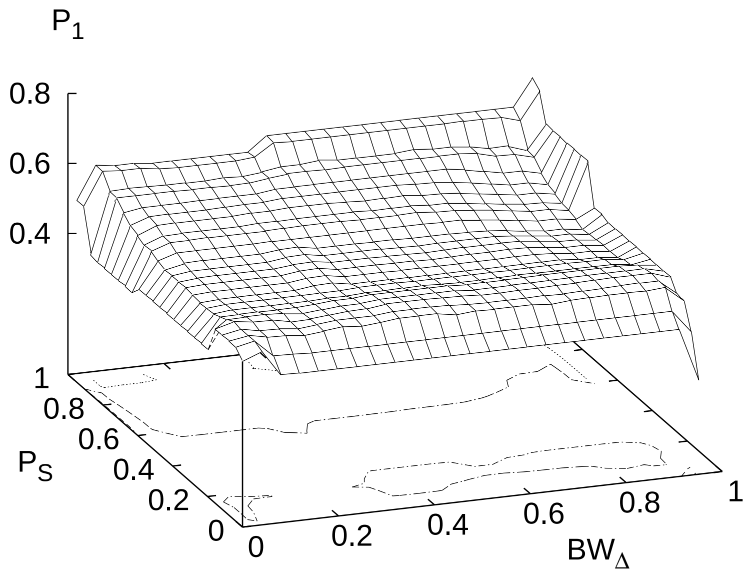



Abbildung D.11: Kennlinienfeld des Fuzzy Controllers $FC_1$

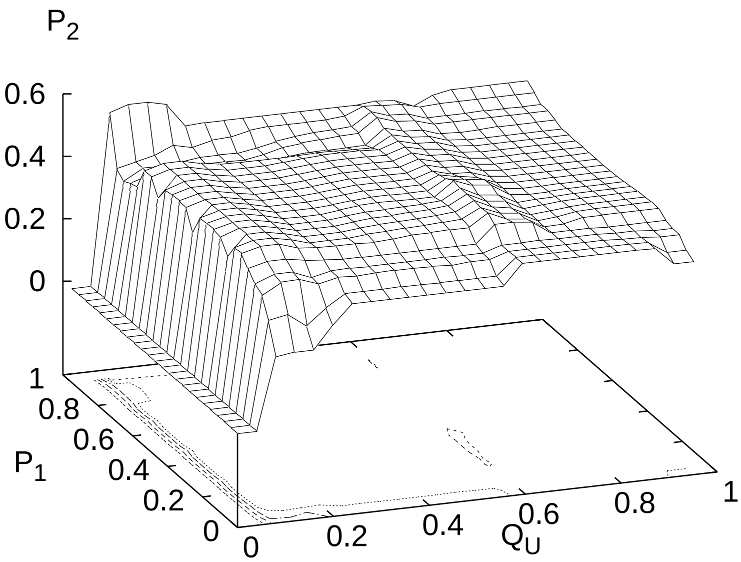



Abbildung D.12: Kennlinienfeld des Fuzzy Controllers $FC_2$

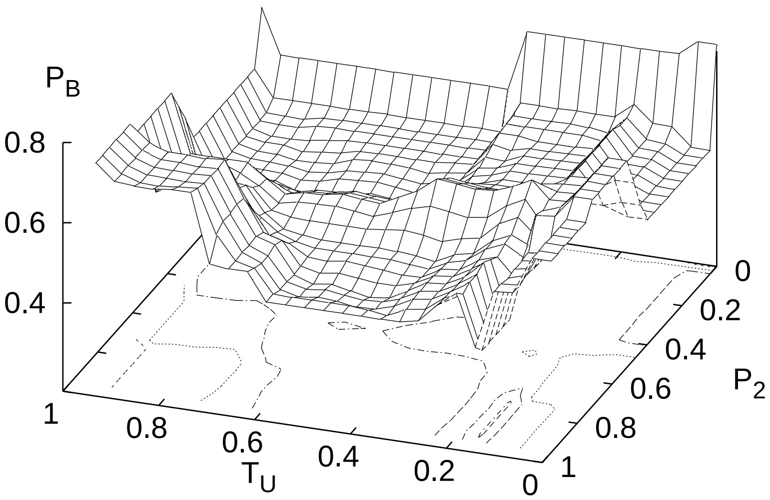



Abbildung D.13: Kennlinienfeld des Fuzzy Controllers $FC_3$

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem genetischen Algorithmus nach 85 Generationen erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.

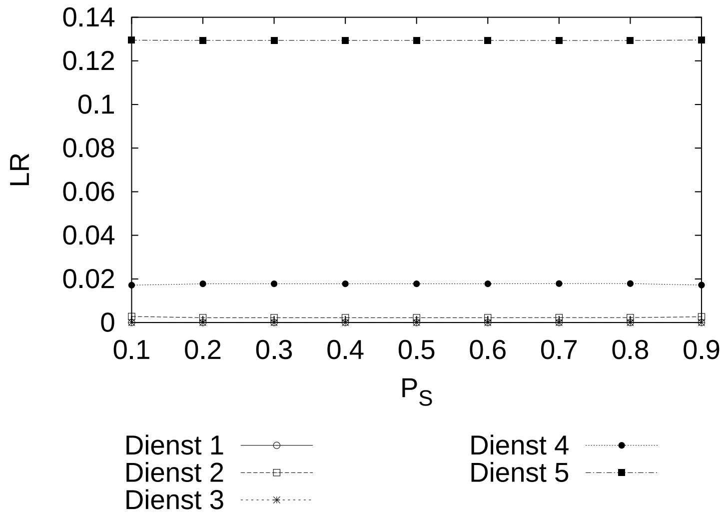



Abbildung D.14: $Verlustrate$

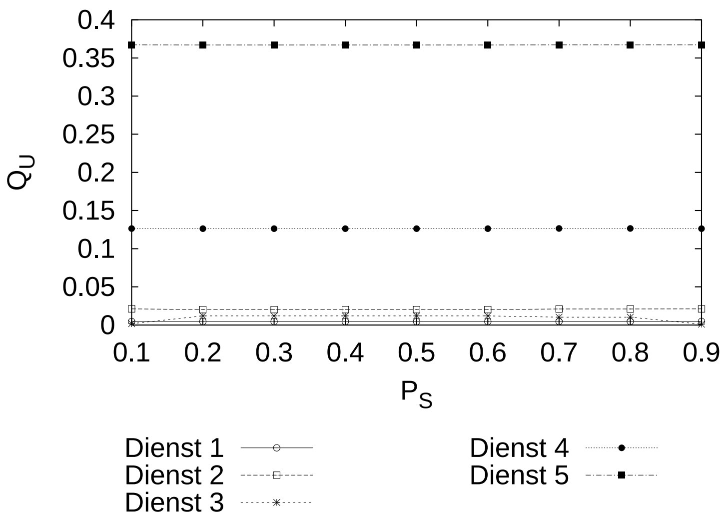



Abbildung D.15: $AuslastungderWarteschlangen$

## D.4 Fitness Funktion 8

### D.4.1 Generation 96

Im Folgenden sind die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ dargestellt. Der Policing Controller ergab sich unter Anwendung der Fitness Funktion Gl.6.13 nach 96 Generationen.

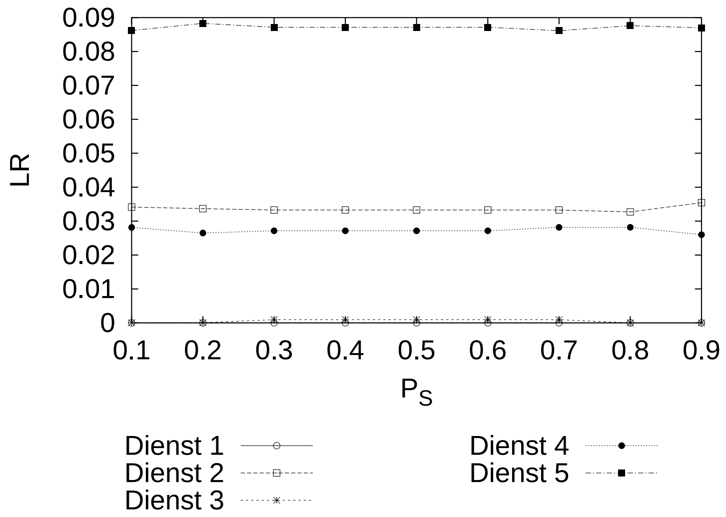



Abbildung D.16: Abhängigkeit der Verluste von $P_{S,3}$

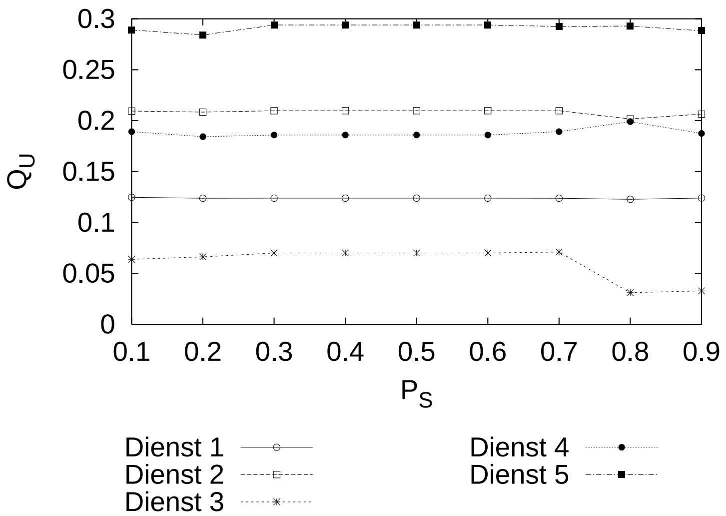



Abbildung D.17: Abhängigkeit der Auslastung von $P_{S,3}$

### D.4.2 Generation 143

Im Folgenden sind die Zusammenhänge, die sich bei der Untersuchung des Verhaltens des Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem genetischen Algorithmus nach 143 Generationen erzeugt wurden, ergaben. Dargestellt sind die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$.

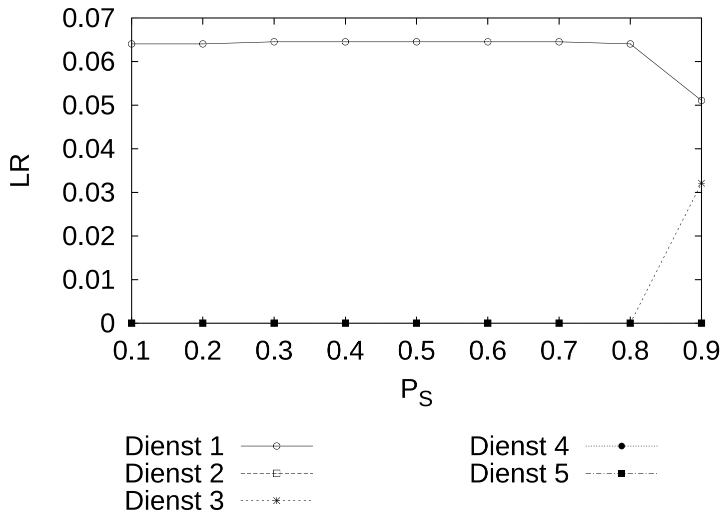



Abbildung D.18: Abhängigkeit der Verlustrate von $P_{S,3}$

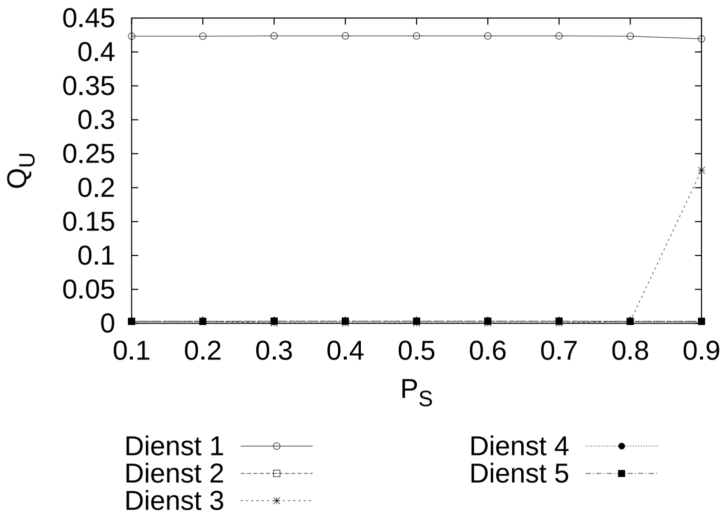



Abbildung D.19: Abhängigkeit der Auslastung von $P_{S,3}$

## D.5 Fitness Funktion 9

Im Folgenden sind die ind die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ dargestellt. Der Policing Controller ergab sich unter Anwendung der Fitness Funktion Gl.6.14.

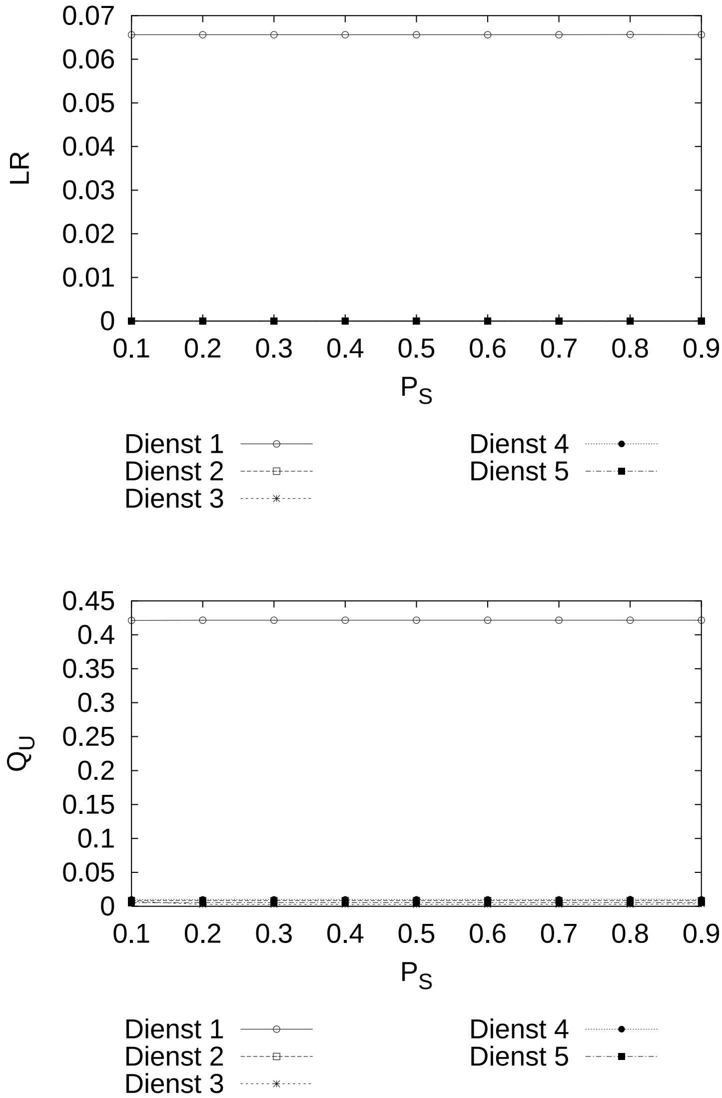



Abbildung D.20: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

## D.6 Fitness Funktion 10

Im Folgenden sind die Übertragungskennlinien der Fuzzy Controller $FC_1$, $FC_2$ und $FC_3$, die mit Hilfe der Fitness Funktion 10 ermittelt wurden, abgebildet.

$$Fitness_{Gesamt} = \sum_{i=1}^{n} \frac{LR_i \cdot P_i^3}{(1 - BW_{\Delta,i})^3} \tag{D.4}$$

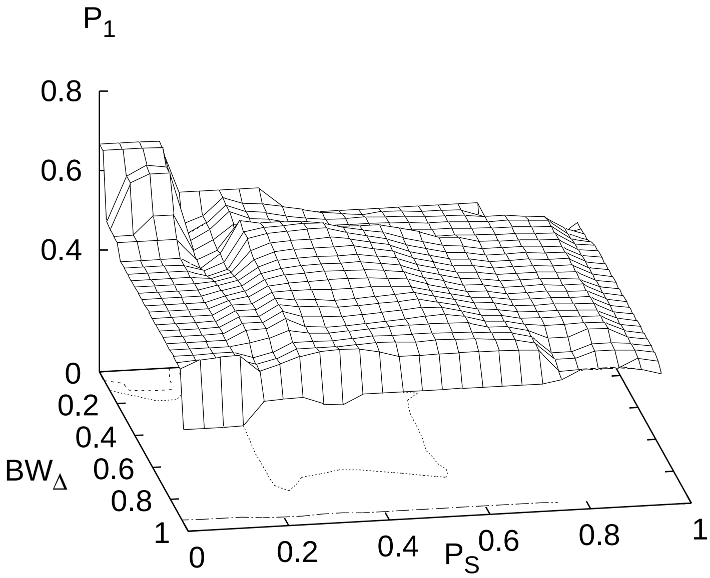



Abbildung D.21: Kennlinienfeld des Fuzzy Controllers $FC_1$

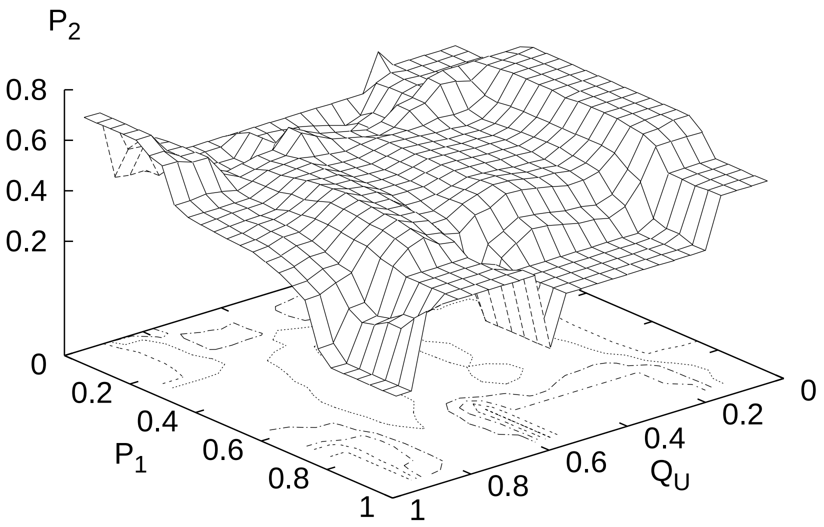



Abbildung D.22: Kennlinienfeld des Fuzzy Controllers $FC_2$

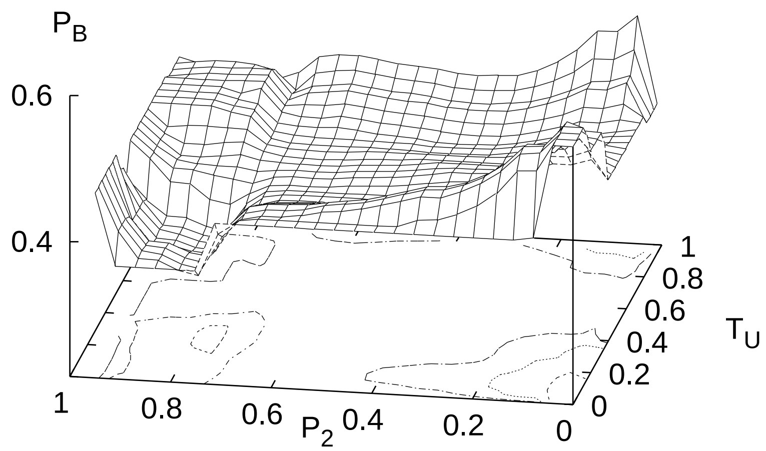



Abbildung D.23: Kennlinienfeld des Fuzzy Controllers $FC_3$

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem genetischen Algorithmus nach 64 Generationen erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.

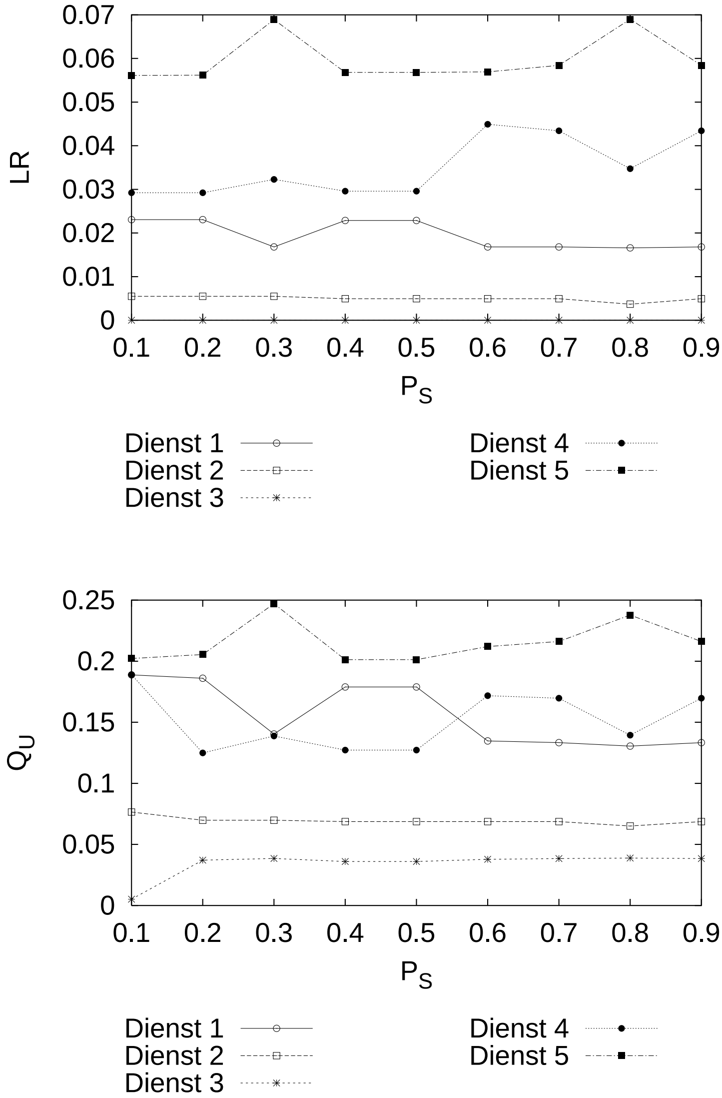
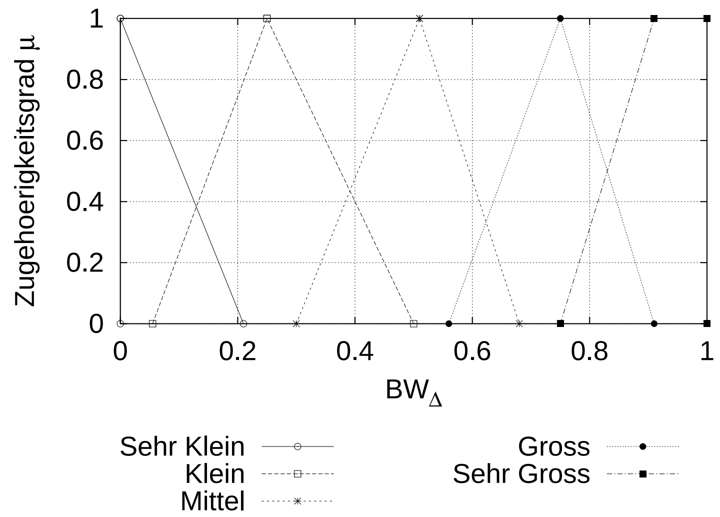



Abbildung D.24: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

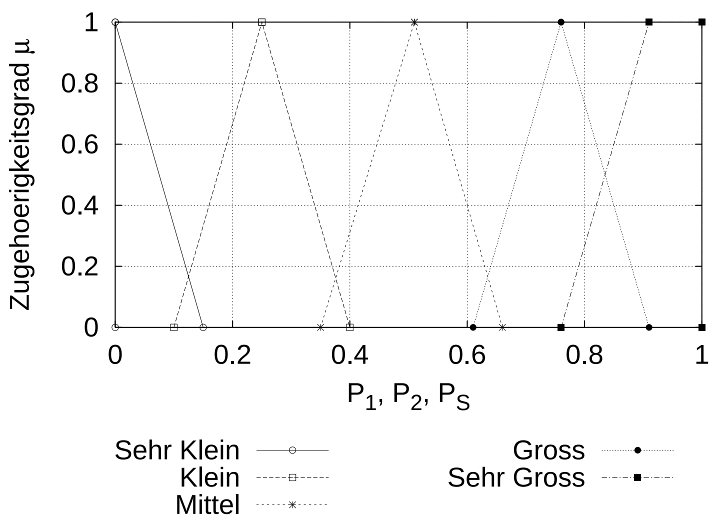
# D.7 Unscharfe Fitness Funktion

## D.7.1 Fitness Controller

Im Folgenden sind die Zugehörigkeitsfunktionen und Regelbasen des Fuzzy Logic basierten Fitness Controllers abgebildet.

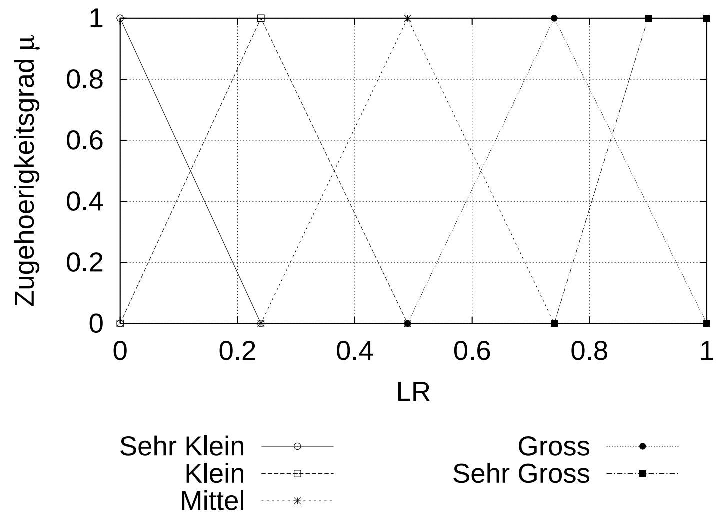



(a) Abweichung von der deklarierten Bandbreite $BW_\Delta$

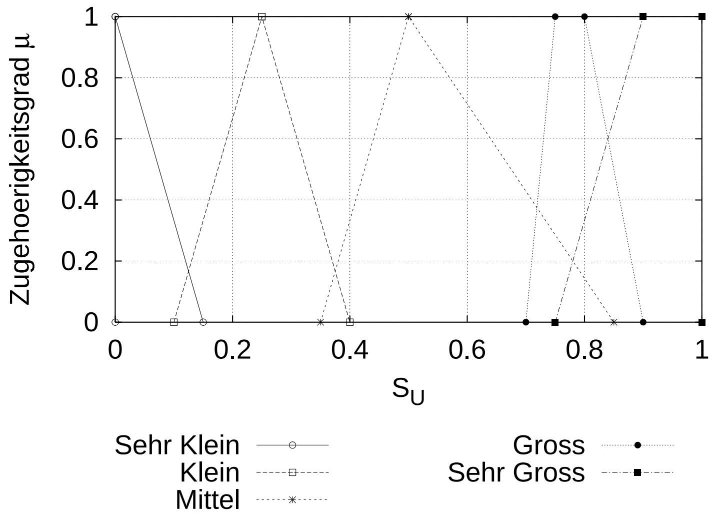



(b) $P_1$, $P_2$ und $P_S$

Abbildung D.25: Linguistische Variablen des Fitness Controllers




(a) Verlustrate




(b) Systemauslastung $S_U$

Abbildung D.26: Linguistische Variablen des Fitness Controllers

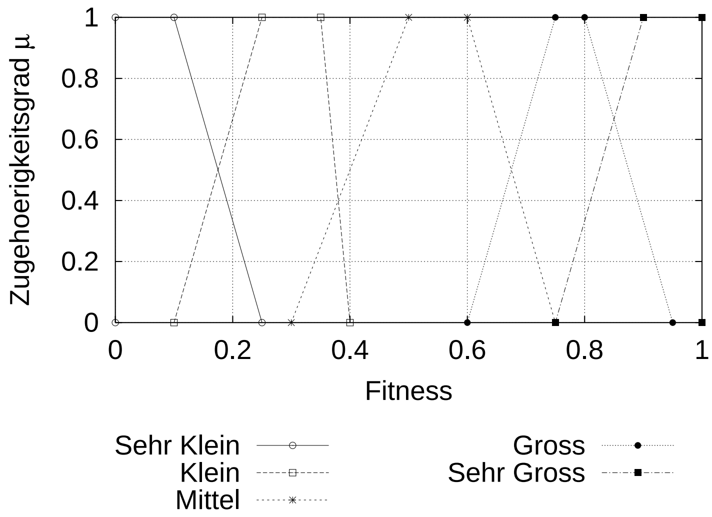



Abbildung D.27: Linguistische Variable Fitness

| $P_1$ | | $BW_\Delta$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_S$ | SK | SK | K | M | G | SG |
| | K | SK | SK | M | G | SG |
| | M | SK | SK | K | M | G |
| | G | SK | SK | K | K | M |
| | SG | SK | SK | SK | K | M |

| $P_2$ | | $LR$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_1$ | SK | SK | SK | SK | SK | SK |
| | K | SK | SK | SK | K | M |
| | M | SK | SK | K | M | G |
| | G | SK | SK | K | G | SG |
| | SG | K | M | G | SG | SG |

| $Fitness$ | | $S_U$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_2$ | SK | SG | SG | SG | SG | SG |
| | K | M | M | G | SG | SG |
| | M | S | M | M | G | SG |
| | G | SK | K | K | M | G |
| | SG | SK | SK | K | K | M |

Tabelle D.1: Regelbasis der in Abb. 7.1 dargestellten Fuzzy Controller

### D.7.2 Generation 8

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem Fuzzy Logic basierten genetischen Algorithmus nach 8 Generationen erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.

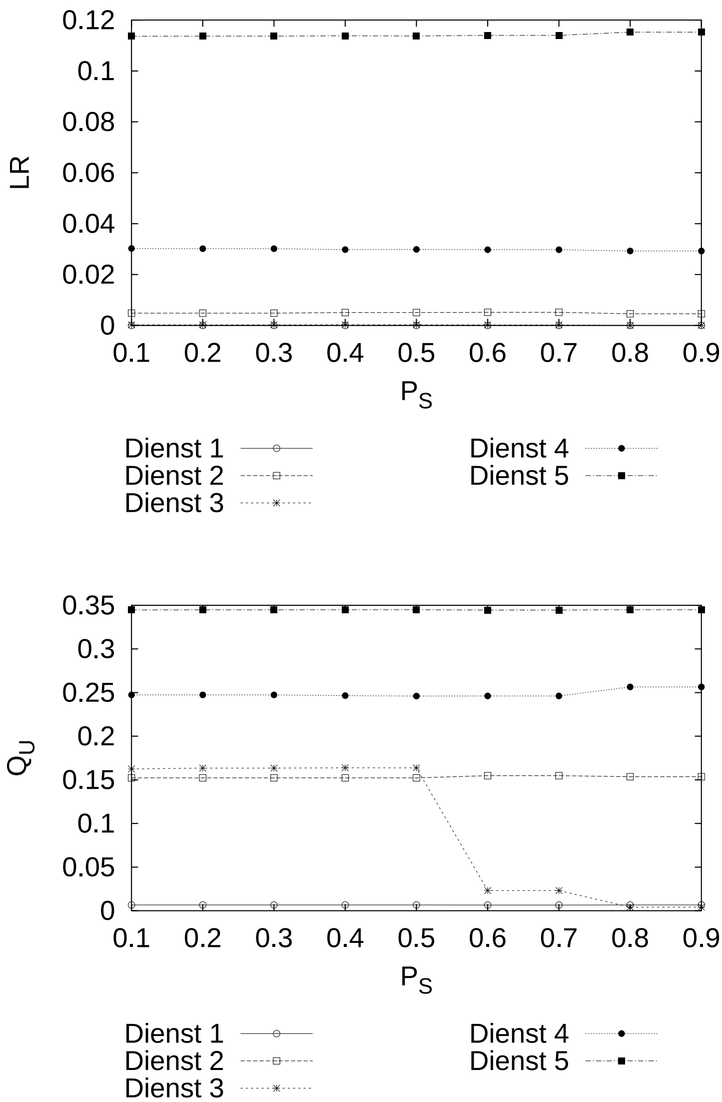



Abbildung D.28: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

### D.7.3 Generation 29

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem Fuzzy Logic basierten genetischen Algorithmus nach 29 Generationen erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.

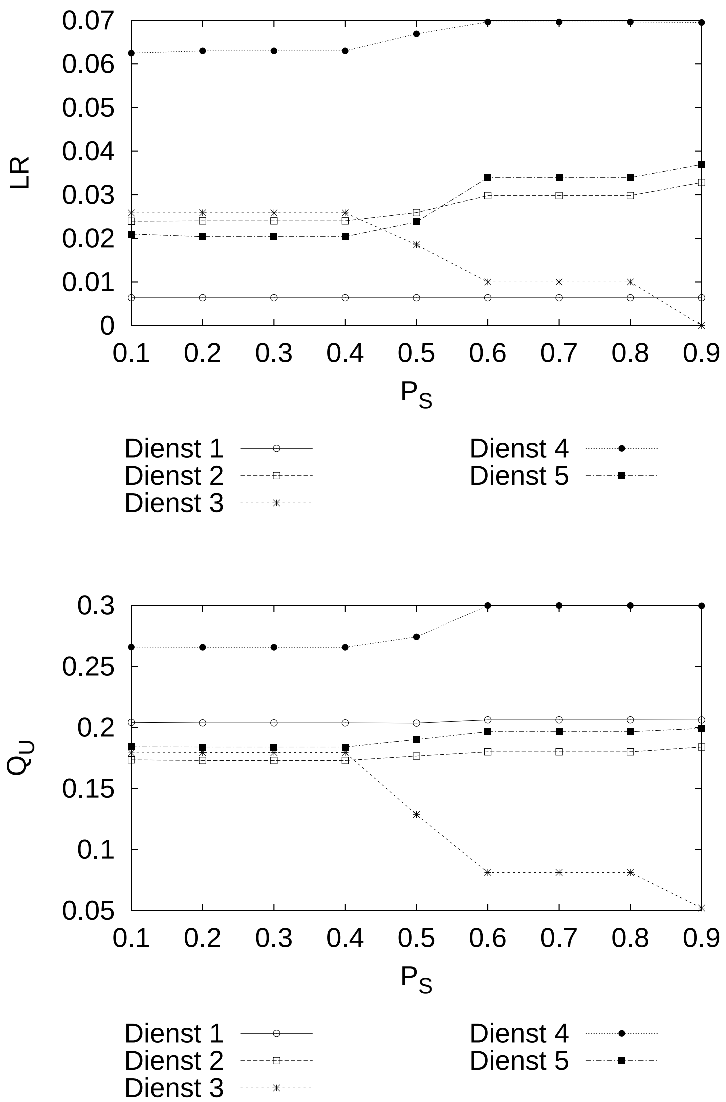



Abbildung D.29: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

### D.7.4 Generation 63

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem Fuzzy Logic basierten genetischen Algorithmus nach 63 Generationen erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.

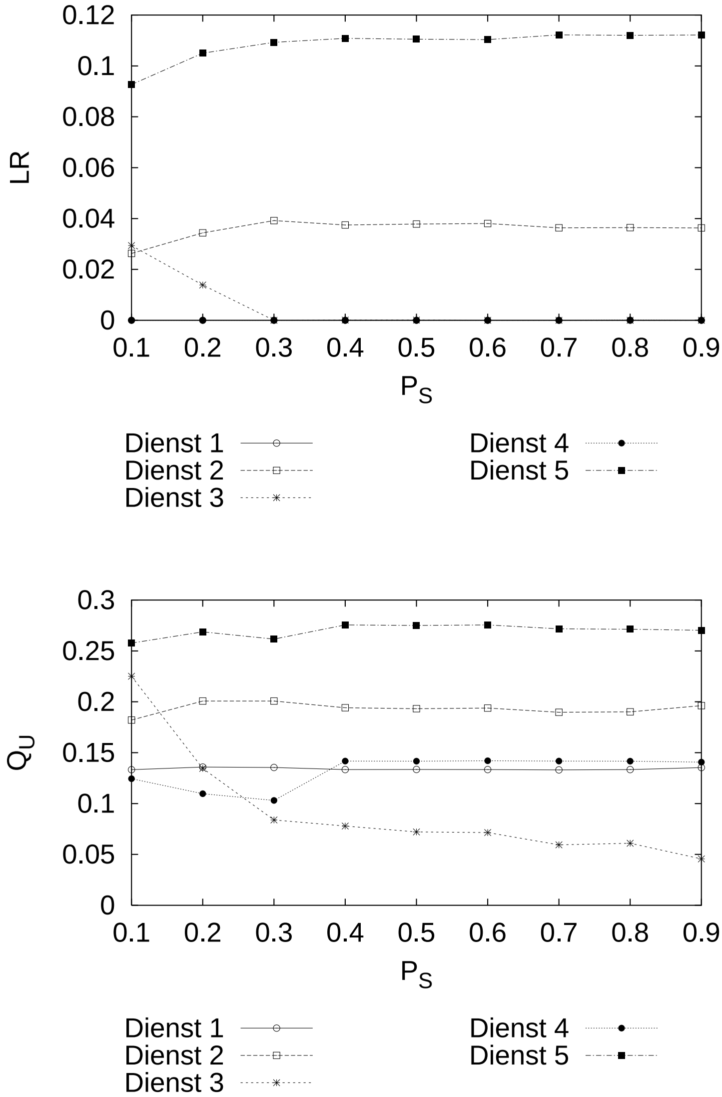



Abbildung D.30: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

### D.7.5 Generation 127

Bei der Untersuchung des Verhaltens des Fuzzy Logic basierten Policing Controllers, dessen Regelbasis und Zugehörigkeitsfunktionen mit einem Fuzzy Logic basierten genetischen Algorithmus nach 127 Generationen erzeugt wurden, ergaben sich für die Verlustrate und die Auslastung in Abhängigkeit von der Dienstpriorität $P_S$ folgende Zusammenhänge.

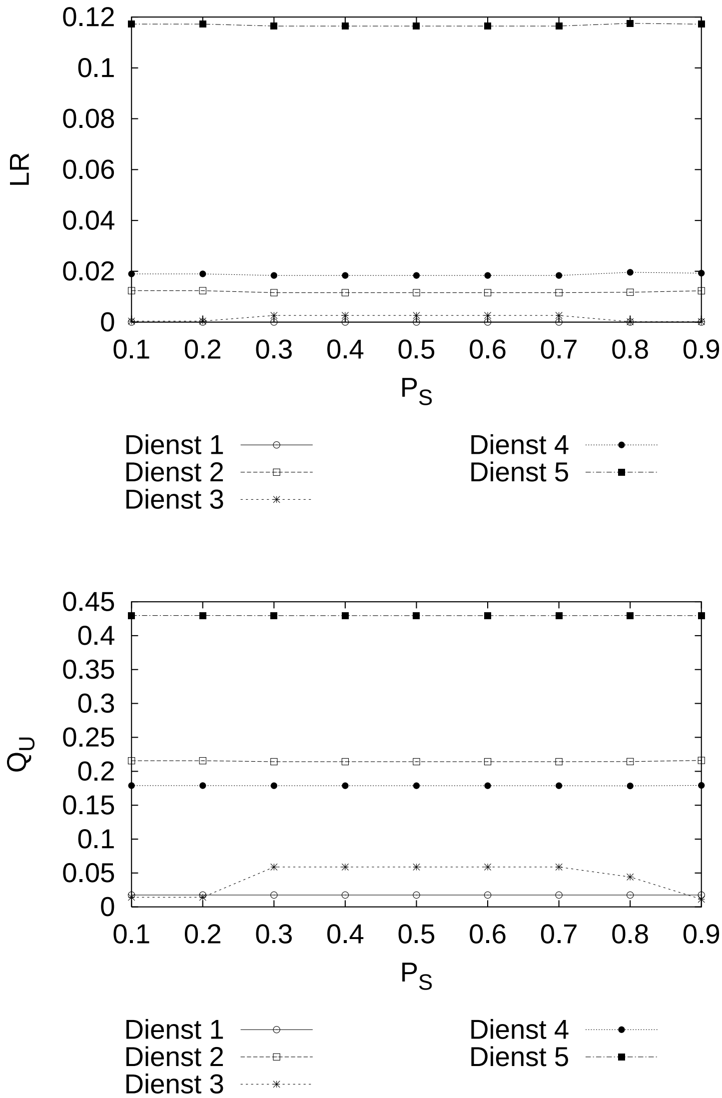



Abbildung D.31: Abhängigkeit der Verluste und Auslastung von $P_{S,3}$

# Anhang E

# Call Admission Controller

## E.1 Traffic Qualifier

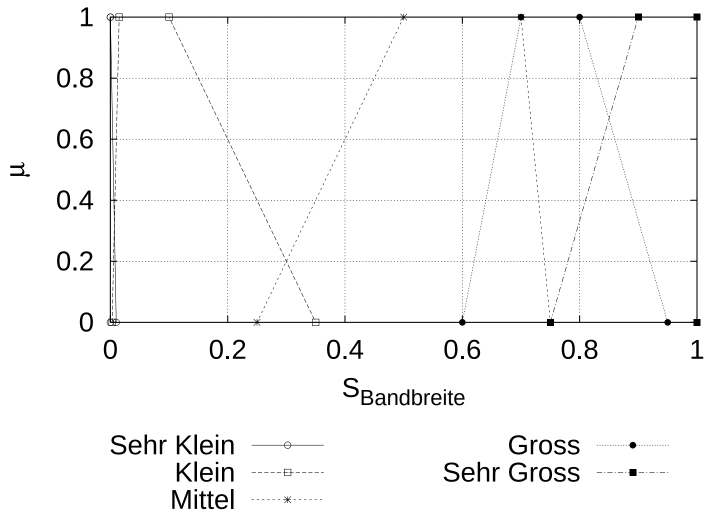



Abbildung E.1: Linguistische Terme der Variablen $S_{Bandbreite}$

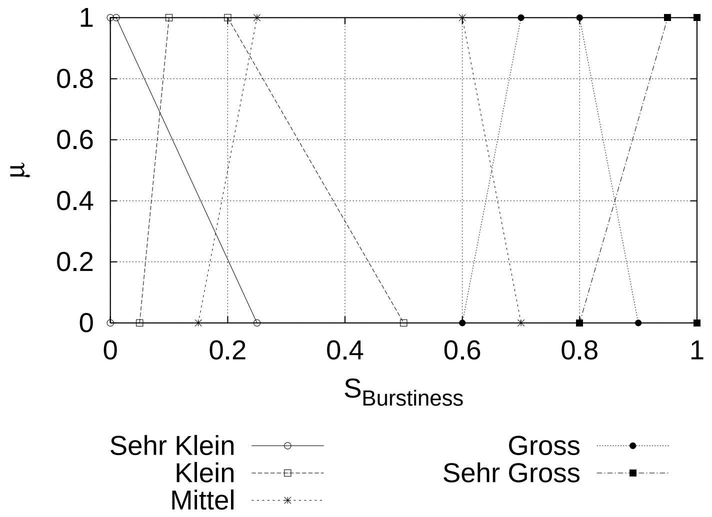



Abbildung E.2: Linguistische Terme der Variablen $S_{Burstiness}$

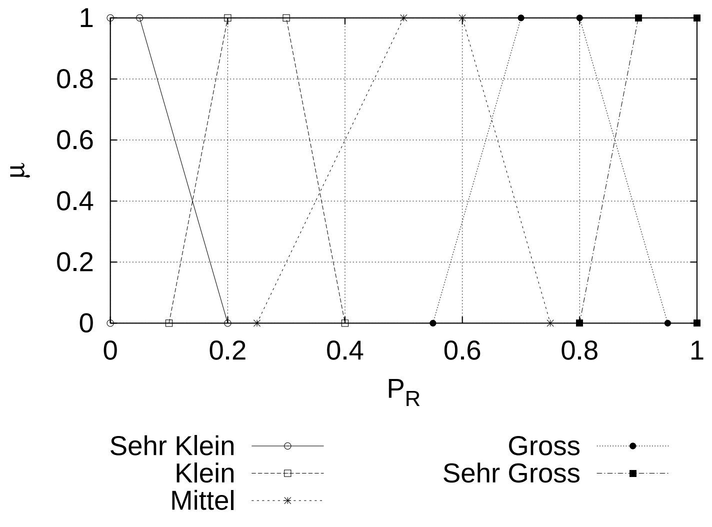



Abbildung E.3: Linguistische Terme der Variablen $P_R$, die als Ausgangswert des Traffic Qualifiers die Anforderung, die ein Dienst an das Übertragungssystem stellt, beschreibt.

# E.2 State Qualifier

## E.2.1 Funktion 1

$P_U = const. = 0.4$

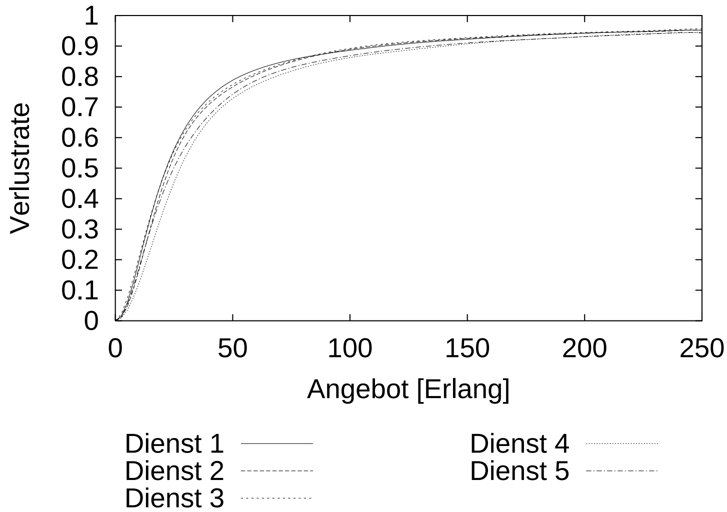



Abbildung E.4: $LR = f(A)\ mit\ P_U = 0.4$

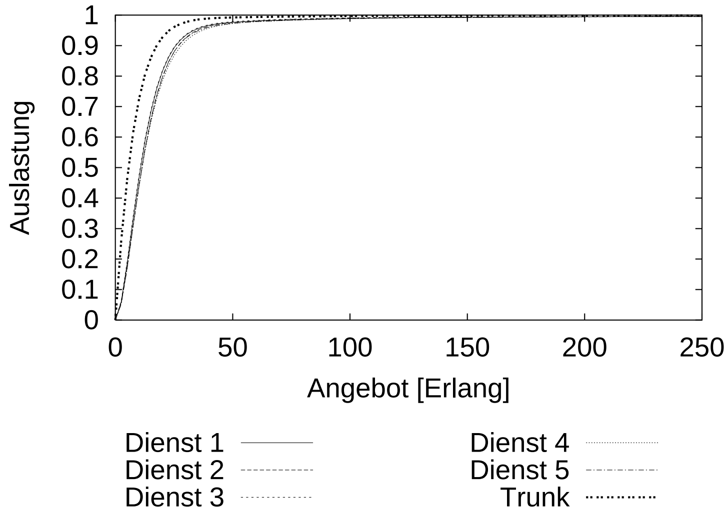



Abbildung E.5: $Auslastung = f(A)\ mit\ P_U = 0.4$

$P_U = const. = 0.9$

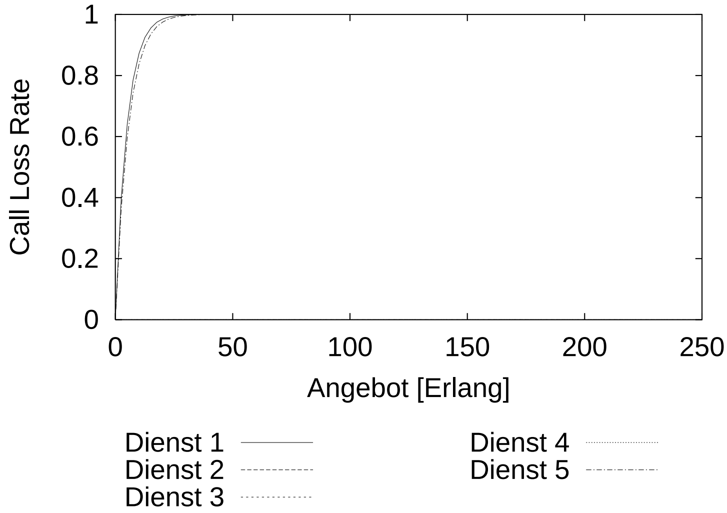



Abbildung E.6: $CLR = f(A)\ mit\ P_U = 0.9$

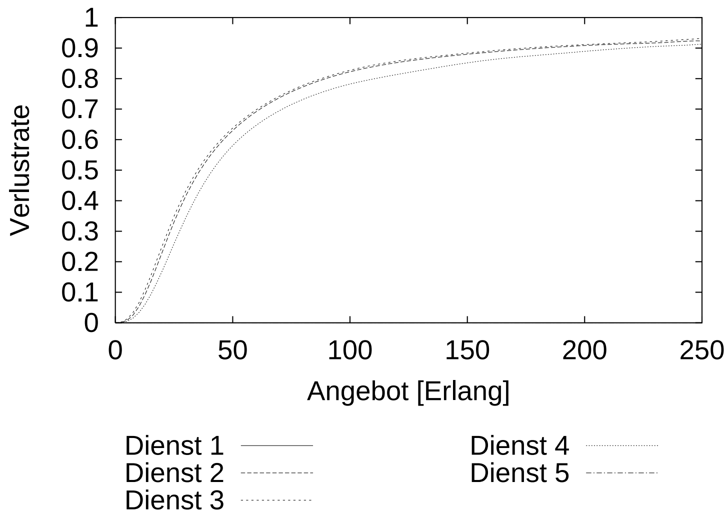



Abbildung E.7: $LR = f(A)\ mit\ P_U = 0.9$

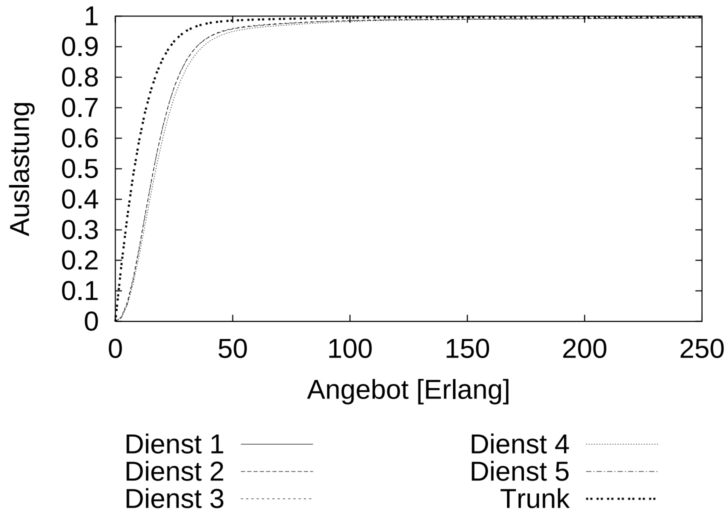



Abbildung E.8: $Auslastung = f(A)\ mit\ P_U = 0.9$

### E.2.2 Funktion 2

$\ddot{U}bertragungsreserve = 1 - T_U$

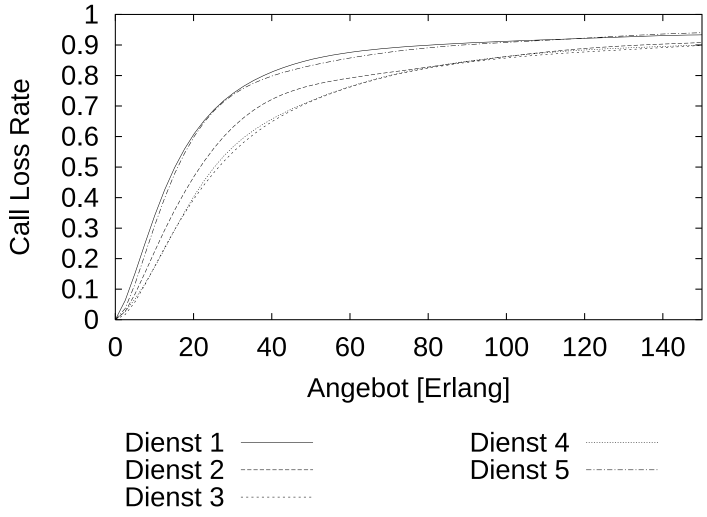



Abbildung E.9: Call Loss Rate

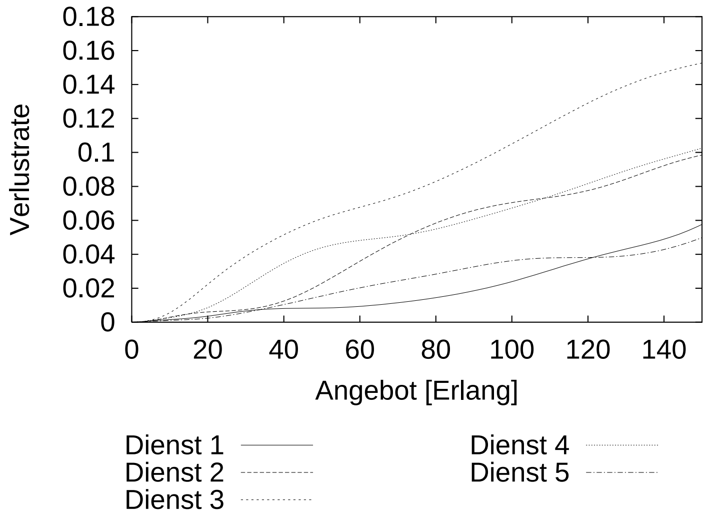



Abbildung E.10: Paketverlustrate

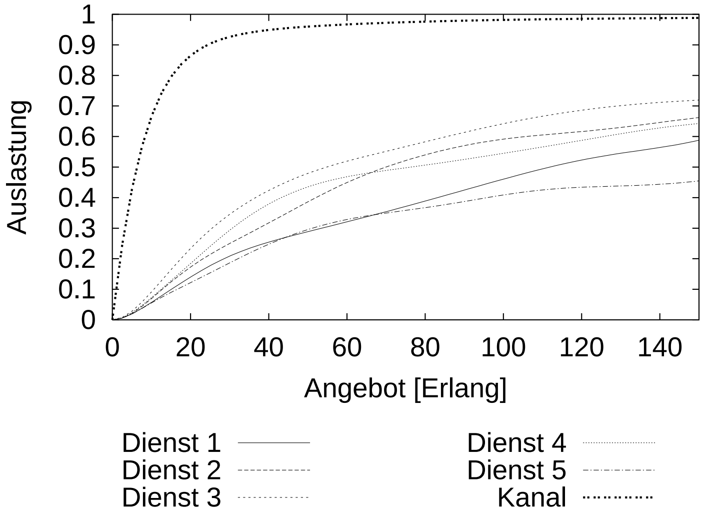



Abbildung E.11: Kanalauslastung

### E.2.3 Funktion 3

$\ddot{U}bertragungsreserve = 1 - Q_U$

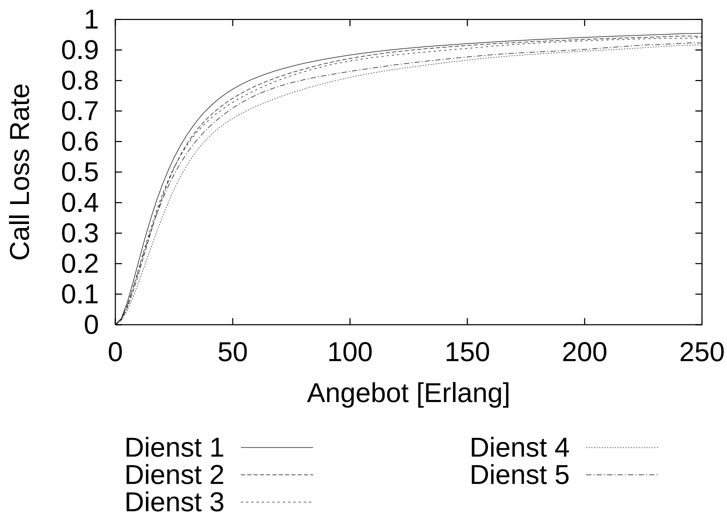



Abbildung E.12: Call Loss Rate

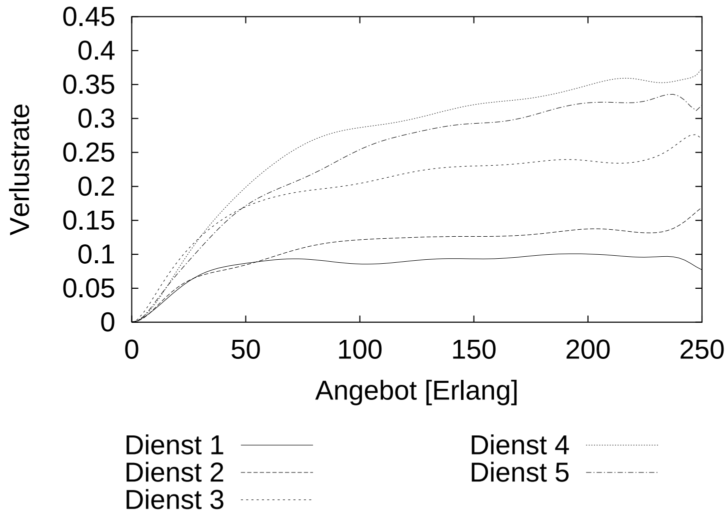



Abbildung E.13: Paketverlustrate

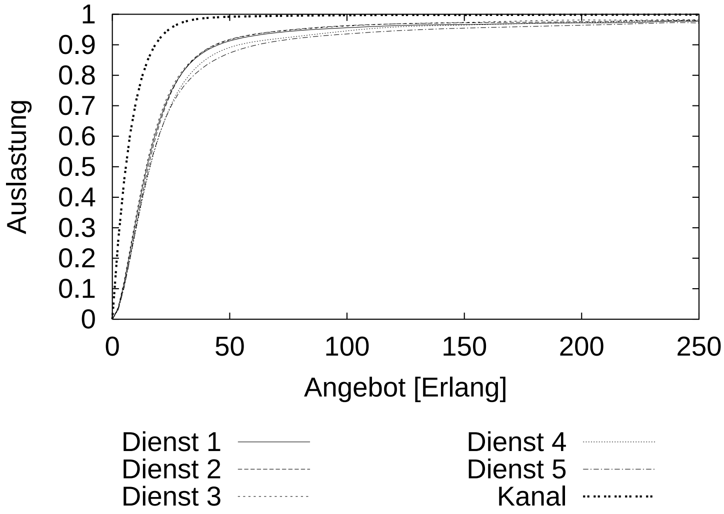



Abbildung E.14: Kanalauslastung

### E.2.4 Funktion 4

$\ddot{U}bertragungsreserve = (1 - Q_U) \cdot (1 - T_U)$

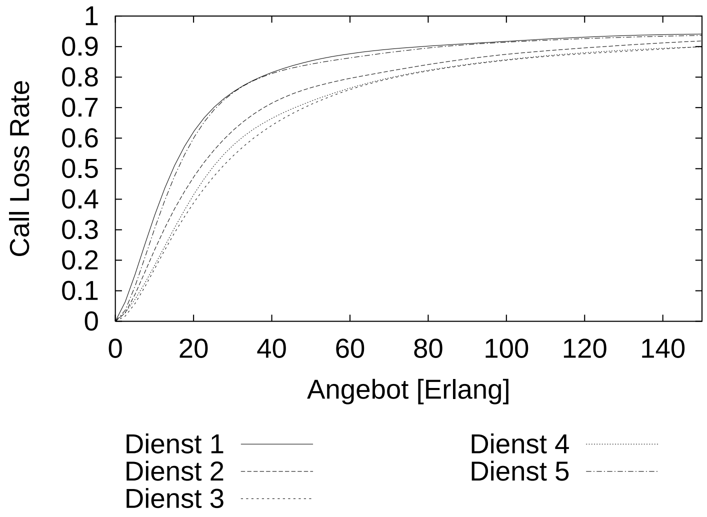



Abbildung E.15: Call Loss Rate

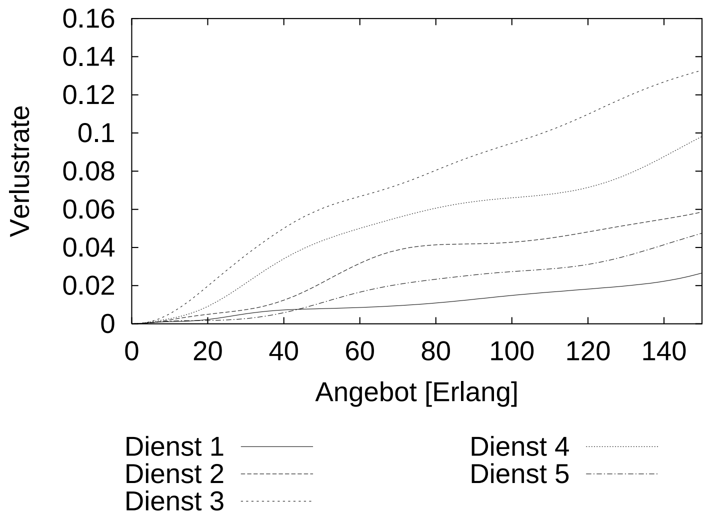



Abbildung E.16: Paketverlustrate

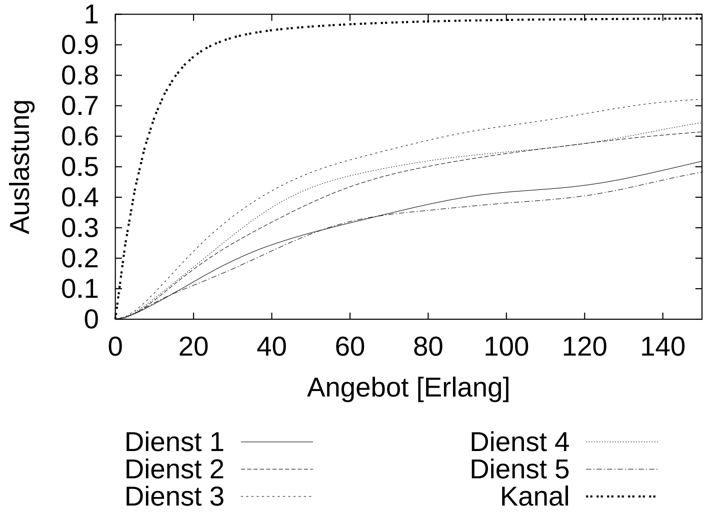



Abbildung E.17: Kanalauslastung

### E.2.5 Funktion 5

$\ddot{U}bertragungsreserve = 1 - S_U$

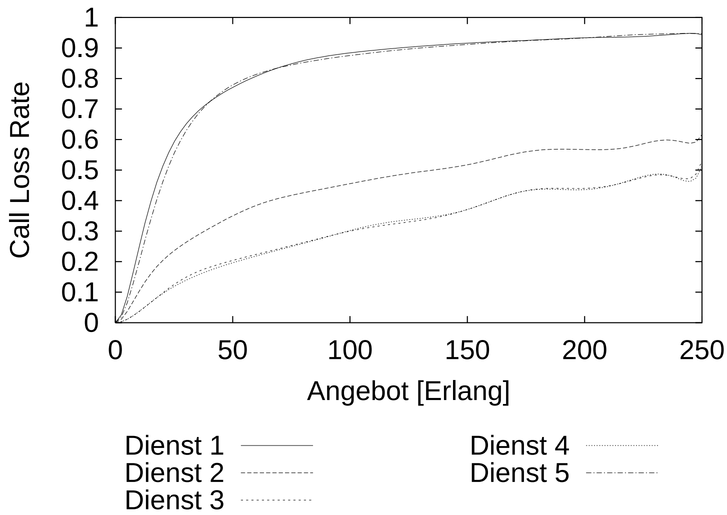



Abbildung E.18: Call Loss Rate

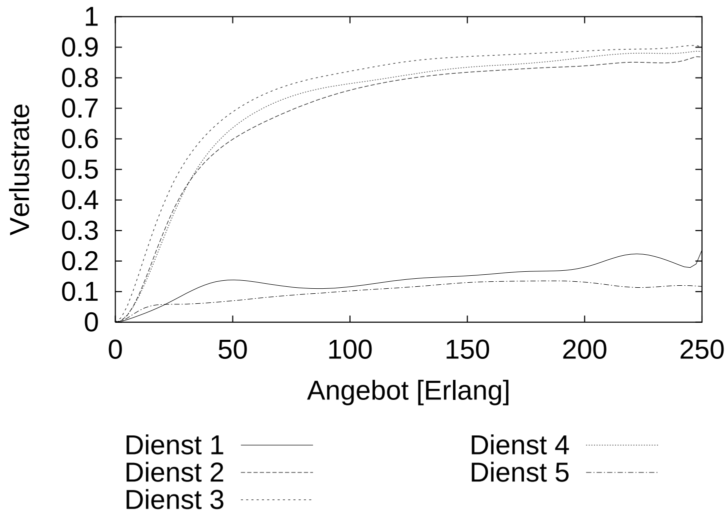



Abbildung E.19: Paketverlustrate

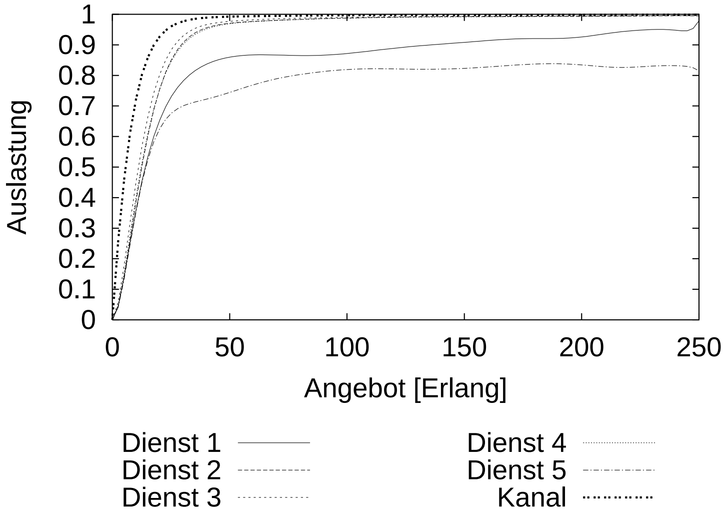



Abbildung E.20: Kanalauslastung

### E.2.6 Funktion 6

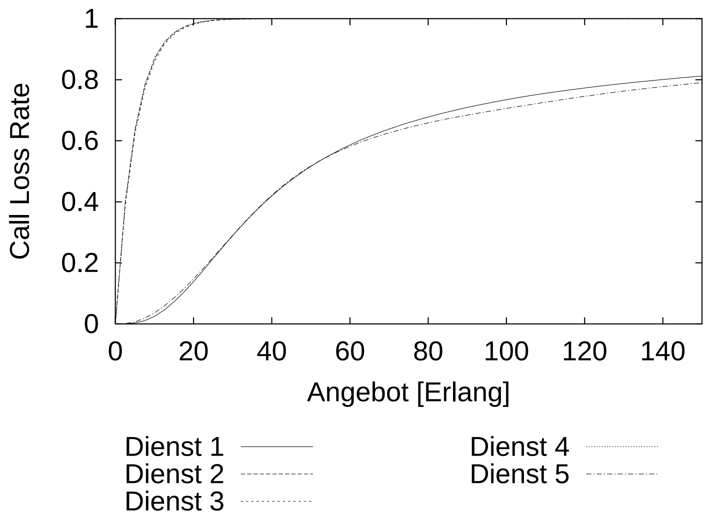



Abbildung E.21: Call Loss Rate

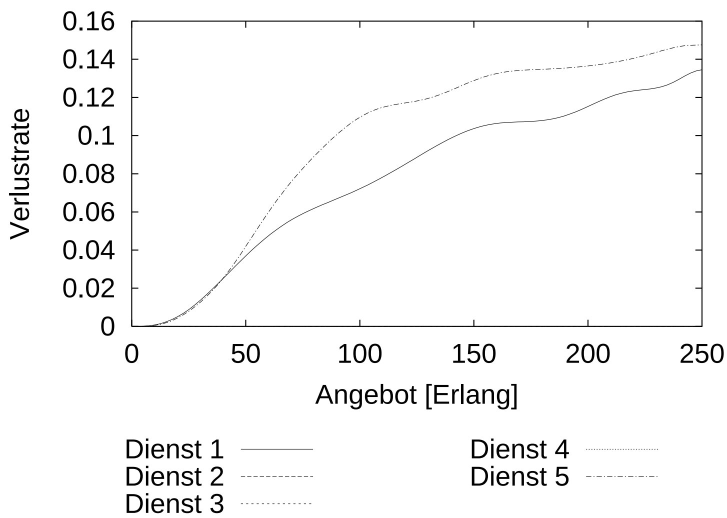



Abbildung E.22: Paketverlustrate

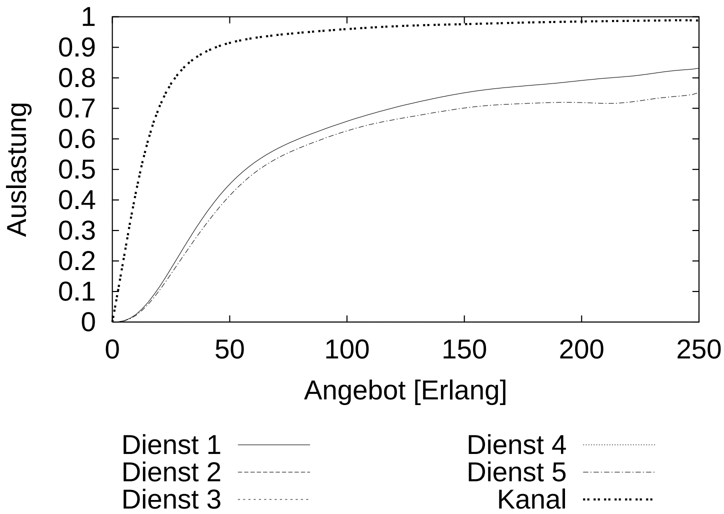



Abbildung E.23: Kanalauslastung

### E.2.7 Funktion 7

| $P_{CLR}$ | | $CLR_{System}$ SK | K | M | G | SG |
|---|---|---|---|---|---|---|
| $CLR_{Dienst}$ | SK | SG | G | M | K | SK |
| | K | SG | G | M | K | SK |
| | M | SG | G | M | K | SK |
| | G | SG | G | M | K | SK |
| | SG | SG | G | M | K | SK |

Tabelle E.1: Regelbasis des Fuzzy Controllers $FC_1$

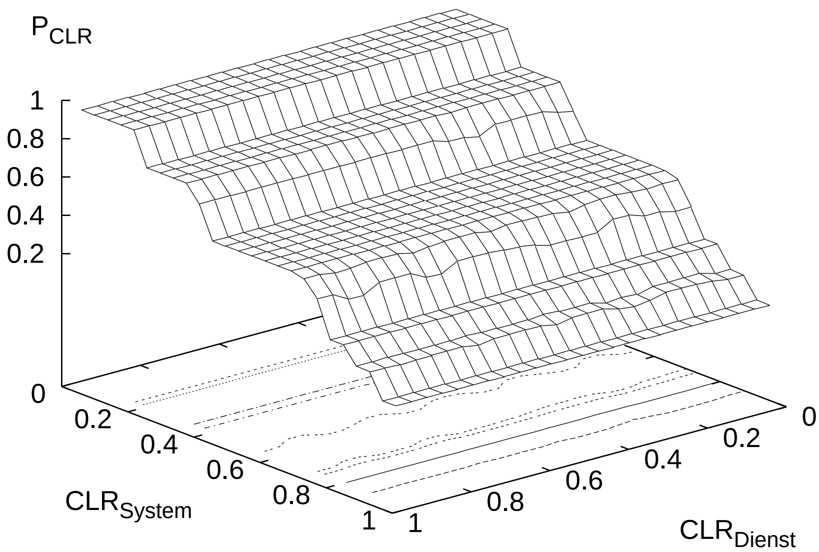



Abbildung E.24: Fuzzy Controllers $FC_1$

| $P_{ATR}$ | | $T_U$ NK | N | K |
|---|---|---|---|---|
| $Q_U$ | SK | SG | SG | G |
| | K | SG | SG | G |
| | M | G | G | K |
| | G | M | M | SK |
| | SG | K | K | SK |

Tabelle E.2: Regelbasis des Fuzzy Controllers $FC_2$

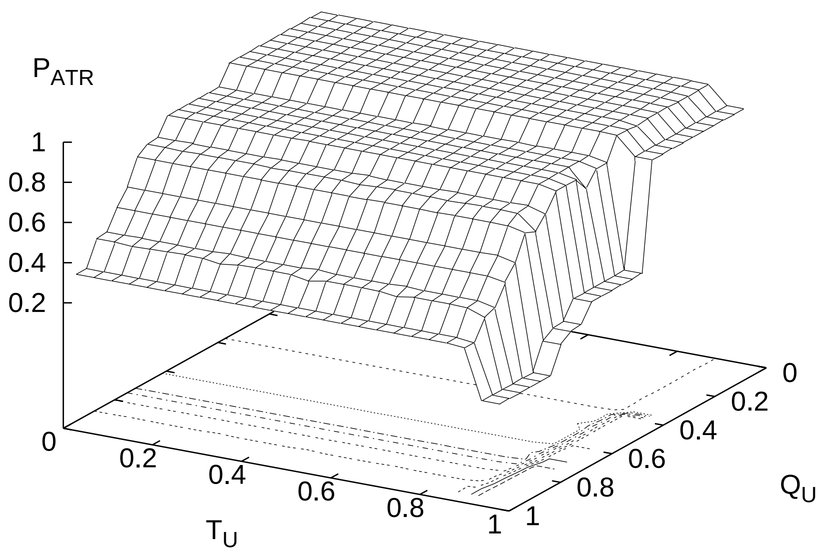



Abbildung E.25: Fuzzy Controllers $FC_2$

| $P_B$ | | $P_{ATR}$ | | | | |
|---|---|---|---|---|---|---|
| | | SK | K | M | G | SG |
| $P_{CLR}$ | SK | SK | SK | SK | SK | SK |
| | K | SK | K | K | K | K |
| | M | SK | K | M | M | M |
| | G | SK | K | M | G | G |
| | SG | SK | K | M | G | SG |

Tabelle E.3: Regelbasis des Fuzzy Controllers $FC_3$

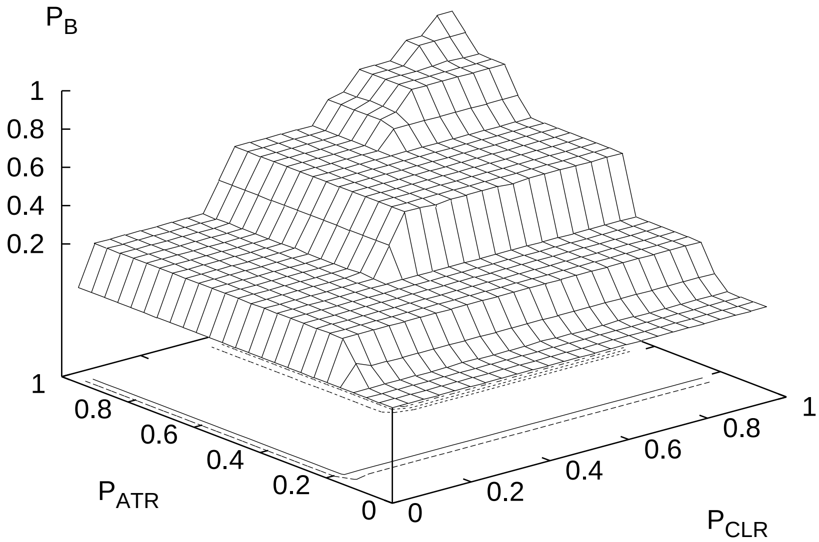



Abbildung E.26: Fuzzy Controllers $FC_3$

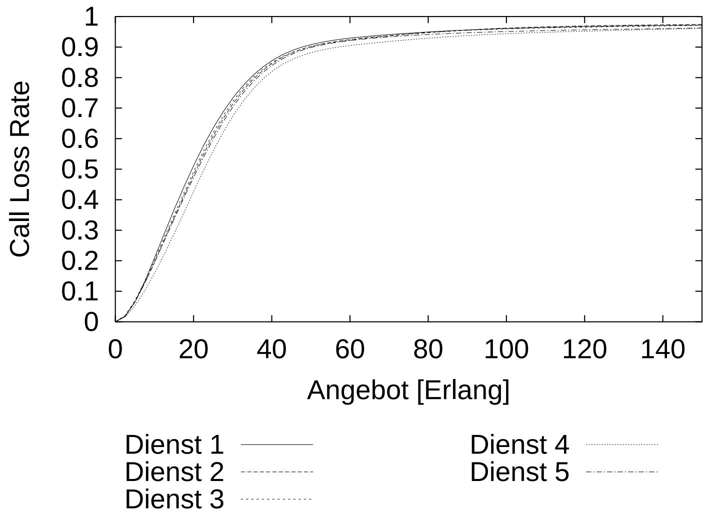



Abbildung E.27: Call Loss Rate

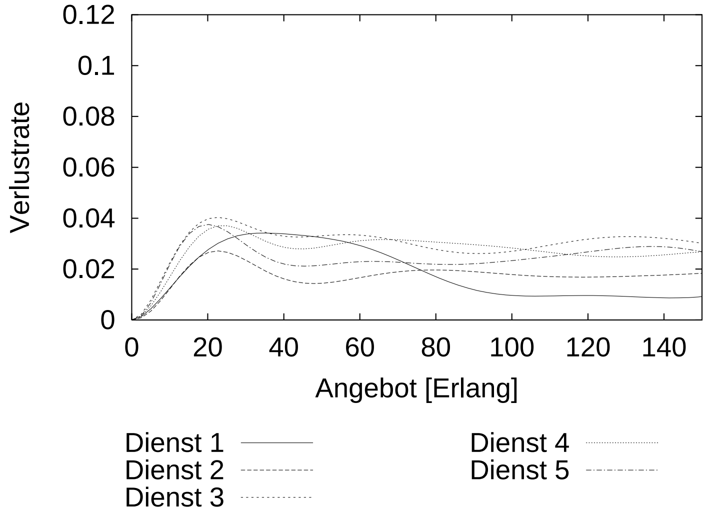



Abbildung E.28: Paketverlustrate

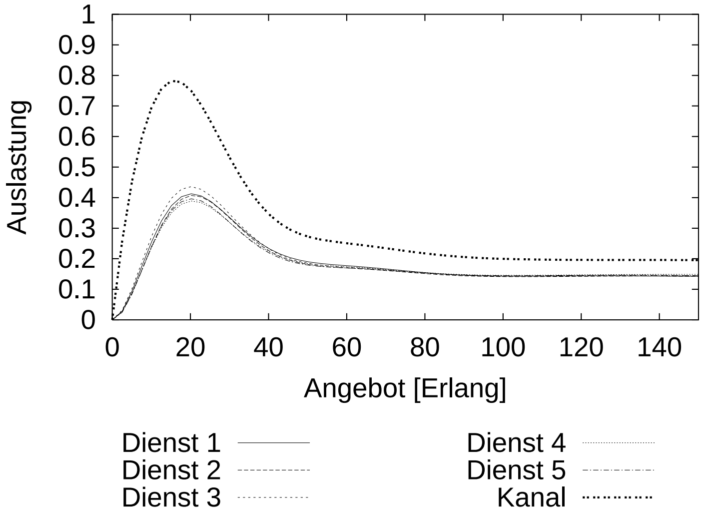



Abbildung E.29: Kanalauslastung

# Anhang F

# Backpropagations-Algorithmus

## F.1 Adaption der Ausgangsgewichtungen

Unter Anwendung der Kettenregel folgt aus Gleichung 8.9:

$$\begin{aligned} \frac{\partial E_m}{\partial w_{ij}} &= \frac{\partial E_m}{\partial net_j} \frac{\partial net_j}{\partial w_{ij}} \\ &= \underbrace{\frac{\partial E_m}{\partial o_j}}_{I} \underbrace{\frac{\partial o_j}{\partial net_j}}_{II} \underbrace{\frac{\partial net_j}{\partial w_{ij}}}_{III} \end{aligned} \tag{F.1}$$

**Term I:** Unter Berücksichtigung von Gl. 8.7 folgt:

$$\frac{\partial E_m}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2} \sum_{k \in Ausgänge} (o_{Soll,k} - o_{Ist,k})^2 \tag{F.2}$$

$$\begin{aligned} \frac{\partial E_m}{\partial o_j} &= \frac{1}{2} \frac{\partial}{\partial o_j} (o_{Soll,j} - o_{Ist,j})^2 \\ &= \frac{1}{2} 2 (o_{Soll,j} - o_{Ist,j}) \frac{\partial (o_{Soll,j} - o_{Ist,j})}{\partial o_j} \\ &= -(o_{Soll,j} - o_{Ist,j}) \end{aligned} \tag{F.3}$$

**Term II:** Mit der Identität als Outputfunktion (Gl. 8.4) und der sigmoiden Aktivierungsfunktion (Gl. 8.3), ergibt sich für den Term $\frac{\partial o_j}{\partial net_j}$ aus Gl. F.1

$$\begin{aligned} \frac{\partial o_j}{\partial net_j} &= \frac{\partial a(net_j)}{\partial net_j} \\ &= o_j(1 - o_j) \end{aligned} \tag{F.4}$$

**Term III:** Term III ergibt sich durch die Ableitung der Input-Funktion 8.1.

$$\begin{aligned} \frac{\partial net_j}{\partial w_{ij}} &= \frac{\partial}{\partial w_{ij}} \sum_{i \in Eing\ddot{a}nge} w_{ij} o_i + \Theta_j \\ &= o_i \end{aligned} \tag{F.5}$$

Substituiert man die Terme I - III in Gleichung F.1, ergibt sich für das Fehlermaß folgender Zusammenhang.

$$\delta = \frac{\partial E_m}{\partial w_{ij}} = -(o_{Soll,j} - o_{Ist,j}) \cdot o_j (1 - o_j) \cdot o_i \tag{F.6}$$

Diese Gleichung ist allerdings nur auf Neuronen der Ausgabeschicht anwendbar, da für die Neuronen der inneren Schichten kein $o_{Soll}$ vorgegeben werden kann.

## F.2 Adaption der Verbindungsgewichte der versteckten Layer

Um das Fehlersignal eines inneren Neurons berechnen zu können, muß diese Größe durch einen äquivalenten Faktor ersetzt werden. Durch Anwendung der Kettenregel kann der Lernfehler eines inneren Neurons rekursiv aus dem Produkt des Fehlersignals nachgeschalteter Neuronen und den zugehörigen Verbindungsgewichten ermittelt werden.

$$\delta_i = -\frac{\partial E_p}{\partial net_i} = \underbrace{\frac{\partial E_p}{\partial o_i}}_{I} \underbrace{\frac{\partial o_i}{\partial net_i}}_{II} \tag{F.7}$$

**Term I:**

$$\frac{\partial E_p}{\partial o_i} = \sum_k \frac{\partial E_P}{\partial net_k} \frac{\partial net_k}{\partial o_i} \tag{F.8}$$

$$\delta_j = -\frac{\partial E_P}{\partial net_j} \tag{F.9}$$

$$\frac{\partial net_k}{\partial o_i} = \frac{\partial}{\partial o_i} \cdot \sum_j w_{ij} \cdot o_i = w_{ij} \tag{F.10}$$

$$\frac{\partial E_P}{\partial o_i} = -\sum_j \delta_j \cdot w_{ij} \tag{F.11}$$

**Term II:** In Anlehnung an die Herleitung der Beziehung Gl. F.4 gilt folgender Zusammenhang

$$\frac{\partial o_j}{\partial net_j} = o_j \cdot (1 - o_j) \tag{F.12}$$

$$\delta = o_j (1 - o_j) \sum_k -\delta w_{kj} \tag{F.13}$$

# Literaturverzeichnis


[1] E. Aboelela and C. Douligeris. Fuzzy multiobjective routing model in B-ISDN. *Computer Communications*, 21:1571–1584, 1998.

[2] M. E. Anagnostou, J. A. Sanchez, and I. S. Venieris. A Multiservice Structured Markovian Traffic Source Model. *0-7803-1825-0/94 IEEE*, pages 1008–1013, 1994.

[3] The ATM Forum. *Traffic Management Specification*, 1999.

[4] J. Aweya and L. Orozco-Barbosa. Neurocontroller for buffer overload control in a packet switch. *IEE Proc.-Commun.*, 145(4):227–233, 1998.

[5] B. Müller and J. Reinhardt. *Neural Networks, An Introduction.* Springer-Verlag, second edition, 1991.

[6] A. G. Barto. Reinforcement learning and adaptive critic methods. In D. A. White and D. A. Sofge, editors, *Handbook of Intelligent Control; Neural, Fuzzy and Adaptive Approaches*, pages 469–491. New York, NY., 1992.

[7] D. Beasley, D. R. Bull, and R. R. Martin. An Overview on Genetic Algorithms: Part 1, Fundamentals. *University Computing*, 15(2), 1993.

[8] H. Benbrahim and J. A. Franklin. Biped dynamic walking using reinforcement learning. *Robotics and Autonomous Systems*, 22:283–302, 1997.

[9] G. Böhme. *Fuzzy-Logik.* Springer, 1993.

[10] F. Borgonovo and L. Fratta. Policing Procedures: Implications, Definitioins and Proposals. In *Teletraffic and Datatraffic in a Period of Change, ITC-13*, pages 859–866, 1991.

[11] H.-H. Bothe. *Fuzzy Logic.* Springer, second edition, 1995.

[12] M. Braae and D. A. Rutherford. Selection of Parameters for a fuzzy logic controller. In *Fuzzy Sets ans Systems 2*, pages 185–199, Tokyo, 1979.

[13] Bronstein and Semendjajew, editors. *Taschenbuch der Mathematik.* Verlag Harri Deutsch, Thun und Frankfurt (Main), 1980.

[14] J. L. Castro, M. Delgado, and F. Herrera. A Learning Method of Fuzzy Reasoning by Genetic Algorithm. In *Proc. of First European Congress on Fuzzy an Intelligent Technologies*, pages 804–809, Aachen, 1993.

[15] V. Catania, G. Ficili, and D. Panno. A fuzzy alogic based approach to multipriority control in ATM networks. *Computer Standards and Interfaces*, (21):19–32, 1999.

[16] V. Catania, G. Ficili, and D. Panno. On the impact of traffic control algorithms on resource management in ATM networks. *computer communications*, (22):258–265, 1999.

[17] H. Chao. Design of Leaky Bucket Access Control Schemes in ATM Networks. In *Proceedings of ICC*, 1991.

[18] R.-G. Cheng and C.-J. Chang. Neural-network connection-admission control for ATM networks. *IEE Proc.-Commun.*, 144(2):227–233, 1997.

[19] L.-D. Chou and J.-L. C. Wu. Bandwidth allocation of virtual paths using neural-network-based genetic algorithms. *IEE Proc.-Commun.*, 145(1):33–39, 1998.

[20] P. K. Dash, H. P. Satpathy, and A. C. Liew. A real-time short-term peak and average load forecasting system using a self-organizing fuzzy neural network. *Engineering Applications of Artificial Intelligence*, (11):307–316, 1998.

[21] F. Denissen, E. Desmet, and G.H. Petit. The Policing Function in an ATM Network. In *Proceedings 1990 Int. Zurich Seminar on Digital Communications*, pages 131–144, 1990.

[22] D. Driankov, H. Hellendoorn, and M. Reinfrank. *An Introduction to Fuzzy Control.* Springer-Verlag, 1993.

[23] E. Nordström and J. Carlström. A Reinforcement Learning Scheme for Adaptive Link Allocation in ATM Networks. Technical report, Department of Computer Systems, Uppsala University, Sweden, 1998.

[24] G. Ficili and D. Panno. A fuzzy algorithm for combined control of traffic parameters: assessment and key issues. *computer communications*, (22):199–210, 1999.

[25] V. S. Frost and B. Melamed. Traffic Modeling For Telecommunications Networks. *IEEE Communications Magazine*, 32(3):70–81, March 1994.

[26] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley Publishing Company, 1998.

[27] J. Hall and P. Mars. Limitations of artificial neural networks for traffic prediction in broadband networks. *IEE Proceedings on Communications*, 147(2):114–118, 2000.

[28] G. Haßlinger and Th. Klein, editors. *Breitband-ISDN und ATM-Netze.* B. G. Teubner, Stuttgart, 1999.

[29] K. Heesche. *Selbstlernende Fuzzy-Systeme mittels neuronaler und genetischer Algorithmen.* PhD thesis, Universität Dortmund, 1996.

[30] J. J. Henry, J. L. Farges, and J. L Gallego. Neuro-fuzzy techniques for traffic control. *Control Engineering Practice*, 6:755–761, 1998.

[31] F. Herrera, M.Lozano, and J. L. Verdegay. Learning and Tuning Fuzzy Contrl Rules using Genetic Algorithms. In *Proc. of 4. Dortmunder Fuzzy Tage*, pages 79 – 91, 1994.

[32] ITU-T Recommendation I.371:. *Traffic control and congestion control in B-ISDN*, 1993.

[33] D. Jensen. Fair Bandwidth Control in ATM-Systems by Artificial Neural Networks. In *ISSLS93.*

[34] D. Jensen. Efficient Training Algorithm for Artificial Neural Networks in ATM-Systems. 1993.

[35] D. Jensen. B-ISDN Network Management by a Fuzzy Logic Controller. In *Proc. of 4. Dortmunder Fuzzy Tage*, pages 255 – 262, 1994.

[36] D. Jensen. B-ISDN Network Management by a Fuzzy Logic Controller. In *IEEE Globecom*, 1994.

[37] D. Jensen. On The Design of a Bandwidth Controller Based on Fuzzy Logic. In *Proc. on the IV Russian-German Seminar on Integrated Networks and Flow Control*, 1994.

[38] D. Jensen. CAC in ATM-Systems by Neural Networks using a Fuzzy Predictor for Reinforcement Parameter Learning. In *EUFIT 97*, 1997.

[39] K. A. De Jong, editor. *Analysis of the Behavior of a Class of Genetic Adaptive Systems.* University of Michigan, 1975.

[40] S.-S. Joo and F. C.-H. Rhee. A Fuzzy Rule Generation Algorithm For A Switching Systems Overload Control. *EUFIT*, pages 1740–1746, 1995.

[41] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[42] J. Kahlert. *Fuzzy Control für Ingenieure.* Vieweg, 1995.

[43] J. Kahlert and H. Frank. *Fuzzy-Logik und Fuzzy-Control.* Vieweg, 1993.

[44] C. Karr and E. J. Gentry. Fuzzy Control of pH Using Genetic Algorithms. *IEEE Transactions on Fuzzy Systems*, 1(1):46–53, 1993.

[45] J. Kinzel, F. Klawonn, and R. Kruse. Anpassung Genetischer Algorithmen zum Erlernen und Optimieren von Fuzzy Reglern. In *Proc. of 4. Dortmunder Fuzzy Tage*, pages 92 – 99, 1994.

[46] R. Kleinewillinghöfer-Kopp and R. Lehnert. ATM Reference Traffic Sources and Traffic Mixes. *RACE 1022, TG III Contribution to the BLNT Workshop*, July 1990.

[47] K. P. Kratzer. *Neuronale Netze.* Hanser, 1990.

[48] K. Kropp and U. G. Baitinger. Optimization of Fuzzy Logic Controller Inference Rules Using a Genetic Algorithm. *Proc. of First European Congress on Fuzzy and Intelligent Technologies*, pages 1090–1096, 1993.

[49] D. Lam, D. C. Cox, and J. Widom. Teletraffic Modeling for Personal Communication Services. *IEEE Communications Magazine*, 35(2):79–87, February 1997.

[50] D. Leitch and P. Probert. Optimization of Fuzzy Logic Controller Inference Rules using a Genetic Algorithm. Technical report, University of Oxford.

[51] C.-T. Lin and M.-C. Kan. Adaptive Fuzzy Command Acquisition with Reinforcement Learning. *IEEE Transactions on Fuzzy Systems*, 6(1):102–121, 1998.

[52] M. Luoni et al. Source Models and Applications for Video. In *Race 1022 Workshop: Traffic and Performance Aspects in IBCN*, Aveiro, Portugal, 1992.

[53] M. Mizumoto. Fuzzy Controls under various fuzzy reasoning methods. In *Proc. Second IFSA Congress*, Tokyo, 1987.

[54] W. M. Moh, M.-J. Chen, N.-M. Chu, and C.-D. Liao. Traffic prediction and dynamic bandwidth allocation over ATM: a neural network approach. *computer communications*, 18(8):563–571, 1995.

[55] R. J. T. Morris and B. Samadi. Neural Network Control of Communications Systems. *IEEE Transactions on Neural Networks*, 5(4):639–650, 1994.

[56] S. Naughton, P. Cunningham, and F. Somers. Asynchronous transfer mode traffic modelling and dimensioning using artificial neural networks. *Engineering Applications of Artificial Intelligence*, (12):321–342, 1999.

[57] C. H. Ng, L. Bai, and B. H. Soong. Modelling multimedia traffic over ATM using MMBP. *IEE Proceedings on Communications*, 144(5):307–310, October 1997.

[58] Q. Pang and S. Cheng. A Novel Fuzzy Priority Manager in ATM Networks. *XVI World Telecom Congress Proceedings*, pages 59–63, 1997.

[59] Y.-K. Park and G. Lee. NN Based ATM Cell Scheduling with Queue Length-Based Priority Scheme. *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, 15(2):261–270, 1997.

[60] D.T. Pham and P.T.N. Pham. Artificial intelligence in engineering. *International Journal of Machine Tools & Manufacture*, 39:937–949, 1999.

[61] S. M. Prabhu and D. P. Garg. Fuzzy-logic-based reinforcement learning of admittance control for automated robotic manufacturing. *Engineering Applications of Artificial Intelligence*, 11:7–23, 1998.

[62] E. Rathgeb. Modeling and Performance Comparison of Policing Mechanisms for ATM Networks. *IEEE Journal on Selected Areas in Communications*, 9(3), 1991.

[63] I. S. Reljin. Neural Network Based Cell Scheduling in ATM Nodes. *IEEE Communications Letters*, 2(3):78–80, 1998.

[64] Q. Ren and H. Kobayashi. Diffusion Approximation Modeling for Markov Modulated Bursty Traffic and its Applicatioons to Bandwidth Allocation in ATM Networks. *IEEE Journal on Selected Areas in Communications*, 16(5):679–691, June 1998.

[65] Q. Ren and G. Ramamurthy. A Real-Time Dynamic Connection Admission Controller Based On Traffic Modeling, Measurement, and Fuzzy Logic Control. *IEEE Journal on Selected Areas in Communications*, 18(2):184–196, 2000.

[66] R.Palm, D. Driankov, and H. Hellendoorn. *Model Based Fuzzy Control*. Springer-Verlag, 1996.

[67] R. Schaphorst. *Videoconferencing and Videotelephony*. Artech House, 1996.

[68] Prof. Dr. Ing. R. G. Schehrer. Vorlesung Vermittlungssysteme I, 2000.

[69] Prof. Dr. Ing. R. G. Schehrer. Vorlesung Vermittlungssysteme II, 2000.

[70] K. Sriram and W. Whitt. Characterizing Superposition Arrival Processes in Packet Multiplexers for Voice and Data. *IEEE Journal on Selected Areas of Communication*, 4(6):833–846, 1986.

[71] R. S. Sutton. Learning to Predict by the Methods of Temporal Differences. *Machine Learning*, 3:9–44, 1988.

[72] N. Swaminathan, J. Srinivasan, and S. V. Raghavan. Bandwidth-demand prediction in virtual path in ATM networks using genetic algorithms. *computer communications*, 22:1127–1135, 1999.

[73] K. Uehara and K. Hirota. Fuzzy Connection Admission Control for ATM Networks Based on Possibility Distribution of Cell Loss Ratio. *IEEE Journal on Selected Areas in Communications*, 15(2):179–190, 1997.

[74] B. Warfield and P. Sember. Prospects for the Use of Artificial Intelligence in Real-Time Network Traffic Management. *Computer Networks and ISDN Systems*, 20:163–169, 1990.

[75] P. J. Werbos. A Menu of Designs for Reinforcement Learning Over Time. In W. T. Miller, R. S. Sutton, and P. J. Werbos, editors, *Neural Networks for Control*, chapter 3, pages 1571–1584. Cambridge: MIT Press, 1990.

[76] R. J. Williams. A Class of Gradient-Estimating Algorithms for Reinforcement Learning in Neural Networks. volume 2, pages 601–608, San Diego, CA, 1987.

[77] R. J. Williams. Simple statistical gradient-following algorithms foe connectionist reinforcement learning. volume 8, pages 229–256, 1992.

[78] M. Woodruff and R. Kositpaiboon. Multimedia Traffic Management Principles for Guaranteed ATM Network Performance. *IEEE Journal on Selected Areas in Communications*, 8(3), 1990.

[79] H. Xu, C. M. Kwan, L. Hayes, and J. D. Pryor. Real-time adaptive on-line traffic incident detection. *computer communications*, 93:173–183, 1998.

[80] M. H. Yaghmaee, M. Safavi, and M. B. Menhaj. An intelligent usage parameter controller based on dynamic rate leaky bucket for ATM networks. *Computer Networks*, 32:17–34, 2000.

[81] S. Yazid and H.T. Mouftah. Congestion Control Methods for BISDN. *IEEE Communications Magazine*, 1992.