

# Signalflußsimulation von Überlastkontroll-Mechanismen für den TCP-Verkehr

Gerhard Bergholz, Dresden

Partiell gefördert durch das Projekt INTERMON

Mit diesem Beitrag werden Signalflußmodelle für die Simulation von Überlastkontroll-Mechanismen des TCP-Verkehrs entwickelt. Die Grundlage für die Signalflußsimulation bilden Flußmodelle, die mathematisch mit Hilfe von Differentialgleichungen dargestellt werden können. Untersuchungsgegenstand der Signalflußsimulation ist das dynamische Verhalten der Signale und dessen Auswirkungen auf die Systemleistung. Im Mittelpunkt der Betrachtungen stehen die prinzipiellen Implementierungsmöglichkeiten der Signalflußmodelle für die Überlastkontrollsysteme und erste Simulationsergebnisse über die Wirkung der Überlastkontrolle. Dabei wird SIMULINK in Verbindung mit MATLAB als Werkzeug genutzt. In diesem Beitrag werden ein erweitertes Stationsmodell und Modelle der beiden Überlastkontrollmechanismen TD (Tail-Drop) und RED (Random Early Detection) entwickelt und der Einfluß der Überlastkontrolle auf die Beseitigung des Staukollapses und die damit verbundenen annähernden Sicherung des Nettodurchsatzes in Höhe der Stationskapazität dargestellt.

## 1 Vorbetrachtungen

Während am Anfang der Nutzung des Internet die Überlastkontrolle (Congestion Control) wegen der geringen Größe und Belastung des Netzes nicht beachtet werden mußte, wurde Ende der 80er Jahre, nachdem Staukollaps aufgetreten war, die Überlastkontrolle eingeführt. Ab Anfang der 90er Jahren gewannen, mit dem stürmisch wachsenden Umfang des Internet, die Anforderungen an die Überlastkontrolle immer mehr an Bedeutung.

Der Internetverkehr kann vor allem durch Überlastkontroll-Mechanismen wie Tail-Drop-Überlastkontrolle und verschiedener AQM-Mechanismen (AQM - Active Queue Management) überwacht und gesteuert werden. Das Ziel der Überlastkontrolle besteht darin, im Fall der Überlastung einer Leitung im Inneren des Netzes, die Rate der gesendeten Daten an die augenblickliche Lastsituation durch die Änderung der Staufenstergröße anzupassen und damit den Staukollaps zu vermeiden..

Dabei erfolgt diese Anpassung bei Tail-Drop (TD) direkt als Funktion der durch die endliche Puffergröße der Engpaßbedienstation bewirkten Verlustrate. Traditionell sind Internet-Router mit Tail-Drop ausgestattet. Im Gegensatz zu Tail-Drop wird beim AQM nicht erst gewartet bis durch die endliche Puffergröße Pakete verloren gehen, sondern es werden vor Erreichen der Puffergröße durch die Warteschlangenlänge zusätzlich Pakete markiert, deren Markierung als Grundlage für die Absenkung der Datenübertragungsrate genutzt werden. Am bekanntesten ist hier der RED-Mechanismus (RED - Random Early Detection), bei dem die markierten Pakete weggeworfen werden. Der

RED-Algorithmus wurde entwickelt um das Tail-Drop zu ersetzen und steht heute in kommerziell verfügbaren Routern bereit.

Im Zusammenhang mit der Entwicklung der verschiedenen Überlastkontrollalgorithmen findet seit einiger Zeit eine umfangreiche Forschung auf dem Gebiet der Überlastkontrolle (Congestion Control) für den Entwurf immer besserer Überlastkontroll-Mechanismen und deren optimale Parameterfestlegung statt. Eine große Anzahl der Untersuchungen wurden mit dem ereignisdiskreten Simulator ns durchgeführt (s. [4], [16], [17], [18], [21], [22], [23] und [24]).

Für die oben genannte Klasse der Netzanwendung wurden parallel dazu neue Wege der Modellierung und Simulation gesucht. Ein solcher Weg besteht darin, einen Wechsel vom traditionellen Paradigma der ereignisorientierten Simulation zu einem Paradigma vorzunehmen, das in der Literatur mit dem Begriff *Flußmodelle* (s. [6], [12] [13] und [14]) verbunden wird. Ein wesentlicher Unterschied zur herkömmlichen Betrachtungsweise besteht darin, daß nicht die Zwischenankunftszeiten und Bedienzeiten, sondern die Ankunftsraten und Bedienraten als Zufallsgrößen eingeführt werden. Dabei treten die Pakete als Forderungen explizit im Modell nicht auf. Der Weg besteht hierbei darin, die Ereignisorientierung der Simulation zu verlassen. Bei der Untersuchung der Überlastkontroll-Mechanismen spielen dynamische Probleme, wie die Stabilität und das Schwingen im stationären Zustand eine wesentliche Rolle. Deshalb bietet sich die Anwendung von Flußmodellen in Form von Differentialgleichungen für die Analyse der Überlastkontrolle an. In der Literatur wurden für RED vor allem analytische regelungstechnische Untersuchungen mit Hilfe von Flußmodellen durchgeführt(s.[8], [9], [10], [11], [15], [16] und [20]).

In diesem Beitrag wird die vom Autor eingeführte Signalflußsimulation (s. [1]), die auf der Grundlage von Flußmodellen (s. [3], [5], [6], [12], [13], [14] und [19]) entwickelt wurde, für die Untersuchung von Überlastkontrollsystemen angewendet. Dabei erfolgt die Signalflußsimulation mit Hilfe von SIMULINK (s. [7]). In [1] wurde als Modellierungsgegenstand eine Bedienstation betrachtet. Jetzt geht es darum dieses Modell auf die Modellierung des Überlastkollapses und der verschiedenen Formen der Überlastkontrolle zu erweitern. Gegenüber den Ansätzen von Holot et al [9] wird ein genaueres Modell für die Warteschlangenlänge, die Durchsatzrate und die Droprate auf der Grundlage der Ergebnisse von [1] für die Modellierung der Überlastkontrolle eingeführt und der Zusammenhang mit den Senderaten hergestellt..

Ausgangspunkt unserer Betrachtungen ist eine Netzwerkkonfiguration entsprechend Bild 1. Wir betrachten ein System mit  $N$  Verkehrsflüssen  $F_i$  ( $i=1,2,\dots,N$ ) durch die gemeinsame Verbindung  $BC$  mit der Kapazität  $C$ . Die Verkehrsflüsse finden zwischen den Hosts  $A_i$  und  $D_i$  statt und transportieren Daten in eine Richtung von  $A_i$  nach  $D_i$ . Der Verkehr in umgekehrter Richtung besteht nur aus ACKs. Wir nehmen an, daß alle Verbindungen  $A_iB$  und  $CD_i$  genügend Kapazität besitzen, so daß die Verbindung  $BC$  der einzige Flaschenhals (Bottleneck) für die Verkehrsflüsse ist.

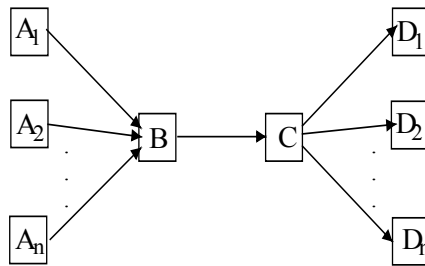


Bild 1: Grundkonfiguration des betrachteten Computernetzes

Nachfolgend betrachten wir die Signalfußsimulation des erweiterten Stationsmodells und der beiden Überlastkontroll-Mechanismen TD und RED (Random Early Detection). Im Abschnitt 2 wird das Einklassenmodell der Engpaßbedienstation, im Abschnitt 3 das erweiterte Stationsmodell der Engpaßbedienstation, im Abschnitt 4 das Signalfußmodell des Tail-Drop-Systems und im Abschnitt 5 das Signalfußmodell des RED-Regelungssystems eingeführt. Abschnitt 6 enthält Simulationsergebnisse für die Überlastkontrolle.

## 2 Einklassen-Stationsmodell

### 2.1 Warte-Verlustsystem

In der Konfiguration des Bildes 1 betrachten wir die Engpaßleitung zusammen mit den Puffern im entsprechenden Router B als Bedienstation.

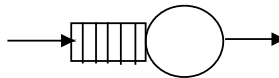


Bild 2: Grundmodell der Bedienstation

Dabei interpretieren wir zunächst die Bedienstation als Warte-Verlustsystem (s. Bild 2). Wir betrachten jetzt das Signalfußmodell der Bedienstation. In diesem Flußmodell sind folgende Zufallsprozesse vorhanden:  $A(t)$  – Ankunftsratenprozeß,  $B(t)$  – Bedienratenprozeß,  $G(t)$  – Verlustratenprozeß und  $D(t)$  – Abgangsratenprozeß. Außerdem ändert sich der innere Zustand  $Z(t)$ , der als aktuelle Warteschlangenlänge oder Pufferinhalt interpretiert wird.

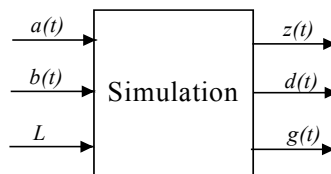


Bild 3: Zusammenhang zwischen den Realisierungen der Zufallsprozesse bei der Simulation

Bei der Simulation müssen Realisierungen der oben genannten Zufallsprozesse und der Zusammenhang zwischen diesen nachgebildet werden. Die Realisierungen werden dementsprechend mit  $a(t)$ ,  $b(t)$ ,  $z(t)$ ,  $d(t)$  und  $g(t)$  bezeichnet. Dabei fassen wir die Ankunftsratenprozesse und Bedienratenprozesse unter dem Namen Eingangsprozesse zusammen. Die Zusammenfassung der Zustandsprozesse, Abgangsratenprozesse und Verlustratenprozesse wird Ausgangsprozesse genannt. In Bild 3 ist der Zusammenhang zwischen den Realisierungen der Eingangsprozesse und Ausgangsprozesse dargestellt, wobei die konstante Puffergröße  $L$  mit eingezeichnet wurde.

Bei dieser Betrachtungsweise treten im Rahmen der Simulation von stochastischen Prozessen auf der Simulationsebene nur Realisierungen auf und keine wahrscheinlichkeitstheoretischen Kenngrößen. Für den Fall, daß es keine zufälligen Eingangsgrößen gibt, kann dieser Modellansatz auch für deterministische Modelle genutzt werden. In diesem Fall verwandeln sich die Realisierungen in deterministische Signale.

## 2.2 Gleichungen des Einklassen-Stationsmodells

Für das Einklassen-Stationsmodell können wir die mathematischen Beziehungen des Signalflußmodells einer Bedienstation [1] nutzen. In [1] wird eine Bedienstation mit einem endlichen Puffer  $P$  als Warteraum für die wartenden Forderungen und einem Einfachserver  $S$  für deren Bedienung betrachtet. Zum Zeitpunkt  $t$  hat der Ankunftsratenprozeß am Eingang des Puffers die Rate  $a(t)$  und wird mit der Bedienrate  $b(t)$  im Server entladen. Der aktuelle Pufferinhalt (Warteschlangenlänge), der als Zustand angesehen werden kann, wird mit  $z(t)$  bezeichnet. Die Pufferkapazität (Warteraumgröße) heißt  $L$ . Die Abflußrate vom Server nach außen wird mit  $d(t)$  bezeichnet und die Verlustrate wird  $g(t)$  genannt. Die Zeitvariable  $t$  soll sich im Intervall  $[0 T]$  befinden. Es wird angenommen, daß zu Beginn der Simulation der Puffer leer ist, was  $z(0)=0$  bedeutet

Des weiteren gilt für ein Signalflußmodell die Annahme, daß die Größen  $a(t)$ ,  $b(t)$ ,  $z(t)$  und  $g(t)$  stückweise kontinuierlich und differenzierbar sind. Die entsprechenden Gleichungen wurden in [1] in Anlehnung an [11] und [16] eingeführt. Das Signalflußmodell der Bedienstation in [1] besteht aus drei Teilmodellen: dem Warteschlangenmodell (dort auch als Zustandsmodell bezeichnet), dem Verlustratenmodell und dem Durchsatzratenmodell (dort auch als Abgangsratenmodell bezeichnet).

Für das *Warteschlangenmodell* (Zustandsmodell) gilt nach [1]

$$\frac{dz}{dt} = \begin{cases} 0 & \text{if } z(t) = 0 \text{ and } a(t) - b(t) \leq 0 \text{ or } (z(t) = L \text{ and } a(t) - b(t) \geq 0) \\ a(t) - b(t) & \text{else} \end{cases} \quad (1)$$

Für das *Verlustratenmodell* übernehmen wir aus [1]

$$g(t) = \begin{cases} a(t) - b(t) & \text{if } z(t) = L \text{ and } a(t) - b(t) \geq 0 \\ 0 & \text{else} \end{cases} \quad (2)$$

Für das *Durchsatzratenmodell* (Abgangsratenmodell) gilt nach [1]

$$d(t) = \begin{cases} \min\{a(t), b(t)\} & \text{if } z(t) = 0 \\ b(t) & \text{else} \end{cases} \quad (3),$$

Diese Beziehung geht als Bestandteil in das erweiterte Durchsatzraten-Modell ein.

In [1] wurde gezeigt, wie für die Gleichungen (1), (2) und (3) ein SIMULINK-Signalfußplan implementiert werden kann.

In diesem Modellansatz wurden die einzelnen Verkehrsflüsse nicht unterschieden. Es handelt sich um ein Einklassenmodell.

### 2.3 Zusammenhang zwischen Bedienrate und Kanalkapazität

Wir nehmen jetzt zur Modellvereinfachung an, daß die Bedienrate  $b$  konstant ist und wir können zwischen der Bedienrate und der Kanalkapazität der Engpaßressource den allgemein bekannten Zusammenhang mit der Formel

$$b = C / (8 * Pl)$$

herstellen. Dabei gelten die Bezeichnungen  $C$  – Kanalkapazität [Mb/s],  $b$  – Bedienrate [ Pakete/s] und  $Pl$  –Paketlänge [ Byte].

## 3 Erweitertes Stationsmodell

Der wichtigste Unterschied des erweiterten Stationsmodell gegenüber dem Einklassen-Stationsmodell des Abschnittes 2 besteht in der Berücksichtigung der wiederholten Übertragungen und der damit verbundenen Betrachtung der einzelnen Verkehrsflüsse.

### 3.1 Verkehrsfluß-Submodell

Bei der Berücksichtigung der wiederholten Übertragungen sind im Gegensatz zum Einklassen-Stationsmodell die einzelnen Verkehrsflüsse zu betrachten und es ist ein Mehrklassenabschnitt (multiples Submodell) einzuführen.

#### 3.1.1 Ankunftsratenmodell eines Verkehrsflusses

Es werden jetzt die Wiederholraten für jeden Verkehrsfluß betrachtet. Dabei gilt für die Wiederholrate  $u_i(t)$  für einen Verkehrsfluß

$$u_i(t) = g_i(t - r_i(t)) \quad (4)$$

mit  $g_i(t)$  als Verlustrate eines Verkehrsflusses  $F_i$ . Die Größe  $r_i(t)$  ist hier die Umlaufzeit (round trip time) des entsprechenden Verkehrsflusses.

Für die Wiederholung der Übertragungen gilt dann folgender Zusammenhang der

$$a_i(t) = q_i(t) + u_i(t) \quad (5)$$

Dabei ist  $q_i(t)$  die *Senderate* und  $a_i(t)$  die *Ankunftsrate* des  $i$ -ten Verkehrsflusses. Diese Beziehung wird *Ankunftsratenmodell* eines Verkehrsflusses der erweiterten Bedienstation genannt.

Für die Senderaten aller Verkehrsflüsse nehmen wir der Bestimmtheit halber ON/OFF-Prozesse oder zeitinvariante deterministische Signale an..

### 3.1.2 Bestimmung der Umlaufzeit

In (4) wird die Umlaufzeit als Eingangssignal genutzt. Wir spezifizieren jetzt die Umlaufzeit.

Die Umlaufzeit setzt sich einerseits aus der Verweilzeit  $v_0(t)$  der Pakete im Abschnitt BC und andererseits aus der Verweilzeit  $v_{ie}(t)$  der Pakete in den Abschnitten  $A_b$  und  $CD_i$  zusammen. Dabei ist zu beachten, daß die Stationsverweilzeit  $v_0(t)$  im Abschnitt BC durch das Modell der Bedienstation bestimmt wird, dabei zeitabhängig und für alle Verkehrsflüsse gleich ist. Es gilt die Abhängigkeit von der Warteschlangenlänge  $z(t)$  und der Bedienrate  $b$  für die Stationsverweilzeit

$$v_0(t) = z(t)/b.$$

Wir bezeichnen die Verweilzeit der Pakete in den Abschnitten  $A_iB$  und  $CD_i$  als *Verzögerungszeit* (propagation delay). Dann ergibt sich für die Umlaufzeit eines Verkehrsflusses

$$r_i(t) = v_i + z(t)/b. \quad (i=1, \dots, N) \quad (6)$$

Für die Beziehungen (6), (7) und (8) kann ein Ankunftsratenmodell als SIMULINK-Signalflußplan implementiert werden.

Das erweiterte Stationsmodell wird zunächst auf einer Ebene betrachtet, auf dem alle Verkehrsflüsse zusammengefaßt sind. Es besteht aus dem multiplen Submodell, dem Warteschlangenmodell und dem erweiterten Durchsatzratenmodell. In Bild 6 ist der Signalflußplan des erweiterten Stationsmodells angegeben.

Das bedeutete, daß alle Umlaufzeiten einen wesentlichen Anteil (nämlich die Stationsverweilzeit) enthalten, der zwar immer zeitabhängig aber für alle Verkehrsflüsse gleich ist. Demgegenüber ist die Verzögerungszeit i.a. für die verschiedenen Verkehrsflüsse unterschiedlich. Im deterministischen Fall ist die Verzögerungszeit jedes Verkehrsflusses zeitlich konstant und im stochastischen Fall nehmen wir an, daß sie stationär ist. In manchen Untersuchungen wird in der Literatur die gesamte Umlaufzeit vorgegeben. Dadurch entstehen Fehler struktureller Natur. Deshalb wird von uns der Ansatz (8) berücksichtigt und die Verzögerungszeit vorgegeben. Beim stochastischen Fall nehmen wir einen stochastischen ON/OFF-Prozeß mit exponentielle Verteilungen der ON- und OFF-Phasen an. Dieser wird in MATLAB erzeugt und über den Block ‚From Workspace‘ in SIMULINK bereitgestellt.

### 3.1.3 Verlustratenmodell eines Verkehrsflusses

Außerdem gehört das Verlustratenmodell zum multiplen Submodell. Im Abschnitt 2.2 wurde das Verlustratenmodell für das Einklassen-Stationsmodell eingeführt. Man kann jetzt ein *Verlustratenmodell* für jeden Verkehrsfluß  $F_i$  ( $i=1, 2, \dots, N$ ) mit

$$g_i(t) = \begin{cases} a_i(t) - b & \text{if } z = L \text{ and } (a_i(t) - b) \geq 0 \\ 0 & \text{else} \end{cases} \quad (7)$$

einführen Dieser Ansatz wird für das multiple Submodell gebraucht. Es gilt dabei der Zusammenhang

$$g(t) = \sum_{i=1}^N g_i(t). \quad (8)$$

Die Gleichung (7) wurde als SIMULINK-Signalflußplan implementiert.

### 3.1.4 Signalflußplan des Verkehrsfluß-Submodells

Das Ankunftsratenmodell bildet für jeden Verkehrsfluß  $F_i$  zusammen mit dem Verlustratenmodell ein Submodell, das wir *Verkehrsfluß-Submodell* der erweiterten Bedienstation nennen. In Bild 8 ist der entsprechende Signalflußplan angegeben. Dabei enthält das Ankunftsratenmodell die Beziehungen (4), (5) und (6) und das Verlustratenmodell die Gleichung (7).

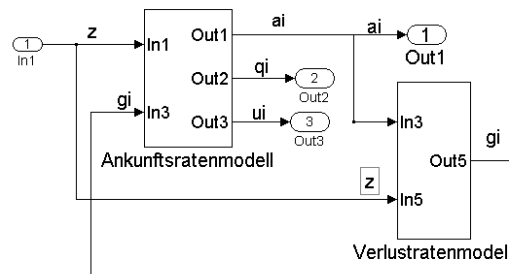


Bild 4: Signalflußplan für das Submodell des Verkehrsflusses  $F_i$

Das Verkehrsfluß-Submodell kann jetzt in einem Block entsprechend Bild 4 zusammengefaßt werden

## 3.2 Multiples Submodell

Als nächstes betrachten wir das multiple Submodell für  $N$  Verkehrsflüsse.

Man kann jetzt die entsprechenden Größen für das den Verkehrsflüssen übergeordnete System einführen mit  $q$  als Quellrate des Systems,  $a$  als Ankunftsrate des Systems und  $u$  als Wiederholrate des Systems. Dabei gelten die Beziehungen

$$u = \sum_{i=1}^N u_i, \quad (9)$$

$$q = \sum_{i=1}^N q_i \quad (10)$$

und

$$a = \sum_{i=1}^N a_i. \quad (11)$$

Auf dieser Grundlage führen wir das *multiple Submodell* ein, wobei die Ankunftsraten, Quellraten und Wiederholraten einbezogen werden. Bild 5 enthält den entsprechenden Signalflußplan des multiplen Submodells für  $N$  Verkehrsflüsse.

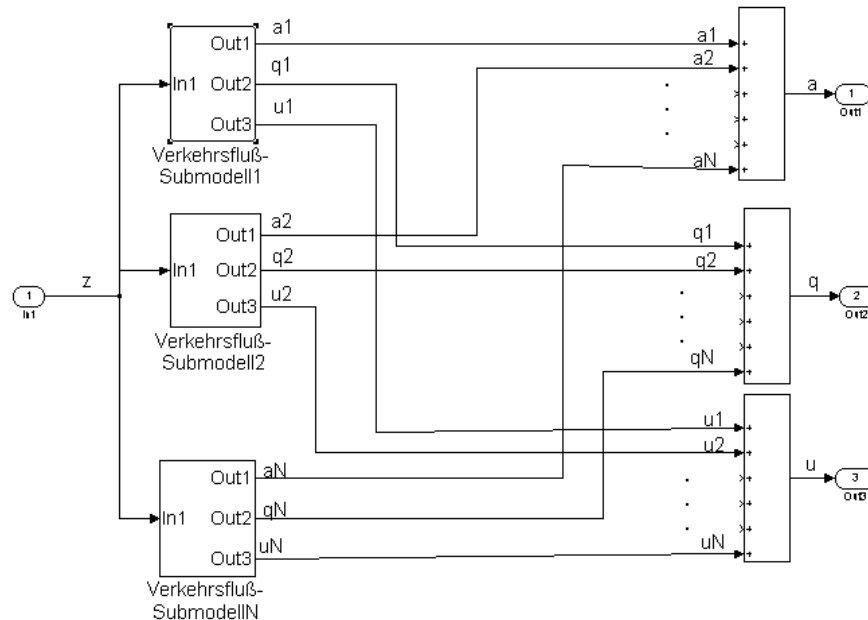


Bild 5: Signalflußplan des multiplen Submodells mit  $N$  Verkehrsflüssen

### 3.3 Erweitertes Durchsatzratenmodell

Die Durchsatzrate ist eine wichtige Kenngröße der Leistungsbewertung. Für die Bewertung der Wirkung des Staukollapses und der Wirksamkeit der Überlastkontrolle ist sie die entscheidende Bewertungsgröße. In (5) ist die Beziehung für die Durchsatzrate  $d(t)$  angegeben. Dabei ist  $d(t)$  die Bruttodurchsatzrate. Sie enthält implizit auch die Übertragungswiederholungen, die wir als Overhead ansehen müssen. Wichtig ist hier aber die Nettodurchsatzrate, d.h. der Anteil der Senderate der wirklich übertragen wird.

Für die Bestimmung der Nettodurchsatzrate führen wir den Wirkungsgrad  $\eta(t)$  mit

$$\eta(t) = q(t) / a(t)$$

ein. Dann erhalten wir die Nettodurchsatzrate  $d_n(t)$  mit

$$d_n(t) = \eta(t) * d(t).$$

Das Modell, daß die Bestimmung der Durchsatzrate und der Nettodurchsatzrate enthält heißt *erweitertes Durchsatzratenmodell*.

### 3.4 SIMULINK-Signalflußplan des erweiterten Stationsmodells

Die Teilmodelle multiples Submodell, Durchsatzratenmodell und Warteschlangenmodell können zu einem erweiterten Stationsmodell zusammengefaßt werden. Bild 10 enthält die Darstellung des hierarchischen Aufbaus für das Stationsmodell.

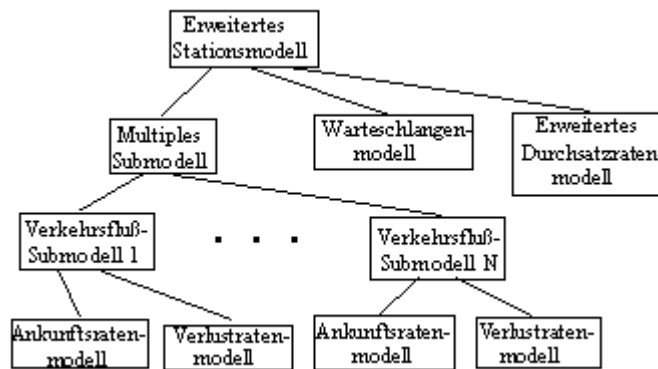


Bild 6 : Hierarchischer Aufbau des Stationsmodells

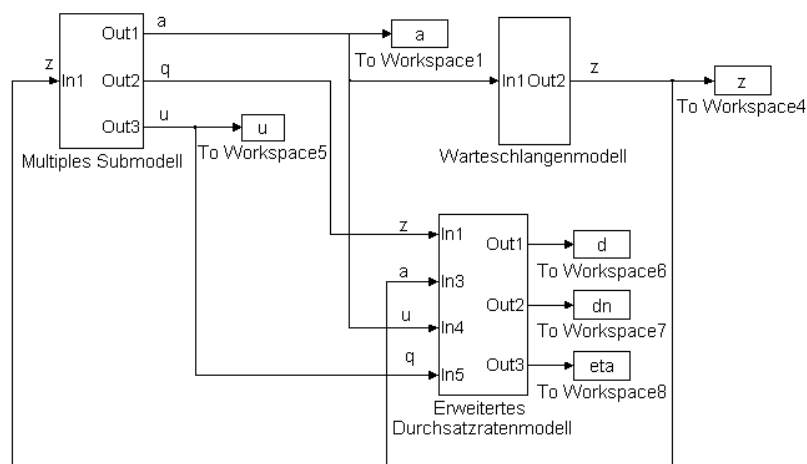


Bild 7: Signalflußplan des erweiterten Stationsmodells

Das Warteschlangenmodell kann aus dem Stationsmodell mit (1) übernommen werden. Das erweiterte Durchsatzratenmodell wurde im Abschnitt 3.3 und das multiple Submodell im Abschnitt 3.2 eingeführt.

### 3.5 Staukollaps-Analyse

Der Staukollaps kann mit Hilfe des erweiterten Stationsmodells analysiert werden. Wir führen jetzt den Verkehrswert  $\rho$  des Stationsmodells mit

$$\rho = E(q)/E(b)$$

ein. Dabei ist  $E(q)$  der Mittelwert der Quellrate und  $E(b)$  der Mittelwert der Bedienrate. Für eine zeitlich invariante Bedienrate gilt

$$E(b) = b.$$

Der Verkehrswert des Stationsmodells kann als Kenngröße für die Systembelastung genutzt werden. In Anlehnung an die herkömmliche Bedienungstheorie können wir jetzt die Hypothese aufstellen, daß bei

$$\rho \geq 1 \tag{12}$$

eine Überlastung des Systems vorhanden ist und bei

$$\rho < 1 \quad (13)$$

sich das System im stationären Gleichgewicht befindet. Diese Hypothese wird durch die Ergebnisse der Signalflußsimulation bestätigt. Uns interessiert vor allem der Fall mit Überlast (s. Ungleichung (12)). Dabei haben wir hier für die einzelnen Verkehrsflüsse ON/OFF-Prozesse für die Quellraten (s. [2]) und  $N=10$  für die Anzahl der Verkehrsflüsse angenommen.

Wir führen jetzt die Untersuchung bei  $\rho=1.4$  durch. Das Bild 8 stellt den Verlauf der Durchsatzraten dar, wobei die grüne Kurve die Bruttodurchsatzrate angibt. Das Bild 8 zeigt deutlich die Stauwirkung, die im erweiterten Stationsmodell bei Überlast eintritt. Im Laufe der Zeit wächst der Wiederholanteil der konstanten Bruttodurchsatzrate an. Dementsprechend sinkt die Nettodurchsatzrate mit der Zeit. Wenn der Ungleichgewichtszustand entsprechend (12) eine längere Zeit anhält, verschwindet der Nettodurchsatz praktisch und wir haben es mit einem echten Stau zu tun. Wie Untersuchungen mit verschiedenen Senderatenprozessen (deterministische Prozesse, ON-OFF-Prozesse) hängt dieses Ergebnis unwesentlich vom Charakter der Senderatenprozesse ab.

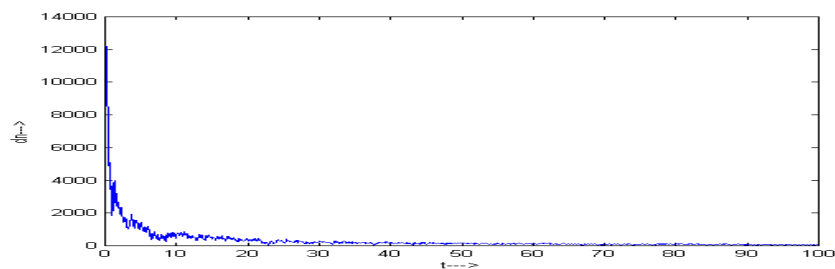


Bild 8: Realisierung der Nettodurchsatzrate bei Überlast

Für die Warteschlangenlänge ergibt sich nach dem Übergangsprozeß ein konstanter Verlauf infolge der Begrenzung durch die Puffergröße mit  $z=L = 4286$ .

### 3.6 Verhalten des erweiterten Stationsmodells im Gleichgewichtszustand

Wir betrachten jetzt das erweiterte Stationsmodell im Gleichgewichtszustand, indem wir  $\rho=0.8$  annehmen (entspricht der Ungleichung (13)). In diesem Fall erhalten wir für den Nettodurchsatz das in Bild 9 dargestellte Simulationsergebnis.

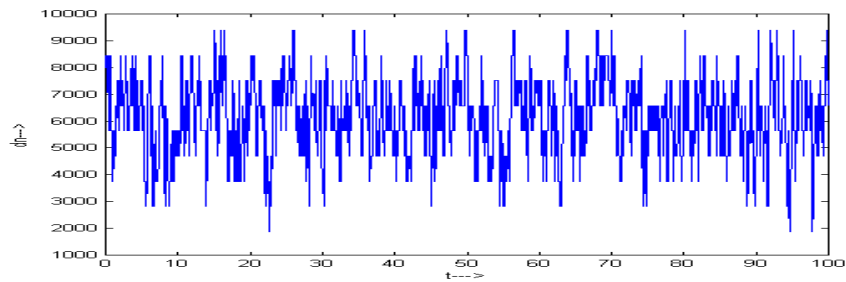


Bild 9: Zeitlicher Verlauf der Nettdurchsatzrate im Gleichgewichtszustand

In diesem Fall ist keine Warteschlange vorhanden und es gilt für die Warteschlangenlänge nach der Beendigung des Übergangsprozesses  $z = 0$ .

### 3.7 Einfluß der Arbeitslast auf die Nettdurchsatzrate

Wir betrachten jetzt die Abhängigkeit der mittleren Nettdurchsatzrate  $D_n = E(d_n)$  vom Verkehrswert  $\rho$  (s. Bild 10). Es zeigt sich deutlich bei einer Arbeitslast von  $\rho = 0.8$  ein Maximum für die mittlere Durchsatzrate.

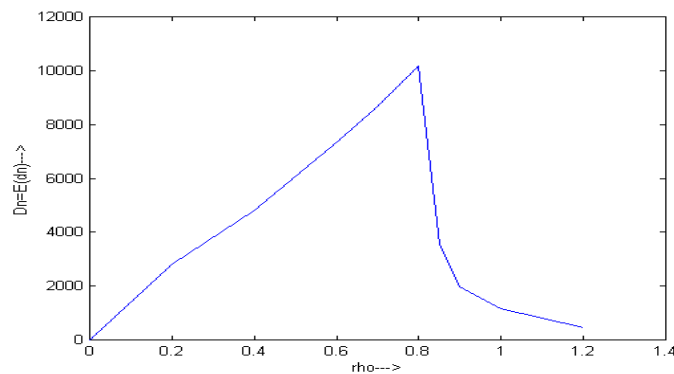


Bild 10: Abhängigkeit der mittleren Nettdurchsatzrate vom Verkehrswert für das erweiterte Stationsmodell

## 4 Signalflußmodell des Tail-Drop-Systems

Die Analyse des Staukollapses zeigt, daß dieser unbedingt vermieden werden muß. Man kann den Staukollaps verhindern, wenn es bei vorhandener Überlast gelingt die wirksame Senderate zu senken. Diesem Ziel dient die Überlastkontrolle mit Tail-Drop und RED.

In diesem Abschnitt wird das Signalflußmodell des Tail-Drop-Systems eingeführt und im Abschnitt 5 das Signalflußmodell des RED-Regelkreises. Das erweiterte Stationsmodell stellt das Modell für die Engpaßstation dar, wenn keine Überlastkontrolle vorhanden ist. Durch die Berücksichtigung der Überlastkontrolle kommen jetzt Modellkomponenten dazu und die bisherigen Modellgleichungen müssen teilweise umgeformt werden.

Das Tail-Drop-System besteht vor allem aus dem AIMD-Mechanismus (Active Increase Multiplicative Decrease) und RED enthält den AIMD-Mechanismus.

## 4.1 Signalflußmodell für die fenstergestützte Überlastkontrolle

### 4.1.1 Staufenstermodell

Wir betrachten jetzt den AIMD-Mechanismus und führen für diesen ein Flußmodell ein. Dabei vernachlässigen wir die Slow-Start-Phase. Unter dieser Bedingung besteht der AIMD-Mechanismus eines Verkehrsflusses aus der zeitlich wiederholten Folge einer AI-Phase und einer MD-Phase. Zu beachten ist, daß dies für jeden Verkehrsfluß gilt. Das Staufenstermodell beschreibt die Veränderung der Staufenstergröße  $w_i(t)$  ( $i=1,2,\dots,N$ ). Dabei kann die Änderung der Staufenstergröße, die wir mit  $dw_i(t)/dt$  bezeichnen, in der AI-Phase proportional dem Reziprokwert der Umlaufzeit  $1/r_i(t)$  und in der MD-Phase proportional dem Produkt aus halber Staufenstergröße und um die Umlaufzeit verzögerter Droprate (Absenkrate)  $-(w_i(t)/2)*s_i(t-r_i(t))$  angesetzt werden. Die Umlaufzeit  $r_i(t)$  und die Droprate  $s_i(t)$  sind dabei i. a. ebenfalls für jeden Verkehrsfluß unterschiedlich. Auf dieser Grundlage ergibt sich die Gleichung für das AIMD-Staufenstermodell

$$dw_i(t)/dt = \begin{cases} 1/r_i(t) & \text{if } s_i(t-r_i(t)) \\ -(w_i(t)/2)*s_i(t-r_i(t)) & \text{else} \end{cases} \quad (14)$$

Hierbei gelten die Bezeichnungen  $r_i(t)$  – Umlaufzeit und  $s_i(t)$  – Droprate für den TCP-Fluß  $F_i$  ( $i=1,2,\dots,N$ ).

Von Misra et al [13] wurde ein anderer Ansatz gemacht, der unter Berücksichtigung der von uns eingeführten Begriffe die Form

$$dw_i(t)/dt = 1/r_i(t) - (w_i(t)/2)*s_i(t-r_i(t))$$

annimmt. Wie praktische Analysen zeigen, kann dieser Ansatz als Approximation der Gleichung (14) angesehen werden. Wir verwenden hier den Ansatz (14).

In (14) muß für die Bestimmung der Staufenstergröße die Droprate als Eingangssignal berücksichtigt werden, die für TD und RED unterschiedlich ist. Für das Tail-Drop-System ist die Droprate jedes Verkehrsflusses gleich der Verlustrate

$$s_i(t) = g_i(t) \quad (i=1,2,\dots,N) \quad (17)$$

die mit (3) bereits angegeben wurde. Unter Berücksichtigung von (4) kann jetzt auf der rechten Seite von (14) an die Stelle von  $s_i(t-r_i(t))$  die Wiederholrate  $u_i(t)$  gesetzt werden.

Dieser Ansatz geht von der Annahme aus, daß die Verlustrate richtig erfaßt wird. Für das RED-System muß bei der Droprate zusätzlich zur Verlustrate die Wegwerfrate berücksichtigt werden. Aus den Gleichungen (6) und (14) wurde der entsprechende SIMULINK-Signalflußplan implementiert.

### 4.1.2 Ankunftsratenmodell des Tail-Drop-Systems

Wir müssen jetzt das Ankunftsratenmodell verändern, indem wir den AIMD-Fenstermechanismus berücksichtigen. Der Fenstermechanismus bewirkt eine Ankunftsrate von  $w_i(t)/r_i(t)$ . Diese ist aber nur wirksam, wenn sie größer als die Summe von Sende- und Wiederholrate  $q_i(t) + u_i(t)$  ist. Andernfalls wirkt die Ankunftsrate  $q_i(t) + u_i(t)$ . Somit gilt

$$a_i(t) = \begin{cases} w_i(t) / r_i(t) & \text{if } (w_i(t) / r_i(t)) < (q_i(t) + u_i(t)) \\ q_i(t) + u_i(t) & \text{else} \end{cases} \quad (15)$$

Falls die erste Zeile der Formel (15) gilt, hängt die Ankunftsrate nicht von der Senderate sondern nur vom Fenstermechanismus ab. Die erste Zeile der Formel (15) bewirkt deshalb, daß in diesem Fall nicht mehr die volle Senderate wirksam wird. Genau das ist der Effekt der TD-Überlastkontrolle. Und das zu berücksichtigen führen wir die *wirksame Senderate* ein. Die wirksame Senderate kann jetzt mit

$$e_i(t) = \begin{cases} a_i(t) - u_i(t) & \text{if } (w_i(t) / r_i(t)) < (q_i(t) + u_i(t)) \\ q_i(t) & \text{else} \end{cases} \quad (16)$$

angesetzt werden.

Aus den Beziehungen (4), (15) und (16) kann der SIMULINK-Signalflußplan für das Ankunftsratenmodell implementiert werden. Dabei wurde die Bestimmung der Umlaufzeit ähnlich wie beim erweiterten Stationsmodell in das Ankunftsratenmodell mit aufgenommen.

### 4.1.3 Verkehrsfluß-Submodell der Überlastkontrolle

Die fenstergestützte Überlastkontrolle findet innerhalb des Verkehrsfluß-Submodells statt. Im Verkehrsflußsubmodell für Tail-Drop sind jetzt das Staufenstermodell des Abschnittes 4.1.1, das Ankunftsratenmodell des Abschnittes 4.1.2 und das Verlustratenmodell des Abschnittes 3.1.3 als Knoten enthalten. Dementsprechend erhalten wir ein Verkehrsfluß-Submodell wie in Bild 12.

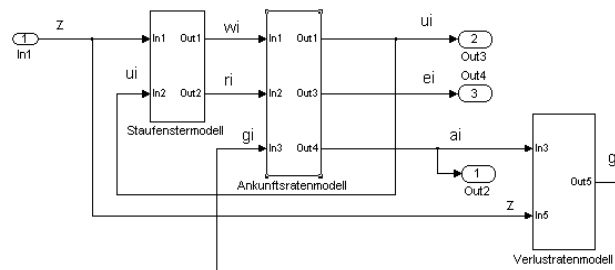


Bild 12: Verkehrsfluß-Submodell für den Verkehrsfluß  $F_i$

#### 4.1.4 Multiples Submodell des Tail-Drop-Systems

Für alle TCP-Flüsse zusammen erhalten wir auf dieser Grundlage ein multiples Submodell, das aus den  $N$  Verkehrsfluß-Submodellen besteht.

In Bild 5 ist bereits für ein multiples Submodell des erweiterten Stationsmodell mit  $N$  Verkehrsflüssen der Signalflußplan angegeben. Beim Tail-Drop-Modell gilt das gleiche multiple Submodell. Allerdings unterscheiden sich die Verkehrsfluß-Submodelle des Tail-Drop-Systems von den Verkehrsfluß-Submodellen des erweiterten Stationsmodells. (vergleiche Bild 12 mit Bild 4)

#### 4.2 Gesamtaufbau des Tail-Drop-Systems

Auf der Grundlage der bisherigen Betrachtungen können wir jetzt den SIMULINK-Signalflußplan des Gesamtmodells für TD aufbauen. In Bild 13 ist der hierarchische Aufbau des Tail-Drop-Modells dargestellt. Dieser Aufbau bildet die Grundlage für den SIMULINK-Signalflußplan. Dabei wurden die SIMULINK-Signalflußpläne für das Staufenstermodell, für das Verlustratenmodell, für das Warteschlangenmodell für das erweiterte Durchsatzratenmodell, für das multiple Submodell und für das Gesamtmodell bereits im vorangehenden Abschnitt eingeführt.

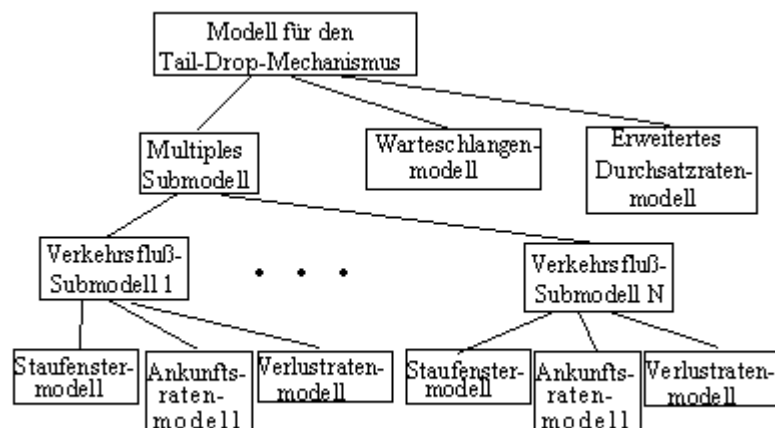


Bild 13: Hierarchischer Aufbau des SIMULINK-Signalflußplans

Der Signalflußplan der nächsten Ebene unterhalb der Wurzelebene besteht aus dem multiplen Submodell, dem Warteschlangenmodell und dem erweiterten Durchsatzratenmodell. In Bild 14 ist der Signalflußplan des erweiterten Stationsmodells angegeben. Der Unterschied zum Gesamtsignalflußplan des erweiterten Stationsmodells besteht darin, daß das multiple Submodell inhaltlich anders aufgebaut ist und daß dabei an die Stelle der Quellrate  $q$  die wirksame Quellrate  $e$  tritt.

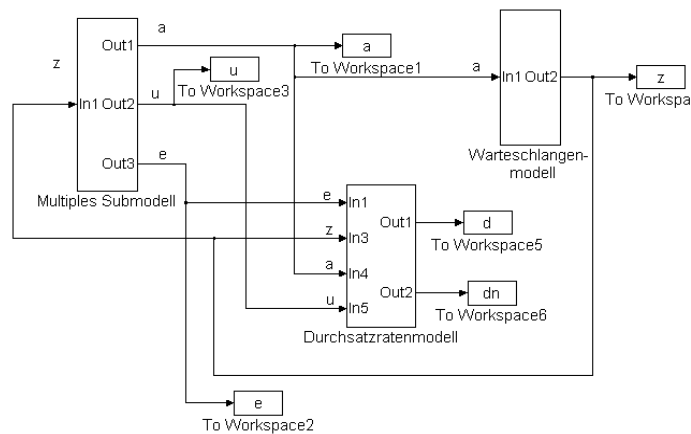


Bild 14: Gesamt-Signalflußplan des Tail-Drop-Modells

## 5 Signalflußmodell des RED-Regelkreises

Das Modell der RED-Überlastkontrolle führt zu einem Regelungssystem mit Rückkopplung. Dabei können die Komponenten des Tail-Drop-Modells als Elemente des Regelstreckenmodell genutzt werden. Das RED-System kann als eine Erweiterung des Tail-Drop-Systems betrachtet werden, bei dem zusätzlich zu den Komponenten des Regelstreckenmodells: multiples Submodell, Warteschlangenmodell und erweitertes Durchsatzratenmodell noch das RED-Regler-Modell kommt.

In Bild 21 ist der SIMULINK-Signalflußplan des RED-Regelkreises dargestellt.

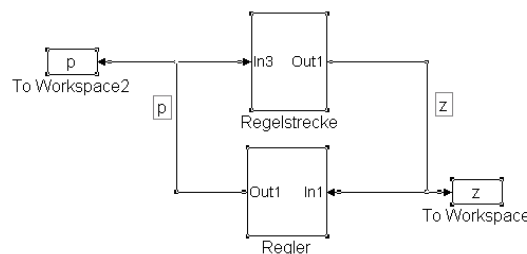


Bild 21: Signalflußplan des RED-Regelkreises

### 5.1 Reglermodell

Wir gehen jetzt zum Reglermodell von RED über. Das Reglermodell der RED-Regelung besteht aus zwei Teilen

- aus dem Block für die gleitende Mittelung
- und dem Block zur Bildung der RED-Drop-Funktion

Dementsprechend erhalten wir den SIMULINK-Signalflußplan des Reglermodell in Bild 22.

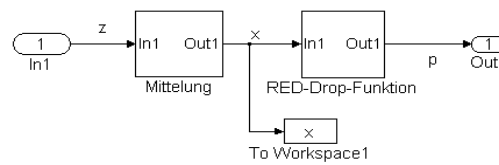


Bild 22: Signalflußplan des Reglermodells

### 5.1.1 Gleitende Mittelwertbildung

Als erstes betrachten wir den Block für die gleitende Mittelwertbildung. Dieser entsteht aus dem diskreten Algorithmus für die gleitende Mittelwertbildung

$$x((k+1)\delta) = (1-\alpha) * x(k\delta) + \alpha * z(k\delta).$$

Wir erhalten jetzt daraus das kontinuierliche Approximationsmodell

$$T_m * (dx(t)/dt) = x(t) + z(t)$$

mit

$$T_m = -\delta / \ln(1-\alpha)$$

Für dieses Modell kann man im Bildbereich der Laplace-Transformation die Übertragungsfunktion

$$F(s) = 1 / (T_m * s + 1)$$

angeben. Es handelt sich hier um ein Tiefpaßfilter, der als SIMULINK-Signalflußplan implementiert wurde.

### 5.1.2 Bildung der RED-Drop-Funktion des Reglers

Die RED-Drop-Funktion bildet den nichtlinearen statischen Block des Reglers mit dem mathematischen Ansatz

$$p(x) = \begin{cases} 0 & \text{if } 0 \leq x < m_i \\ \frac{x - m_i}{m_a - m_i} p_{ma} & \text{elseif } m_i \leq x \leq m_a \\ 1 & \text{else} \end{cases}.$$

In Bild 31 ist die RED-Drop-Funktion graphisch dargestellt.

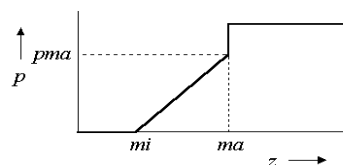


Bild 31: RED-Drop-Funktion

Auch hier wurde der SIMULINK-Signalflußplan für die RED-Drop-Funktion implementiert.

## 5.2 Regelstreckenmodell

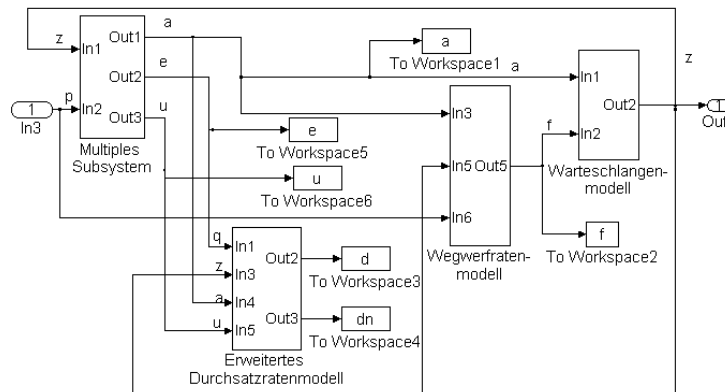


Bild 23: Signalflußplan des Regelstreckenmodells

Auf der Grundlage der für das Tail-Drop-Modell eingeführten Submodelle können wir jetzt den SIMULINK-Signalflußplan des Streckenmodells entsprechend Bild 23 darstellen. Innerhalb dieses Modells ändert sich das erweiterte Durchsatzratenmodell gegenüber dem TD-Modell nicht. Das geänderte multiple Submodell, das geänderte Warteschlangenmodell und das Wegwerfratenmodell wird nachfolgend beschrieben.

### 5.2.1 Multiples Submodell bei RED

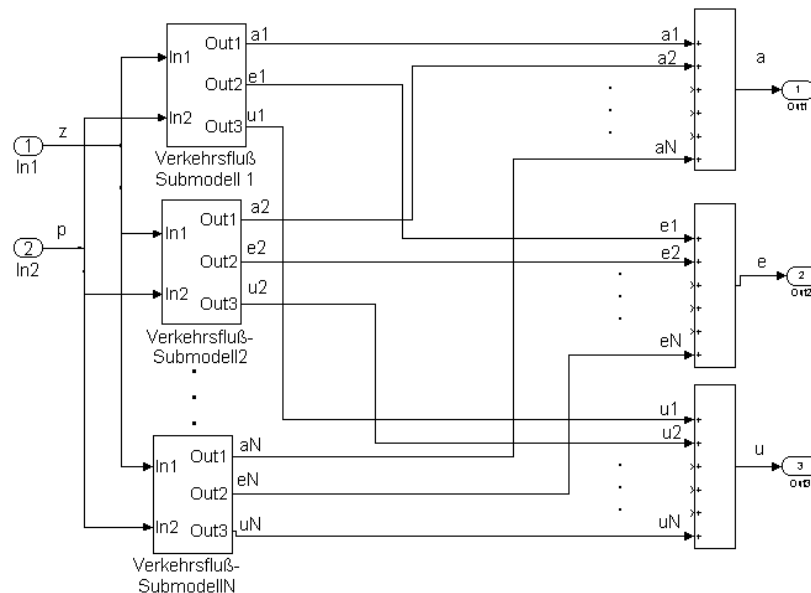


Bild 24: Signalflußplan des multiplen Submodells

Wir können jetzt die SIMULINK-Signalflußpläne für das multiple Submodell für RED einführen. Für das multiple Submodell erhalten wir den Signalflußplan in Bild 24. Die einzelnen Knoten sind wieder Verkehrsfluß-Submodelle. Der Unterschied zum multiplen Submodell des Bildes 5 besteht darin, daß die Verkehrsfluß-Submodelle

anders aufgebaut sind und dementsprechend einen zweiten Signaleingang mit dem Signal  $p$  haben.

### 5.2.2 Verkehrsfluß-Submodell für RED

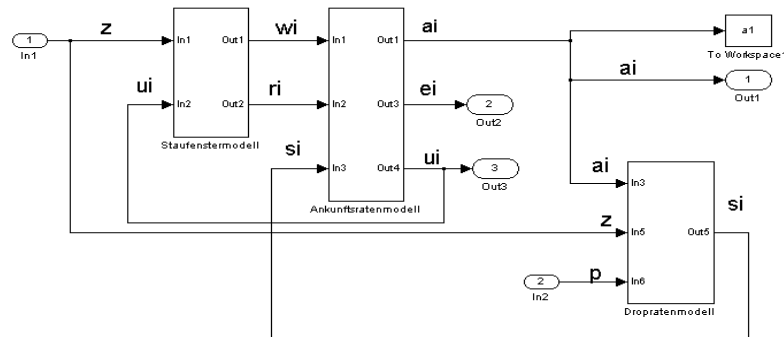


Bild 25: Signalflußplan des Verkehrsfluß-Submodells bei RED

Das Verkehrsfluß-Submodell von RED nimmt jetzt die Form des Bildes 25 an.

Im Verkehrsfluß-Submodell kann das gleiche Staufenstermodell genutzt werden wie bei Tail-Drop. Der strukturelle Aufbau des Ankunftsratemodells ändert sich nicht. Dabei tritt aber an die Stelle des Eingangs  $g_i(t)$  der Eingang  $s_i(t)$ . Das Verlustratemodell wird jetzt durch das Dropratenmodell ersetzt, das nachfolgend eingeführt wird.

### 5.2.3 Dropratenmodell für RED

Für das Dropratenmodell gilt der Ansatz

$$s_i(t) = \begin{cases} a_i(t) - b/N & \text{if } (z = L) \text{ and } ((a_i(t) - b/N) \geq 0) \\ a_i(t) * p(t) & \text{else} \end{cases} \quad (14)$$

Ähnlich wie beim Verlustratemodell, wird jetzt davon ausgegangen, daß sowohl die Verlustrate als auch die Wegwerfrate richtig erfaßt werden und als Droprate auftreten. Der Unterschied zum Verlustratemodell in Gleichung (3) besteht darin, daß auf der rechten Seite in der zweiten Zeile nicht Null sondern  $a_i(t) * p(t)$  tritt. Dabei ist in der ersten Zeile der rechten Seite die Wirkung der endlichen Puffergröße und in der zweiten Zeile der rechten Seite die Wirkung des RED-Reglers über die RED-Drop-Funktion enthalten. Die RED-Drop-Funktion  $p(t)$  befindet sich als Steuergröße am Ausgang des Reglers. Als Regelgröße wird die Warteschlangenlänge  $z(t)$  verwendet. Das Dropratenmodell für RED unterscheidet sich also vom Verlustratemodell für TD dadurch, daß die Wirkung des RED-Reglers dazu kommt.

Für diese Gleichung wurde der SIMULINK-Signalflußplan implementiert.

### 5.2.4 Warteschlangenmodell für RED

Auf die Warteschlangenlänge hat bei RED nicht nur die Ankunftsrate und Bedienrate Einfluß, sondern auch die Wegwerfrate. Deshalb muß das Warteschlangenmodell hier verändert und das Wegwerfratenmodell eingeführt werden.

Wir formen jetzt die Gleichung (1) für das Warteschlangenmodell in eine für unsere Zwecke geeignetere Form um. Dabei nutzen wir die Tatsache, daß die Wegwerfrate die gleiche Wirkung auf den Behälterstand  $z$  hat wie die Bedienrate. Dementsprechend erhalten wir anstelle von (1) für das Warteschlangenmodell

$$dz(t)/dt = \begin{cases} 0 & \text{if } ((z(t) = 0) \text{ and } (a(t) - b(t) - f(t) \leq 0)) \\ & \text{or } ((z(t) = L) \text{ and } (a(t) - b(t) - f(t) \geq 0)). \\ a(t) - b(t) - f(t) & \text{else} \end{cases} \quad (6)$$

Für diese Beziehung wurde der entsprechende SIMULINK-Signalflußplan implementiert.

### 5.2.5 Wegwerfratenmodell für RED

Als Bestandteil von RED tritt im Warteschlangenmodell die Wegwerfrate  $f$  des Systems auf, die sich aus der Wegwerfrate der Verkehrsflüsse  $f_i(t)$  Dabei gilt

$$f = \sum_{i=1}^N f_i(t).$$

Die Wegwerfrate  $f$  beeinflusst also die Warteschlangenlänge. Dabei gilt bei RED

$$f(t) = \begin{cases} 0 & \text{if } z(t) = L \text{ and } ((a(t) - b) \geq 0) \\ p * a(t) & \text{else} \end{cases} \quad (6a)$$

Der entsprechende Signalflußplan wurde in SIMULINK implementiert.

## 5.3 Gesamtaufbau des RED-Regelkreismodells

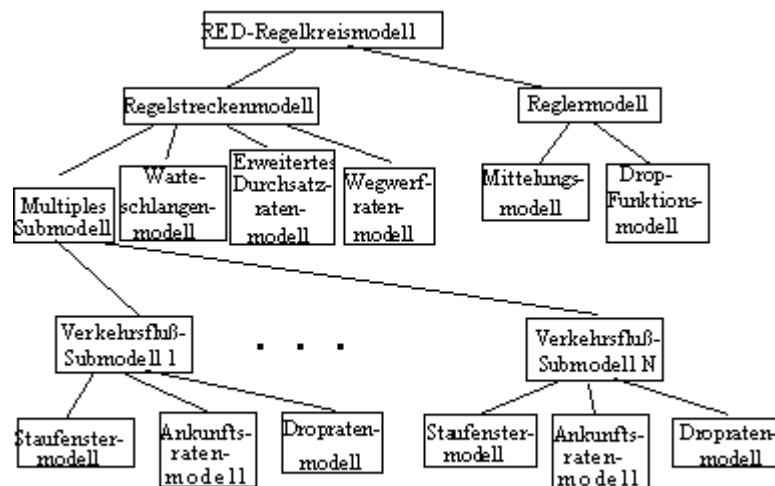


Bild 26: Hierarchischer Aufbau des RED Regelkreismodells

Das RED-Regelkreismodell ebenso wie die beiden bereits eingeführten Modelle hierarchisch aufgebaut werden, wobei das Bild 26 diesen Aufbau angibt.

## 6 Simulationsergebnisse der Überlastkontroll-Modelle

Wir haben Simulationsexperimente mit dem TD- und dem RED-Modell durchgeführt und die nachfolgenden Ergebnisse erzielt. Dabei haben wir für die Übertragungszeiten der verschiedenen Verkehrsflüsse zeitinvariante Signale angenommen. Für die Senderaten der Verkehrsflüsse haben wir zwei Klassen betrachtet: deterministische Senderaten und stochastische ON/OFF-Senderaten. Das Ziel dieser Untersuchungen ist es, die qualitative Wirkung der Überlastkontrolle aufzuzeigen. Da wir die Parameter für RED willkürlich gewählt haben, sollten diese Ergebnisse nicht zum Vergleich zwischen TD und RED herangezogen werden.

### 6.1 Überlastverhalten

Wir betrachten jetzt das Überlastverhalten von TD und RED. Dabei geht es um die Nettodurchsatzrate und die Warteschlangenlänge. Wir nehmen hier  $\rho = 5$  an.

#### 6.1.1 Nettodurchsatzrate für TD und RED

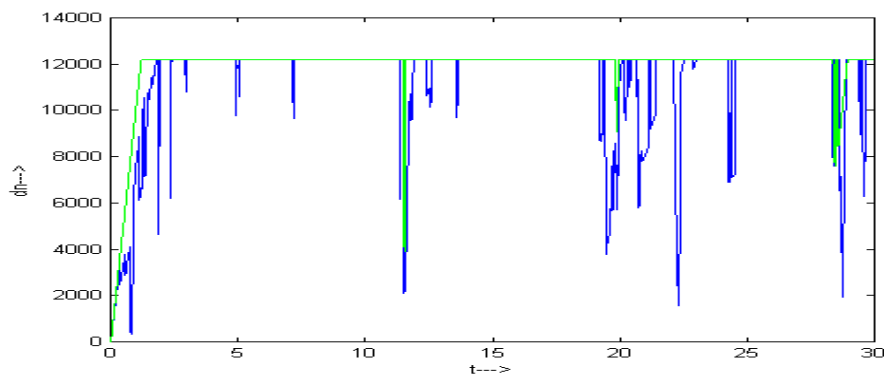


Bild 27 : Nettodurchsatzrate bei der TD-Überlastkontrolle ( $\rho=5$ )

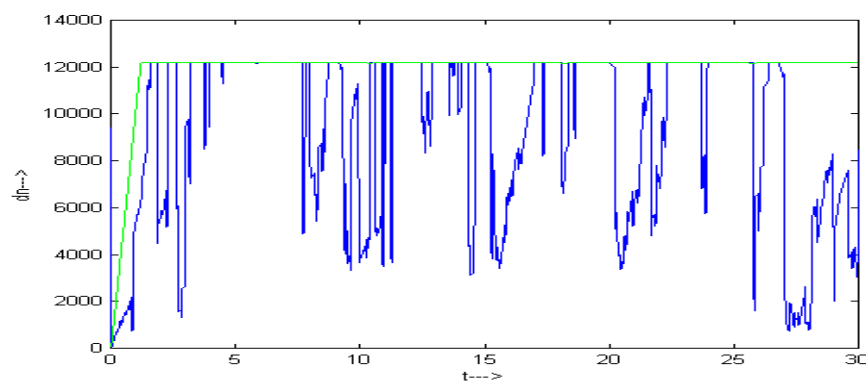


Bild 28 : Nettodurchsatzrate bei der RED-Überlastkontrolle ( $\rho=5$ )

In den Bildern 27 und 28 ist die Nettodurchsatzrate sowohl für stochastische ON/OFF-Sendeprozesse (blau) als auch für deterministische Senderaten. (grün) bei der TD- und RED-Überlastkontrolle eingetragen. Es zeigt sich, daß die Nettodurchsatzrate in beiden

Fällen nahezu der Bedienrate  $b=12207$  Pakete/s entspricht. Das bedeutet daß die Kanalkapazität auch bei Überlast ( $\rho > 1$ ) im Gegensatz zum Fall ohne Staukontroll-Mechanismen praktisch voll ausgeschöpft wird. Verloren geht vor allem der Anteil der Senderate, der über die Kanalkapazität hinaus geht.

### 6.1.2 Warteschlangenlänge für TD und RED

In den Bildern 31 und 32 sind die Warteschlangenlängen für die TD- und RED-Überlastkontrolle aufgezeichnet. Hier entsprechen wieder die blauen Kurven dem Modell mit stochastischen ON/OFF-Sendeprozessen und die grünen Kurven dem Modell mit deterministischen Sendeprozessen. In beiden Fällen oszillieren die Warteschlangenlängen. Die Oszillation bei den Modellen mit deterministischen Sendeprozessen wird durch die Taildrop-Raten hervorgerufen. Bei den Modellen mit stochastische ON/OFF-Sendeprozessen kommen zusätzlich zum den Einfluß der Taildrop-Raten noch der Einfluß der ON/OFF-Sprünge.

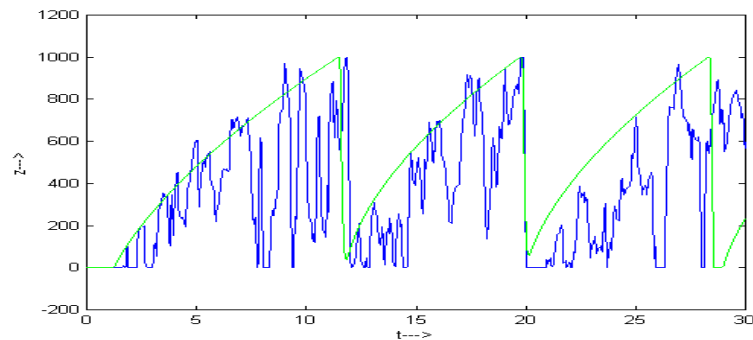


Bild 31: Warteschlangenlänge bei der TD-Überlastkontrolle ( $\rho=5$ )

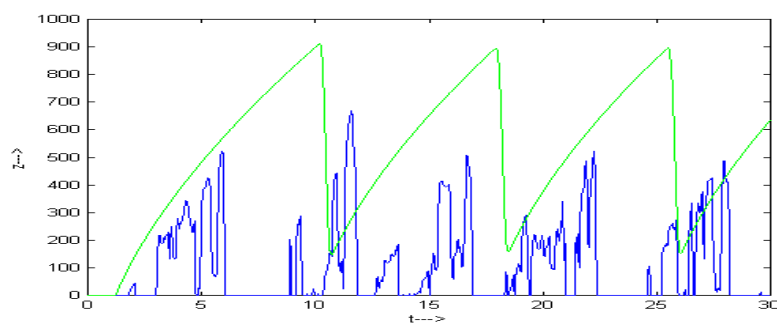


Bild 32: Warteschlangenlänge bei der RED-Überlastkontrolle ( $\rho=5$ )

## 6.2 Verhalten der Überlastkontroll-Modelle im Gleichgewichtszustand

Wir betrachten jetzt das Verhalten der Überlastkontroll-Modelle im Gleichgewichtszustand. Dabei nehmen wir als Gleichgewichtszustand an, daß keine Überlast vorhanden ist. Konkret nehmen wir  $\rho=0.5$  an.

### 6.2.1 Nettodurchsatzrate bei TD und RED

In die Bildern 33 und 34 ist der Verlauf der Nettodurchsatzrate bei TD und RED eingetragen. Dabei gilt wieder die blaue Kurve für das Modell mit stochastischer ON/Off-Senderate und die grüne Kurve für das Modell mit deterministischer Senderate.

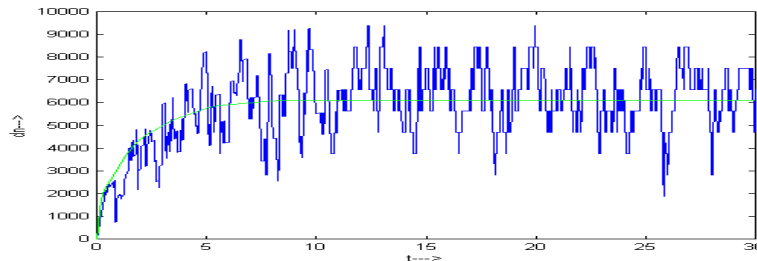


Bild 33 : Verlauf Nettodurchsatzrate bei TD ( $\rho=0.5$ )

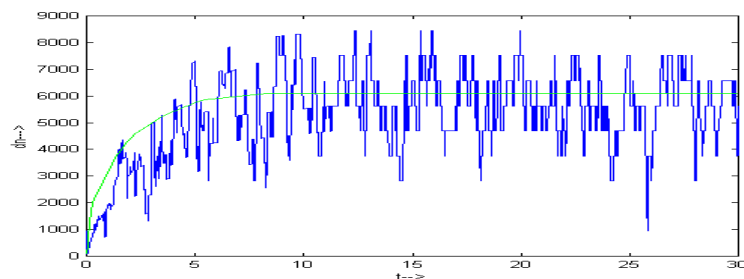


Bild 34: Verlauf der Nettodurchsatzrate bei RED ( $\rho=0.5$ )

### 6.2.2 Warteschlangenlänge bei TD und RED

Sowohl bei TD als auch bei RED verschwindet im Gleichgewichtszustand für Formen der Senderate die Warteschlange und es gilt in beiden Fällen  $z = 0$ .

## 6.3 Mittlere Nettodurchsatzraten und Warteschlangenlängen

Bei der Leistungsbewertung werden zur Erzielung kompakter Ergebnisse die Mittelwerte der Leistungskenngrößen herangezogen.

### 6.3.1 Mittlere Nettodurchsatzraten

Oben wurden die zeitliche Verläufe der einzelnen Signale betrachtet. Dabei wurden die unterschiedlichen Arbeitslasten (ausgedrückt durch den Verkehrswert) nur an zwei Punkten ( $\rho=0.5$  und  $\rho=5$ ) berücksichtigt. Um die Wirkung der Überlastkontrolle in Abhängigkeit von der Arbeitslast zu zeigen ist es zweckmäßig die mittlere Nettodurchsatzrate  $E(dn)$  in Abhängigkeit von der angebotenen Arbeitslast zu untersuchen. In den Bildern 35 und 36 sind die entsprechenden Abhängigkeiten graphisch dargestellt.

Interessant ist dabei, daß sowohl bei stochastischer ON/OFF-Senderaten als auch bei deterministischen Senderaten bei Überlast ( $\rho>1$ ), die Arbeitslast im Gegensatz zum

erweiterten Stationsmodell keinen Einfluß mehr auf die Nettodurchsatzrate hat. Damit bestätigt der hier eingeführte Modellansatz die intuitiven Erwartungen.

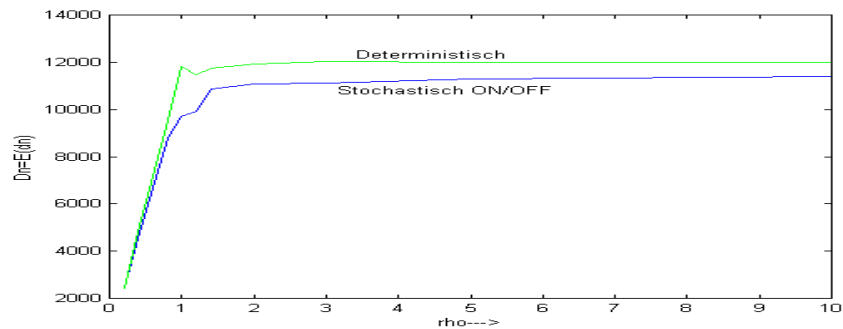


Bild 35: .Abhängigkeit der mittleren Nettodurchsatzrate vom Verkehrswert bei TD

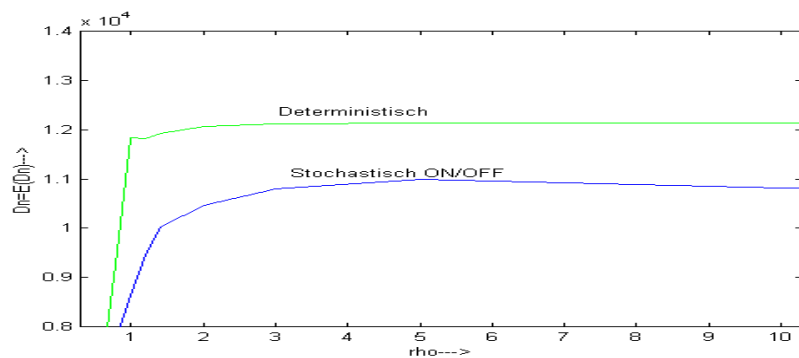


Bild 36: Abhängigkeit der mittleren Nettodurchsatzrate vom Verkehrswert bei RED

### 6.3.2 Mittlere Warteschlangenlängen

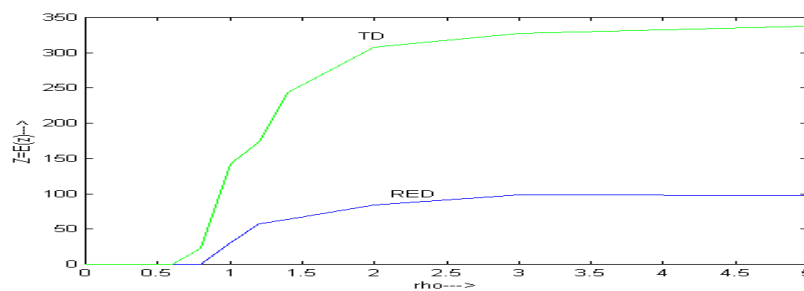


Bild 37: Abhängigkeit der mittleren Warteschlangenlänge vom Verkehrswert bei TD und RED für stochastische ON/OFF-Senderaten

Eine Auswertung der Bilder 35, 36 und 37 zeigt, daß für die angenommenen Werte der Reglerparameter und der Puffergröße das Warteschlangenverhalten bei RED erheblich besser ist als bei TD. Ein Teil dieser Verbesserung durch RED wird aber durch eine Verschlechterung des Nettodurchsatzverhaltens erkauft. Dabei ist zu beachten, daß die hier erzielten Simulationsergebnisse nicht für einen Vergleich zwischen TD und RED gedacht sind.

## 7 Schlußbemerkungen

Wir haben hier ein Modell für Simulation des Staukollapses und zwei Modelle für die Signalflußsimulation von Überlastkontrollmechanismen entwickelt. Im Vordergrund der Betrachtungen stand hier nicht der Vergleich von TD und RED, sondern die Entwicklung von Signalflußmodellen und ihre Implementierung mit SIMULINK und die Analyse Wirksamkeit der Überlastkontroll-Mechanismen. Die Simulationsergebnisse zeigen, daß sowohl bei der TD-Überlastkontrolle als auch bei der RED-Überlastkontrolle der Staukollaps beseitigt wird und bei Überlast nahezu die durch die Kanalkapazität bestimmte Durchsatzrate erreicht wird. Man kann mit Hilfe der Modelle das Verhalten aller darin auftretenden Signale ohne wesentlichen zusätzlichen Aufwand untersuchen. Es zeigt sich, daß der Modellansatz der Signalflußsimulation gut für die Analyse des Überlastverhaltens geeignet ist. Dieser Modellansatz ist ausbaufähig und bestimmte Einschränkungen dieses Beitrags sind nicht von prinzipieller Natur, sondern wurden der konkreten Implementierung wegen gemacht.

In nächster Zeit werden vom Autor systematische Untersuchungen der Wirksamkeit der verschiedenen Überlastkontrollmechanismen auf die Wartezeit, die Nettodurchsatzrate und andere Leistungskenngrößen durchgeführt. Besonders ist hier zusätzlich ein Modell für ECN (Explicit Congestion Notification) einzuführen und zu analysieren.

## 8 Literatur

- [1] G. Bergholz: Signalflußsimulation von Nachrichtenverkehrsmodellen. Arbeitsbericht (2002) [www.ist-intermon.org/download/sigflu5d.pdf](http://www.ist-intermon.org/download/sigflu5d.pdf)
- [2] G. Bergholz: Bereitstellung geeigneter Ankunftsratenprozesse für die Signalflußsimulation Arbeitsbericht (2003). [www.ist-intermon.org/download/bereit02.pdf](http://www.ist-intermon.org/download/bereit02.pdf)
- [3] J. C. Bolot and A. U. Shankar: Analysis of a Fluid Approximation to Flow Control Dynamics (1992)
- [4] C. Brandauer, G. Iannaccone, C. Diot, T. Ziegler, S.Fdida and M. May: Comparison of Tail Drop and Active Management Performance for bulk-data and Web-like Internet Traffic (2001)
- [5] V. Firoiu and M. Borden: A Study of Active Queue Management for Congestion Control . (2000)
- [6] Y.Guo et al.: Time-stepped Hybrid Simulation (TSHS) For Large Scale Networks. Proc. of INFCOM (2000)
- [7] I. Hoffmann: MATLAB und SIMULINK. Addison-Wesley (1998)
- [8] C. V. Hollot, V. Misra, D. Towsley and W. B. Gong: Analysis and Design of Controllers for AQM Routers Supporting TCP Flows 10 (2000)
- [9] C. V. Hollot, V Misra, D. Towsley and W. B. Gong: A Control Theoretical Analysis of RED. (2001)
- [10] C. V. Hollot, V. Misra, D. Towsley and W. B. Gong: On Designing Improved Controllers fpr AQM Router Supporting TCP Flows. (2001)
- [11] Y. Liu, F. L. Presti, V. Misra, D. Towsley an Y. Go: Fluid models an Solution for Large Scale Ip Networks SIGMETRICS03 6(2003)
- [12] B. Liu et al.: A Study of Network Simulation Efficiency: Fluid Flow Simulation vs. Packet level Simulation. IEEE INFOCOM (2001)
- [13] B. Liu et al.: Fluid Simulation of Large Scale Networks: Issues and Tradeoffs. Las Vegas, NV, June (1999)

- 
- [14] B. Melamed et al: Hybrid discret-continuous fluid-flow simulation. Rutgers University and Georgia Institut of Technology (2001)
- [15] V. Misra, W. B. Gong and D. Towsley: Fluid-based Analysis of a Network of AQM Routers supporting TCP Flow with an Application to RED. (2000)
- [16] V. Misra, W. B. Gong and D. Towsley: Stochastic Differential Equations Modeling and Analysis of TCP-Window size Behavior. (1999).
- [17] E. Plasser and T. Ziegler: An Analytical RED Function Design Guaranteeing Stable System Behavior (2003)
- [18] E. Plasser, T. Ziegler and P. Reichel: On the Non-Linearity of the RED Drop Function (2001)
- [19] Y. Wardi and B. Melamed: Loss Volume In Continuous Flow Models. Fast Simulation And Sensitivity Analysis Via IPA. Georgia Institut of Technologiy (2000)
- [20] Zhang Shu, M. Bauman and D. Marandin: Stability of TCP Congestion Control Networks with Active Queue Management (2002)
- [21] T. Ziegler, S. Fdida and C. Brandauer: Stability Criteria of RED with TCP Traffic (2000)
- [22] T. Ziegler, S. Fdida and C. Brandauer: Stability Criteria of RED with TCP-Traffic. Technical Report (2000)
- [23] T. Ziegler, C. Brandauer and S. Fdida: A quantitative Model for the Parameter Setting of RED with TCP-Traffic (2001)
- [24] T. Ziegler: On Avering for Active Queue Management Congestion Avoidance. (2003)