CONGESTION/FLOW CONTROL AND ROUTING IN DATA NETWORKS: A CONTROL-THEORETIC APPROACH

FINAL REPORT

SEMYON M. MEERKOV, Principal Investigator

LOTFI BENMOHAMED, Research Assistant

June 25, 1993

U. S. ARMY RESEARCH OFFICE

Grant Number DAAL03-90-G-0094

THE UNIVERSITY OF MICHIGAN

Department of Electrical Engineering and Computer Science
Ann Arbor, MIchigan 48109-2122
USA

APPROVED FOR PUBLIC RELEASE;

DISTRIBUTION UNLIMITED.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
June 25, 1993

3. REPORT TYPE AND DATES COVERED
Final 1 Apr 90 - 31 Mar 93

4. TITLE AND SUBTITLE
Congestion/Flow Control and Routing in Data Networks: A Control-Theoretic Approach

5. FUNDING NUMBERS
DAAL03-90-G-0094

6. AUTHOR(S)
Semyon M. Meerkov, P.I.
Lotfi Benmohamed, Res. Assist.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Electrical Engineering and Computer Science
The University of Michigan
Ann Arbor, MI 48109-2122

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P. O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ARO 27834.1-EL

11. SUPPLEMENTARY NOTES
The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an offi ial Department of the Army position, policy, or decision, unless so designated by other documentation.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

In this project, a novel architecture for congestion control in store-and-forward networks has been developed and design techniques for congestion controllers has been derived. In the framework of this architecture, each switching node is equiped with a congestion controller which calculates the throttled source admission rates using only local information available at the node. Each source receives the throttled admission rates and responds by transmitting with the lowest one. The design methodology developed in this project leads to the choice of the controller gains that ensure a stable congestion-free behavior of the network. No similar results are available in the current literature.

14. SUBJECT TERMS
Congestion control, Decentralized control architecture, Stable transmission rate control

15. NUMBER OF PAGES
71

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

CONGESTION/FLOW CONTROL AND ROUTING IN DATA NETWORKS: A CONTROL-THEORETIC APPROACH

FINAL REPORT

SEMYON M. MEERKOV, PRINCIPAL INVESTIGATOR

LOTFI BENMOHAMED, RESEARCH ASSISTANT

June 25, 1993

U. S. ARMY RESEARCH OFFICE

Grant Number DAAL03-90-G-0094

THE UNIVERSITY OF MICHIGAN

1 FOREWORD

In this project, a novel architecture for congestion control in store-and-forward networks has been developed and design techniques for congestion controllers has been derived. In the framework of this architecture, each switching node is equiped with a congestion controller which calculates the throttled source admission rates using only local information available at the node. Each source receives the throttled admission rates and responds by transmitting with the lowest one. The design methodology developed in this project leads to the choice of the controller gains that ensure a stable congestion-free behavior of the network. No similar results are available in the current literature.

2 TABLE OF CONTENTS

3 LIST OF APPPENDICES, ILLUSTRATIONS AND TABLES

List of Appendices

List of Figures

List of Tables

4 STATEMENT OF THE PROBLEM

Store-and-forward datagram networks consist of switching nodes and communication links connecting them according to a certain topology. Each link is characterized by its packet transmission capacity (packets per unit time). Each node is characterized by its packet storing capacity (buffer length) and its packet processing capacity (packets per unit time). A node which reaches its maximum storing capacity due to the saturation of its processors or one or more of its outgoing transmission links is called **congested**. Some of the packets, arriving at a congested node, cannot be accepted and have to be retransmitted at a later time. This leads to a deterioration of the network's throughput (number of packets delivered per unit time). Therefore, congestion prevention is an important problem of store-and-forward networks management (see [1]-[9] for several classical and recent publications).

As it follows from the above, congestion is a result of a mismatch between the network resources (buffer space, processing and transmission capacity) and the amount of traffic admitted for transmission. Consequently, congestion prevention can be interpreted as the problem of matching the admitted traffic to the network resources. This, in turn, could be viewed as a classical problem of feedback control (matching the output to the input of dynamical systems) and, indeed, has been recognized as such by many researchers [10]-[12]. The literature, however, does not offer rigorous methods for congestion controllers design. Nevertheless, these methods are of substantial importance since many designs result in non-decaying oscillation of the traffic admitted to the network even when the offered traffic is constant [13]-[15]. The main goal of this report is to describe an analytical method for the design of congestion controllers that ensures good dynamic characteristics and performance of datagram networks along with some fairness in resource allocations.

The approach used here is quite standard [16]: The level of network congestion is monitored through the buffers occupancy. When the occupancy of the buffer at some link reaches a given threshold x^0, the congestion controller associated with this link calculates a traffic admission rate and supplies all the sources of its traffic with the result of this calculation. In their turn, traffic sources reduce their transmission rate to the lowest level allowed by intermediate links.

Given this architecture, the main question is how the traffic admission rates should be calculated so that the network performance is sufficiently good. We give here the answer to this question for a backbone network. Part 1 of this report deals with the case of a single congested node where the congestion is due to the saturation of one of its outgoing links. The extension to multiple congested nodes and links is presented in Part 2.

5 SUMMARY OF THE MOST IMPRTANT RESULTS - PART 1 : CASE OF A SINGLE CONGESTED NODE

5.1 THE MODEL

5.1.1 Assumptions

The network

(i). The network employs a store-and-forward datagram (or connectionless) service where users are serviced without prior reservation.

(ii). The network consists of S switching nodes and N communication links. Let $\mathcal{S} = \{1, 2, \ldots, S\}$ denote the set of nodes and $\mathcal{N} = \{1, 2, \ldots, N\}$ denote the set of links. For each node $i \in \mathcal{S}$, let $\mathcal{O}(i) \subset \mathcal{N}$ denote the set of its outgoing links.

(iii). Each link has a transmission capacity of of $1/\tau_s$ packets/sec, where τ_s is the transmission time of a packet, and a propagation delay of τ_p sec. We assume that packets have the same fixed length.

(iv). Each node has a processing capacity $1/\tau_{pr}$ packets/sec, where τ_{pr} is the processing time of a packet. The processing capacity of each node is assumed to be larger than the total transmission capacity of its incoming links.

(v). The network traffic consists of flows corresponding to each source-destination pair (ij), i and $j \in \mathcal{S}$. The pair (ij) will be referred to as the (ij)-type traffic or the (ij) connection. Let $\mathcal{C}$ denote the set of all such connections.

(vi). For each (ij) connection, the source at node i sends packets to the destination at node j through a sequence of links referred to as the path of the connection and denoted $p(ij)$. The routing policy which determines the path of each connection is assumed to be static. Let $C(k)$ be the set of all connections (ij) which flow through link k.

(vii). Each node has a buffer for storing packets waiting to be transmitted on one of its outgoing links. Let $x_j, j \in \mathcal{N}$, denote the number of packets buffered for transmission on link j and referred to as link j's buffer.

The control architecture

(viii). Each node i has a congestion controller which computes, for each link $j \in \mathcal{O}(i)$, a quenched transmission rate q_j based on the difference between x_j and some threshold x^0 and on q_j at present and in the finite past. This information is local to node i.

(ix). Control algorithm updates take place every T sec where T is the time to transmit c packets, i.e., $T = c\tau_s$. Thus, the controller time is slotted with the slot duration, $[n, n+1), n = 0, 1, \ldots,$ equal to T.

(x). Each node sends the computed control information (q_j's) to the sources along the reverse direction of the traffic. This control information is serviced with high priority and is carried either in separate packets or along with data or acknowledgment packets.

The input traffic

(xi). For each connection (jk), let r_{jk}^0 denote the rate (in packets per slot) of its offered traffic. Let $f_i^0 = \sum_{(jk)\in C(i)} r_{jk}^0$ be the total rate of all connections flowing through link i. We assume that the input traffic is such that only one link is overloaded, i.e., there exists a link i_0 such that $f_{i_0}^0 > c$ whereas $f_i^0 < c$ for all $i \neq i_0$.

(xii). During the settling time of the system, the input traffic, satisfying (xi), is constant but otherwise arbitrary and unknown.

Remarks

(1). Assumption (i) means that the sources of the network traffic can be slowed down and do not need to reserve bandwidth. In integrated service networks, where some traffic types, such as voice or real-time video, cannot be slowed down and require a guaranteed transmission rate, we believe that our feedback congestion control scheme can coexist with a traffic admission algorithm. The latter makes decisions regarding bandwidth assignment to sources that require a guaranteed transmission rate whereas the former controls the sharing of some portion of the bandwidth among traffic sources that tolerate a certain level of time delay. The interaction between these two traffic management procedures needs to be investigated.

(2). Assumption (iii) assumes that the network is homogeneous. A generalization to heterogeneous networks, where each link has a different transmission capacity and/or propagation delay, is straightforward.

(3). Assumption (iv) means that links rather than processors are the bottlenecks. The case where processing is the bottleneck can be approached similarly. Note that τ_p/τ_s is the delay bandwidth product and is negligible for a slow speed network. Since τ_p is constant for a given transmission line but τ_s decreases as the transmission speed increases, the delay-bandwidth product is significant for high speed networks and represents the number of packets being propagated on the transmission line ("in-the-pipe" packets). The approach developed in this work is applicable to both processing and transmission bottlenecks.

(4). According to assumption (v), we distinguish between traffic types on the basis of their source and destination nodes so that traffic from multiple users at some node i sent to one or more users at some node j constitutes a single connection between i and j. The congestion control algorithm proposed here can equally use other definitions of traffic types.

(5). The cardinality of $C(k)$ in (vi) is the maximum number of connections that might simultaneously compete for the transmission capacity of link k. The congestion controller has to evenly share this capacity among the competing connections.

(6). Although with current technology buffers can be made very large, the threshold x^0 in assumption (viii) is introduced to insure a desired bound on the time delay in the steady state and to prevent congestion or underutilization of the transmission capacity during the transient periods of control.

(7). A tradeoff is involved in the choice of the update period T, or equivalently of c (since $T = c\tau_s$), as will be discussed in Section 4.4. Shorter T's lead to better responsiveness to changes in the input traffic but require the processors to devote more time computing the feedback signals. In this report, we choose T such that it is much less than the typical duration of a communication session so that assumption (xii) holds.

(8). Assumption (xii) could be understood in the sense that the input traffic is piecewise constant with the jumps occurring seldom enough so that the transients of the system have time to settle down between two consecutive jumps.

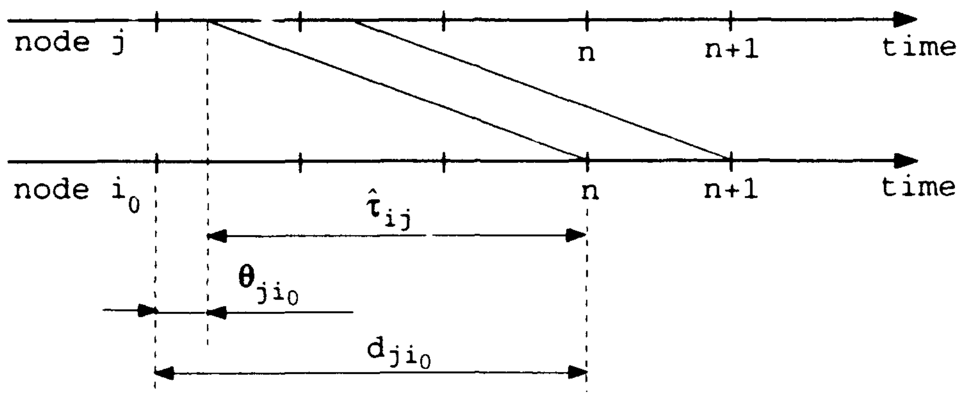

Figure 5.1: Network delay.

5.1.2 Buffer equations

Although the offered traffic in backbone networks is typically random, we use a deterministic fluid flow description. The reason is that, under certain assumptions, the stochastic nature of the offered traffic can be, in the first approximation, averaged using asymptotic techniques [17]-[18]. The resulting equations coincide with the fluid approximation used here. Therefore, to simplify the presentation and eliminate technical details, we begin directly from the fluid flow description.

Let $x_i(n)$ denote the occupancy of the buffer of link i at time n. Let $f_i(n)$ denote the total rate at which traffic, both new and transit, is flowing through link i during the $[n, n+1)$ slot. Then, the dynamics of buffer i is described by the following difference equation:

$$x_i(n+1) = \text{sg}\{x_i(n) + f_i(n) - c\}, \quad i \in \mathcal{N}, \tag{5.1}$$

where

$$\text{sg}(x) = \begin{cases} x & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

Since, $f_i(n) \leq f_i^0$ (admitted traffic $\leq$ offered traffic) and $f_i^0 < c$ for $i \neq i_0$ (assumption (xi)), then

$$x_i(n) = 0,\ n = 0, 1, \ldots,\ \forall i \neq i_0, \tag{5.2}$$

assuming that buffers are initially empty. This implies that traffic will incur no queueing delay at buffers of non-overloaded links.

Let $r_{jk}(n)$ denote the rate at which connection (jk) has been admitted to the network during $[n, n+1)$. Then, the total amount of traffic arriving at buffer i_0 during $[n, n+1)$ is

$$f_{i_0}(n) = \sum_{(jk)\in C(i_0)} r_{jk}(n - \hat{\tau}_{ji_0}), \tag{5.3}$$

where $C(i_0)$ is defined in assumption (vi) and $\hat{\tau}_{ji_0} = \tau_{ji_0}/T$ is the delay, measured in time slots, that the (jk) traffic incurs from its point of entry to the network at node j until it arrives at buffer i_0, while τ_{ji_0} is the sum, along the path of (jk) from j to i_0, of all transmission, propagation and processing delays. Since the network time is slotted, when τ_{ji_0} is not an integer multiple of T, we rewrite (5.3) as follows (see Figure 5.1):

$$f_{i_0}(n) = \sum_{(jk)\in C(i_0)} (1 - \theta_{ji_0})\, r_{jk}(n - d_{ji_0}) + \theta_{ji_0}\, r_{jk}(n - d_{ji_0} + 1), \tag{5.4}$$

where

$$\begin{aligned} d_{ji_0} &= \lceil \hat{\tau}_{ji_0} \rceil : \text{the smallest integer} \geq \hat{\tau}_{ji_0}, \\ \theta_{ji_0} &= \lceil \hat{\tau}_{ji_0} \rceil - \hat{\tau}_{ji_0}, \quad \theta_{ji_0} \in [0, 1). \end{aligned} \tag{5.5}$$

Note that d_{ji_0} and θ_{ji_0} are physical parameters of the network. Along one hop (i.e., through one link) we have

$$\hat{\tau} = \frac{\tau_s + \tau_p + \tau_{pr}}{c\tau_s} = \frac{1}{c} + \frac{\tau_p + \tau_{pr}}{c\tau_s} \triangleq d - \theta. \tag{5.6}$$

Therefore, as the network speed increases (τ_s decreases), d increases and results in an increase in the dimension of the closed loop equations (see Section 3 below).

Substituting (5.4) in (2.1) we obtain the buffer equation for the overloaded link:

$$x_{i_0}(n+1) = \text{sg}\left\{ x_{i_0}(n) + \sum_{(jk)\in C(i_0)} (1-\theta_{ji_0}) r_{jk}(n - d_{ji_0}) + \theta_{ji_0} r_{jk}(n - d_{ji_0} + 1) - c \right\}. \tag{5.7}$$

5.1.3 Controller

Idea of the control law

The idea of the congestion controller employed in this work can be illustrated as follows:

Consider a single-node network with a traffic source of rate r and an outgoing link with capacity c, where $r > c$. A congestion controller is supposed to quench the offered traffic so that the steady state buffer occupancy at the node is a given number x^0. The quenched source rate $q(t) < r$, defines the part of the offered traffic that will be admitted to the network. Assuming, for simplicity, that all variables are continuous, the dynamics of the buffer occupancy, $x(t)$, can be described as follows:

$$\dot{x} = q - c.$$

If the proportional control law is used, the quenched transmission rate obeys the equation

$$\dot{q} = -\alpha_0(x - x^0).$$

This results in

$$\ddot{x} + \alpha_0 x = \alpha_0 x^0,$$

and both the buffer occupancy, $x(t)$, and the admitted traffic, $q(t)$, exhibit non-decaying oscillations (if $\alpha_0 > 0$; otherwise the system is unstable).

To eliminate this problem, a proportional-plus-derivative control law can be utilized:

$$\dot{q} = -\alpha_0(x - x^0) - \alpha_1 \dot{x}. \tag{5.8}$$

Then,

$$\ddot{x} + \alpha_1 \dot{x} + \alpha_0 x = \alpha_0 x^0, \tag{5.9}$$

and $x(t)$ and $q(t)$ converge to x^0 and c, respectively (if α_0 and $\alpha_1 > 0$). Moreover, the speed of the convergence can be assigned arbitrarily by an appropriate choice of α_0 and α_1.

In networks with multiple switching nodes, however, PD-controllers will not work either. The reason is that, due to delays in information transmission between the nodes, the buffer occupancy at each node is described by a time-delay equation of the form

$$\dot{x} = q(t - \tau) - c,$$

where $\tau > 0$. In this case, the closed loop system may be oscillatory even if the PD controller (5.8) is employed. The main idea of this work is to combat this difficulty by constructing a controller of the form

$$\dot{q}(t-\tau) = -\alpha_0(x(t) - x^0) - \alpha_1 \dot{x}(t).$$

In this case, the closed loop behavior is still described by (5.9), and any desired dynamics can be achieved. The question is whether such a controller can be constructed for various types of input traffic patterns that result in different τ's. It turns out, however, that it is, indeed, possible, and the equations for such a controller are given below.

Controller equation

The congestion controller includes two parts: the control algorithm and the control protocol. The control algorithm calculates the quenched source transmission rate and the control protocol specifies how the traffic sources react to these quenched transmission rates.

The following **control protocol** is used in this work: Assume that $q_i(n+1), i \in \mathcal{N}$, is the quenched transmission rate calculated at time n by the node for which link i is an outgoing link. Then, during the time slot $[n, n+1)$, the (jk)-type traffic is admitted with the rate

$$r_{jk}(n) = \min\{q_m(n+1-\hat{\tau}_{mj}), m \in p(jk); r^0_{jk}\}, \quad (jk) \in \mathcal{C}, \tag{5.10}$$

where $\hat{\tau}_{mj}$ is the delay incurred by the feedback signal q_m to reach node j. In compliance with this expression, the source admission rate is defined by the smallest among all transmission rates corresponding to the links along the path $p(jk)$ and the desired transmission rate. Since q_m is generated every T, the most recent feedback signal available at node j at time n is $q_m(n+1-d_{mj})$ where $d_{mj} = \lceil \hat{\tau}_{mj} \rceil$. Therefore, equation (5.10) becomes

$$r_{jk}(n) = \min\{q_m(n+1-d_{mj}), m \in p(jk);\ r^0_{jk}\}, \quad (jk) \in \mathcal{C}. \tag{5.11}$$

The **control algorithm** is defined by the following equations:

$$q_i(n+1) = \mathrm{Sat}_{q^0}\left\{q_i(n) - \sum_{j=0}^{J} \alpha_j [x_i(n-j) - x^0] - \sum_{k=0}^{K} \beta_k q_i(n-k)\right\}, \quad i \in \mathcal{N}, \tag{5.12}$$

where $q^0 \geq c$, J and K are non-negative integers and

$$\mathrm{Sat}_a(z) = \begin{cases} 0 & \text{if } z \leq 0, \\ a & \text{if } z \geq a, \\ z & \text{otherwise}. \end{cases}$$

The saturation function $\mathrm{Sat}_{q^0}(.)$ in (5.12) is introduced to impose bounds on q_i's. The lower bound zero keeps $q_i \geq 0$ whereas the upper bound q^0 limits the initial sending rate of connections with uncongested path.

When $J = 1$ and $K = 0$, (5.12) results in the PD-controller

$$q_i(n+1) = \mathrm{Sat}_{q^0}\{q_i(n) - a(x_i(n) - x^0) - b(x_i(n) - x_i(n-1))\}, \tag{5.13}$$

where $a = \alpha_0 + \alpha_1$ and $b = -\alpha_1$.

As it is shown below, the control gains, α_j and β_k in (5.12), must satisfy, among others, the following requirements:

$$\sum_{j=0}^{J} \alpha_j > 0\,, \qquad \sum_{k=0}^{K} \beta_k = 0. \tag{5.14}$$

Since, according to (5.2), $x_i(n) = 0, n \geq 0$, for $i \neq i_0$, solving (5.12) under conditions (5.14) we obtain $q_i(n) = q^0,\ n \geq 0,\ i \neq i_0$ (assuming $q_i(n) = q^0,\ n \leq 0,\ i \neq i_0$). Therefore, underloaded links have their buffers at zero and their control signals equal to q^0. This implies that, for all connections $(jk) \in C(i_0)$, equation (5.11) becomes

$$r_{jk}(n) = \min\{q_{i_0}(n+1-d_{i_0 j}), r^0_{jk}\}, \quad (jk) \in C(i_0). \tag{5.15}$$

As it follows from (5.12), the control algorithm is defined by $J+K+2$ control gains α_i and β_i. In the next section, the analysis of the closed loop behavior of (5.7), (5.12) and (5.15) for given α's and β's is described and a method for choosing α's and β's that ensure high delay-throughout characteristics is given.

5.2 ANALYSIS

5.2.1 Closed loop equations

As it follows from the previous section, the closed loop behavior of the network with i_0 as the single overloaded link is described by the following equations:

$$x_{i_0}(n+1) = \mathrm{sg}\left\{x_{i_0}(n) + \sum_{(jk)\in C(i_0)} (1-\theta_{ji_0}) r_{jk}(n-d_{ji_0}) + \theta_{ji_0} r_{jk}(n-d_{ji_0}+1) - c\right\} \tag{5.16}$$

$$r_{jk}(n) = \min\{q_{i_0}(n+1-d_{i_0 j}), r^0_{jk}\}, \quad (jk) \in C(i_0), \tag{5.17}$$

$$q_{i_0}(n+1) = \mathrm{Sat}_{q^0}\left\{q_{i_0}(n) - \sum_{j=0}^{J} \alpha_j [x_{i_0}(n-j) - x^0] - \sum_{k=0}^{K} \beta_k q_{i_0}(n-k)\right\}. \tag{5.18}$$

Equation (5.17) can be rewritten as follows:

$$r_{jk}(n) = \delta_{jk}(n)\, q_{i_0}(n+1-d_{i_0 j}) + (1-\delta_{jk}(n))\, r^0_{jk}, \tag{5.19}$$

where

$$\delta_{jk}(n) = \begin{cases} 1 & \text{if } r^0_{jk} > q_{i_0}(n+1-d_{i_0 j}), \\ 0 & \text{else.} \end{cases} \tag{5.20}$$

Combining equations (5.16) and (5.19), we obtain

$$\begin{aligned} x_{i_0}(n+1) \;=\; \mathrm{sg}\Big\{ & x_{i_0}(n) + \sum_{(jk)\in C(i_0)} [\delta_{jk}(n-d_{ji_0})(1-\theta_{ji_0}) q_{i_0}(n+1-d_j) + \\ & \delta_{jk}(n-d_{ji_0}+1)\theta_{ji_0} q_{i_0}(n+1-(d_j-1))] + r^0_{i_0}(n) - c\Big\}, \end{aligned} \tag{5.21}$$

where

$$\begin{aligned} r^0_{i_0}(n) &= \sum_{(jk)\in C(i_0)} (1-\delta_{jk}(n-d_{ji_0}))(1-\theta_{ji_0}) r^0_{jk} + (1-\delta_{jk}(n-d_{ji_0}+1))\theta_{ji_0} r^0_{jk}, \\ d_j &= d_{ji_0} + d_{i_0 j}. \end{aligned} \tag{5.22}$$

When $\delta_{jk}(n) = 1$, the (jk) traffic is being quenched during $[n, n+1)$, otherwise (when $\delta_{jk}(n) = 0$) it is admitted with its desired transmission rate and $r^0_{i_0}$ is the cumulative rate of all unquenched flows. We will refer to d_j as the round trip delay of connection (jk).

Equation (5.21) can be rewritten as

$$x_{i_0}(n+1) = \mathrm{sg}\left\{x_{i_0}(n) + \sum_{i=0}^{D} l_i(n)\, q_{i_0}(n+1-i) + r^0_{i_0}(n) - c\right\}, \tag{5.23}$$

here

$$\begin{aligned} l_i(n) &= \sum_{(jk)\in C^i} \delta_{jk}(n-d_{ji_0})(1-\theta_{ji_0}) + \sum_{(jk)\in C^{i+1}} \delta_{jk}(n-d_{ji_0}+1)\theta_{ji_0}, &(5.24)\\ D &= \max_{(jk)\in C(i_0)} d_j, &(5.25)\end{aligned}$$

where

$$C^d = \{(jk)\in C(i_0) : d_j = d\},$$

i.e., C^d is the set of all connections flowing through link i_0 with round trip delay equal to d. The l_i's can be interpreted as the number of quenched flows with round trip delay equal to i time slots. Note that $l \triangleq \sum_{i=0}^{D} l_i = \sum_{(jk)\in C(i_0)} \delta_{jk}$ is the total number of connections being quenched by link i_0 and is a positive number since link i_0 is overloaded and therefore at least one connection is quenched.

Omitting the index i_0, the closed loop equations become

$$\begin{aligned} x(n+1) &= \mathrm{sg}\left\{x(n) + \sum_{i=0}^{D} l_i(n) q(n+1-i) + r^0(n) - c\right\}, &(5.26)\\ q(n+1) &= \mathrm{Sat}_{q^0}\left\{q(n) - \sum_{j=0}^{J}\alpha_j(x(n-j)-x^0) - \sum_{k=0}^{K}\beta_k q(n-k)\right\}. &(5.27)\end{aligned}$$

Note that since (5.26) involves delayed versions of $q(n)$ up to $q(n-(D-1))$, it is natural to assume that $K \geq D-1$. Equations (5.26)-(5.27) describe a $(J+K+2)$-dimensional system with the state

$$Y(n) = \left[x(n)-x^0, x(n-1)-x^0, \ldots, x(n-J)-x^0, q(n), q(n-1), \ldots, q(n-K)\right]^T.$$

The steady states and dynamic properties of this system are described below.

5.2.2 Steady states

Let x_s and q_s be the steady state values corresponding to equations (5.26) and (5.27) under the assumption that the input traffic is constant. Then,

$$\begin{aligned} x_s &= \mathrm{sg}\left\{x_s + (\sum_{i=0}^{D} l_i) q_s + r^0 - c\right\}, &(5.28)\\ q_s &= \mathrm{Sat}_{q^0}\left\{(1-\sum_{k=0}^{K}\beta_k) q_s - (\sum_{j=0}^{J}\alpha_j)(x_s - x^0)\right\}. &(5.29)\end{aligned}$$

Figures 5.2.a and 5.2.b show the solutions to equations (5.28) and (5.29), respectively. The combined solution is shown in Figure 5.2.c and corresponds to

$$\begin{aligned} q_s &= \frac{c-r^0}{l}, &(5.30)\\ x_s &= x^0 - \frac{\sum_{k=0}^{K}\beta_k}{\sum_{j=0}^{J}\alpha_j} q_s. \end{aligned}$$

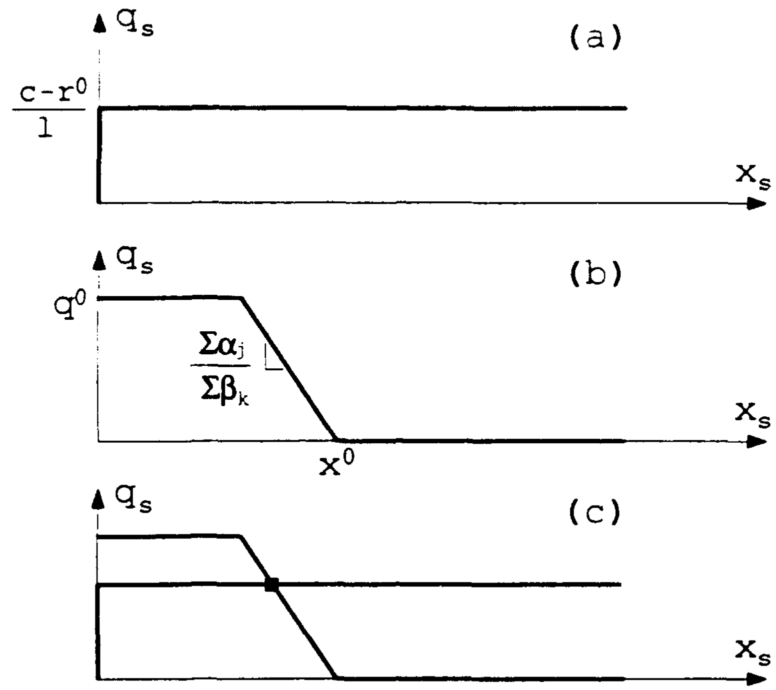

Figure 5.2: Steady state.

In order to ensure that $x_s = x^0$, we choose β_k's so that

$$\sum_{k=0}^{K} \beta_k = 0. \tag{5.31}$$

Thus, the steady state $Y_s = (0, \ldots, 0, q_s, \ldots, q_s)^T$ exists if and only if the vector of control parameters

$$Q = (\alpha_0, \ldots, \alpha_J, \beta_0, \ldots, \beta_K)^T \tag{5.32}$$

satisfies constraint (5.31). Note that q_s in (5.30) can be rewritten as follows:

$$q_s = \frac{c - r^0}{l} = \frac{c}{M} + \frac{(M-l)\frac{c}{M} - r^0}{l}.$$

This expression presents the following fairness property: If M connections share a transmission link, then each gets $1/M$ of the bandwidth and if $(M-l)$ connections use less than their share (i.e., only l connections are quenched), then the unused portion $((M-l)\frac{c}{M} - r^0)$ is equally distributed among the rest.

5.2.3 Transient analysis

To study the local stability properties, we simplify the dynamic equations for Y in the neighborhood of Y_s by

(a) removing the saturation-type nonlinearities (sg(.) and $\mathrm{Sat}_{q^0}(.)$) in (5.26) and (5.27) since they are not activated for small deviations around Y_s ((x_s, q_s) is in the interior of $[0, \infty) \times [0, q^0]$);

(b) treating $l_i(n)$ and $r^0(n)$ as constants since, based on equation (5.20), there exists a ball around Y_s within which δ_{jk} is constant (assuming that $r^0_{jk} \neq q_s, (jk) \in C(i_0)$).

The resulting equations are

$$x(n+1) = x(n) + \sum_{i=0}^{D} l_i\, q(n+1-i) + r^0 - c, \tag{5.33}$$

$$q(n+1) = q(n) - \sum_{j=0}^{J} \alpha_j (x(n-j) - x^0) - \sum_{k=0}^{K} \beta_k\, q(n-k). \tag{5.34}$$

Theorem 5.1 *For any given $l_0, l_1, \ldots, l_D$, the poles of the closed loop system (5.33)-(5.34) can be placed at will by an appropriate choice of the gains $\alpha_0, \ldots, \alpha_J, \beta_0, \ldots, \beta_K$, $\sum_{k=0}^{K} \beta_k = 0$, if $J = 1$ and $K = D$. In other words, a PD controller with respect to x and a controller of order D with respect to q are sufficient to ensure any desired dynamics of the closed loop system (5.33)-(5.34).* □

Proof: See Appendix 1.

As it follows from the proof, the characteristic polynomial of the closed loop system can be represented as

$$P(\lambda) = (-1)^{D+1} \lambda P_\Gamma(\lambda),$$

where

$$P_\Gamma(\lambda) = \lambda^{D+2} + \gamma_1 \lambda^{D+1} + \gamma_2 \lambda^D + \ldots + \gamma_{D+2}, \tag{5.35}$$

and the coefficients γ_i, $i = 1, 2, \ldots, D+2$, are uniquely defined by the control gains

$$Q = (\alpha_0, \alpha_1, \beta_0, \beta_1, \ldots, \beta_D)^T.$$

Therefore, as it follows from (5.35), one of the closed loop poles is at 0 and the remaining $D+2$ poles can be chosen at will. It is shown in the Proof of Theorem 5.1 that for any desired set of coefficients γ_i, $i = 1, 2, \ldots, D+2$, the corresponding control gain Q is

$$Q = [M(L)]^{-1} \tilde{\Gamma}, \tag{5.36}$$

where

$$M(L) = \begin{pmatrix} l_0 & 0 & 1 & & & & \\ l_1 & l_0 & -1 & 1 & & & \\ l_2 & l_1 & & -1 & \ddots & & \\ \vdots & \vdots & & & \ddots & 1 & \\ l_D & l_{D-1} & & & & -1 & 1 \\ 0 & l_D & & & & & -1 \\ 0 & 0 & 1 & 1 & \ldots & & 1 \end{pmatrix}, \quad L = \begin{pmatrix} l_0 \\ l_1 \\ \vdots \\ l_D \end{pmatrix}, \quad \tilde{\Gamma} = \begin{pmatrix} \gamma_1 + 2 \\ \gamma_2 - 1 \\ \gamma_3 \\ \gamma_4 \\ \vdots \\ \gamma_{D+1} \\ \gamma_{D+2} \\ 0 \end{pmatrix}.$$

As it is shown in Appendix 1, expression (5.36) can be rewritten in an alternative form shown below:

$$\Gamma = \Gamma_\beta + W_\alpha L, \tag{5.37}$$

where

$$\Gamma = \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \vdots \\ \gamma_{D+1} \\ \gamma_{D+2} \end{pmatrix}, \Gamma_\beta = \begin{pmatrix} \beta_0 - 2 \\ \beta_1 - \beta_0 + 1 \\ \beta_2 - \beta_1 \\ \beta_3 - \beta_2 \\ \vdots \\ \beta_D - \beta_{D-1} \\ -\beta_D \end{pmatrix}, W_\alpha = (V_0 \, V_1 \ldots V_D) = \begin{pmatrix} \alpha_0 & & & \\ \alpha_1 & \alpha_0 & & \\ & \alpha_1 & \ddots & \\ & & \ddots & \alpha_0 \\ & & & \alpha_1 \end{pmatrix}.$$

Both of these expressions, (5.35) and (5.37), will be used in Section 4 below for design purposes.

Concluding this section, we note that the sum of the coefficients of $P_\Gamma(\lambda)$, $1 + \sum_{i=1}^{D+2} \gamma_i$, is equal to $(\alpha_0 + \alpha_1) \sum_{i=0}^{D} l_i$. Since it is necessary for asymptotic stability that this sum be positive, $\alpha_0 + \alpha_1$ has to be positive as stated in (5.14).

5.3 DESIGN

5.3.1 Adaptive vs. robust design

The design of the congestion controller (5.12) involves the computation of the parameters of the control law, i.e., the vector of control gains $Q = (\alpha_0, \alpha_1, \beta_0, \beta_1, \ldots, \beta_D)^T$. As it has been shown in Section 3, the closed loop dynamics depend on the number of quenched flows $L = (l_0, l_1, \ldots, l_D)^T$, where l_i is the equivalent number of quenched flows with round trip delay equal to i. Therefore, if each node can take into account the variations of L when computing the control law, the resulting design is referred to as **adaptive**; here Q is updated whenever L changes. If information about L is not available, a **robust** design has to be considered; here the control law is implemented with a fixed Q so that stability and performance requirements are satisfied for all admissible values of L.

Note that an adaptive design has the advantage of achieving good performance but at the expense of higher computational requirements (the updating algorithm for Q) whereas a robust design requires less computations but the achieved performance may be inferior. Also, in order to be able to implement an adaptive design, the congestion control protocol should be implemented in such a way that L can be known by the switching nodes. One way of doing so is to stamp each packet with the identity of the quenching link, if any. In this way, every node can maintain a table of quenched flows. Each of these design approaches are investigated below.

5.3.2 Adaptive design

Suppose that the desired performance of the congestion control system is specified in terms of closed loop poles that correspond to the vector of characteristic polynomial coefficients $\Gamma = (\gamma_1, \gamma_2, \ldots, \gamma_{D+2})^T$. Then, as it follows from (5.36), the adaptation of the control gain $Q = (\alpha_0, \alpha_1, \beta_0, \ldots, \beta_D)^T$ to changes in L takes the following form:

$$Q(n) = [M(L(n-1))]^{-1} \tilde{\Gamma}, \tag{5.38}$$

where $L(n) = (l_0(n), l_1(n), \ldots, l_D(n))^T$ and $\tilde{\Gamma} = (\gamma_1 + 2, \gamma_2 - 1, \gamma_3, \ldots, \gamma_{D+2}, 0)^T$. Note that $Q(n)$ depends on L at time $n-1$ since at time n only $L(n-1)$ is available. The performance of the network utilizing the adaptive control law (5.38) is illustrated in Section 5 below.

5.3.3 Robust design

The goal of this subsection is to find a fixed gain $Q = (\alpha_0, \alpha_1, \beta_0, \beta_1, \ldots, \beta_D)^T$ that ensures stability of the closed loop system (5.33)-(5.34) for all values of L belonging to an admissible set $\mathcal{L}$ defined as

$$\mathcal{L} = (\mathcal{L}_0 \times \mathcal{L}_1 \times \ldots \times \mathcal{L}_D) - \{\mathbf{0}\},\ \mathcal{L}_i = [0, \bar{l}_i],\ i = 0, 1, \ldots, D. \tag{5.39}$$

where $\bar{l}_i$ is the maximum equivalent number of quenched flows with round trip delay equal to i. From (5.24) it is given by

$$\bar{l}_i = \sum_{(jk)\in C^i} (1 - \theta_{ji_0}) + \sum_{(jk)\in C^{i+1}} \theta_{ji_0}, \tag{5.40}$$

where

$$C^d = \{(jk) \in C(i_0) : d_j = d\}.$$

The solution of this problem is given by the following two theorems:

Theorem 5.2 *There exists a positive number $k^*(\bar{L})$ where $\bar{L} = (\bar{l}_0, \bar{l}_1, \ldots, \bar{l}_D)^T$ such that the closed loop system (5.33)-(5.34) is asymptotically stable for any $L \in \mathcal{L}$ if the control gains are chosen as*

$$Q = Q_k \triangleq (k\alpha_0, k\alpha_1, \beta_0, \beta_1, \ldots, \beta_D)^T,\ k \in (0, k^*),$$

where

$$\begin{aligned} \alpha_0 &= \frac{D+4}{2(D+1)}, \\ \alpha_1 &= -\frac{D+2}{2(D+1)}, \\ \beta_0 &= \frac{3D}{2(D+1)}, \\ \beta_i &= \frac{D-2(i+1)}{2(D+1)},\ i = 1, 2, \ldots, D. \end{aligned}$$

□

Proof: See Appendix 2.

Thus the robust control gain, Q_k, is the $(D+3)$-dimensional vector with components defined in Theorem 5.2.

Let S_θ be the surface in $\mathbb{R}^{D+2}$ defined by the equations

$$S_\theta : \begin{cases} \sum_{i=1}^{D+2} \gamma_i \cos(i\theta) = -1, \\ \sum_{i=1}^{D+2} \gamma_i \sin(i\theta) = 0,\ \theta \in (0, \pi), \end{cases}$$

and let the inequality $L \leq \bar{L}$ imply the component wise inequalities $l_i \leq \bar{l}_i$, $i = 0, 1, \ldots, D$.

Theorem 5.3 *The positive number k^*, referred to in Theorem 4.1, is defined as*

$$k^* = \min\{k_1, k_2\}.$$

Here,

$$k_1 = \frac{3D + 5 + (-1)^D(D+3)}{2(D+3)\sum_{i=0}^{\lfloor D/2 \rfloor} \bar{l}_{2i}},$$

and k_2 is the solution of the following optimization problem:

$$\textit{minimize } k \textit{ subject to the constraints} \tag{5.41}$$

$$\begin{array}{l} k \geq 0, \\ \Gamma_\beta + W_\alpha L \in S_\theta, \\ 0 \leq L \leq k\bar{L}, \end{array}$$

where Γ_β and W_α are defined in (5.37). If the solution of this optimization problem does not exist, $k_2 = +\infty$.

Proof: See Appendix 2.

Thus, the design of the robust congestion controller can be accomplished in two stages. First, the optimization problem of Theorem 5.3 is solved and k^* is determined. At the second stage, a specific gain Q_k is chosen from the family defined in Theorem 5.2 so that the performance requirements are satisfied as much as possible. This design procedure is illustrated in Section 5 below.

For each θ, the optimization problem of Theorem 5.3 is a linear programming problem. Indeed, we minimize a linear function, k, subject to the following liner equality and inequality constraints obtained by substituting in (5.41) the expressions of Γ_β, W_α and S_θ:

$$\begin{array}{l} k \geq 0, \\ 0 \leq l_i \leq k\bar{l}_i, \; i = 0, 1, \ldots, D, \\ \sum_{i=0}^{D} a_i(\theta) l_i = c_1(\theta), \\ \sum_{i=0}^{D} b_i(\theta) l_i = c_2(\theta), \end{array} \tag{5.42}$$

where

$$\begin{array}{l} a_i(\theta) = \alpha_0 \cos((i+1)\theta) + \alpha_1 \cos((i+2)\theta), \\ b_i(\theta) = \alpha_0 \sin((i+1)\theta) + \alpha_1 \sin((i+2)\theta), \\ c_1(\theta) = -1 - 2\cos(\theta) + \cos(2\theta) + \sum_{i=0}^{D} \beta_i (\cos((i+1)\theta) - \cos((i+2)\theta)), \\ c_2(\theta) = -2\sin(\theta) + \sin(2\theta) + \sum_{i=0}^{D} \beta_i (\sin((i+1)\theta) - \sin((i+2)\theta)). \end{array}$$

Therefore, if $k_2(\theta)$ denotes the result of the above minimization, the solution of the optimization problem (5.41) can be found as

$$k_2 = \inf_{\theta \in (0,\pi)} k_2(\theta).$$

Practically, an approximate solution can be found by discretizing the interval $(0, \pi)$ as $\theta_i = i\frac{\pi}{N+1}$, $i = 1, 2, \ldots, N$, solving the N resulting LP problems, and choosing the smallest of the N numbers. This procedure is illustrated below.

To conclude this subsection, we consider an example of calculating k^* for a particular network. Consider a network with $2D+3$ nodes connected according to a bidirectional ring topology with the minimum-hop routing strategy. If the controller update period T is equal to the delay along one hop (transmission + propagation + processing), then $\bar{L}$ is given by

$$\begin{cases} \bar{l}_{2i} = D + 1 - i, \\ \bar{l}_{2i+1} = 0, \quad i = 0, 1, \ldots, D. \end{cases}$$

Table (5.1) shows k_1, k_2 and k^* for different values of D. Note that k^* is a decreasing function of D since, as D increases, the network size increases and therefore the number of flows that might be active simultaneously increases and this requires a smaller gain in order to insure stability for any admissible vector L of quenched flows.

D	1	2	3	4	5	6	7	8	9
k_1	0.533	0.286	0.178	0.121	0.088	0.067	0.052	0.042	0.035
k_2	0.640	0.340	0.196	0.137	0.089	0.069	0.050	0.049	0.029
k^*	0.533	0.286	0.178	0.121	0.088	0.067	0.050	0.042	0.029

Table 5.1: Values of k^* for a ring network.

5.3.4 The update period

Both the robust and the adaptive design approaches require the selection of the update period T for the controller (see assumption (ix) in Section 2). The value of T affects both the transient response and the control overhead due to the computation and transmission of the feedback information.

The settling time and the buffer overshoot are the main characteristics of the transient response. The settling time, t_s, is the time interval between the start of transmission and the time when the transmission rate reaches and remains in the 5% neighborhood of its steady state value. Obviously,

$$t_s \geq RTD + \kappa_1 T,$$

where RTD is the round trip delay and $\kappa_1 > 0$ is a constant defined by the control law. If N is the average number of packets to be transmitted per connection and M is the average number of connections sharing the link, the ratio of settling time to transmission time of a connection, t_c, can be characterized as

$$\frac{t_s}{t_c} \geq \frac{RTD + \kappa_1 T}{NM\tau_s}. \tag{5.43}$$

The buffer overshoot Δx can be evaluated as follows:

$$\Delta x = x - x^0 \geq (RTD + \kappa_2 T)\, k\, c, \tag{5.44}$$

where x^0 is the desired buffer occupancy, $k\,c = k/\tau_s$ is the excess rate beyond the link capacity c, and $\kappa_2 > 0$ again depends on the control law . The inequalities in (5.43) and (5.44) are due to the fact that the control action cannot start before the round trip delay has elapsed and would require a number (κ_1 or κ_2) of update periods to reach either the steady state or the maximum buffer occupancy. From these inequalities we see that faster updates (i.e., smaller T) lead to shorter settling time and smaller buffer overshoot which are some of the desirable features for the network.

On the other hand, the decrease in T leads to an increase in the control overhead. Indeed, if τ_0 is the time necessary for the node to compute and transmit the control signals (feedback information) every update period T, then τ_0/T is the control overhead. The time τ_0 is smaller when the network speed is higher and therefore, it can be assumed proportional to τ_s ($\tau_0 = \kappa_3\tau_s$).

Given these performance criteria, the choice of the update period T can be approached as a multicriteria optimization problem and an appropriate Pareto set can be calculated. However, if we assume that there exist penalties ν_1, ν_2, and ν_3 for settling delay, queueing delay, and control overhead, respectively, then the following single function of T should be minimized in order to determine the compromise value of the controller update period:

$$J(T) = \nu_1 \frac{RTD + \kappa_1 T}{NM\tau_s} + \nu_2(RTD + \kappa_2 T)\frac{k}{\tau_s} + \nu_3 \frac{\kappa_3 \tau_s}{T}.$$

The solution, obviously, is

$$T_0 = \tau_s \sqrt{\frac{\nu_3 \kappa_3}{\nu_1 \frac{\kappa_1}{NM} + \nu_2 \kappa_2 k}}.$$

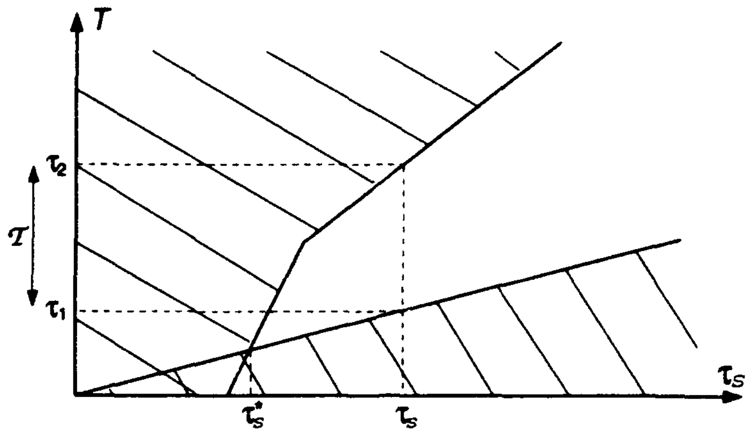

Figure 5.3: Admissible set of update periods.

This solution assumes that no constraints on the selection of T are imposed. Often it is the case, and the minimization of $J(T)$ should take place over some admissible set $\mathcal{T}$ resulting from the following requirements:

(a). Ratio of settling time to transmission time of a connection less than some constant $\delta_1 < 1$:

$$\frac{RTD + \kappa_1 T}{NM\tau_s} \leq \delta_1. \tag{5.45}$$

(b). Buffer overshoot less than δ_2:

$$(RTD + \kappa_2 T)\frac{k}{\tau_s} \leq \delta_2. \tag{5.46}$$

(c). Control overhead less than $\delta_3 < 1$:

$$\frac{\kappa_3 \tau_s}{T} \leq \delta_3. \tag{5.47}$$

From constraints (5.45)-(5.47), we obtain the admissible set $\mathcal{T}$ as a function of τ_s (see Figure 5.3). This figure is obtained under the assumption

$$\frac{\kappa_3}{\delta_3} < \min\left\{\frac{NM\delta_1}{\kappa_1}, \frac{\delta_2}{k\kappa_2}\right\}. \tag{5.48}$$

If (5.48) is not satisfied, the set $\mathcal{T}$ is empty, otherwise, $\mathcal{T} = [\tau_1, \tau_2]$ where

$$\begin{aligned} \tau_1 &= \frac{\kappa_3}{\delta_3}\tau_s, \\ \tau_2 &= \min\left\{\frac{NM\delta_1\tau_s - RTD}{\kappa_1}, \frac{\frac{\delta_2}{k}\tau_s - RTD}{\kappa_2}\right\}. \end{aligned}$$

Let τ_s^* be the value of τ_s for which $\tau_1 = \tau_2$ (see Figure 5.3). Then, if $\tau_s \geq \tau_s^*$, the constrained optimal update period is

$$T^* = \begin{cases} T_0, & \text{if } T_0 \in [\tau_1, \tau_2], \\ \tau_1, & \text{if } T_0 < \tau_1, \\ \tau_2, & \text{if } T_0 > \tau_2. \end{cases}$$

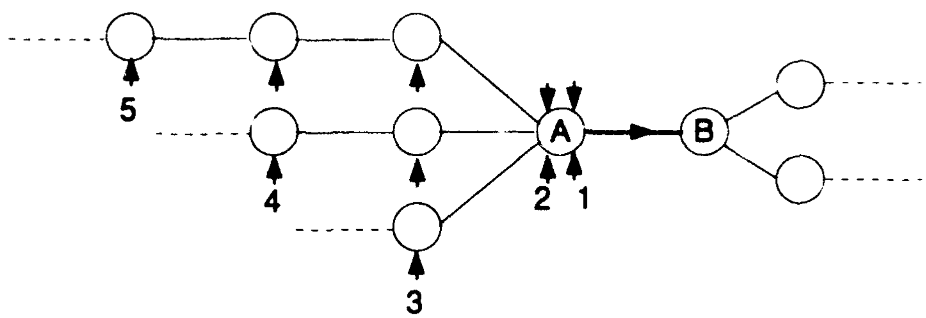

Figure 5.4: Network topology.

Section 5 below illustrates the effects of T on the transient behavior of the network using numerical simulations.

5.4 DESIGN EXAMPLE AND NUMERICAL SIMULATIONS

This section presents a design example and simulation results. We start by describing the network and the input traffic. Then, we carry out both the robust and adaptive designs followed by an examination of the effects of the update period T and of the randomness in the input traffic. Finally, we illustrate the recovery of the network from congestion.

5.4.1 The network and the input traffic

Consider a network where links have a delay-bandwidth product of 10 packets, i.e., the propagation delay over a link corresponds to the time for transmitting 10 packets. This is the case, for instance, of 45 Mbps channels when 1 kbyte packets are sent over a 340-mile link. Let τ_s be the transmission time of a packet and let c be the transmission capacity in packets/slot where the slot duration is equal to the update period T ($T = c\tau_s$).

For this design example, we assume that the network topology, and the input traffic, are such that the following properties are satisfied:

(a). Only one link is overloaded.

(b). The input traffic is such that

$$n_0 = 4,$$
$$n_1 = 3,$$
$$n_2 = 2,$$
$$n_3 = 1,$$
$$n_i = 0, \; i > 3,$$

where n_i is the maximum number of connections that can flow through the overloaded link with the source node located i hops from this link.

These properties are satisfied, for instance, by the network and the traffic shown in Figure 5.4 and Table 5.2, respectively. Figure 5.4 shows only a subset of the network nodes along with the sources of the five connections labeled according to the first column of Table 5.2. This table describes the activity of these sources during the 1000 time slots of the experiment where the slot duration, T_1, is defined below. The presence of the other five unlabeled connections as potential traffic sources is taken into account in the design of the controller. The destination nodes, which are not shown in Figure 5.4, can be any of the nodes downstream of node A. Note that the input traffic becomes heavy, i.e., larger than the link capacity, starting from time $t = 100T_1$.

Connection #	# of hops	initiated at time	terminated at time	input rate ($\times c$)
1	0	50	700	0.6
2	0	250	1000	0.7
3	1	100	850	0.6
4	2	400	1000	0.5
5	3	550	1000	0.4

Table 5.2: Input traffic.

5.4.2 Robust design

As it follows from Figure 5.4, the largest round trip propagation delay, RTD, corresponds to 6 hops: 3 hops in each of the forward and reverse paths of connection 5. Since the delay bandwidth product is assumed to be 10 and the transmission time of a packet is τ_s, this implies that $RTD = 6 \times 10\tau_s = 60\tau_s$. We choose the update period to be equal to the largest round trip propagation delay, $T_1 = 60\tau_s$, i.e., $c = 60$, and choose $x^0 = 30$ packets.

Let $\hat{\tau}_{iA}$ denote the delay, measured in time slots T_1, that connections originating i hops from node A incur until they arrive at node A. Then, neglecting the processing delay, and using equations (5.6), (5.5) and (5.22) we have, respectively,

$$\begin{cases} \hat{\tau}_{0A} = 0 \\ \hat{\tau}_{1A} = \frac{T_s+T_p}{cT_s} = \frac{11T_s}{60T_s} = \frac{11}{60} \\ \hat{\tau}_{2A} = 2\hat{\tau}_{1A} = \frac{22}{60} \\ \hat{\tau}_{3A} = 3\hat{\tau}_{1A} = \frac{33}{60} \end{cases} \Rightarrow \begin{cases} d_{0A} = 0,\ \theta_{0A} = 0 \\ d_{1A} = \lceil\frac{11}{60}\rceil = 1,\ \theta_{1A} = 1 - \frac{11}{60} = \frac{49}{60} \\ d_{2A} = \lceil\frac{22}{60}\rceil = 1,\ \theta_{2A} = 1 - \frac{22}{60} = \frac{38}{60} \\ d_{3A} = \lceil\frac{33}{60}\rceil = 1,\ \theta_{3A} = 1 - \frac{33}{60} = \frac{27}{60} \end{cases} \Rightarrow \begin{cases} d_0 = 0, \\ d_1 = 2, \\ d_2 = 2, \\ d_3 = 2. \end{cases}$$

Thus, the maximum time delay D, as defined by (5.25) is equal to 2. Therefore,

$$Q_k = (k\alpha_0, k\alpha_1, \beta_0, \beta_1, \beta_2)^T,\ k \in (0, k^*), \tag{5.49}$$

where, as it follows from Theorem 5.2,

$$\begin{aligned} \alpha_0 &= \tfrac{D+4}{2(D+1)} = 1, \\ \alpha_1 &= -\tfrac{D+2}{2(D+1)} = -2/3, \\ \beta_0 &= \tfrac{3D}{2(D+1)} = 1, \\ \beta_1 &= \tfrac{D-4}{2(D+1)} = -1/3, \\ \beta_2 &= \tfrac{D-6}{2(D+1)} = -2/3, \end{aligned}$$

and

$$\bar{L} = (\bar{l}_0, \bar{l}_1, \bar{l}_2)^T,$$

where, as it follows from (5.40),

$$\begin{aligned} \bar{l}_0 &= n_0(1-\theta_{0A}) = 4, \\ \bar{l}_1 &= n_1\theta_{1A} + n_2\theta_{2A} + n_3\theta_{3A} = 4.17, \\ \bar{l}_2 &= n_1(1-\theta_{1A}) + n_2(1-\theta_{2A}) + n_3(1-\theta_{3A}) = 1.83. \end{aligned}$$

With this $\bar{L}$, the solution of the optimization problem of Theorem 5.3 is $k^* = 0.156$.

The behavior of the network with Q_k defined by (5.49) for $k = 0.15$ and $k = 0.075$ is shown in Figure 5.5.a and Figure 5.5.b, respectively. These figures were obtained by a numerical solution of equations (5.16)-(5.18) with corresponding Q_k's. The results can be summarized as follows:

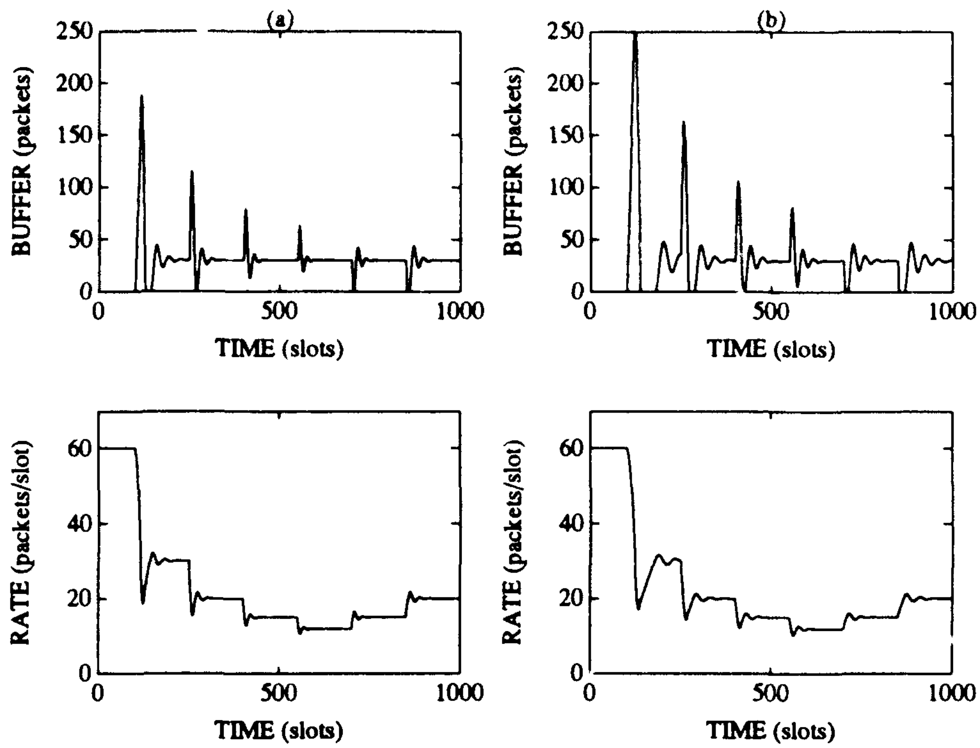

Figure 5.5: Robust design.

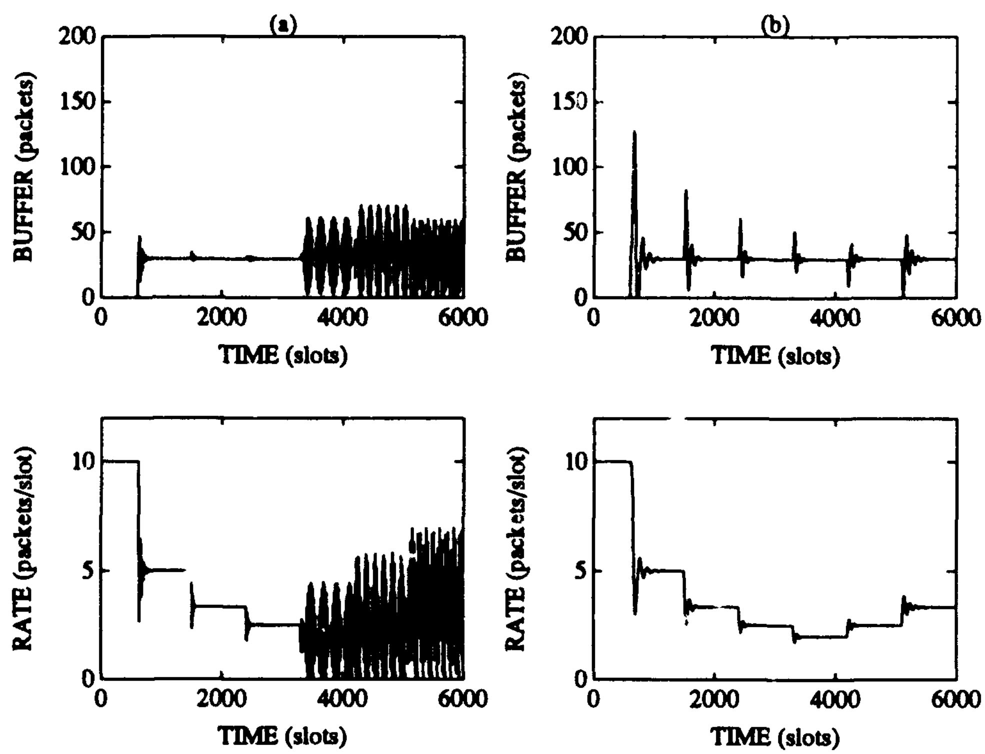

Figure 5.6: Effect of T.

(a). Smaller values of k result in a slower response and larger buffer overshoot. As it will be discussed below, faster response and smaller buffer overshoot can be achieved by adopting smaller update periods. Another way of reducing the buffer overshoot, which is not discussed here, is to impose a slow-start on connections to avoid sharp increases in the input traffic.

(b). When the traffic is heavy (starting at time $t = 100$), the buffer is kept at $x^0 = 30$, with fluctuations whenever a connection is initiated (at $t =100$, 250, 400 and 550) or terminated (at $t = 700$ and 850).

(c). When the input traffic changes, the rate $q(n)$ converges to a new steady state where the link bandwidth is reallocated in order to accommodate all incoming traffic in a fair way.

5.4.3 Effect of the update period T

Controller (5.49) was derived for the update period $T_1 = 60\tau_s$. As it follows from Figure 5.5, this T results in quite large overshoot and slow response. It might seem reasonable to decrease T in order to eliminate these problems. According to the theory developed in Sections 3 and 4, this, however, is not the case: the utilization of a controller designed for a larger T in a network with smaller T may bring about an instability. This effect is illustrated in Figure 5.6.a where Q_k of (5.49) with $k = 0.15$ is used when T is chosen to be $T_2 = 10\tau_s$ instead of $60\tau_s$. Obviously, the oscillatory behavior observed is unacceptable in most applications.

The way to improve the speed of the response is not only to reduce T but also to redesign the controller appropriately (Theorems 5.1, 5.2, and 5.3). Specifically, for $T_2 = 10\tau_s$, repeating the design steps described above, we obtain

$$\begin{cases} \hat{\tau}_{0A} = 0 \\ \hat{\tau}_{1A} = \frac{T_s+T_p}{cT_s} = 1.1 \\ \hat{\tau}_{2A} = 2\hat{\tau}_{1A} = 2.2 \\ \hat{\tau}_{3A} = 3\hat{\tau}_{1A} = 3.3 \end{cases} \Rightarrow \begin{cases} d_{0A} = 0,\ \theta_{0A} = 0 \\ d_{1A} = 2,\ \theta_{1A} = 0.9 \\ d_{2A} = 3,\ \theta_{2A} = 0.8 \\ d_{3A} = 4,\ \theta_{3A} = 0.7 \end{cases} \Rightarrow \begin{cases} d_0 = 0, \\ d_1 = 4, \\ d_2 = 6, \\ d_3 = 8. \end{cases}$$

Thus, the maximum time delay D is equal to 8 and therefore

$$Q_k = (k\alpha_0, k\alpha_1, \beta_0, \beta_1, \ldots, \beta_8)^T,\ k \in (0, k^*), \tag{5.50}$$

where,

$$\begin{aligned} &\alpha_0 = 2/3,\ \alpha_1 = -5/9, \\ &\beta_0 = 4/3,\ \beta_1 = 2/9,\ \beta_2 = 1/9, \\ &\beta_3 = 0,\ \beta_4 = -1/9,\ \beta_5 = -2/9, \\ &\beta_6 = -1/3,\ \beta_7 = -4/9,\ \beta_8 = -5/9, \end{aligned}$$

and

$$\bar{L} = (\bar{l}_0, \bar{l}_1, \ldots, \bar{l}_8)^T,$$

where,

$$\begin{aligned} &\bar{l}_0 = n_0(1-\theta_{0A}) = 4,\ \bar{l}_1 = 0,\ \bar{l}_2 = 0, \\ &\bar{l}_3 = n_1\theta_{1A} = 2.7,\ \bar{l}_4 = n_1(1-\theta_{1A}) = 0.3, \\ &\bar{l}_5 = n_2\theta_{2A} = 1.6,\ \bar{l}_6 = n_2(1-\theta_{2A}) = 0.4, \\ &\bar{l}_7 = n_3\theta_{3A} = 0.7,\ \bar{l}_8 = n_3(1-\theta_{3A}) = 0.3. \end{aligned}$$

The solution of the optimization problem gives $k^* = 0.160$.

Equations (5.16)-(5.18) with $T = 10\tau_s$ and Q_k given in (5.50) with $k = 0.15$ have been solved numerically. The results, shown in Figure 5.6.b, lead to the following conclusions:

(a). Adopting a smaller update period T along with the appropriate controller results in a stable operation of the network with faster response and smaller buffer overshoot.

(b). When the network operates with a controller designed for larger T, an instability could be brought about (see Figure 5.6.a where large oscillations take place when connection 5 is active).

Note that the total simulation time ($6000T_2$) is the same as in Figure 5.5 ($1000T_1$) since $T_2 = T_1/6$.

5.4.4 Adaptive design

In this section we design an adaptive controller for both update periods considered above, T_1 and T_2.

In the case of T_1, the characteristic polynomial $P_\Gamma(\lambda)$ in (5.35) has dimension $D+2=4$ and therefore, the gains are specified by the choice of 4 desired closed loop poles. For instance, if we choose the desired characteristic polynomial as

$$P_\Gamma(\lambda) = \lambda^2(\lambda-\lambda_1)(\lambda-\lambda_2),\ \lambda_{1,2} = 0.4 \pm j0.3,$$

then, as it follows from (5.38), the adaptation of the controller gains $Q = (\alpha_0, \alpha_1, \beta_0, \beta_1, \beta_2)^T$ to changes in L takes the following form

$$Q(n) = [M(L(n-1))]^{-1}\tilde{\Gamma}, \tag{5.51}$$

where

$$\tilde{\Gamma} = (1.2,\ -0.75,\ 0,\ 0,\ 0)^T.$$

In the case $T_2 = 10\tau_s$, we choose the desired behavior according to

$$P_\Gamma(\lambda) = \lambda^8(\lambda-\lambda_1)(\lambda-\lambda_2),\ \lambda_{1,2} = 0.4 \pm j0.3\,.$$

As a result, $Q = (\alpha_0, \alpha_1, \beta_0, \beta_1, \ldots, \beta_8)^T$ is given by

$$Q(n) = [M(L(n-1))]^{-1}\tilde{\Gamma}, \tag{5.52}$$

where

$$\tilde{\Gamma} = (1.2,\ -0.75,\ 0,\ 0,\ \ldots,\ 0)^T.$$

Networks with controllers (5.51) and (5.52) have been simulated by a numeric solution of equations (5.26)-(5.27). The results are shown in Figure 5.7.a and Figure 5.7.b, respectively. From these figures we conclude:

(a). The transient behavior of the network under adaptive control is much better than under robust control.

(b). Higher updating rates result in faster response and smaller buffer overshoot.

(c). The superior performance exhibited by the adaptive controller justifies its selection for implementation despite the extra computational time it requires.

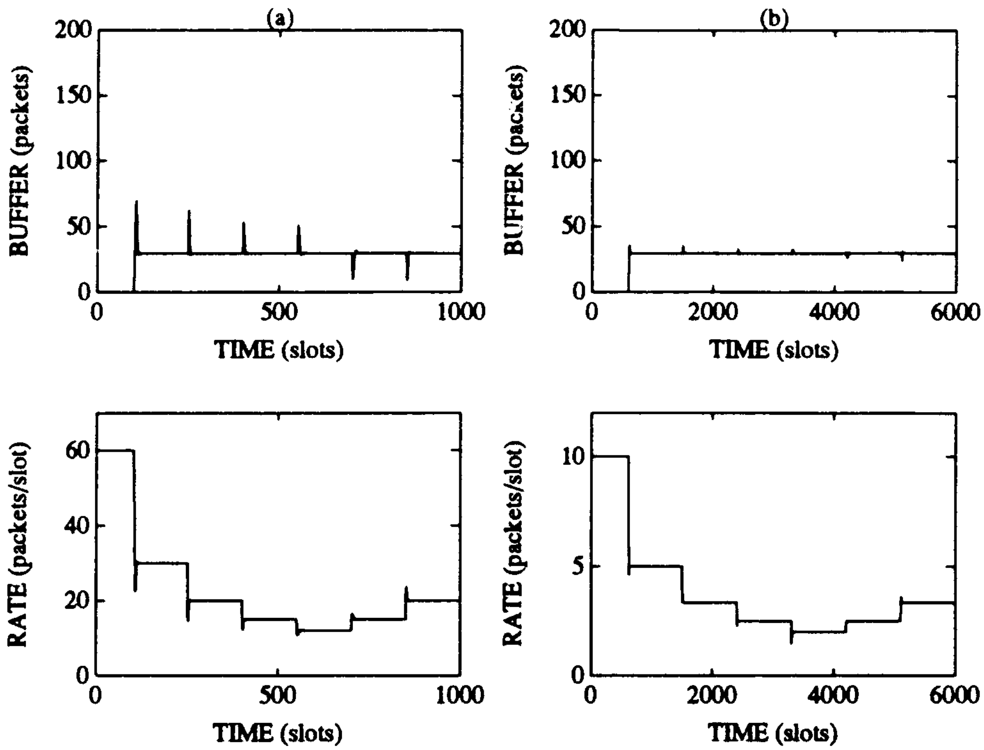

Figure 5.7: Adaptive design.

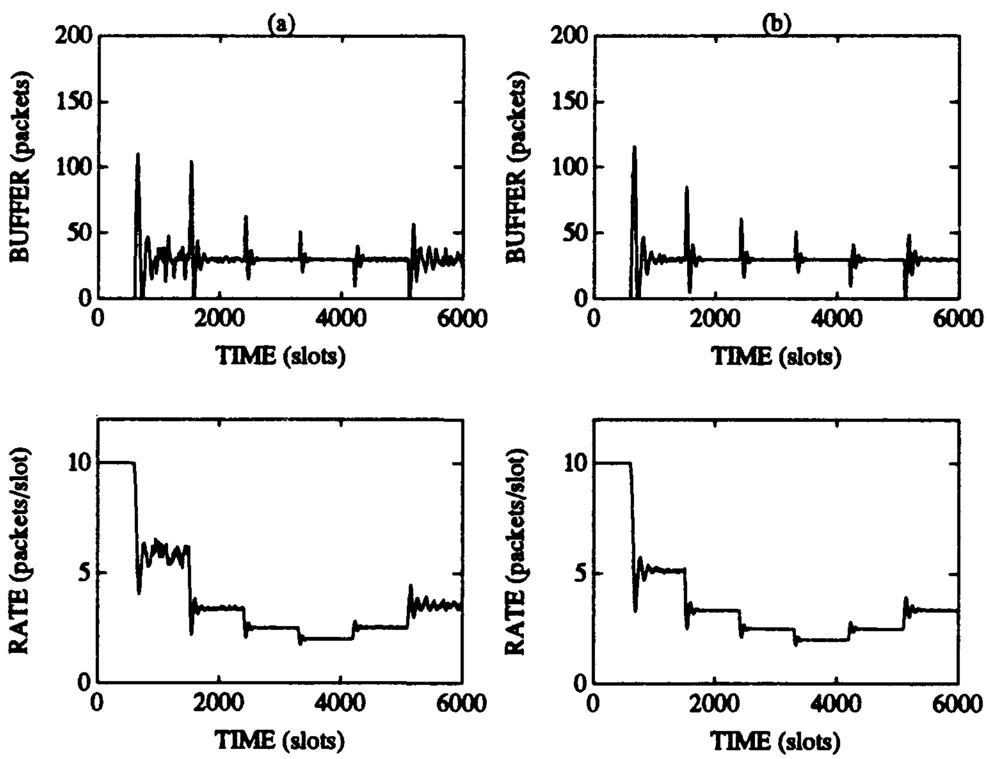

Figure 5.8: Random input.

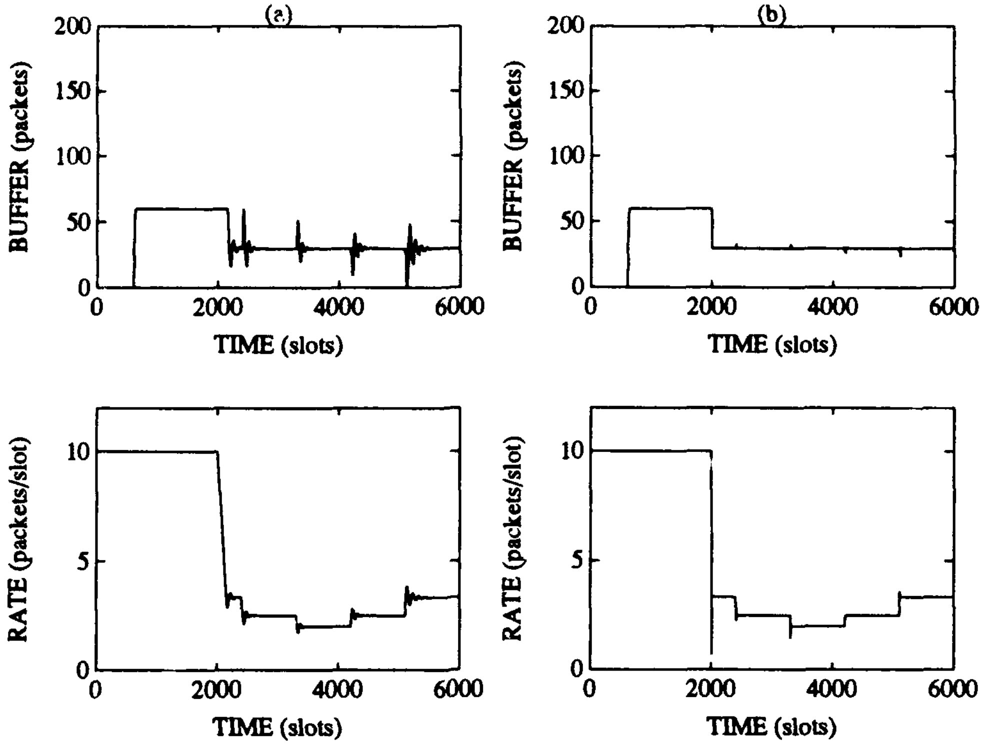

Figure 5.9: Congestion recovery.

5.4.5 Effect of random traffic

We have assumed throughout this report that the input traffic is specified by a constant rate that may change from time to time in a random manner. In reality, however, the input traffic is random in the sense that the rate r_i^0 of connection i is a random number $r_i^0(n)$. In order to investigate the behavior of the control scheme developed in this report, we assume that $r_i^0(n)$, $n = 1, 2, \ldots$, is a sequence of independent random variables uniformly distributed over the range $[(1-\varepsilon)r_i^0, (1+\varepsilon)r_i^0]$. The behavior of the network with controller (5.50) and $k = 0.15$ is shown in Figure 5.8.a, for $\varepsilon = 0.6$, and Figure 5.8.b, for $\varepsilon = 0.3$. The results can be summarized as follows:

(a). Due to the structure of the control protocol in (5.17), the randomness will not appear in the state trajectories if $r_i^0(n)$ remains greater than $q(n)$ for all active connections i. This is more likely to happen if more connections are being quenched since, as it follows from (5.30), the steady state q is inversely proportional to the number l of quenched flows. This is the reason why the plots in Figure 5.8 exhibit more fluctuations when smaller number of connections are active.

(b). During the time intervals when the randomness in the input is reflected in the dynamics, the state trajectories fluctuate but remain close to the deterministic trajectories shown in Figure 5.6.b.

(c). As the level of randomness increases, i.e., as ε gets larger, the level of fluctuations increases.

5.4.6 Congestion recovery

The previous simulations dealt with congestion prevention where, starting from an empty network, the role of the controller is to prevent congestion from taking place. In this final experiment, we examine how the controller ensures recovery from congestion. Under the same input traffic pattern described above, we operate the network without the controller. The buffer, with a capacity of 60 packets, fills up and overflows. When the robust controller (5.50), with slot duration $T = T_2$ and $k = 0.15$, is turned on at time $t = 2000T_2$, we obtain the dynamic behavior shown in Figure 5.9.a. Figure 5.9.b corresponds to the case where the adaptive controller (5.52) is utilized. From these figures we conclude that

(a). Both the robust and adaptive controllers are capable of ensuring congestion recovery under heavy traffic situations.

(b). With the robust controller, the node recovers from congestion in about 80 time slots whereas a much faster recovery is achieved with the adaptive controller.

6 SUMMARY OF THE MOST IMPORTANT RESULTS - PART 2 : THE CASE OF MULTIPLE CONGESTED NODES

In this second part or the report we discuss the case of multiple congested nodes. The network, the input traffic and the control architecture, analyzed here, are defined as follows where most of the assumptions have been introduced in Part 1 and are restated here with a slight change in notation:

The network

(i). The network employs a store-and-forward datagram service where users are serviced without prior reservation of resources.

(ii). The network consists of S switching nodes and N communication links. Let $\mathcal{S} = \{1, 2, \ldots, S\}$ denote the set of nodes and $\mathcal{N} = \{1, 2, \ldots, N\}$ denote the set of links. For each node $a \in \mathcal{S}$, let $\mathcal{O}(a) \subset \mathcal{N}$ denote the set of its outgoing links and for each link $i \in \mathcal{N}$, let s_i be the node of which i is an outgoing link, i.e., the origin node of link i.

(iii). Each link i has a transmission capacity of $1/\tau_s$ packets/sec, where τ_s is the transmission time of a packet, and a propagation delay of τ_p^i sec which depends on the transmission medium and on the length of the link. We assume that packets have the same fixed size.

(iv). Each node has a processing capacity of $1/\tau_{pr}$ packets/sec, where τ_{pr} is the processing time of a packet. The processing capacity of each node is assumed to be larger than the total transmission capacity of its incoming links. The case where processing is the bottleneck can be approached similarly.

(v). The network traffic consists of flows corresponding to each source-destination pair (ab), where a and $b \in \mathcal{S}$. The pair (ab) will be referred to as the (ab)-type traffic or the (ab) connection. Let $\mathcal{C}$ denote the set of all such connections.

(vi). For each (ab) connection, the source at node a sends packets to the destination at node b through a sequence of links referred to as the path of the connection and denoted $p(ab)$. The routing policy which determines the path of each connection is assumed to be static. Let $C(i)$ be the set of all connections (ab) which traverse link i and let e_{ab} denote the first link in the path of connection (ab).

(vii). Each node has a buffer for storing packets waiting to be transmitted on one of its outgoing links. Let $x_i, i \in \mathcal{N}$, denote the number of packets buffered for transmission on link i and referred to as link i's buffer.

The control architecture

(viii). Each node a has a congestion controller associated with each outgoing link $i \in \mathcal{O}(i)$. This controller periodically computes a traffic admission rate q_i based on a **control algorithm** (see below) that uses local information to node a: the difference between x_i and some threshold x^0 and the control decision q_i at present and in the finite past. The threshold x^0 is introduced to insure a desired bound on the time delay in the steady state and to prevent congestion or underutilization of the transmission capacity during the transient periods of control.

(ix). The control algorithm updates take place every T sec, where T is the time to transmit c packets ($T = c\tau_s$), i.e., a new control update is generated after every c packets are transmitted. Thus, the controller time is slotted with the slot duration, $[n, n+1)$, $n = 0, 1, \ldots$, equal to T. Note that a tradeoff is involved in the choice of the update period T since shorter T's lead to better responsiveness to changes in the input traffic but require the processors to devote more time updating the feedback signals.

(x). Each node sends the computed control information (q_i's) to the sources along a fixed feedback path, usually the reverse direction of the traffic (assuming bidirectional links which is often the case). This control information is serviced with high priority and is carried either in separate packets or along with data or acknowledgment packets. The sources respond to the control information received according to the **control protocol** defined below.

The input traffic

(xi). The input traffic is viewed as a fluid where packet boundaries are ignored. This fluid model approximation is a typical assumption in the analysis of dynamic congestion control mechanisms [15]-[16].

(xii). For each connection (ab), let r^0_{ab} denote the rate (in packets per slot) of its traffic demand. We assume that during the settling time of the system, the traffic demand is constant but otherwise arbitrary and unknown. This assumption could be understood in the sense that the input traffic is piecewise constant with the jumps occurring seldom enough so that the transients of the system have time to settle down between two consecutive jumps.

System (i)-(xii) has been introduced and analyzed in Part 1 for the case of a single congested node. Specifically, Part 1 used the following rate-based control algorithm and control protocol:

The control algorithm

(xiii). The traffic admission rate $q_i(n+1)$ calculated by link i at time n is given by

$$q_i(n+1) = \mathrm{Sat}_{q^0}\left\{ q_i(n) - \sum_{j=0}^{J_i} \alpha_{ij}\left(x_i(n-j) - x^0\right) - \sum_{k=0}^{K_i} \beta_{ik}\, q_i(n-k) \right\}, \quad i \in \mathcal{N}, \qquad (6.1)$$

where $q^0 > c$, α_{ij} and β_{ik} are the controller gains, J_i and K_i are non-negative integers and

$$\mathrm{Sat}_a(z) = \begin{cases} 0 & \text{if } z \le 0, \\ a & \text{if } z \ge a, \\ z & \text{otherwise}. \end{cases}$$

The control protocol

(xiv). During the time slot $[n, n+1)$, the (ab)-type traffic is admitted with the rate

$$r_{ab}(n) = \min\left\{ q_m(n+1-d^{ab}_{ma}), m \in p(ab);\ r^0_{ab} \right\}, \quad (ab) \in \mathcal{C}, \qquad (6.2)$$

where d^{ab}_{ma} is the feedback delay of connection (ab) with respect to link m. When the minimum in (6.2) corresponds to the admission rate of link $j \in p(ab)$, connection (ab) is said to be **throttled** by link j.

(xv). Traffic at each link is serviced according to the First-Come-First-Serve (FCFS) priority discipline.

Unfortunately, the methods developed in Part 1 do not seem to generalize directly to the case of multiple congested nodes, which is the main topic of this part of the report. Two indirect generalizations are, however, possible. The first one is based on a modification of the FCFS priority scheme and the second on the assumption that the control update period is large. This part of the report is devoted to the former case. Specifically, assume that:

(xv′). Traffic at each link is transmitted according to the Remote-Throttled-First-Serve (RTFS) priority discipline: Each link i first transmits traffic of connections not throttled by itself including traffic not throttled at all. When no such traffic is waiting for service, it starts transmitting traffic of connections throttled by itself. Connections within each of the two categories (throttled and nonthrottled by link i) are serviced on a first-come-first-serve basis. To implement this strategy, each packet carries a stamp identifying the throttling link of the connection it belongs to.

Given this discipline, this report presents a method for choosing the gains, α_{ij} and β_{ik} for each link $i \in \mathcal{N}$, which result in a controller that prevents network congestion. Moreover, we show by simulations that the same gains ensure a congestion-free stable operation of the network under the FCFS strategy. This result, however, is not proved analytically.

6.1 EVOLUTION EQUATIONS

6.1.1 Preliminaries

Let $x_i^{ab}(n)$ denote the amount of (ab)-type traffic at the buffer of link i at time n. Let $\psi_i^{ab}(n)$ and $\psi_{i^-}^{ab}(n)$ denote the rate at which the (ab)-type traffic is transmitted on link i and on the link that precedes link i along the path of connection (ab), respectively, during the time slot $[n, n+1)$. Note that, in terms of this notation, $\psi_{i^-}^{ab}(n)$ is equal to $r_{ab}(n)$ if link i is the first link in the path of connection (ab), i.e., $\psi_{e_{ab}^-}^{ab}(n) = r_{ab}(n)$. Then, the evolution of the system defined by assumptions (i)-(xiv) is described by the following equations:

$$x_i^{ab}(n+1) = x_i^{ab}(n) + \psi_{i^-}^{ab}(n - \tau_{i^-}^{ab}) - \psi_i^{ab}(n), \tag{6.3}$$

$$q_i(n+1) = \mathrm{Sat}_{q^0}\left\{ q_i(n) - \sum_{j=0}^{J_i} \alpha_{ij}\left(x_i(n-j) - x^0\right) - \sum_{k=0}^{K_i} \beta_{ik}\, q_i(n-k) \right\}, \tag{6.4}$$

$$\psi_{e_{ab}^-}^{ab}(n) = r_{ab}(n) = \min\left\{ q_m(n+1-d_{ma}^{ab}), m \in p(ab); r_{ab}^0 \right\},\ i \in \mathcal{N}, (ab) \in C(i), \tag{6.5}$$

where $\tau_{i^-}^{ab}$ is the one-hop delay for the link that precedes link i on the path of connection (ab), d_{ma}^{ab} is the feedback delay of connection (ab) with respect to link m, and α_{ij} and β_{ik} are, as before, the controller gains associated with link i. We use the following notations for time delays:

τ_{ia}^{ab} : feedback delay from node s_i to node a,

τ_{ai}^{ab} : forward delay from node a to node s_i,

τ_{ij}^{ab} : forward delay from node s_i to node s_j,

$\tau_{iaj}^{ab} = \left\lceil \tau_{ia}^{ab} \right\rceil + \tau_{aj}^{ab}$: round trip delay from node s_i to node s_j, where $\lceil z \rceil$ denotes the smallest integer greater or equal to z. The forward (feedback) delay from node a_1 to node a_2 is equal to the sum of the one-hop delays of all links between node a_1 and node a_2 in the forward (feedback) path of connection (ab). The one-hop delay for link i is defined as the sum of the packet transmission delay at the origin node of link i, the propagation delay on link i, and the packet processing delay at the destination node of link i, all expressed in time slots T, i.e.,

$$\tau_i = \frac{\tau_s + \tau_p^i + \tau_{pr}}{T}. \tag{6.6}$$

The feedback delay d_{ma}^{ab} in (6.5) is equal to $\lceil \tau_{ma}^{ab} \rceil$ since, due to the fact that q_m is generated every T, the most recent feedback information available at node a at time n is $q_m(n + 1 - d_{ma}^{ab})$.

Equations (6.3)-(6.5) are not closed with respect to the unknowns since ψ_i^{ab} is not expressed in terms of the states x_i^{ab} and q_i and the demands r_{ab}^0. They can be made closed under the FCFS discipline since, in this case, $\psi_i^{ab}(n)$ can be modeled as

$$\psi_i^{ab}(n) = \min\left\{ x_i^{ab}(n), \frac{x_i^{ab}(n)}{x_i(n)} c \right\},$$

where $x_i = \sum_{(ab)\in C(i)} x_i^{ab}$ is the total occupancy of the buffer of link i. However, the substitution of this expression in (6.3) results in a system that seems to be impossible to analyze. Even the linearized form of these equations seem to be too complex for analytical investigation due to its high dimensionality and linearized dynamics complexity. These difficulties are avoided when the RTFS discipline, introduced in (xv′), is used. However, under this priority discipline, equations (6.3)-(6.5) cannot be made closed since ψ_i^{ab} does not admit a simple representation. Therefore, we proceed as follows: First, we consider the steady state form of (6.3)-(6.5) which allows us to determine the steady state values of x_i and q_i irrespective of the priority discipline. Then, we show that, in some neighborhood of this steady state, the dynamic equations of only a subset of the network links deviate from the steady state, and the consideration of only this subset is sufficient for the network analysis. This subset is referred to as the **bottleneck subnetwork**. Finally, we show that, under the RTFS discipline, a closed form expression for ψ_i^{ab} can be derived and, due to the resulting decoupling of the dynamics, we show that it is sufficient to consider the dynamic equations for x_i and q_i only, without involving the buffer occupancies x_i^{ab}. The bottleneck subnetwork is introduced below followed by the derivation of the local dynamic equations.

6.1.2 The bottleneck subnetwork

The balance equation for $x_i(n)$, derived from (6.3), is

$$x_i(n+1) = x_i(n) + f_i(n) - \psi_i(n),\ i \in \mathcal{N}, \tag{6.7}$$

where

$$f_i(n) \triangleq \sum_{(ab)\in C(i)} \psi_{i^-}^{ab}(n - \tau_{i^-}^{ab})$$

is the aggregate input flow to link i, and

$$\psi_i(n) \triangleq \sum_{(ab)\in C(i)} \psi_i^{ab}(n)$$

is the total transmission rate of link i during $[n, n+1)$. This rate is equal to the link capacity c as long as the buffer is not empty. Therefore, (6.7) is equivalent to

$$x_i(n+1) = \mathrm{sg}\{x_i(n) + f_i(n) - c\}, i \in \mathcal{N}, \tag{6.8}$$

where

$$\mathrm{sg}(x) = \begin{cases} x & \text{if } x \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

Let $\bar{z}$ denote the steady state value of $z(n)$. Then, (6.3)-(6.5) and (6.8) imply, respectively,

$$\bar{\psi}_{i^-}^{ab} = \bar{\psi}_i^{ab}, \tag{6.9}$$

$$\bar{q}_i = \mathrm{Sat}_{q^0}\left\{\bar{q}_i - \left(\sum_{j=0}^{J_i}\alpha_{ij}\right)\left(\bar{x}_i - x^0\right) - \left(\sum_{k=0}^{K_i}\beta_{ik}\right)\bar{q}_i\right\}, \tag{6.10}$$

$$\bar{\psi}_{e_{ab}^-}^{ab} = \bar{r}_{ab} = \min\left\{\bar{q}_m, m \in p(ab); r_{ab}^0\right\}, \tag{6.11}$$

$$\bar{x}_i = \mathrm{sg}\left\{\bar{x}_i + \bar{f}_i - c\right\},\ i \in \mathcal{N},\ (ab) \in C(i). \tag{6.12}$$

Equations (6.9) and (6.11) imply that

$$\bar{\psi}_i^{ab} = \bar{r}_{ab},\ (ab) \in \mathcal{C},\ i \in p(ab).$$

Therefore,

$$\bar{f}_i = \sum_{(ab)\in C(i)} \bar{\psi}_{i^-}^{ab} = \sum_{(ab)\in C(i)} \bar{r}_{ab} = \sum_{(ab)\in C(i)} \min\left\{\bar{q}_m, m \in p(ab); r_{ab}^0\right\}.$$

Let C_{im} be the set of connections in $C(i)$ that are throttled by link m and let l_{im} be its cardinality. Let C_{i0} be the set of connections in $C(i)$ that are not throttled by any link. Then, if we assume that

$$\bar{q}_m \neq \bar{q}_n \neq r_{ab}^0,\ (ab) \in \mathcal{C},\ m,n \in p(ab),\ m \neq n, \tag{6.13}$$

then C_{im}, $i = 1,2,\ldots,N, m = 0,1,\ldots,N$, are uniquely defined and we have

$$\bar{f}_i = \sum_{m=1}^{N} l_{im}\bar{q}_m + r_i^0, \tag{6.14}$$

where $r_i^0 = \sum_{(ab)\in C_{i0}} r_{ab}^0$ is the cumulative rate of nonthrottled connections in $C(i)$. We adopt the following assumption about the input traffic which is less conservative than (6.13):

$$\forall (ab) \in C(i): \quad \begin{array}{l} \text{if } \min\{\bar{q}_m, m \in p(ab), r_{ab}^0\} = \bar{q}_n, \text{ then } \bar{q}_n \neq r_{ab}^0, \bar{q}_n \neq \bar{q}_m, m \in p(ab), m \neq n, \\ \text{if } \min\{\bar{q}_m, m \in p(ab), r_{ab}^0\} = r_{ab}^0, \text{ then } r_{ab}^0 \neq \bar{q}_m, m \in p(ab). \end{array} \tag{6.15}$$

This assumption is used below to linearize the system in the vicinity of its steady state since the function $h(q) = \min\{q, a\}$ is not differentiable at $q = a$.

Obviously, the steady state flow $\bar{f}_i$ in (6.14) can be either smaller or larger than the link capacity c. If $\bar{f}_i < c$, then (6.12) implies that $\bar{x}_i = 0$. In addition, since, as it is shown below, the controller gains must satisfy, among others, the following requirements:

$$\sum_{j=0}^{J_i}\alpha_{ij} > 0 \ \text{ and } \ \sum_{k=0}^{K_i}\beta_{ik} = 0\,,\ \ i \in \mathcal{N}, \tag{6.16}$$

it follows from (6.10) that $\bar{q}_i = q^0$. Therefore, any link i with steady state flow less than its capacity (**underloaded** link) has its buffer at zero and its control signal saturated at q^0.

Furthermore, since $\bar{f}_i = l_{ii}\bar{q}_i + \sum_{j=1, j\neq i}^{N} l_{ij}\bar{q}_j + r_i^0$, $\bar{q}_i = q^0 > c$, and l_{ii} is a nonnegative integer, we conclude that $l_{ii} = 0$ (and as a result $l_{ji} = 0, j \in \mathcal{N}$) for all underloaded links.

On the other hand, if $\bar{f}_i \geq c$, then (6.12) implies that $\bar{f}_i = c$. When $\bar{f}_i = c$ and $l_{ii} > 0$, the link is said to be **overloaded**. There may exist a link i with $\bar{f}_i = c$ but $l_{ii} = 0$. Such a link is neither overloaded nor underloaded since the steady state flow is equal to the link capacity but no traffic is throttled. We assume that no link operates in this regime. For overloaded links, since $\bar{f}_i = c$ and $l_{ii} > 0$, (6.14) implies that $0 < \bar{q}_i \leq c < q^0$ and therefore, (6.10) gives

$$\bar{x}_i = x^0 - \frac{\sum_{k=0}^{K_i} \beta_{ik}}{\sum_{j=0}^{J_i} \alpha_{ij}} \bar{q}_i.$$

In order to achieve $\bar{x}_i = x^0$, we impose the constraint

$$\sum_{k=0}^{K_i} \beta_{ik} = 0. \tag{6.17}$$

Let $\mathcal{N}_1$ and $\mathcal{N}_2$ denote the sets of overloaded and underloaded links, respectively, with $\mathcal{N} = \mathcal{N}_1 \cup \mathcal{N}_2$. Assume, without loss of generality, that $\mathcal{N}_1 = \{1, 2, \ldots, N_1\}$, $N_1 \leq N$. Then, in the neighborhood of the steady state, $f_i(n) < c$, $i \in \mathcal{N}_2$, since $\bar{f}_i < c$ and $f_i(n)$ is a continuous function of the network state (see Lemma 6.2 below). This implies that, in that neighborhood, $x_i(n)$ and $q_i(n)$ do not deviate from their steady state values $\bar{x}_i = 0$ and $\bar{q}_i = q^0$, and as a result, no queueing delay is experienced at buffers of underloaded links. Therefore, for the local analysis, it suffices to consider the bottleneck subnetwork consisting of all links $i \in \mathcal{N}_1$.

6.1.3 Local dynamic equations

As it follows from the previous subsection, the steady state flow through an overloaded link i is given by

$$\bar{f}_i = \sum_{j=1}^{N_1} l_{ij}\bar{q}_j + r_i^0, \; i \in \mathcal{N}_1,$$

or in matrix notation

$$\bar{F} = \Lambda \bar{Q} + R^0, \tag{6.18}$$

where $\bar{F}$, $\bar{Q}$, and R^0 are N_1-dimensional vectors with the i-th entry being $\bar{f}_i$, $\bar{q}_i$, and r_i^0, respectively, and Λ is the $N_1 \times N_1$-matrix of the l_{ij}'s with positive diagonal entries l_{ii}'s. In this subsection, we derive an expression of the flow $F(n)$ in the neighborhood of its steady state value $\bar{F}$.

Lemma 6.1 *Under the control protocol introduced in assumption (xiv), Λ is an essentially triangular matrix, i.e., there exists a relabeling of $q_i, i = 1, 2, \ldots, N_1$, that results in Λ being triangular.* □

Proof: See Appendix 3.

Assume that the links have been relabeled in such a way that Λ is a lower triangular matrix. Therefore,

$$\bar{f}_i = \sum_{j=1}^{i} l_{ij}\bar{q}_j + r_i^0 = \sum_{j=1}^{i} \bar{f}_{ij} + r_i^0 = \bar{f}_{ii} + \bar{f}_{\overline{ii}}, \tag{6.19}$$

where $\bar{f}_{ij} \triangleq l_{ij}\bar{q}_j$ is the steady state contribution to the flow of link i by the traffic throttled by link j and $\bar{f}_{i\bar{i}} \triangleq \bar{f}_i - \bar{f}_{ii}$ corresponds to traffic which is not throttled by link i. Let the vector $Y_i(n)$ denote the state of link i at time n consisting of the admission rate q_i and buffer occupancies x_i^{ab}'s at present and in the finite past. The exact form of $Y_i(n)$ will be given below.

Lemma 6.2 *Under the RTFS priority discipline introduced in assumption (xv'), there exists a neighborhood $\mathcal{B}$ of the steady state where the link flow, $f_i(n)$, is given by*

$$f_i(n) = f_{ii}(n) + f_{i\bar{i}}(n), \; i \in \mathcal{N}_1 ,$$

where

$$f_{ii}(n) = \sum_{(ab)\in C_{ii}} q_i(n+1-\tau_{iai}^{ab}), \tag{6.20}$$

and $f_{i\bar{i}}(n)$ is a function of the states $Y_j(n)$ of links j, $j = 1, 2, \ldots, i-1$. Moreover, $f_i(n)$, $i \in \mathcal{N}$, depends continuously on the network state. □

Proof: See Appendix 3.

Since the rates are constant during any time slot $[n, n+1)$, when the delay τ is not an integer, we rewrite $q(n-\tau)$, which represents the rate during $[n-\tau, n+1-\tau)$, as

$$q(n-\tau) = (1-\theta)\, q(n-d) + \theta\, q(n+1-d), \tag{6.21}$$

where $d = \lceil \tau \rceil$, the smallest integer greater or equal to τ, and $\theta = \lceil \tau \rceil - \tau$ is in the interval $[0,1)$. Using transformation (6.21), equation (6.20) becomes

$$f_{ii}(n) = \sum_{k=0}^{D_i} l_i^k q_i(n+1-k), \tag{6.22}$$

where

$$\begin{aligned} l_i^k &= \sum_{(ab)\in C_{ii}^k} \left(1-\theta_{iai}^{ab}\right) + \sum_{(ab)\in C_{ii}^{k+1}} \theta_{iai}^{ab}, \\ C_{ii}^k &= \left\{(ab) \in C_{ii} : \; \left\lceil \tau_{iai}^{ab} \right\rceil = k \right\}, \\ D_i &= \max_{(ab)\in C(i)} \left\lceil \tau_{iai}^{ab} \right\rceil . \end{aligned} \tag{6.23}$$

Here, l_i^k is the equivalent number of connections throttled by link i with round trip delay equal to k satisfying $\sum_{k=0}^{D_i} l_i^k = l_{ii}$ and D_i is the largest round trip delay for all connections having link i in their path. Note that the sets C_{ii}^k may not remain constant outside the neighborhood $\mathcal{B}$ and therefore l_i^k is generally a function of time $\left(l_i^k(n)\right)$.

Since, as it is shown before, the steady state values of $x_i(n)$ and $q_i(n)$, $i \in \mathcal{N}_1$, satisfy $\bar{x}_i = x^0 > 0$ and $0 < \bar{q}_i < q^0$, the saturation-type nonlinearities in (6.4) and (6.8) are not activated in the neighborhood $\bar{\mathcal{B}}$ of the steady state where $x_i > 0$ and $0 < q_i < q^0$. Therefore, the network dynamics is described in $\mathcal{B}_0 \triangleq \bar{\mathcal{B}} \cap \mathcal{B}$ by the equations

$$\begin{aligned} x_i(n+1) &= x_i(n) + f_i(n) - c, \\ q_i(n+1) &= q_i(n) - \sum_{j=0}^{J_i} \alpha_{ij} \left(x_i(n-j) - x^0\right) - \sum_{k=0}^{K_i} \beta_{ik}\, q_i(n-k), \; i \in \mathcal{N}_1, \end{aligned}$$

where $f_i(n)$ is given in Lemma 6.2. The analysis of these equations is described in the next section where a method of choosing the gains α_{ij}'s and β_{ik}'s is presented.

6.2 ANALYSIS

6.2.1 The decoupling property

As it follows from the previous section, the local (i.e., in the neighborhood $\mathcal{B}_0$ of the steady state) closed loop behavior of the network under the RTFS priority scheme is described by the following equations:

$$x_i(n+1) = x_i(n) + \sum_{k=0}^{D_i} l_i^k \, q_i(n+1-k) + f_{\overline{ii}}(n) - c\,, \tag{6.24}$$

$$q_i(n+1) = q_i(n) - \sum_{j=0}^{J_i} \alpha_{ij} \left(x_i(n-j) - x^0\right) - \sum_{k=0}^{K_i} \beta_{ik}\, q_i(n-k), \;\; i \in \mathcal{N}_1\,. \tag{6.25}$$

Theorem 6.1 *Under assumptions (i)-(xiv) and (xv'), equations (6.24)-(6.25) are decoupled in the following sense: System (6.24)-(6.25) is asymptotically stable if and only if each of the following N_1 independent pairs of equations which corresponds to a separate link is asymptotically stable:*

$$\begin{aligned} x_i(n+1) &= x_i(n) + \sum_{k=0}^{D_i} l_i^k \, q_i(n+1-k) - c\,, & (6.26)\\ q_i(n+1) &= q_i(n) - \sum_{j=0}^{J_i} \alpha_{ij} \left(x_i(n-j) - x^0\right) - \sum_{k=0}^{K_i} \beta_{ik}\, q_i(n-k), & (6.27) \end{aligned}$$

$$\forall\, i \in \mathcal{N}_1\,. \qquad \square$$

Proof: See Appendix 4.

Note that such a decoupling is not possible under the FCFS priority discipline due to the interaction among all traffic types queued for transmission. The RTFS discipline limits such interaction by giving priority to nonthrottled traffic.

It follows from Theorem 6.1 that the design of the network controller can be accomplished by designing a controller for each link separately without regard to the others. In other words, Theorem 6.1 reduces the solution of the multiple congested node case to that of the single congested node case. The problem of analysis and design for the single congested node case has been carried out in Part 1. The main results will be presented here without proofs and the reader is referred to Part 1 for the complete derivations.

Note that, since (6.24) involves delayed versions of $q_i(n)$ up to $q_i(n-(D_i-1))$, it is natural to assume that $K_i \geq D_i - 1$ and, as a result, we choose the network state $Y(n)$ as

$$Y(n) = (Y_1(n)\, Y_2(n) \ldots Y_{N_1}(n))^T\,, \tag{6.28}$$

where

$$Y_i(n) = \left(x_i(n) - x^0,\, x_i(n-1) - x^0, \ldots, x_i(n-J_i) - x^0,\, q_i(n),\, q_i(n-1), \ldots, q_i(n-K_i)\right)^T.$$

The steady states and dynamic properties of the network are described below.

6.2.2 Steady states

As it follows from the previous section, the steady state flow through any overloaded link i, $\bar{f}_i = \sum_{j=1}^{i} l_{ij}\bar{q}_j + r_i^0$, is equal to the link capacity c and, under constraint (6.17), the steady state buffer occupancy $\bar{x}_i$ is equal to x^0.

Using the notation in (6.18), the steady state flow can be written in matrix form as

$$\bar{F} = \Lambda\,\bar{Q} + R^0 = C,$$

where C is the N_1-dimensional vector with entries equal to c. Being lower triangular with positive diagonal elements, Λ is nonsingular and therefore

$$\bar{Q} = \Lambda^{-1}\left(C - R^0\right). \tag{6.29}$$

Thus, under constraint (6.17), the unique steady state is characterized by the steady state buffer occupancies $\bar{x}_i = x^0$, $i \in \mathcal{N}_1$, and the steady state rates $\bar{Q} = \Lambda^{-1}(C - R^0)$. The fairness property of these steady state rates is presented below.

In allocating the network resources among the competing connections, one of the objectives is to maximize the network utilization while achieving some fairness in resource allocation. An intuitive notion of fairness is that any connection is entitled to as much network use as any other connection, irrespective to the geographical separation of the connection's origin and destination. Since connections follow different paths and therefore links are traversed by different numbers of connections, this intuitive notion leads to the max-min fairness [24] described below.

We start by describing some terms. An **allocation** r is a function that assigns each connection (ab) a rate r_{ab}. The allocation is **feasible** if it is consistent with the demand r_{ab}^0 and the link capacities, i.e.,

$$0 \le r_{ab} \le r_{ab}^0,\ (ab) \in \mathcal{C}, \tag{6.30}$$

$$f_i = \sum_{(ab)\in C(i)} r_{ab} \le c,\ i \in \mathcal{N}. \tag{6.31}$$

A feasible allocation r is **max-min fair** if for each connection (ab) and feasible allocation r' for which $r'_{ab} > r_{ab}$, there exists a connection $a'b'$ with $r_{a'b'} \le r_{ab}$ and $r'_{a'b'} < r_{a'b'}$. In other words, r is max-min fair if for every connection $(ab) \in \mathcal{C}$, r_{ab} cannot be increased, while maintaining feasibility, without decreasing $r_{a'b'}$ for some connection $a'b'$ for which $r_{a'b'} \le r_{ab}$.

The term "max-min fair allocation" comes from the fact that for such a strategy, the smallest rate assigned to any connection is as large as possible and, subject to this constraint, the second smallest assigned rate is as large as possible, etc. Each of these nested max-min optimization problems can be formulated as a linear programming problem and it can be shown that there exists a unique allocation that solves them [24].

Theorem 6.2 *The allocation resulting from the steady state rates (6.29) is max-min fair.* □

Proof: See Appendix 4.

6.2.3 Transient analysis

As it follows from the decoupling property, the local dynamics of link i is given by

$$x_i(n+1) = x_i(n) + \sum_{k=0}^{D_i} l_i^k q_i(n+1-k) + f_{\bar{i}\bar{i}}(n) - c, \tag{6.32}$$

$$q_i(n+1) = q_i(n) - \sum_{j=0}^{J_i} \alpha_{ij}(x_i(n-j) - x^0) - \sum_{k=0}^{K_i} \beta_{ik} q_i(n-k)\,, \tag{6.33}$$

where the effect of the other links reduces to a disturbance $f_{\bar{i}\bar{i}}(n)$.

Theorem 6.3 *For given $l_i^0, l_i^1, \ldots, l_i^{D_i}$, the poles of the system (6.32)-(6.33) can be placed arbitrarily by an appropriate choice of the gains $\alpha_{i0}, \ldots, \alpha_{iJ_i}, \beta_{i0}, \ldots, \beta_{iK_i}$, $\sum_{k=0}^{K_i} \beta_{ik} = 0$, if $J_i = 1$ and $K_i = D_i$.* □

Proof: See Part 1.

Thus, a PD controller with respect to x_i and a controller of order D_i with respect to q_i are sufficient to ensure any desired dynamics or the closed loop system (6.32)-(6.33).

As it follows from the proof in Part 1, the characteristic polynomial of the closed loop system can be represented as

$$P_i(\lambda) = (-1)^{D_i+1} \lambda P_{\Gamma_i}(\lambda),$$

where

$$P_{\Gamma_i}(\lambda) = \lambda^{D_i+2} + \gamma_{i1}\lambda^{D_i+1} + \gamma_{i2}\lambda^{D_i} + \ldots + \gamma_{i,D_i+2}, \tag{6.34}$$

and the vector of coefficients $\Gamma_i = \left(\gamma_{i1}\, \gamma_{i2} \, \ldots \, \gamma_{i,D_i+2}\right)^T$ is related to the controller gains as follows:

$$\tilde{\Gamma}_i = M(L_i) G_i, \tag{6.35}$$

where

$$\tilde{\Gamma}_i = \begin{pmatrix} \gamma_{i1} + 2 \\ \gamma_{i2} - 1 \\ \gamma_{i3} \\ \vdots \\ \gamma_{i,D_i+2} \\ 0 \end{pmatrix}, M(L_i) = \begin{pmatrix} l_i^0 & 0 & 1 & & & & \\ l_i^1 & l_i^0 & -1 & 1 & & & \\ l_i^2 & l_i^1 & & -1 & \ddots & & \\ \vdots & \vdots & & & \ddots & 1 & \\ l_i^{D_i} & l_i^{D_i-1} & & & & -1 & 1 \\ 0 & l_i^{D_i} & & & & & -1 \\ 0 & 0 & 1 & 1 & \ldots & & 1 \end{pmatrix}, \ L_i = \begin{pmatrix} l_i^0 \\ l_i^1 \\ \vdots \\ l_i^{D_i} \end{pmatrix},$$

and $G_i = \left(\alpha_{i0}\, \alpha_{i1}\, \beta_{i0}\, \beta_{i1} \, \ldots \, \beta_{iD_i}\right)^T$. Therefore, as it follows from (6.34), one of the closed loop poles is at 0 and the remaining $D_i + 2$ poles can be chosen at will. Indeed, for any desired set of coefficients γ_{ij}, $j = 1, 2, \ldots, D_i + 2$, the corresponding control gain G_i is obtained from (6.35) as

$$G_i = [M(L_i)]^{-1}\tilde{\Gamma}_i, \tag{6.36}$$

since $M(L_i)$ is shown to be a nonsingular matrix with determinant equal to $\left(\sum_{k=0}^{D_i} l_i^k\right)^2 = (l_{ii})^2$. Expression (6.35) can be rewritten in the following alternative form which is used below for design purposes:

$$\Gamma_i = \Gamma_{i\beta} + W_{i\alpha} L_i\,, \tag{6.37}$$

where

$$\Gamma_{i\beta} = \begin{pmatrix} \beta_{i0} - 2 \\ \beta_{i1} - \beta_{i0} + 1 \\ \beta_{i2} - \beta_{i1} \\ \beta_{i3} - \beta_{i2} \\ \vdots \\ \beta_{iD_i} - \beta_{i,D_i-1} \\ -\beta_{iD_i} \end{pmatrix}, \ W_{i\alpha} = (V_{i0}\, V_{i1} \, \ldots \, V_{iD_i}) = \begin{pmatrix} \alpha_{i0} & & & \\ \alpha_{i1} & \alpha_{i0} & & \\ & \alpha_{i1} & \ddots & \\ & & \ddots & \alpha_{i0} \\ & & & \alpha_{i1} \end{pmatrix}.$$

Concluding this section, we note that the sum of the coefficients of $P_{\Gamma_i}(\lambda)$, $1 + \sum_{j=1}^{D_i+2} \gamma_{ij}$, is equal to $(\alpha_{i0} + \alpha_{i1}) \sum_{k=0}^{D_i} l_i^k$. Since it is necessary for asymptotic stability that this sum be positive, $\alpha_{i0} + \alpha_{i1}$ has to be positive as stated in (6.16).

The next section presents ways of choosing the vectors of control gains G_i, $i \in \mathcal{N}$.

6.3 DESIGN

6.3.1 Adaptive vs. robust design

The design of the congestion controller involves the computation of the parameters of the control law, i.e., the vector of control gains $G_i = \left(\alpha_{i0}\, \alpha_{i1}\, \beta_{i0}\, \beta_{i1} \, \ldots \, \beta_{iD_i}\right)^T$, $i \in \mathcal{N}$. As it has been shown in Section 3, the closed loop dynamics depend on the number of throttled connections $L_i = (l_i^0 \, l_i^1 \, \ldots \, l_i^{D_i})^T$, where l_i^k is the equivalent number of throttled connections with round trip delay equal to k. Therefore, if each link i can take into account the variations of L_i when computing the control law, the resulting design is referred to as **adaptive**; here G_i is updated whenever L_i changes. If information about L_i is not available, a **robust** design has to be considered; here the control law is implemented with a fixed G_i so that stability and performance requirements are satisfied for all admissible values of L_i.

Note that an adaptive design has the advantage of achieving good performance but at the expense of higher computational requirements (the updating algorithm for G_i) whereas a robust design requires less computations but the achieved performance may be inferior. Each of these design approaches are investigated below.

6.3.2 Adaptive design

Suppose that the desired performance of the congestion control system is specified in terms of closed loop poles that correspond to the vector of characteristic polynomial coefficients $\Gamma_i = (\gamma_{i1}\, \gamma_{i2} \, \cdots \, \gamma_{i,D_i+2})^T$. Then, as it follows from (6.36), the adaptation of the control gain G_i to changes in L_i takes the following form:

$$G_i(n) = [M(L_i(n-1))]^{-1} \tilde{\Gamma}_i \,, \tag{6.38}$$

where $L_i(n) = (l_i^0(n)\, l_i^1(n) \, \ldots \, l_i^{D_i}(n))^T$ and $\tilde{\Gamma}_i = (\gamma_{i1}+2,\, \gamma_{i2}-1,\, \gamma_{i3}, \ldots, \gamma_{i,D_i+2},\, 0)^T$. Note that $G_i(n)$ depends on L_i at time $n-1$ since at time n only $L_i(n-1)$ is available. Also, in order to be able to implement an adaptive design, the congestion control protocol should be implemented in such a way that L_i can be known to the switching nodes. One way of doing so is to stamp each packet with the identity of the throttling link, if any. In this way, every node can maintain a table of throttled connections. The performance of the network utilizing the adaptive control law (6.38) is illustrated in Section 5 below.

6.3.3 Robust design

The goal of this subsection is to find a fixed gain $G_i = (\alpha_{i0}\, \alpha_{i1}\, \beta_{i0}\, \beta_{i1} \, \ldots \, \beta_{iD_i})^T$ that ensures stability of the closed loop system (6.32)-(6.33) for all values of L_i belonging to an admissible set $\mathcal{L}_i$ defined as

$$\mathcal{L}_i = (\mathcal{L}_{i0} \times \mathcal{L}_{i1} \times \ldots \times \mathcal{L}_{iD_i}) - \{\mathbf{0}\},\ \mathcal{L}_{ik} = [0, \bar{l}_i^k],\ k = 0, 1, \ldots, D_i, \tag{6.39}$$

where $\bar{l}_i^k$ is the maximum equivalent number of throttled connections with round trip delay equal to k. The solution of this problem is given by the following two theorems:

Theorem 6.4 *There exists a positive number $k_i^*(\bar{L}_i)$ where $\bar{L}_i = (\bar{l}_i^0\, \bar{l}_i^1\, \dots\, \bar{l}_i^{D_i})^T$ such that the closed loop system (6.32)-(6.33) is asymptotically stable for any $L_i \in \mathcal{L}_i$ if the control gains are chosen as*

$$G_i = (k_i\alpha_{i0}\; k_i\alpha_{i1}\; \beta_{i0}\, \beta_{i1}\, \dots\, \beta_{iD_i})^T,\; k_i \in (0, k_i^*),$$

where

$$\begin{aligned}
\alpha_{i0} &= \frac{D_i + 4}{2(D_i + 1)}, \\
\alpha_{i1} &= -\frac{D_i + 2}{2(D_i + 1)}, \\
\beta_{i0} &= \frac{3D_i}{2(D_i + 1)}, \\
\beta_{ik} &= \frac{D_i - 2(k+1)}{2(D_i + 1)},\; k = 1, 2, \dots, D_i.
\end{aligned}$$

□

Proof: See Part 1.

Thus the robust control gain, G_i, is the $(D_i + 3)$-dimensional vector with components defined in Theorem 6.4.

Let S_θ^i be the surface in $\mathbb{R}^{D_i+2}$ defined by the equations

$$S_\theta^i : \begin{cases} \sum_{j=1}^{D_i+2} \gamma_j \cos(j\theta) = -1, \\ \sum_{j=1}^{D_i+2} \gamma_j \sin(j\theta) = 0,\; \theta \in (0, \pi), \end{cases}$$

and let the inequality $L_i \le \bar{L}_i$ imply the component wise inequalities $l_i^k \le \bar{l}_i^k$, $k = 0, 1, \dots, D_i$.

Theorem 6.5 *The positive number k_i^*, referred to in Theorem 4.1, is defined as*

$$k_i^* = \min\{k_i^1, k_i^2\}.$$

Here,

$$k_i^1 = \frac{3D_i + 5 + (-1)^{D_i}(D_i + 3)}{2(D_i + 3)\sum_{k=0}^{\lfloor D_i/2 \rfloor} \bar{l}_i^{2k}},$$

and k_i^2 is the solution of the following optimization problem:

minimize k subject to the constraints (6.40)

$$\begin{aligned}
&k \ge 0, \\
&\Gamma_{i\beta} + W_{i\alpha} L_i \in S_\theta^i, \\
&0 \le L_i \le k\bar{L}_i,
\end{aligned}$$

where $\Gamma_{i\beta}$ and $W_{i\alpha}$ are defined in (6.37). If the solution of this optimization problem does not exist, $k_i^2 = +\infty$. □

Proof: See Part 1.

Thus, the design of the robust congestion controller can be accomplished in two stages. First, the optimization problem of Theorem 6.5 is solved and k_i^* is determined. At the second stage, a specific gain G_i is chosen from the family defined in Theorem 6.4 so that the performance requirements are satisfied as much as possible. This design procedure is illustrated in the next section.

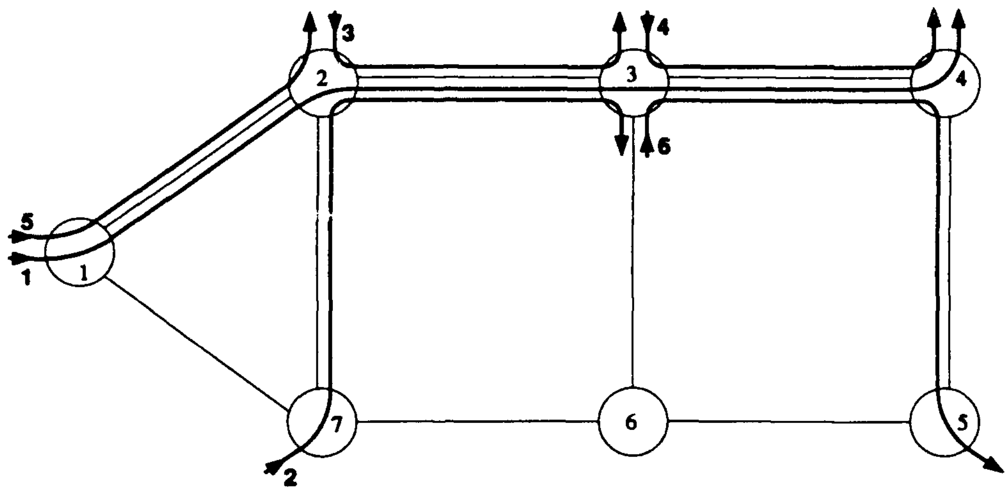

Figure 6.1: Network topology.

Connection #	initiated at time	terminated at time	input rate ($\times c$)
1	20	3000	0.9
2	50	2500	0.8
3	500	3000	0.9
4	1000	2500	0.9
5	1500	3000	0.9
6	2000	3000	0.8

Table 6.1: Network load.

6.4 DESIGN EXAMPLE AND NUMERICAL SIMULATIONS

6.4.1 Preliminaries

This section presents a design example and simulation results Although the theoretical results presented in this second part of the report refer exclusively to RTFS priority discipline, the simulations, described below, have been carried out for both RTFS and FCFS schemes. The reason for presenting the FCFS results is two fold. First the RTFS priority exhibits, in some regimes, undesirable oscillations not revealed by the local analysis; therefore, some remedy to this phenomenon is desirable. Second, although we were unable to investigate the FCFS case analytically, it turns out that, as it is illustrated by the simulation described below, it does indeed offer the remedy to the problem at hand. In other words, we show below that the congestion controller, designed for the RTFS discipline, works well if the FCFS priority is used. Moreover, the dynamic performance of the latter is better than the former in the sense that FCFS produces no undesirable nonlinear oscillations.

The robust and adaptive designs for both the RTFS and FCFS disciplines are presented below. We begin by describing the simulated network and the activity of the input traffic.

6.4.2 The network and the input traffic

Consider a seven-node network shown in Figure 6.1. The links are identical with bidirectional channels having a delay-bandwidth product of 10 packets, i.e., the propagation delay over a link corresponds to the time for transmitting 10 packets. This is the case, for instance, of 45 Mbps channels when 1 kbyte packets are sent over a 340-mile link. The activity of the six labeled connections in Figure 6.1 is shown in Table 6.1 for a duration of 3000 time slots T where T is equal to 10 τ_s (equal to 1.8 msec for the case of 45 Mbps links with 1 Kbyte packets). This choice of T corresponds to one control update every 10 transmitted packets. The set point for buffers is chosen as $x^0 = 30$ packets.

The activity of the input traffic, as shown in Table 6.1, is such that only links 1, 2, and 3 (from node 1 to 2, 2 to 3, and 3 to 4, respectively), will become overloaded and therefore, only the states of these links need to be monitored during the simulation. Note that, although the simulation involves only 6 connections, the fact that the network can carry up to 42 connections (one for each source-destination pair of nodes) is taken into account in the design of the controller.

Assume a balanced minimum-hop routing policy where connections having more than one minimum-hop route are assigned a route in such a way as to evenly spread (balance) the load among the routes. Then, the set of potential connections traversing each overloaded link i, $C(i)$, is

$$\begin{aligned} C(1) &= \{(1,2),\ (1,3),\ (1,4)\}\,, \\ C(2) &= \{(1,3),\ (1,4),\ (2,3),\ (2,4),\ (2,5),\ (7,3)\}\,, \\ C(3) &= \{(1,4),\ (2,4),\ (2,5),\ (3,4),\ (3,5)\}\,, \end{aligned}$$

where (a, b) is the connection from node a to node b. We assume that the feedback information of connection (a, b) is sent along the path of connection (b, a). Note that

$$N_1 = \begin{pmatrix} 3 \\ 0 \\ 0 \end{pmatrix},\ N_2 = \begin{pmatrix} 3 \\ 3 \\ 0 \end{pmatrix},\ N_3 = \begin{pmatrix} 2 \\ 2 \\ 1 \end{pmatrix},$$

where N_i is the vector having its j-th entry equal to the maximum number of connections that can traverse link i with the source node located j hops from this link.

In order to obtain more realistic results, we perform the simulation using a discrete-flow model. In such a model, the traffic flow consists of discrete rather than infinitesimal packets as in the fluid model used in the previous sections. Specifically, at the network interface, the number of packets of every connection (ab) admitted to the network during the time slot $[n, n+1)$, $\hat{r}_{ab}(n)$, is given by the following integrator-quantizer equations:

$$\begin{aligned} \hat{r}_{ab}(n) &= \lfloor s_{ab}(n) + r_{ab}(n) \rfloor\,, \\ s_{ab}(n+1) &= s_{ab}(n) + r_{ab}(n) - \hat{r}_{ab}(n),\ s_{ab}(0) = 0\,, \end{aligned}$$

where $r_{ab}(n)$ is defined by the control protocol (6.2), $s_{ab}(n) < 1$ is the residual traffic of connection (ab) at time n, and $\lfloor z \rfloor$ denotes the integer part of z. In addition, within the network, the buffer of each link is searched, every τ_s sec, and a discrete packet is selected for transmission on each link according to the service discipline in use (RTFS or FCFS).

6.4.3 Robust design

The data for the robust design, based on Theorem 6.4 and Theorem 6.5, consists of the largest round trip delay D_i and the vector of maximum number of throttled connections $\bar{L}_i$ for each link i. Let h^{ab}_{ai} denote the number of hops, along the path of connection (ab), between node a and the origin node of link i. Then, as it follows from (6.23),

$$\begin{aligned} D_i &= \max_{(ab)\in C(i)} \left\lceil \tau^{ab}_{iai} \right\rceil, \\ \bar{l}^k_i &= \sum_{(ab)\in C^k(i)} (1 - \theta^{ab}_{iai}) + \sum_{(ab)\in C^{k+1}(i)} \theta^{ab}_{iai}, \end{aligned}$$

where

$$\tau_{iai}^{ab} = \left\lceil h_{ai}^{ab}\tau \right\rceil + h_{ai}^{ab}\tau,$$
$$\theta_{iai}^{ab} = \left\lceil \tau_{iai}^{ab} \right\rceil - \tau_{iai}^{ab},$$
$$C^k(i) = \left\{(ab) \in C(i) : \left\lceil \tau_{iai}^{ab} \right\rceil = k\right\},$$

and τ is the one-hop delay measured in time slots T. Neglecting processing delay, (6.6) gives

$$\tau = \frac{\tau_s + \tau_p}{T} = \frac{\tau_s + 10\tau_s}{10\tau_s} = 1.1\,.$$

Carrying out the computations, we obtain

$$D_1 = 0,\ \bar{L}_1 = (3),$$
$$D_2 = 4,\ \bar{L}_2 = (3,\ 0,\ 0,\ 2.7,\ 0.3)^T,$$
$$D_3 = 6,\ \bar{L}_3 = (2,\ 0,\ 0,\ 1.8,\ 0.2,\ 0.8,\ 0.2)^T\,.$$

The values of the D_i's determine the structure of the robust controller according to Theorem 6.4 as follows:

$$\begin{aligned}
G_1 &= (k_1\alpha_{10}\ \ k_1\alpha_{11})^T,\ k_1 \in (0, k_1^*),\ \text{with} \\
&\quad \alpha_{10} = 2,\ \alpha_{11} = -1, \\
G_2 &= (k_2\alpha_{20}\ \ k_2\alpha_{21}\ \beta_{20}\ \beta_{21}\ \cdots\ \beta_{24})^T,\ k_2 \in (0, k_2^*),\ \text{with} \\
&\quad \alpha_{20} = .8,\ \alpha_{21} = -.6,\ \beta_{20} = 1.2,\ \beta_{21} = 0,\ \beta_{22} = -.2,\ \beta_{23} = -.4,\ \beta_{24} = -.6, \\
G_3 &= (k_3\alpha_{30}\ \ k_3\alpha_{31}\ \beta_{30}\ \beta_{31}\ \cdots\ \beta_{36})^T,\ k_3 \in (0, k_3^*),\ \text{with} \\
&\quad \alpha_{30} = .71,\ \alpha_{31} = -.57,\ \beta_{30} = 1.29,\ \beta_{31} = .14,\ \beta_{32} = 0, \\
&\quad \beta_{33} = -.14,\ \beta_{34} = -.29, \beta_{35} = -.43,\ \beta_{36} = -.57\,.
\end{aligned} \tag{6.41}$$

The values of k_i^* are functions of $\bar{L}_i$ and are determined by solving the optimization problem of Theorem 6.5 to obtain

$$k_1^* = 0.444,\ k_2^* = 0.178,\ k_3^* = 0.264\,.$$

Note that the controller gains for the other links can be determined in a similar way but for the purpose of this simulation any set of gains satisfying (6.16) is appropriate.

The behavior of the network with G_i defined by (6.41) for $k_i = 0.95\,k_i^*$, i=1, 2, and 3, is shown in Figure 6.2.a. Figure 6.2.b corresponds to the case where the service discipline is FCFS instead of RTFS. Each figure includes simulation traces for the changes over time of the buffer occupancy and the admission rate for each of the links 1, 2, and 3. The results can be summarized as follows:

(a). When a link is overloaded (starting at time t=50, 1000, and 1500 for links 2, 3, and 1, respectively), the buffer is kept in the vicinity of $x^0 = 30$, with fluctuations whenever a new connection is initiated or an established connection is terminated.

(b). As the input traffic to the network changes, the admission rates $q_i(n)$ converge to a new steady state in order to maintain max-min fairness or the rate allocation.

(c). The FCFS discipline exhibits stable (non-oscillatory) behavior whereas oscillations are observed during the time interval [2000,3000) when the RTFS discipline is used. This better performance justifies the choice of the FCFS discipline although its stable behavior was shown by simulation and not analytically.

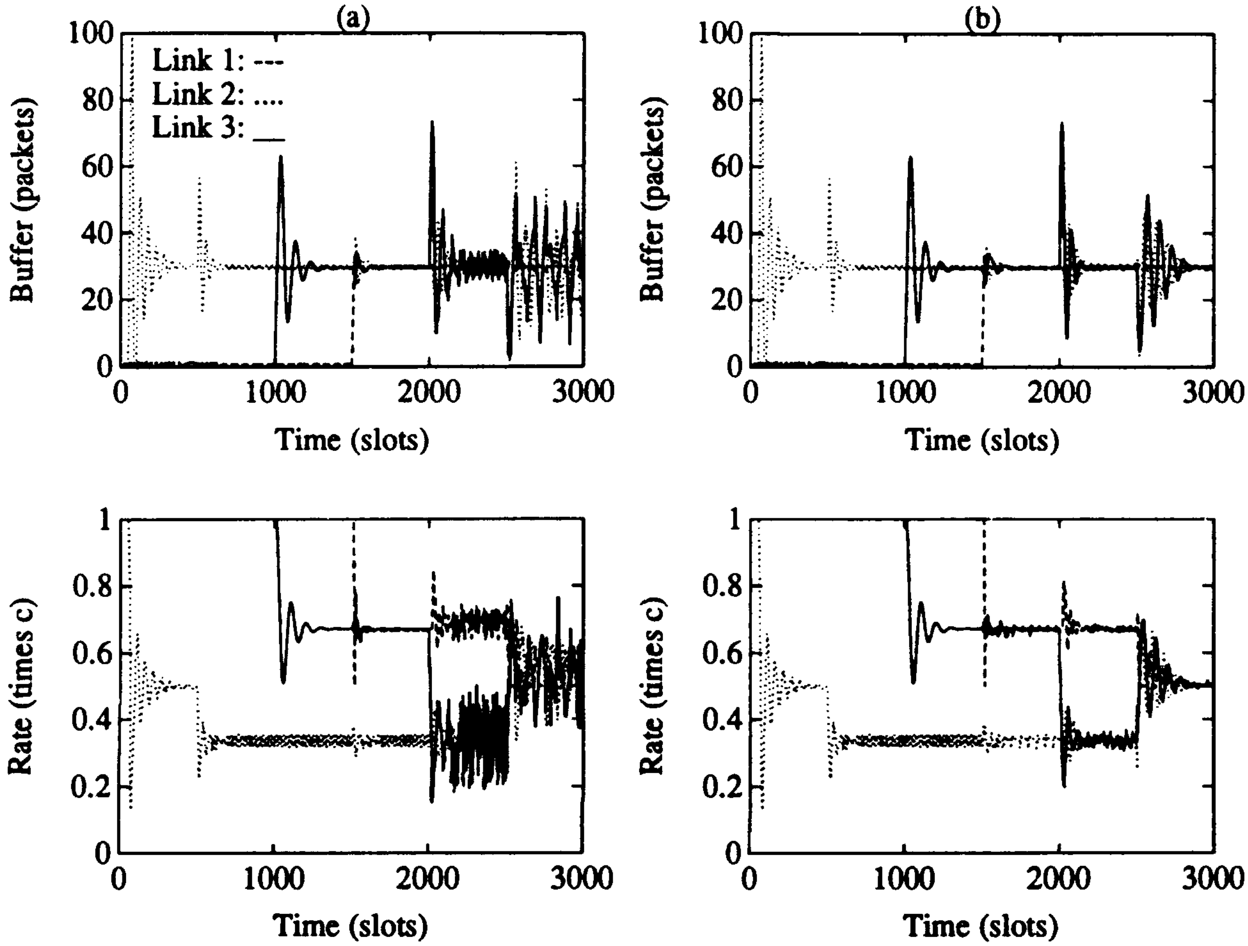

Figure 6.2: Robust design.

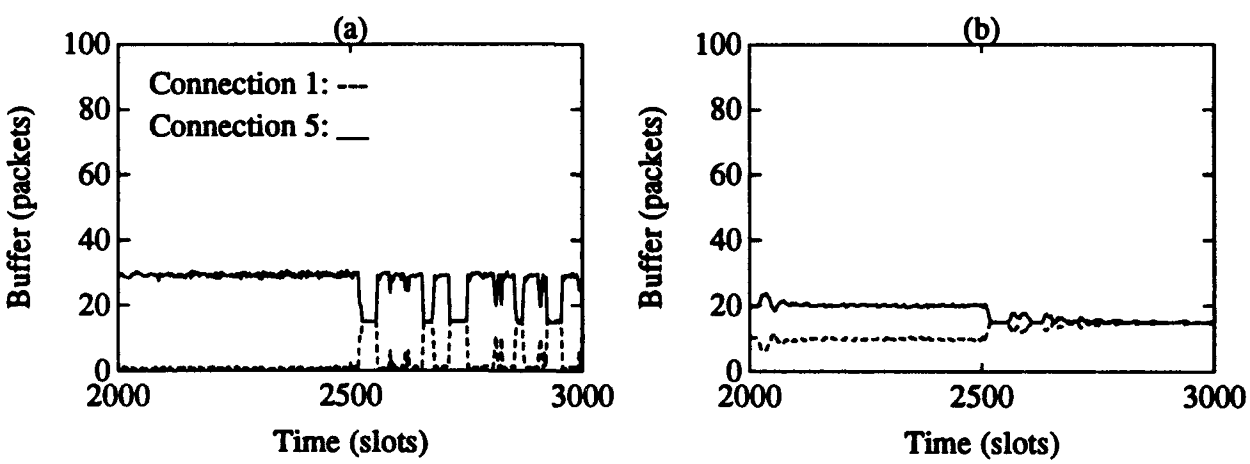

Figure 6.3: Buffer oscillation.

The reason for the oscillations is the following: During the time interval [2000,3000), the throttling link for connection 1, i.e., the link which has the minimum admission rate q among all links in the path of connection 1 (links 1, 2, and 3), is not unique since the steady state rates $\bar{q}_2$ and $\bar{q}_3$ are equal ($\bar{q}_2 = \bar{q}_3 = c/3$) during [2000,2500) and $\bar{q}_1 = \bar{q}_2 = \bar{q}_3 = c/2$ during [2500,3000). These steady states correspond to corner points in the piece-wise linear nonlinearity (6.5). The oscillations are due to this fact and to the fact that, under the RTFS discipline, connections are treated differently by the transmission links depending on whether or not they are throttled by that link. In fact, Figure 6.3.a shows the result of the interaction between the nonlinearity and the RTFS priority scheme. This figure is a plot of the buffer occupancies of traffic belonging to connections 1 and 5 at the buffer of link 1. During the time interval [2500,3000) when $\bar{q}_1 = \bar{q}_2 = \bar{q}_3$, connection 1 is either given priority by link 1 and therefore its buffer occupancy is zero or equally shares the link capacity with connection 2 and therefore both have an occupancy of $x^0/2$. Figure 6.3.b shows that no such oscillations take place under the FCFS priority discipline.

6.4.4 Adaptive design

In this section, we design an adaptive controller and examine its performance for both priority disciplines considered above, RTFS and FCFS.

The characteristic polynomial $P_{\Gamma_i}(\lambda)$ in (6.34) has dimension $D_i + 2$ and therefore the gains are specified by the choice of $D_i + 2$ desired closed loop poles. For instance, if we choose the desired characteristic polynomials as

$$P_{\Gamma_1}(\lambda) = (\lambda - \lambda_1)(\lambda - \lambda_2),$$
$$P_{\Gamma_2}(\lambda) = (\lambda - \lambda_1)^3(\lambda - \lambda_2)^3,$$
$$P_{\Gamma_3}(\lambda) = \lambda^2(\lambda - \lambda_1)^3(\lambda - \lambda_2)^3, \ \lambda_{1,2} = 0.6 \pm j0.1,$$

then, as it follows from (6.38), the adaptation of the controller gains

$$G_i = \left(\alpha_{i0}\ \alpha_{i1}\ \beta_{i0}\ \beta_{i1}\ \ldots\ \beta_{iD_i}\right)^T$$

to changes in L_i takes the following form:

$$G_i(n) = [M(L_i(n-1))]^{-1}\tilde{\Gamma}_i, \ i = 1,2,3, \tag{6.42}$$

where

$$\tilde{\Gamma}_1 = (0.8,\ -0.63)^T,$$
$$\tilde{\Gamma}_2 = (-1.6,\ 4.43,\ -4.39,\ 2.01,\ -.49,\ .05)^T,$$
$$\tilde{\Gamma}_3 = (-1.6,\ 4.43,\ -4.39,\ 2.01,\ -.49,\ .05,\ 0,\ 0)^T.$$

The simulation results of the network with controller (6.42) and the service disciplines RTFS and FCFS are shown in Figure 6.4.a and Figure 6.4.b, respectively. From these figures we conclude:

(a). The adaptive approach achieves better transient behavior in terms of speed of the response and buffer overshoot than the robust approach.

(b). The superior performance of the adaptive controller justifies its selection for implementation despite the extra computational time it requires.

(c). As in the case with robust design, the network with the RTFS discipline is prone to oscillations whereas stable behavior is achieved with the FCFS discipline.

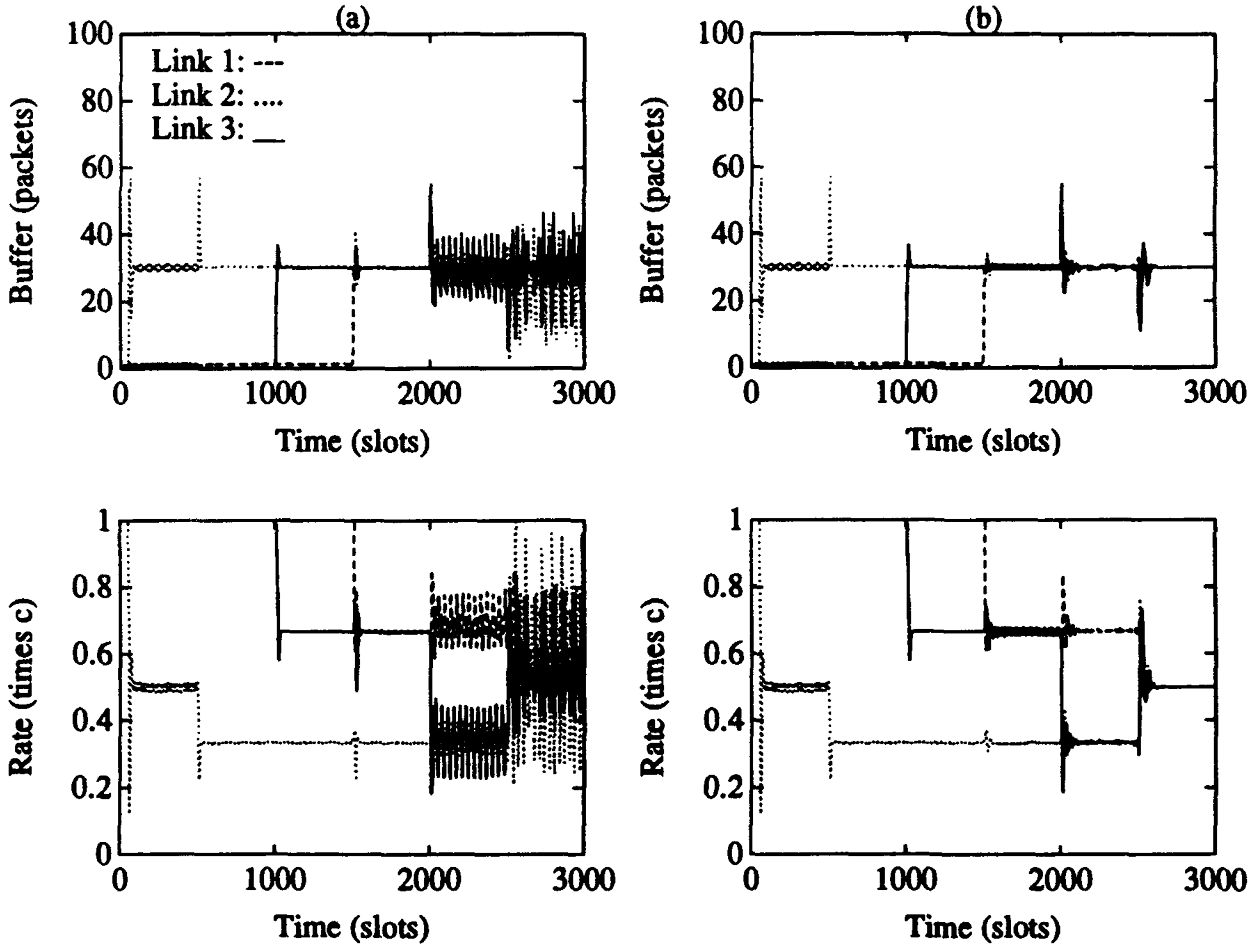

Figure 6.4: Adaptive design.

6.5 CONCLUSIONS

This report presents a theory for analysis and design of a congestion control system in store-and-forward networks with multiple congested nodes. Within the framework of the controller structure and system architecture introduced here, it is shown that the network is decoupled in the sense that its analysis and design can be carried out for each link separately without regard to the others. Based on this decoupling property, we show that the design of the congestion controller for each link is accomplished as follows:

1. Using Theorem 6.3, choose the order of the controller that guarantees the existence of stabilizing gains.

2. If the adaptive controller is used (i.e., the gains are adapted to the specific input traffic and network conditions), select the controller gains based on formula (6.38).

3. If the robust controller is used (i.e., the gain ensures stability for all admissible input traffics and network conditions), select the parameterized controller gain as indicated in Theorem 6.4 and a specific gain based on Theorem 6.5.

Both the adaptive and the robust designs, with the gains selected appropriately, ensure asymptotic stability of the network. Numerical simulations show that the network with the RTFS discipline exhibits, in some regimes, undesirable oscillations that are avoided when the FCFS discipline is utilized. Moreover, it is shown that the adaptive controller is capable of insuring much higher performance characteristics as compared with the robust one. This may justify the additional computation burden necessary for the implementation of the adaptive approach.

7 LIST OF PUBLICATIONS AND TECHNICAL REPORTS

[1] L. Benmohamed, S. M. Meerkov, "Feedback Control of Congestion in Store-and-Forward Datagram Networks: The Case of a Single Congested Node," *Proc. IEEE Conference on Decision and Control*, pp. 991-996, 1992. An extended version of this paper was submitted for publication in *IEEE/ACM Transactions on Networking*. The full paper is in Control Group Report No. CGR-92-12, The University of Michigan, 1992.

[2] L. Benmohamed, S. M. Meerkov, "Feedback Congestion Control in Store-and-Forward Datagram Networks: Analysis and Design Techniques," *Second ORSA Telecommunications Conference*, Boca Raton, Florida, March 1992.

[3] L. Benmohamed, S. M. Meerkov, "Feedback Control of Congestion in Store-and-Forward Networks: The Case of Multiple Congested Nodes," Control Group Report No. CGR-93-3, The University of Michigan, 1993. An extended abstract of this report was submitted for publication in *Computer Networks and ISDN Systems*.

8 LIST OF PARTICIPATING SCIENTIFIC PERSONNEL

1. Semyon M. Meerkov, P.I., Professor.

2. Lotfi Benmohamed, Research Assistant, Ph.D. Candidate, Expected Graduation Date August 1993.

9 BIBLIOGRAPHY

[1] M. Gerla, L. Kleinrock, "Flow Control: A Comparative Survey," *IEEE Trans. Comm.*, Vol. COM-28, pp. 553–574, 1980.

[2] L. Zhang, "Some Thoughts on the Packet Network Architecture," *ACM Comp. Comm. Review*, Vol. 17, No. 1 & 2, pp. 3–17, 1987.

[3] R. Jain, K. Ramakrishnan, "Congestion Avoidance in Computer Networks with Connectionless Network Layer: Concepts, Goals and Methodology," *Proc. Comp. Net. Symp.*, Washington, DC, April 1988, pp. 134–143.

[4] M. Schwartz, *Telecommunication Networks, Protocols, Modeling and Analysis*, Addison-Wesley, 1987.

[5] A. S. Tannenbaum, *Computer Networks*, Prentice Hall, 1988.

[6] D. Bertsekas, R. Gallager, *Data Networks*, Prentice Hall, 1992.

[7] D. Mitra, " Asymptotically Optimal Design of Congestion Control for High Speed Data Networks," *IEEE Trans. Comm.*, Vol. 40, No. 2, pp. 301–311, 1992.

[8] J. Robinson, D. Friedman, M. Steenstrup, "Congestion Control in BBN Packet Switched Networks," *Comp. Comm. Review*, Vol. 20, No. 1, pp. 76–90, 1990.

[9] V. Jacobson, "Congestion Avoidance and Control," *Proc. ACM SigComm*, pp. 314–329, 1988.

[10] K. K. Ramakrishnan, R. Jain, "A Binary Feedback Scheme for Congestion Avoidance in Computer Networks with a Connectionless Network Layer," *Proc. ACM SIGCOMM*, Stanford, CA, 1988, pp. 314–329.

[11] S. Shenker, "A Theoretical Analysis of Feedback Flow Control," *Proc. ACM SigComm 1990*, pp. 156–165, 1990.

[12] S. Keshav, "A Control-Theoretic Approach to Flow Control," *ACM Comp. Comm. Review*, Vol. 21, No. 4, pp. 3–15, 1991.

[13] L. Zhang, "A New Architecture for Packet Switching Network Protocols," MIT Ph.D. Thesis, 1989.

[14] K. W. Fendick, D. Mitra, I. Mitrani, M. A. Rodrigues, J. B. Seery, A. Weiss, "An Approach to High-Performance, High-Speed Data Networks," *IEEE Comm. Magazine*, Vol. 29, No. 10, pp. 74–82, 1991.

[15] K. W. Fendick, M. A. Rodrigues, A. Weiss, "Analysis of a Rate-Based Control Strategy with Delayed Feedback," *Proc. ACM SigComm*, pp. 314–329, 1992.

[16] D. Chiu, R. Jain, "Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Network," *Computer Networks and ISDN systems*, Vol. 17, pp. 1–14, 1989.

[17] S.M. Meerkov, "Simplified Description of slow Markov Walks-Part I," *Automation Remote Contr.*, Vol. 33, pp. 404–414, 1972.

[18] J.-T. Lim and S. M. Meerkov, "Simplified Description of Slow-in-the-Average Markov Walks," *J. Math. Analysis and Appl.*, Vol. 158, No. 2, pp. 476–486, 1991.

[19] A. T. Fam, J. S. Meditch, "A Canonical Parameter Space for Linear Systems Design," *IEEE Trans. Automat. Contr.*, Vol. AC-23, No. 3, pp. 454–458, 1978.

[20] A. Vicino, "Some Results on Robust Stability of Discrete-Time Systems," *IEEE Trans. Automat. Contr.*, Vol. AC-33, No. 9, pp. 844–847, 1988.

[21] R. H. Raible, "A Simplification of Jury's Tabular Form," *IEEE Trans. Automat. Contr.*, Vol. AC-19, pp. 248–250, June 1974.

[22] L. P. Eisenhart, *Transformations of Surfaces*, Princeton University Press, 1923.

[23] J. C. Bolot, A. U. Shankar, "Dynamical Behavior of Rate-Based Flow Control Mechanisms," *ACM Comp. Comm. Review*, Vol. 20, No. 2, pp. 35–49, 1990.

[24] E. L. Hahne, R. G. Gallager, "Round Robin Scheduling for Fair Flow Control in Data Communication Networks," *Proc. IEEE Int. Conf. Comm.*, pp. 103–107, 1986.

10 APPENDICES

APPENDIX 1: PROOFS FOR SECTION 5.2

The proof of the Theorem 5.1 is based on the two lemmas that follow. To simplify the notations, rewrite (5.33) and (5.34) as follows:

$$Y(n+1) = A(L)Y(n) + B \tag{A1.1}$$

where

$$A(L) = \begin{pmatrix} 1-l_0\alpha_0 & -l_0\alpha_1 & \ldots\ldots & -l_0\alpha_{J-1} & -l_0\alpha_J & -b_0 & -b_1 & -b_2 & \ldots\ldots & -b_K \\ 1 & & & & & & & & & \\ & 1 & & & & & & & & \\ & & & & & & & & & \\ & & & 1 & & & & & & \\ -\alpha_0 & -\alpha_1 & \ldots\ldots & -\alpha_{J-1} & -\alpha_J & 1-\beta_0 & -\beta_1 & -\beta_2 & \ldots\ldots & -\beta_K \\ & & & & & 1 & & & & \\ & & & & & & 1 & & & \\ & & & & & & & & & \\ & & & & & & & & 1 & 0 \end{pmatrix},$$

$$Y(n) = \begin{pmatrix} x(n)-x^0 \\ x(n-1)-x^0 \\ \vdots \\ x(n-J)-x^0 \\ q(n) \\ q(n-1) \\ \vdots \\ q(n-K) \end{pmatrix}, \quad B = \begin{pmatrix} x^0+r^0-c \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad L = \begin{pmatrix} l_0 \\ l_1 \\ \vdots \\ l_D \end{pmatrix},$$

and

$$\begin{aligned} b_0 &\triangleq l_0(\beta_0 - 1) - l_1, \\ b_i &\triangleq l_0\beta_i - l_{i+1}, \; i = 1, \ldots, D-1, \\ b_i &\triangleq l_0\beta_i, \; i = D, \ldots, K. \end{aligned} \tag{A1.2}$$

Lemma A1.1: The characteristic equation corresponding to equation (A1.1) is

$$\det[A(L) - \lambda I] = (-1)^{J+K}[\lambda^{J+K}(\lambda-1)^2 + \lambda^J(\lambda-1)P_\beta + \lambda^{K-D+1}P_\alpha P_L], \tag{A1.3}$$

where P_α, P_β, and P_L are polynomials in λ given by

$$\begin{aligned} P_\alpha &= \alpha_0\lambda^J + \alpha_1\lambda^{J-1} + \ldots + \alpha_J, \\ P_\beta &= \beta_0\lambda^K + \beta_1\lambda^{K-1} + \ldots + \beta_K, \\ P_L &= l_0\lambda^D + l_1\lambda^{D-1} + \ldots + l_D. \end{aligned}$$

□

Proof: Using notations (A1.2), the characteristic polynomial of matrix $A(L)$ can be written as follows:

$$\det[A(L) - \lambda I] \triangleq \det(A_K) =$$

$$\det \begin{pmatrix} 1 - l_0\alpha_0 - \lambda & -l_0\alpha_1 & \ldots & -l_0\alpha_{J-1} & -l_0\alpha_J & -b_0 & -b_1 & \ldots & -b_K \\ 1 & -\lambda & & & & & & & \\ & & & & & & & & \\ & & & 1 & -\lambda & & & & \\ -\alpha_0 & -\alpha_1 & \ldots & -\alpha_{J-1} & -\alpha_J & 1 - \beta_0 - \lambda & -\beta_1 & \ldots & -\beta_K \\ & & & & & 1 & -\lambda & & \\ & & & & & & & & \\ & & & & & & & 1 & -\lambda \end{pmatrix}.$$

Expanding A_K with respect to the last row, we obtain the following recursive equation

$$\det(A_K) = -\lambda\det(A_{K-1}) + (-1)^K\det(D_K), \tag{A1.4}$$

where A_i is the matrix obtained by deleting the last $K - i$ rows and the last $K - i$ columns of matrix $A(L)$, and $D_i, i = 1, \ldots, K$, is the following matrix:

$$D_i = \begin{pmatrix} 1 - l_0\alpha_0 - \lambda & -l_0\alpha_1 & \ldots & -l_0\alpha_{J-1} & -l_0\alpha_J & -b_i \\ 1 & -\lambda & & & & \\ & & & & & \\ & & & 1 & -\lambda & \\ -\alpha_0 & -\alpha_1 & \ldots & -\alpha_{J-1} & -\alpha_J & -\beta_i \end{pmatrix}.$$

The solution of equation (A1.4) is given by

$$\det(A_K) = (-1)^K[\lambda^K\det(A_0) + \sum_{i=1}^{K} \lambda^{K-i}\det(D_i)]. \tag{A1.5}$$

Therefore, to complete the proof, we have to calculate $\det(A_0)$ and $\det(D_i)$.

To compute $\det(D_i)$, we expand D_i with respect to the last column and obtain

$$\det(D_i) = (-1)^J b_i\det(M_1) - \beta_i\det(M_2),$$

where

$$M_1 = \begin{pmatrix} 1 & -\lambda & & & \\ & & & & \\ & & & & \\ & & & 1 & -\lambda \\ -\alpha_0 & -\alpha_1 & \dots & -\alpha_{J-1} & -\alpha_J \end{pmatrix}, M_2 = \begin{pmatrix} 1 - l_0\alpha_0 - \lambda & -l_0\alpha_1 & \dots & -l_0\alpha_{J-1} & -l_0\alpha_J \\ 1 & -\lambda & & & \\ & & & & \\ & & & & \\ & & & 1 & -\lambda \end{pmatrix}.$$

After successive expansions with respect to columns, we obtain

$$\begin{aligned} \det(M_1) &= -(\alpha_0\lambda^J + \alpha_1\lambda^{J-1} + \dots + \alpha_J) = -P_\alpha, \\ \det(M_2) &= (-1)^{J+1}[(l_0\alpha_0 + \lambda - 1)\lambda^J + l_0\alpha_1\lambda^{J-1} + \dots + l_0\alpha_J], \\ &= (-1)^{J+1}[\lambda^J(\lambda - 1) + l_0P_\alpha]. \end{aligned}$$

Therefore,

$$\det(D_i) = (-1)^{J+1}[b_iP_\alpha - \beta_i\lambda^J(\lambda - 1) - \beta_i l_0 P_\alpha].$$

The determinant of A_0 can be evaluated analogously. Indeed, as it follows from its definition, A_0 is the following matrix:

$$\begin{pmatrix} 1 - l_0\alpha_0 - \lambda & -l_0\alpha_1 & \dots & -l_0\alpha_{J-1} & -l_0\alpha_J & -b_0 \\ 1 & -\lambda & & & & \\ & & & & & \\ & & & & & \\ & & & 1 & -\lambda & \\ -\alpha_0 & -\alpha_1 & \dots & -\alpha_{J-1} & -\alpha_J & 1 - \beta_0 - \lambda \end{pmatrix}.$$

Notice that this matrix is the same as the matrix corresponding to D_i except that the last column has b_0 instead of b_i and $\beta_0 + \lambda - 1$ instead of β_i. Therefore,

$$\det(A_0) = (-1)^{J+1}[b_0P_\alpha - (\beta_0 + \lambda - 1)\lambda^J(\lambda - 1) - (\beta_0 + \lambda - 1)l_0P_\alpha].$$

As a final step, we carry out the summation in (A1.5) and obtain

$$\begin{aligned} \det(A_K) &= (-1)^{K+J}[\lambda^{K+J}(\lambda - 1)^2 + \lambda^K(\lambda - 1)l_0P_\alpha + \sum_{i=0}^{K}\lambda^{K-i}(-b_iP_\alpha + \beta_i\lambda^J(\lambda - 1) + \beta_i l_0P_\alpha)], \\ &= (-1)^{K+J}[\lambda^{K+J}(\lambda - 1)^2 + \lambda^K(\lambda - 1)l_0P_\alpha + \lambda^J(\lambda - 1)P_\beta + l_0P_\alpha P_\beta - P_\alpha\sum_{i=0}^{K} b_i\lambda^{K-i}]. \end{aligned} \tag{A1.6}$$

The last term in this expression can be rewritten using the definition of b_i's to give

$$\begin{aligned} P_\alpha\sum_{i=0}^{K} b_i\lambda^{K-i} &= P_\alpha[l_0P_\beta - ((l_0 + l_1)\lambda^K + l_2\lambda^{K-1} + l_3\lambda^{K-2} + \dots + l_D\lambda^{K-D+1})], \\ &= P_\alpha[l_0P_\beta + l_0\lambda^K(\lambda - 1) - \lambda^{K-D+1}P_L]. \end{aligned} \tag{A1.7}$$

Substituting this expression into (A1.6) and canceling appropriate terms, we, finally, obtain the statement of the lemma. ■

Let $M(L)$ be the following matrix:

$$M(L) = \begin{pmatrix} l_0 & 0 & 1 & & & & \\ l_1 & l_0 & -1 & 1 & & & \\ l_2 & l_1 & & -1 & & & \\ \vdots & \vdots & & & & 1 & \\ l_D & l_{D-1} & & & & -1 & 1 \\ 0 & l_D & & & & & -1 \\ 0 & 0 & 1 & 1 & \dots & & 1 \end{pmatrix}.$$

Lemma A1.2: In the system under consideration,

$$\det[M(L)] = \left(\sum_{i=0}^{n} l_i\right)^2 .$$

□

Proof: Expanding $M(L)$ with respect to the row next to the last, we obtain the following expression

$$\det[M(L)] = (-1)^D l_D \det(B) + \det(C), \tag{A1.8}$$

where

$$B = \begin{pmatrix} l_0 & 1 & & & & \\ l_1 & -1 & 1 & & & \\ l_2 & & -1 & & & \\ \vdots & & & & & \\ l_{D-1} & & & & 1 & \\ l_D & & & & -1 & 1 \\ 0 & 1 & 1 & \dots & 1 & 1 \end{pmatrix}, C = \begin{pmatrix} l_0 & 0 & 1 & & & & \\ l_1 & l_0 & -1 & 1 & & & \\ l_2 & l_1 & & -1 & & & \\ \vdots & \vdots & & & & 1 & \\ l_{D-1} & l_{D-2} & & & & -1 & 1 \\ l_D & l_{D-1} & & & & & -1 \\ 0 & 0 & 1 & 1 & \dots & & 1 \end{pmatrix}.$$

Next, we calculate the determinant of B and C. Expanding B with respect to the row next to the last, we obtain

$$\det(B) = (-1)^D l_D \det(B_1) - \det(B_2) - \det(B_3), \tag{A1.9}$$

where

$$B_1 = \begin{pmatrix} 1 & & & & & & \\ -1 & 1 & & & & & \\ & -1 & & & & & \\ & & & & & & \\ & & & & 1 & & \\ & & & & -1 & 1 & \\ 1 & 1 & 1 & \dots & 1 & 1 & 1 \end{pmatrix}, B_2 = \begin{pmatrix} l_0 & 1 & & & & & & \\ l_1 & -1 & 1 & & & & & \\ l_2 & & -1 & & & & & \\ l_3 & & & & & & & \\ \vdots & & & & & 1 & & \\ l_{D-2} & & & & & -1 & 1 & \\ l_{D-1} & & & & & & -1 & 0 \\ 0 & 1 & 1 & 1 & \dots & 1 & 1 & 1 \end{pmatrix},$$

$$B_3 = \begin{pmatrix} l_0 & 1 & & & & & & \\ l_1 & -1 & 1 & & & & & \\ l_2 & & -1 & & & & & \\ \vdots & & & & & 1 & & \\ l_{D-2} & & & & & -1 & 1 & \\ l_{D-1} & & & & & & -1 & 1 \\ 0 & 1 & 1 & 1 & \dots & 1 & 1 & 1 \end{pmatrix}.$$

Note that B_1 is a triangular matrix and has a determinant equal to 1. Expanding B_2 with respect to the last column and then the first column of the resulting matrix, we obtain

$$\det(B_2) = \sum_{i=0}^{D-1} (-1)^i l_i \det(B_2^i),$$

where B_2^i is a block diagonal matrix, $B_2^i = \text{diag}\{B_{21}^i, B_{22}^i\}$, with B_{21}^i and B_{22}^i as $(i \times i)$- and $[(D-1-i) \times (D-1-i)]$-dimensional matrices, respectively, given by

$$B_{21}^i = \begin{pmatrix} 1 & & & & \\ -1 & 1 & & & \\ & -1 & & & \\ & & & 1 & \\ & & & -1 & 1 \end{pmatrix}, \quad B_{22}^i = \begin{pmatrix} -1 & 1 & & & \\ & -1 & & & \\ & & & 1 & \\ & & & -1 & 1 \\ & & & & -1 \end{pmatrix}.$$

Therefore,

$$\det(B_2) = \sum_{i=0}^{D-1} (-1)^i l_i (-1)^{D-1-i} = -(-1)^D \sum_{i=0}^{D-1} l_i.$$

Expanding B_3 with respect to the last column, we obtain

$$\det(B_3) = \det(B_3^1) - \det(B_3^2), \tag{A1.10}$$

where

$$B_3^1 = \begin{pmatrix} l_0 & 1 & & & & & \\ l_1 & -1 & 1 & & & & \\ l_2 & & -1 & 1 & & & \\ \vdots & & & & & & \\ l_{D-3} & & & & -1 & 1 & \\ l_{D-2} & & & & & -1 & 1 \\ l_{D-1} & & & & & & -1 \end{pmatrix}, B_3^2 = \begin{pmatrix} l_0 & 1 & & & & & & \\ l_1 & -1 & 1 & & & & & \\ l_2 & & -1 & & & & & \\ \vdots & & & & & 1 & & \\ l_{D-3} & & & & & -1 & 1 & \\ l_{D-2} & & & & & & -1 & 1 \\ 0 & 1 & 1 & 1 & \dots & 1 & 1 & 1 \end{pmatrix}.$$

Note that B_3^1 has the same determinant as B_2, and B_3^2 has the same structure as B_3. Let $b_{D-1} \triangleq \det(B_3)$ and $b_{D-2} \triangleq \det(B_3^2)$. Then, (A1.10) results in the recursive equation

$$b_{D-1} = -(-1)^D \sum_{i=0}^{D-1} l_i - b_{D-2},$$

with the initial value $b_0 = \det\begin{pmatrix} l_0 & 0 \\ 0 & 1 \end{pmatrix} = l_0$. The solution to this equation is

$$\begin{aligned} b_{D-1} &= -(-1)^D \sum_{i=0}^{D-1} l_i + (-1)^{D-1} \sum_{i=0}^{D-2} l_i - (-1)^{D-2} \sum_{i=0}^{D-3} l_i + \ldots + (-1)^{D-1} l_0, \\ &= -(-1)^D \sum_{j=0}^{D-1} \sum_{i=0}^{j} l_i. \end{aligned}$$

Therefore, (A1.9) finally becomes

$$\begin{aligned} \det(B) &= (-1)^D l_D + (-1)^D \sum_{i=0}^{D-1} l_i + (-1)^D \sum_{j=0}^{D-1} \sum_{i=0}^{j} l_i, \\ &= (-1)^D l_D + 2(-1)^D \sum_{i=0}^{D-1} l_i + (-1)^D \sum_{j=0}^{D-2} \sum_{i=0}^{j} l_i. \end{aligned} \qquad \text{(A1.11)}$$

Now we calculate $\det(C)$ of (A1.8). Expanding matrix C with respect to the row next to the last, we obtain

$$\det(C) = (-1)^D l_D \det(C_1) - (-1)^D l_{D-1} \det(C_2) + \det(C_3), \qquad \text{(A1.12)}$$

where

$$C_1 = \begin{pmatrix} 0 & 1 & & & & & & \\ l_0 & -1 & 1 & & & & & \\ l_1 & & -1 & & & & & \\ \vdots & & & & & 1 & & \\ l_{D-3} & & & & & -1 & 1 & \\ l_{D-2} & & & & & & -1 & 1 \\ 0 & 1 & 1 & 1 & \ldots & 1 & 1 & 1 \end{pmatrix}, C_2 = \begin{pmatrix} l_0 & 1 & & & & & & \\ l_1 & -1 & 1 & & & & & \\ l_2 & & -1 & & & & & \\ \vdots & & & & & 1 & & \\ l_{D-2} & & & & & -1 & 1 & \\ l_{D-1} & & & & & & -1 & 1 \\ 0 & 1 & 1 & 1 & \ldots & 1 & 1 & 1 \end{pmatrix},$$

$$C_3 = \begin{pmatrix} l_0 & 0 & 1 & & & & & \\ l_1 & l_0 & -1 & 1 & & & & \\ l_2 & l_1 & & -1 & & & & \\ \vdots & \vdots & & & & & 1 & \\ l_{D-2} & l_{D-3} & & & & & -1 & 1 \\ l_{D-1} & l_{D-2} & & & & & & -1 \\ 0 & 0 & 1 & 1 & 1 & \ldots & 1 & 1 \end{pmatrix}.$$

Note that $\det(C_1) = -\det(B_3^2) = -b_{D-2}$ (after expanding C_1 with respect to the first row). In addition, $\det(C_2) = \det(B_3) = b_{D-1}$, and finally, matrix C_3 has the same structure as matrix C. Let $c_D \triangleq \det(C)$ and $c_{D-1} \triangleq \det(C_3)$, then from (A1.12) we have the following recursive equation

$$\begin{aligned} c_D &= (-1)^D l_D (-1)^{D-1} \sum_{j=0}^{D-2} \sum_{i=0}^{j} l_i - (-1)^D l_{D-1} \left[-(-1)^D \sum_{j=0}^{D-1} \sum_{i=0}^{j} l_i \right] + c_{D-1}, \\ &= -l_D \sum_{j=0}^{D-2} \sum_{i=0}^{j} l_i + l_{D-1} \sum_{j=0}^{D-1} \sum_{i=0}^{j} l_i + c_{D-1}, \end{aligned}$$

with the initial value

$$c_1 = \det \begin{pmatrix} l_0 & 0 & 1 \\ l_1 & l_0 & -1 \\ 0 & 0 & 1 \end{pmatrix} = (l_0)^2.$$

After some algebra, the solution of this equation can be known to be

$$c_D = -l_D \sum_{j=0}^{D-2} \sum_{i=0}^{j} l_i + \left(\sum_{j=0}^{D-1} l_j \right)^2. \tag{A1.13}$$

Finally, substituting (A1.13) and (A1.11) into (A1.8), we obtain

$$\begin{aligned} \det[M(L)] &= (l_D)^2 + 2l_D \sum_{i=0}^{D-1} l_i + \left(\sum_{i=0}^{D-1} l_i \right)^2, \\ &= \left(l_D + \sum_{i=0}^{D-1} l_i \right)^2, \\ &= \left(\sum_{i=0}^{D} l_i \right)^2. \end{aligned}$$

■

Proof of Theorem 5.1: Due to Lemma A1.1, with $J = 1$ and $K = D$, equation (A1.3) can be factored out as follows:

$$\begin{aligned} \det[A(L) - \lambda I] &= (-1)^{D+1}\lambda[\lambda^D(\lambda-1)^2 + (\lambda-1)P_\beta + P_\alpha P_L], \\ &= (-1)^{D+1}\lambda P_\Gamma(\lambda), \end{aligned} \tag{A1.14}$$

where

$$P_\Gamma(\lambda) \triangleq \lambda^{D+2} + \gamma_1\lambda^{D+1} + \gamma_2\lambda^D + \ldots + \gamma_{D+2}.$$

The coefficients of $P_\Gamma(\lambda)$ are related to Q and L through the following equation:

$$\begin{pmatrix} \gamma_1 + 2 \\ \gamma_2 - 1 \\ \gamma_3 \\ \gamma_4 \\ \vdots \\ \gamma_{D+1} \\ \gamma_{D+2} \\ 0 \end{pmatrix} = \begin{pmatrix} l_0 & 0 & 1 & 0 & \ldots & 0 \\ l_1 & l_0 & -1 & 1 & \ldots & 0 \\ l_2 & l_1 & 0 & -1 & & \\ l_3 & l_2 & & & & \\ & \vdots & & & & \\ l_D & l_{D-1} & & & & 1 \\ 0 & l_D & & & & -1 \\ 0 & 0 & 1 & 1 & \ldots & 1 \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \beta_0 \\ \beta_1 \\ \vdots \\ \\ \\ \beta_D \end{pmatrix} = M(L)Q\,,$$

where the last row corresponds to constraint (5.31). Due to Lemma A1.2, $M(L)$ is nonsingular since $\det(M(L)) = (\sum_{i=0}^{D} l_i)^2$, and $\sum_{i=0}^{D} l_i \geq 1$ (at least one flow is being quenched). Therefore, for any given characteristic polynomial coefficients $\Gamma = (\gamma_1, \gamma_2, \ldots, \gamma_{D+2})^T$, the corresponding vector of feedback gains exists and is given by

$$Q = [M(L)]^{-1}\tilde{\Gamma},$$

where $\tilde{\Gamma} \triangleq (\gamma_1 + 2, \gamma_2 - 1, \gamma_3, \ldots, \gamma_{D+2}, 0)^T$. ■

APPENDIX 2: PROOFS FOR SECTION 5.3

Consider again polynomial $P_\Gamma(\lambda)$ defined in (6.34):

$$P_\Gamma(\lambda) = \lambda^{D+2} + \gamma_1\lambda^{D+1} + \gamma_2\lambda^D + \ldots + \gamma_{D+2}. \tag{A2.1}$$

The closed loop system is asymptotically stable if and only if $P_\Gamma(\lambda)$ is Hurwitz, i.e., has all zeros located in the interior of the unit circle on the complex plane. The coefficients of $P_\Gamma(\lambda)$, $\gamma_i, i = 1, 2, \ldots, D+2$, depend on the vector of quenched traffics, $L = (l_0, l_1, \ldots, l_D)^T$, and on the controller gain, $Q = (\alpha_0, \alpha_1, \beta_0, \beta_1, \ldots, \beta_D)^T$. The problem of robust controller design is to choose $Q \in \mathbb{R}^{D+3}$ so that $P_\Gamma(\lambda)$ is Hurwitz for all $L \in \mathcal{L}$ where $\mathcal{L}$ is defined in (6.39).

Let Γ denote the vector of coefficients of $P_\Gamma(\lambda)$, i.e.,

$$\Gamma = (\gamma_1, \gamma_2, \ldots, \gamma_{D+2})^T.$$

Note that the coefficient of the highest power of $P_\Gamma(\lambda)$ is equal to 1 and therefore is not included as a component of Γ. Let $\mathcal{D}$ denote the stability domain in the coefficient space, i.e., a set in $\mathbb{R}^{D+2}$ such that for all $\Gamma \in \mathcal{D}$, the resulting polynomial $P_\Gamma(\lambda)$ is Hurwitz. The nature of the stability domain has been analyzed in a number of publications (see [19]-[20]). The following facts are of importance to our analysis:

(i). Every point on the boundary of $\mathcal{D}$ corresponds to a polynomial with at least one zero on the unit circle in the complex plane.

(ii). Let S be the set of all $\Gamma \in \mathbb{R}^{D+2}$ such that the resulting polynomial has at least one zero on the unit circle. Then S is generated by the equation

$$e^{j(D+2)\theta} + \sum_{i=1}^{D+2} \gamma_i e^{j(D+2-i)\theta} = 0,\ \theta \in [0, \pi],$$

or

$$1 + \sum_{i=1}^{D+2} \gamma_i e^{-ij\theta} = 0,\ \theta \in [0, \pi]. \tag{A2.2}$$

(iii). The set S is composed of three hypersurfaces. The first two are $(D+1)$-dimensional hyperplanes and correspond to $\theta = 0$ and $\theta = \pi$ in (A2.2):

$$S_0: \ 1 + \textstyle\sum_{i=1}^{D+2} \gamma_i = 0,$$
$$S_\pi: \ 1 + \textstyle\sum_{i=1}^{D+2} (-1)^i \gamma_i = 0,$$

and the third surface is generated by the movement of a D-dimensional hyperplane when the parameter θ takes its values in $(0, \pi)$:

$$S_\theta: \begin{cases} 1 + \sum_{i=1}^{D+2} \gamma_i \cos(i\theta) = 0, \\ \sum_{i=1}^{D+2} \gamma_i \sin(i\theta) = 0. \end{cases}$$

(iv). The set S partitions $\mathbb{R}^{D+2}$ into $(2D+5)$ open sets. The stability domain, $\mathcal{D}$, is one of them, specifically, the one containing the origin.

To utilize these facts, we represent the vector of coefficients, Γ, as follows:

$$\Gamma = \Gamma_\beta + \Gamma_\alpha^L, \tag{A2.3}$$

where, as it follows from(6.37),

$$\Gamma_\beta = \begin{pmatrix} \beta_0 - 2 \\ \beta_1 - \beta_0 + 1 \\ \beta_2 - \beta_1 \\ \beta_3 - \beta_2 \\ \vdots \\ \beta_D - \beta_{D-1} \\ -\beta_D \end{pmatrix}, \ \Gamma_\alpha^L = W_\alpha L = \begin{pmatrix} l_0\alpha_0 \\ l_1\alpha_0 + l_0\alpha_1 \\ l_2\alpha_0 + l_1\alpha_1 \\ l_3\alpha_0 + l_2\alpha_1 \\ \vdots \\ l_D\alpha_0 + l_{D-1}\alpha_1 \\ l_D\alpha_1 \end{pmatrix}.$$

Vector Γ_α^L depends on L whereas Γ_β does not. In addition, since the sum of the coefficients of $P_{\Gamma_\beta}(\lambda)$ is zero, no choice of the β_i's results in $P_{\Gamma_\beta}(\lambda)$ being Hurwitz. Therefore, the idea of the proof is as follows: First we choose $Q = Q_1$ so that the resulting $P_{\Gamma_\beta}(\lambda)$ is as "good" as possible, i.e., is defined by Γ_β belonging to the boundary of $\mathcal{D}$. At the second stage, using Γ_α^L, we modify this Q_1 so that $\Gamma_\beta + \Gamma_\alpha^L \in \mathcal{D}$ for all $L \in \mathcal{L}$. The following two lemmas implement this idea.

Lemma A2.1: Assume that Q_1 is given by

$$\begin{aligned} \alpha_0 &= \frac{D+4}{2(D+1)}, \\ \alpha_1 &= -\frac{D+2}{2(D+1)}, \\ \beta_0 &= \frac{3D}{2(D+1)}, \\ \beta_i &= \frac{D-2(i+1)}{2(D+1)}, \ i = 1, 2, \ldots, D. \end{aligned} \tag{A2.4}$$

Then, Γ_β belongs to the boundary of the stability domain $\mathcal{D}$. Moreover, Γ_β lies on the part of the boundary where S_θ and S_0, defined in (iii), intersect. □

Proof: Every point on the boundary of the domain $\mathcal{D}$ corresponds to a polynomial with zeros being either on or inside the unit circle (at least one is on the unit circle). We will show that P_{Γ_β} has two zeros equal to 1 and D zeros inside the unit circle.

The expression of $P_{\Gamma_\beta}(\lambda)$ can be obtained using the coefficients of Γ_β in (A2.3):

$$\begin{aligned} P_{\Gamma_\beta}(\lambda) &= \lambda^{D+2} + \sum_{i=1}^{D+2} \left(\Gamma_\beta\right)_i \lambda^{i-1}, \\ &= \lambda^D(\lambda-1)^2 + (\lambda-1)P_\beta(\lambda), \end{aligned} \tag{A2.5}$$

where

$$P_{\beta}(\lambda) = \sum_{i=0}^{D} \beta_{D-i}\lambda^i.$$

Since $P_{\beta}(1) = \sum_{i=0}^{D} \beta_i = 0$, we can factor out $(\lambda - 1)$ in $P_{\beta}(\lambda)$ to obtain

$$\begin{aligned} P_{\beta}(\lambda) &= (\lambda - 1)\sum_{i=0}^{D-1}\left(\sum_{j=0}^{D-1-i} \beta_j\right)\lambda^i, \\ &= (\lambda - 1)\sum_{i=0}^{D-1} \tilde{\beta}_i\lambda^i, \end{aligned} \quad \text{(A2.6)}$$

where $\tilde{\beta}_i \triangleq \sum_{j=0}^{D-1-i} \beta_j$. Substituting (A2.6) in (A2.5), we have

$$\begin{aligned} P_{\Gamma_{\beta}}(\lambda) &= (\lambda - 1)^2[\lambda^D + \sum_{i=0}^{D-1} \tilde{\beta}_i\lambda^i], \\ &= (\lambda - 1)^2 P_{\tilde{\beta}}(\lambda), \end{aligned} \quad \text{(A2.7)}$$

where $P_{\tilde{\beta}}(\lambda) = \lambda^D + \sum_{i=0}^{D-1} \tilde{\beta}_i\lambda^i$. Equation (A2.7) shows that $P_{\Gamma_{\beta}}$ has two zeros on the unit circle, and we now show that the zeros of $P_{\tilde{\beta}}(\lambda)$ are inside the unit circle. We start by evaluating $\tilde{\beta}_i$:

$$\begin{aligned} \tilde{\beta}_i &= \sum_{j=0}^{D-1-i} \beta_j = \frac{3D}{2(D+1)} + \sum_{j=1}^{D-1-i} \frac{D - 2(j+1)}{2(D+1)}, \\ &= \frac{1}{2(D+1)}\left[3D + \sum_{j=1}^{D-1-i}(D-2) - 2\sum_{j=1}^{D-1-i} j\right], \\ &= \frac{1}{2(D+1)}\left[3D + (D-1-i)(D-2) - 2\frac{(D-1-i)(D-i)}{2}\right], \\ &= \frac{1}{2(D+1)}(D+2-i)(1+i). \end{aligned}$$

Therefore,

$$\begin{aligned} 2(D+1)P_{\tilde{\beta}}(\lambda) &= 2(D+1)\lambda^D + \sum_{i=0}^{D-1}(D+2-i)(1+i)\lambda^i, \\ &= \sum_{i=0}^{D}(D+2-i)(1+i)\lambda^i, \\ &= \sum_{i=0}^{D} a_i\lambda^i, \end{aligned} \quad \text{(A2.8)}$$

where $a_i \triangleq (D+2-i)(1+i)$. To show that the polynomial $\sum_{i=0}^{D} a_i\lambda^i$ has all its zeros inside the unit circle, we use Raible's tabular form for Jury's stability test [21]. It is a $(D+1) \times (D+1)$

triangular table U (u_{ij} is defined for $j \leq D+2-i$), where the first row consists of the coefficients of the polynomial:

$$u_{1j} = a_{D+1-j},\ j = 1,2,\ldots,D+1, \tag{A2.9}$$

and the entries of each row i, $i = 2,3,\ldots,D+1$, are defined in terms of the entries of row $i-1$ as follows:

$$u_{ij} = u_{i-1,j} - \frac{u_{i-1,D+3-i}}{u_{i-1,1}} u_{i-1,D+4-i-j},\ j = 1,2,\ldots,D+2-i. \tag{A2.10}$$

The stability condition is that all the entries of the first column have the same sign.

With the a_i's as defined in(A2.8), the entries of the table are given by

$$u_{ij} = (D+2)\frac{v_{ij}}{v_{i-1,1}},$$

where

$$v_{ij} = (D+3-i-j)\left[(i+j)D^2 - (i^2+2ij-6i-5j+2)D + \frac{1}{3}(i^3+3i^2j-9i^2-15ij+26i+18j-12)\right]$$

$$i = 0,1,\ldots,D+1,\ j = 1,2,\ldots,D+2-i.$$

Indeed, after some algebraic manipulations, one can show that u_{ij} as defined above satisfies (A2.9) and (A2.10). The entries of the first column are

$$u_{i1} = (D+2)\frac{v_{i1}}{v_{i-1,1}},\ i = 1,2,\ldots,D+1.$$

Here

$$\begin{aligned} v_{i1} &= (D+2-i)\left[(i+1)D^2 - (i^2-4i-3)D + \frac{1}{3}(i^3-6i^2+11i+6)\right], \\ &= (D+2-i)w(i), \end{aligned}$$

where $w(i) \triangleq (i+1)D^2-(i^2-4i-3)D+\frac{1}{3}(i^3-6i^2+11i+6)$. The sign of v_{i1}, $i = 0,1,\ldots,D+1$, is equal to the sign of $w(i)$ which can be rewritten as follows:

$$w(i) = \frac{1}{3}i^3 - (D+2)i^2 + (D^2+4D+\frac{11}{3})i + D^2+3D+2.$$

To determine the sign of $w(i)$, we take its derivative:

$$\begin{aligned} w'(i) &= i^2 - 2(D+2)i + D^2 + 4D + \frac{11}{3}, \\ &= (i-i_1)(i-i_2), \end{aligned}$$

where

$$\begin{aligned} i_1 &= D+2-\frac{1}{\sqrt{3}}, \\ i_2 &= D+2+\frac{1}{\sqrt{3}}. \end{aligned}$$

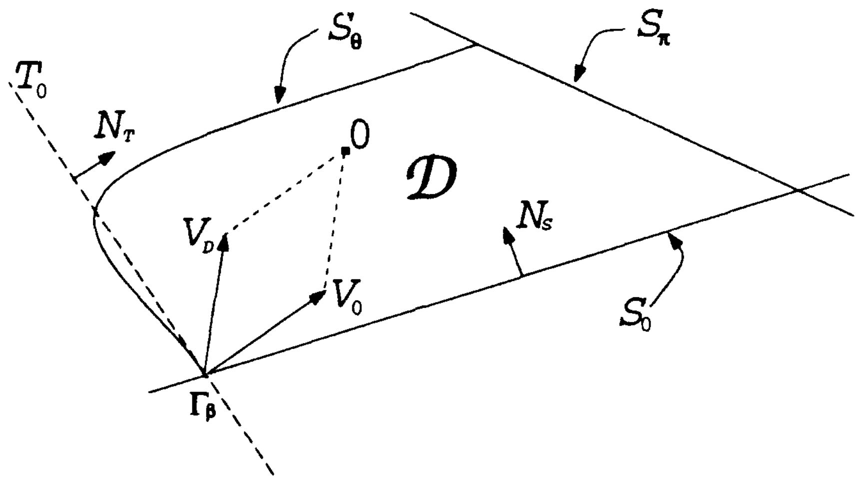

Figure A2.1: Stability domain in coefficient space.

Therefore, $w(i)$ is increasing for $i \leq D+1 < i_1$ and since $w(0) = D^2 + 3D + 2 > 0$, then $w(i) > 0$, $i = 0, 1, \ldots, D+1$. Thus, u_{i1}, $i = 1, 2, \ldots, D+1$ is positive. Since there is no change of sign in the first column of the table, $P_{\bar{\beta}}(\lambda)$ has all its roots inside the unit circle.

To complete the proof, we show that Γ_β lies on the part of the boundary where S_θ and S_0 intersect. Indeed, let S_θ^0 denote that part of the boundary, i.e., $S_\theta^0 = \lim_{\theta \to 0} S_\theta$. Then, using first order approximations of $\cos\theta$ and $\sin\theta$ ($\cos\theta \sim 1$, $\sin\theta \sim \theta$), S_θ^0 is defined by the equations

$$\begin{cases} \sum_{i=1}^{D+2} \gamma_i & = & -1\,, \\ \sum_{i=1}^{D+2} i\,\gamma_i & = & 0\,. \end{cases}$$

It can easily be shown that Γ_β satisfies these two equations. ■

As it follows from (A2.3), vector Γ_α^L can be represented as

$$\Gamma_\alpha^L = \sum_{i=0}^{D} l_i V_i\,,$$

where

$$(V_0\, V_1\, \ldots\, V_D) = W_\alpha = \begin{pmatrix} \alpha_0 & & & \\ \alpha_1 & \alpha_0 & & \\ & \alpha_1 & & \\ & & & \alpha_0 \\ & & & \alpha_1 \end{pmatrix}.$$

Let T_0 be the plane tangent to S_θ at the point Γ_β. Then, the two planes, S_0 and T_0, divide the space $\mathbb{R}^{D+2}$ into four quadrants (see Figure A2.1).

Lemma A2.2: Points $\Gamma_\beta + V_i$, $i = 0, 1, \ldots, D$ belong to the interior of the quadrant that contains the origin. □

Proof: If N_S and N_T are vectors normal to the hyperplanes S_0 and T_0, respectively, as shown in Figure A2.1, then we need to show that

$$\langle N_S, V_i \rangle \;>\; 0,\; i = 0, 1, \ldots, D, \qquad \text{(A2.11)}$$

$$\langle N_T, V_i \rangle \;>\; 0,\; i = 0, 1, \ldots, D, \qquad \text{(A2.12)}$$

where $\langle U, V\rangle$ denotes the dot product of the vectors U and V. Inequality (A2.11) can be easily shown to hold since, from the equation of the hyperplane S_0, we have $N_S = (1, 1, \ldots, 1)^T$ and therefore

$$\langle N_S, V_i\rangle = \alpha_0 + \alpha_1 = \frac{1}{D+1} > 0.$$

To find N_T, we first compute $N_\theta(\Gamma)$ which is a vector normal to S_θ at a point Γ, and then calculate

$$N_T = \lim_{\theta\to 0} N_\theta(\Gamma)\Big|_{\Gamma=\Gamma_\beta} \tag{A2.13}$$

Since S_θ is defined by the equations

$$\begin{cases} f_1(\Gamma,\theta) \triangleq \sum_{i=1}^{D+2} \gamma_i \cos i\theta + 1 = 0, \\ f_2(\Gamma,\theta) \triangleq \sum_{i=1}^{D+2} \gamma_i \sin i\theta = 0, \quad \theta \in (0,\pi), \end{cases} \tag{A2.14}$$

the j-th component of $N_\theta(\Gamma)$ is given by [22]

$$\begin{aligned} N_\theta^j(\Gamma) &= \frac{\partial f_1}{\partial \gamma_j}\frac{\partial f_2}{\partial \theta} - \frac{\partial f_1}{\partial \theta}\frac{\partial f_2}{\partial \gamma_j}, \\ &= (\cos j\theta)\Big(\sum_{i=1}^{D+2} i\gamma_i \cos i\theta\Big) - (\sin j\theta)\Big(-\sum_{i=1}^{D+2} i\gamma_i \sin i\theta\Big), \\ &= \sum_{i=1}^{D+2} i\gamma_i(\cos j\theta \cos i\theta + \sin j\theta \sin i\theta), \\ &= \sum_{i=1}^{D+2} i\gamma_i \cos(i-j)\theta. \end{aligned} \tag{A2.15}$$

Since $\sum_{i=1}^{D+2} i(\gamma_\beta)_i = 0$, N_θ vanishes in the limit $\theta = 0$. To avoid this, we factor out the term $\frac{-\sin^3\theta}{\sin(D+1)\theta}$ and show that N_T can be characterized as follows:

$$N_T = \lim_{\theta\to 0} -\frac{\sin(D+1)\theta}{\sin^3\theta} N_\theta\Big|_{\Gamma=\Gamma_\beta}.$$

Indeed, from (A2.14) we have

$$\begin{aligned} \gamma_1 &= -\frac{\sum_{i=2}^{D+1} \gamma_i \sin(D+2-i)\theta + \sin(D+2)\theta}{\sin(D+1)\theta}, \\ \gamma_{D+2} &= -\frac{\sum_{i=2}^{D+1} \gamma_i \sin(i-1)\theta - \sin\theta}{\sin(D+1)\theta}. \end{aligned}$$

Substituting in (A2.15) we obtain

$$N_\theta^j = \frac{-1}{\sin(D+1)\theta}\Big(\psi_j + \sum_{i=2}^{D+1} \gamma_i \psi_{ij}\Big),\ j = 1, 2, \ldots, D+2,$$

where

$$
\begin{aligned}
\psi_j &= \sin(D+2)\theta\cos(j-1)\theta - (D+2)\sin\theta\cos(D+2-j)\theta, \\
\psi_{ij} &= \sin(D+2-i)\theta\cos(j-1)\theta + (D+2)\sin(i-1)\theta\cos(D+2-j)\theta - i\sin(D+1)\theta\cos(i-j)\theta.
\end{aligned}
$$

For positive integers a and b, the following identities hold:

$$
\begin{aligned}
\sin a\theta &= \sum_{k=0}^{\lfloor\frac{a-1}{2}\rfloor}(-1)^k\binom{a}{2k+1}(\cos\theta)^{a-2k-1}(\sin\theta)^{2k+1}, \\
\cos b\theta &= \sum_{l=0}^{\lfloor\frac{b}{2}\rfloor}(-1)^l\binom{b}{2l}(\cos\theta)^{b-2l}(\sin\theta)^{2l},
\end{aligned}
$$

where

$$
\binom{n}{m} = \frac{n(n-1)(n-2)\ldots(n-(m-1))}{m!}.
$$

Therefore,

$$
\begin{aligned}
\sin a\theta\cos b\theta &= \sum_{k=0}^{\lfloor\frac{a-1}{2}\rfloor}\sum_{l=0}^{\lfloor\frac{b}{2}\rfloor}(-1)^{k+l}\binom{a}{2k+1}\binom{b}{2l}(\cos\theta)^{a+b-2(k+l)-1}(\sin\theta)^{2(k+l)+1}, \\
&= a(\cos\theta)^{a+b-1}\sin\theta - \left[\binom{a}{3}+b\binom{b}{2}\right](\cos\theta)^{a+b-3}\sin^3\theta + U^\theta_{ab}\sin^5\theta,
\end{aligned}
$$

where

$$
U^\theta_{ab} = \sum_{k=1}^{\lfloor\frac{a-1}{2}\rfloor}\sum_{l=1}^{\lfloor\frac{b}{2}\rfloor}(-1)^{k+l}\binom{a}{2k+1}\binom{b}{2l}(\cos\theta)^{a+b-2(k+l)-1}(\sin\theta)^{2(k+l-2)}.
$$

Using this identity, we obtain

$$
\begin{aligned}
\psi_j = &\left\{(D+2)((\cos\theta)^{D+j}-(\cos\theta)^{D+2-j})\right\}\sin\theta \\
&-\left\{\left[\binom{D+2}{3}+(D+2)\binom{j-1}{2}\right](\cos\theta)^{D+j-2}-(D+2)\binom{D+2-j}{2}(\cos\theta)^{D-j}\right\}\sin^3\theta \\
&+\left\{U^\theta_{D+2,j-1}-(D+2)U^\theta_{1,D+2-j}\right\}\sin^5\theta.
\end{aligned}
$$

The first term can be further factored out since

$$
\begin{aligned}
(\cos\theta)^{D+j}-(\cos\theta)^{D+2-j} &= (\cos\theta)^{D+2-j}\left[(\cos\theta)^{2(j-1)}-1\right], \\
&= (\cos\theta)^{D+2-j}\left[(1-\sin^2\theta)^{j-1}-1\right], \\
&= (\cos\theta)^{D+2-j}\sum_{k=1}^{j-1}(-1)^k\binom{j-1}{k}\sin^{2k}\theta, \\
&= (\cos\theta)^{D+2-j}\left[-(j-1)\sin^2\theta+\sin^4\theta\sum_{k=2}^{j-1}(-1)^k\binom{j-1}{k}(\sin\theta)^{2(k-2)}\right].
\end{aligned}
$$

Therefore,

$$\lim_{\theta\to 0}\frac{1}{\sin^3\theta}\psi_j = -(D+2)(j-1)-\binom{D+2}{3}-(D+2)\binom{j-1}{2}+(D+2)\binom{D+2-j}{2},$$
$$= \frac{1}{3}(D+1)(D+2)(D+3-3j) \triangleq \xi_j .$$

For ψ_{ij} we have

$$\begin{aligned}
\psi_{ij} &= \left\{(D+2-i)(\cos\theta)^{D-i+j}+(D+2)(i-1)(\cos\theta)^{D+i-j}-i(D+1)(\cos\theta)^{D+|i-j|}\right\}\sin\theta \\
&-\left\{\left[\binom{D+2-i}{3}+(D+2-i)\binom{j-1}{2}\right](\cos\theta)^{D-i+j-2}\right. \\
&\quad+(D+2)\left[\binom{i-1}{3}+(i-1)\binom{D+2-j}{2}\right](\cos\theta)^{D+i-j-2} \\
&\quad\left.-i\left[\binom{D+1}{3}+(D+1)\binom{|i-j|}{2}\right](\cos\theta)^{D+|i-j|-2}\right\}\sin^3\theta \\
&+\left\{U^\theta_{D+2-i,j-1}+(D+2)U^\theta_{i-1,D+2-j}-iU^\theta_{D+1,|i-j|}\right\}\sin^5\theta.
\end{aligned}$$

If $i-j\geq 0$, the coefficient of $\sin\theta$ is equal to

$$\begin{aligned}
&(D+2-i)\left[(\cos\theta)^{D-i+j}-(\cos\theta)^{D+i-j}\right] \\
&= (D+2-i)(\cos\theta)^{D-i+j}\left[1-(\cos\theta)^{2(i-j)}\right] \\
&= (D+2-i)(\cos\theta)^{D-i+j}\left[-\sum_{k=1}^{i-j}(-1)^k\binom{i-j}{k}\sin^{2k}\theta\right] \\
&= (D+2-i)(\cos\theta)^{D-i+j}\left[(i-j)\sin^2\theta-\sin^4\theta\sum_{k=2}^{i-j}(-1)^k\binom{i-j}{k}\sin^{2(k-2)}\theta\right].
\end{aligned}$$

If $i-j\leq 0$, it is equal to

$$\begin{aligned}
&(D+2)(i-1)\left[(\cos\theta)^{D+i-j}-(\cos\theta)^{D-i+j}\right] \\
&= (D+2)(i-1)(\cos\theta)^{D+i-j}\left[1-(\cos\theta)^{2(j-i)}\right] \\
&= (D+2)(i-1)(\cos\theta)^{D+i-j}\left[(j-i)\sin^2\theta-\sin^4\theta\sum_{k=2}^{j-i}(-1)^k\binom{j-i}{k}\sin^{2(k-2)}\theta\right].
\end{aligned}$$

Therefore,

$$\begin{aligned}
\lim_{\theta\to 0}\frac{1}{\sin^3\theta}\psi_{ij} &= (D+2-i)(i-j)I_{i-j>0}+(D+2)(i-1)(j-i)I_{i-j<0} \\
&-\binom{D+2-i}{3}-(D+2-i)\binom{j-1}{2}-(D+2)\binom{i-1}{3} \\
&-(i-1)\binom{D+2-j}{2}+i\binom{D+1}{3}+i(D+1)\binom{|i-j|}{2} \\
&\triangleq \xi_{ij},
\end{aligned}$$

where I is the indicator function. Thus,

$$\lim_{\theta\to 0} -\frac{\sin(D+1)\theta}{\sin^3\theta} N_\theta^j = \xi_j + \sum_{i=2}^{D+1} \gamma_i \xi_{ij}. \tag{A2.16}$$

To obtain N_T, we need to substitute $\Gamma = \Gamma_\beta$ in (A2.16). Since

$$\Gamma_\beta = \frac{-1}{2(D+1)}(D+4, 2, 2, \ldots, 2, -D-2)^T,$$

the j-th entry of N_T is given by

$$N_T^j = \lim_{\theta\to 0} -\frac{\sin(D+1)\theta}{\sin^3\theta} N_\theta^j \Big|_{\Gamma=\Gamma_\beta} = \xi_j - \frac{1}{D+1}\sum_{i=2}^{D+1} \xi_{ij}.$$

After some algebraic manipulations, it can be shown that

$$\sum_{i=2}^{D+1} \xi_{ij} = -\frac{1}{12} D(D+1)^2(D+2)(D+3-2j),$$

and as a result

$$\begin{aligned} N_T^j &= \frac{(D+1)(D+2)}{12}\left[4(D+3-3j) + D(D+3-2j)\right], \\ &= \frac{(D+1)(D+2)}{12}\left[(D+3)(D+4) - 2(D+6)j\right]. \end{aligned}$$

Finally, the dot product in (A2.12) is

$$\begin{aligned} \langle N_T, V_i\rangle &= \alpha_0 N_T^{i+1} + \alpha_1 N_T^{i+2}, \\ &= \frac{1}{2(D+1)}\left[(D+4)N_T^{i+1} - (D+2)N_T^{i+2}\right], \\ &= \frac{D+2}{24}\left[2(D+3)(D+4) - 2(D+6)((D+4)(i+1) - (D+2)(i+2))\right], \\ &= \frac{D+2}{12}\left[(D+3)(D+4) + (D+6)(D-2i)\right], \\ &\triangleq h(i). \end{aligned} \tag{A2.17}$$

Since $h(i)$ is a decreasing function of i and $h(D) = (D+2)(D+12)/12 > 0$, then $h(i) > 0$, $i = 0, 1, \ldots, D$.

Therefore, the set of points $\{\Gamma \in \mathbb{R}^{D+2} : \Gamma = \Gamma_\beta + V_i,\ i = 0, 1, \ldots, D\}$ belongs to the interior of one of the four quadrants determined by S_0 and T_0. Since $\Gamma_\beta + \sum_{i=0}^{D} V_i = 0$, then they belong to the quadrant containing the origin. ■

Proof of Theorem 5.2: The vector of control gains Q_1 defined in (A2.4) guarantees asymptotic stability of the network for all $L \in \mathcal{L}$ if and only if the set

$$\mathcal{R}(Q_1) = \left\{\Gamma \in \mathbb{R}^{D+2} : \Gamma = \Gamma_\beta + \sum_{i=0}^{D} l_i V_i = \Gamma_\beta + W_\alpha L,\ \forall L \in \mathcal{L}\right\} \triangleq \mathcal{R}_1$$

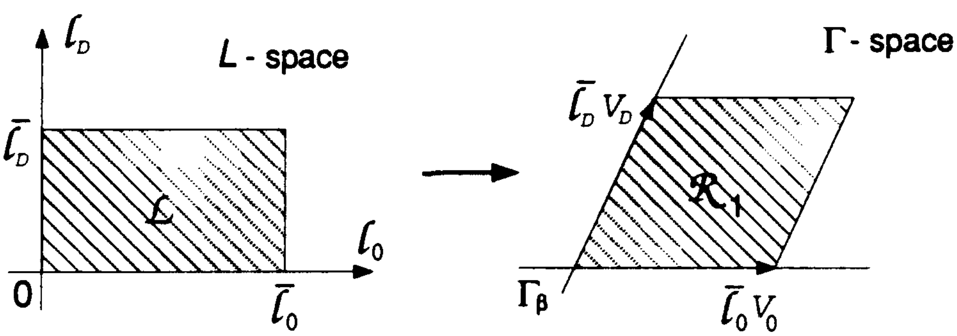

Figure A2.2: Mapping of $\mathcal{L}$ into $\mathcal{R}_1$.

lies inside $\mathcal{D}$. The set $\mathcal{L}$, as defined in (6.39), is a rectangle and $\mathcal{R}_1$ is obtained from $\mathcal{L}$ through an affine map since for every point $\Gamma \in \mathcal{R}_1$, there exists $L \in \mathcal{L}$ such that $\Gamma = \Gamma_\beta + W_\alpha L$. Therefore, $\mathcal{R}_1$ is a polytope, i.e., it is the convex hull of 2^{D+1} polynomials obtained by setting L to an extreme point in $\mathcal{L}$ (see Figure A2.2). Note that since the extreme point $\mathbf{0}$ is excluded from $\mathcal{L}$, the point Γ_β is excluded from $\mathcal{R}_1$.

From the foregoing discussion regarding the characterization of the stability domain $\mathcal{D}$ and the location of Γ_β (Lemma A2.1) and the V_i's (Lemma A2.2) resulting from the choice of $Q = Q_1$, we conclude that there exists k_1 sufficiently small such that the polytope

$$\mathcal{R}(Q_{k_1}) = \{\Gamma \in \mathbb{R}^{D+2} : \Gamma = \Gamma_\beta + k_1 W_\alpha L = \Gamma_\beta + \sum_{i=0}^{D} l_i k_1 V_i, \ \forall L \in \mathcal{L}\} \triangleq \mathcal{R}_{k_1}$$

is such that $\mathcal{R}_{k_1} \subset \mathcal{D}$ and that there exists k_2 sufficiently large such that the polytope $\mathcal{R}_{k_2} \not\subset \mathcal{D}$. More precisely, there exists $k^* > 0$ such that $\mathcal{R}_{k^*}$ intersects the boundary of $\mathcal{D}$ and for any $k \in (0, k^*)$, the polytope $\mathcal{R}_k$ is inside $\mathcal{D}$. The number k^* is the smallest k for which the polytope $\mathcal{R}_k$ comes in contact with the boundary of $\mathcal{D}$. That is, $\mathcal{R}_{k^*}$ is the largest polytope whose interior points correspond to stable polynomials. ■

Proof of Theorem 5.3: As it follows from the proof of Theorem 5.2, k^* in the smallest k for which the polytope $\mathcal{R}_k$ intersects the boundary of $\mathcal{D}$ which consists of the surfaces S_0, S_π and S_θ. The polytope $\mathcal{R}_k$ does not intersect the hyperplane S_0 for any k since, due to (A2.11), $\mathcal{R}_k$, $k \in \mathbb{R}_+$, lies in one of the two half planes generated by S_0. Therefore, $\mathcal{R}_k$ can intersect only S_π or S_θ or both. Let k_1 (k_2, respectively) be the smallest k for which $\mathcal{R}_k$ intersects S_π (S_θ, respectively). Then,

$$k_1 = \inf_{\substack{k \geq 0 \\ \Gamma \in \mathcal{R}_k \\ \Gamma \in S_\pi}} k, \tag{A2.18}$$

and

$$k_2 = \inf_{\substack{k \geq 0 \\ \Gamma \in \mathcal{R}_k \\ \Gamma \in S_\theta}} k. \tag{A2.19}$$

If $\mathcal{R}_k$ does not intersect S_π (S_θ, respectively) we set the solution of (A2.18) ((A2.19), respectively) to $+\infty$. From the above, it is clear that

$$k^* = \min\{k_1, k_2\}.$$

The optimization problem (A2.18) and (A2.19) can be rewritten in the form used in Theorem 5.3. Indeed, first we rewrite them as

$$k_1 = \inf_{\substack{k \geq 0 \\ L \in \mathcal{L}^k \\ \Gamma_\beta + W_\alpha L \in S_\pi}} k, \tag{A2.20}$$

and

$$k_2 = \inf_{\substack{k \geq 0 \\ L \in \mathcal{L}^k \\ \Gamma_\beta + W_\alpha L \in S_\theta}} k, \tag{A2.21}$$

respectively, where

$$\mathcal{L}^k = (\mathcal{L}_0^k \times \mathcal{L}_1^k \times \ldots \times \mathcal{L}_D^k) - \{\mathbf{0}\},\ \mathcal{L}_i^k = [0, k\bar{l}_i],\ i = 0, 1, \ldots, D.$$

Since for $L = \mathbf{0}$, Γ is equal to Γ_β which does not belong to S_π or S_θ, we can include $L = \mathbf{0}$ in the feasible set without changing the infimum in (A2.20) and (A2.21). Therefore,

$$k_1 = \inf_{\substack{k \geq 0 \\ \mathbf{0} \leq L \leq k\bar{L} \\ \Gamma_\beta + W_\alpha L \in S_\pi}} k, \tag{A2.22}$$

and

$$k_2 = \inf_{\substack{k \geq 0 \\ \mathbf{0} \leq L \leq k\bar{L} \\ \Gamma_\beta + W_\alpha L \in S_\theta}} k, \tag{A2.23}$$

where the constraint $\mathbf{0} \leq L \leq k\bar{L}$ denotes the component wise inequalities $0 \leq l_i \leq k\bar{l}_i$, $i = 0, 1, \ldots, D$.

The infimum in (A2.22) can be obtained in closed form. Indeed, substituting $\Gamma = \Gamma_\beta + W_\alpha L$ in the equation of S_π, the constraint $\Gamma_\beta + W_\alpha L \in S_\theta$ becomes

$$\sum_{i=0}^{D} l_i = \frac{3D + 5 + (-1)^D(D+3)}{2(D+3)} \triangleq a. \tag{A2.24}$$

The constraint $0 \leq l_i \leq k\bar{l}_i$ is equivalent to

$$l_i = k\bar{l}_i x_i,\ 0 \leq x_i \leq 1. \tag{A2.25}$$

Combining (A2.24) and (A2.25) and solving for k, (A2.22) becomes

$$k_1 = \inf_{0 \leq x_i \leq 1} \frac{a}{\sum_{i=0}^{D} \bar{l}_i x_i}. \tag{A2.26}$$

The solution of (A2.26) is obviously

$$k_1 = \frac{a}{\sum_{\substack{i=0 \\ i \text{ even}}}^{D} \bar{l}_i} = \frac{a}{\sum_{i=0}^{\lfloor D/2 \rfloor} \bar{l}_{2i}}.$$

Note that when $\bar{l}_i$, for all even i, are equal to zero, $k_1 = +\infty$, which means that $\mathcal{R}_k$ intersects S_θ but not S_π. ■

APPENDIX 3: PROOF FOR SECTION 6.1

Proof of Lemma 6.1: Obviously, there exists a permutation $\sigma(1), \sigma(2), \ldots, \sigma(N_1)$ of the set $\{1, 2, \ldots, N_1\}$, not necessarily unique, such that

$$\bar{q}_{\sigma(1)} \leq \bar{q}_{\sigma(2)} \leq \cdots \leq \bar{q}_{\sigma(N_1)}.$$

By relabeling $q_{\sigma(i)}$ as $q_i, i = 1, 2, \ldots, N_1$, Λ becomes a lower triangular matrix. Indeed, for the relabeled q_i's we have

$$\bar{q}_1 \leq \bar{q}_2 \leq \cdots \leq \bar{q}_{N_1}.$$

We need to show that $l_{ij} = 0$ for $i < j$. Assume that $i < j$ and therefore $\bar{q}_i \leq \bar{q}_j$. If $l_{ij} \neq 0$, then, at the steady state, l_{ij} connections throttled by link j traverse link i. According to (6.11), this means that $\bar{q}_j < \bar{q}_i$ which contradicts the assumption. Therefore, $l_{ij} = 0$ for $i < j$ and the statement of the lemma follows. ■

Proof of Lemma 6.2: Let $\mathcal{B}$ be the neighborhood of the steady state where the following holds:

(a). The sets C_{ij}, $i = 1, 2, \ldots, N_1$, $j = 0, 1, \ldots, N_1$, are constant.

(b). The rate of the remote throttled traffic remains less than the link capacity, i.e., $f_{\bar{i}\bar{i}}(n) < c$, $i \in \mathcal{N}_1$.

(c). The rate of traffic traversing underloaded links remains less than the link capacity, i.e., $f_i(n) < c$, $i \in \mathcal{N}_2$.

We show below that $\mathcal{B}$ is not empty. Note that, as it follows from (a), the values of r_i^0 and l_{ij}, $i, j \in \mathcal{N}_1$, are constant.

Assume that the network state is in $\mathcal{B}$. Then, according to the RTFS priority discipline introduced in (xv′), the remote throttled traffic does not incur any queueing delay at the buffer of link i since it is given priority and it has a rate $f_{\bar{i}\bar{i}}$ less than the link capacity. Traffic of throttled connections, on the other hand, might be queued at the buffer of the throttling link and share the residual capacity $c - f_{\bar{i}\bar{i}}$ on a first-come-first-serve basis. Therefore, connections see a change in their rate only at two points along their path: at their point of entry to the network and when they traverse the buffer of their throttling link. Thus, during the time slot $[n, n+1)$, since any connection (ab) throttled by link i is admitted to the network with rate $q_i(n+1-d_{ia}^{ab})$, it will reach the buffer of link j with rate

$$\begin{cases} q_i(n+1-\tau_{iaj}^{ab}), \text{ if the buffer of link } j \text{ precedes link } i \text{ on the path of connection } (ab), \\ \min\left\{x_i^{ab}(n-\tau_{ij}^{ab}), \dfrac{x_i^{ab}(n-\tau_{ij}^{ab})}{x_i(n-\tau_{ij}^{ab})}\left(c-f_{\overline{ii}}(n-\tau_{ij}^{ab})\right)\right\}, \text{ otherwise.} \end{cases} \quad \text{(A3.1)}$$

Therefore, as it follows from the definition of $f_{ii}(n)$, we have

$$f_{ii}(n) = \sum_{(ab)\in C_{ii}} q_i(n+1-\tau_{iai}^{ab}).$$

Let δ_{ij}^{ab} be equal to 1 if link i precedes link j on the path of connection (ab) and 0 otherwise. Then,

$$f_{\overline{ii}}(n) = \sum_{j=1}^{i-1} f_{ij}(n) + r_i^0\,, \quad \text{(A3.2)}$$

where

$$f_{ij}(n) = \sum_{(ab)\in C_{ij}} \delta_{ij}^{ab} q_j(n+1-\tau_{jai}^{ab}) + (1-\delta_{ij}^{ab}) \min\left\{x_j^{ab}(n-\tau_{ji}^{ab}), \frac{x_j^{ab}(n-\tau_{ji}^{ab})}{x_j(n-\tau_{ji}^{ab})}\left(c-f_{\overline{jj}}(n-\tau_{ji}^{ab})\right)\right\}.$$

Since f_{ij} is a function of $f_{\overline{jj}}$, $f_{\overline{ii}}$ can be obtained recursively starting from $f_{\overline{11}} = r_1^0$. It is clear from (A3.2) that $f_{\overline{ii}}$ it is a function of the states of link j (q_j and x_j^{ab} at present and in the finite past), $j = 1, 2, \ldots, i-1$.

Now we show that $\mathcal{B}$ is a nonempty neighborhood of the steady state, i.e., that there exists a ball of positive radius around the steady state where (a)-(c) hold true. Due to assumption (6.15), there exists a neighborhood $\mathcal{B}_1$ of the steady state where (a) is true. In addition, it follows from (A3.1) that the rate of any connection through any link depends continuously on the network state and as a result, $f_{\overline{ii}}(n)$ and $f_i(n)$ are continuous functions of the network state. Moreover, since $f_{\overline{ii}}(n)$ and $f_i(n)$ have their steady state values less than c, there exists neighborhoods $\mathcal{B}_2$ and $\mathcal{B}_3$ of the steady state where (b) and (c) hold true, respectively. Therefore, $\mathcal{B} = \mathcal{B}_1 \cap \mathcal{B}_2 \cap \mathcal{B}_3$ is a nonempty neighborhood of the steady state satisfying (a)-(c). ■

APPENDIX 4: PROOFS FOR SECTION 6.2

Proof of Theorem 6.1: Indeed, for each $i \in \mathcal{N}_1$, the pair of equations (6.24) and (6.25) represents the dynamics of link i with $f_{\overline{ii}}(n)$ being the only term that depends on the states of the other links. According to Lemma 6.2, $f_{\overline{ii}}(n)$ is a function of the states $Y_j(n)$ of links j, $j = 1, 2, \ldots, i-1$; therefore, the dynamics of link i is not influenced by the state of any link j with $j > i$. This means that there can be no dynamic coupling between the links since, for a coupling to take place, there must exist a set of links $i_1, i_2, \ldots, i_k$ in $\mathcal{N}_1$ such that the state of link i_j affects the dynamics of link i_{j+1}, $j = 1, 2, \ldots, k-1$, and the state of link i_k affects the dynamics of link 1. In other words, the linearized form of (6.24)-(6.25) has a block triangular structure where the i-th diagonal block corresponds to link i and is described by (6.26) and

(6.27). Therefore, the analysis of the network dynamics reduces to the analysis of (6.24) and (6.25) separately for each $i \in \mathcal{N}_1$ with $f_{\overline{ii}}(n)$ treated as a disturbance to the dynamics of link i. ■

Proof of Theorem 6.2: Obviously, the rate of nonthrottled connections cannot be increased without violating the feasibility constraint (6.30). To increase the rate of a connection (ab) throttled by link i while maintaining feasibility, we must decrease the rate of some other connection $(a'b')$ traversing link i in order to maintain the feasibility constraint (6.31). But since $\bar{r}_{ab} = \bar{q}_i$ (by definition of throttling link) and $\bar{r}_{a'b'} \leq \bar{q}_i$ for all connections $a'b' \in C(i)$ (by definition of the control protocol), the steady state allocation satisfies the requirements for max-min fairness. ■