b16145203

A NOVEL CONTROL THEORETIC MODEL FOR RESILIENT PACKET RING (RPR) FAIRNESS

A Thesis
Submitted To
The Graduate School Of
Ryerson University
In Partial Fulfillment of the Requirements
for the Degree Of
Master Of Applied Science

By:
Hossein Ghandehari-Alavijeh
BSc. in Electrical and Computer Engineering, IUST, Iran, 1992

Department of Electrical and Computer Engineering
University of Ryerson, Toronto, Canada
January 2004

© Copyright by Hossein Ghndehari-Alavijeh, 2004

PROPERTY OF
RYERSON UNIVERSITY LIBRARY

UMI Number: EC53463

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI®

UMI Microform EC53463
Copyright 2009 by ProQuest LLC
All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Author's Declaration

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Signature: .. ....

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Signature: ... ...

Abstract

A Novel Control Theoretic Model for Resilient Packet Ring (RPR) Fairness, Hossein Ghandehari-Alavijeh, Ryerson University, MSc. in Electrical and Computer Engineering, 2004.

Resilient Packet Ring (RPR) is a new Data Link Layer ring protocol. In RPR, the ring is a shared medium for multiple nodes compete to get a portion of shared bandwidth. Fairness algorithm is responsible for allocating fair bandwidth among competing nodes. In our research, we address the stability problems of the current RPR Fairness and introduce a new solution.

The present work is the first control theoretic approach to RPR Fairness and Congestion Control that rigorously models the dynamics of RPR Fairness algorithm by using control theory. The key idea is to involve the active nodes in the Fairness and Queue Congestion Control process which means developing a decentralized control system.

In RPR, when the number of nodes or the distance between the RPR nodes is high, the delay plays an important role in the behavior of the fairness which may lead to oscillation, instability and packet loss. We propose the implementation of Smith predictor as a valuable technique to overcome the effects of this delay and achieve higher throughput.

Our new theoretical insights allow us to design fairness and congestion control algorithms that achieve fair bandwidth allocation and high throughput with small buffer requirement even in presence of large delay and large number of active nodes in the ring.

Acknowledgements

First of all I would like to thank my thesis supervisor, Professor Kaamran Raahemifar for his invaluable direction and support. He played a fundamental role in shaping this research. He is a man of dream, insight and charm. Working with him is always rewarding. Also, I would like to thank Professor N. W. Ma, my co-supervisor. His knowledge, intelligence, dedication and experience have been truly the driving force of RPR research.

Also, I am grateful to Professor Stalin Boctor, the Dean of the Electrical and Computer Engineering, for his unconditional support of the RPR Project. I would like to thank Professor M. Jaseemuddin, Professor G. N. Khan and Professor S. Krishnan who have contributed tremendously to my learning throughout my graduate education and served on my graduate committee.

This research is made possible by the generous grant of Cisco Systems Inc. Cisco Systems, as the major supporter of development and standardization of RPR, has donated the RPR Lab to Ryerson University High Speed Networking research group.

Contents

Chapter 4

Chapter 5

Chapter 6

Chapter 7

Chapter 8

Chapter 9

List of Figures

Chapter 1

Introduction

One of the major challenges in a ring protocol like RPR is to allocate fair bandwidth for the nodes. As nodes need to use the ring to send their traffic, at some pint, they have to compete for the limited available link bandwidth. So there must be a mechanism on the nodes to adapt their traffic rate to a fair rate. In RPR all nodes are the same and no master node is selected, so RPR nodes need to communicate by a special control packet called fairness packet in case of congestion. This packet is used to send fairness information to the active nodes. Active nodes are the nodes that contribute traffic to the congested node and their traffic rate is adapted to the fair rate.

At any given time the active nodes form a distributed system. A distributed system in here is the programs running on a number of nodes connected via the network, that work together to accomplish a common goal which is fairness. As the number of active nodes varies dynamically, the system could be considered as a dynamic distributed system. Fairness packets are sent to upstream active nodes by the congested node as soon as it detects the congestion. This packet is used by active nodes to adjust their traffic rate to a fair rate.

RPR traffic adaptation in fairness mode can be modeled as an adaptive closed loop feedback system. Some researchers have conducted research during the last decade to analyze the closed loop dynamics and stability properties of congestion control algorithms by utilizing a control theoretical approach [23,24,25].

1.1 Importance of Fairness and Congestion Control

Since RPR is proposed as a MAN and WAN Protocol, it needs to overcome the problems of Wide Area Networks such as Fairness and Congestion control while the system has high link delay and large number of nodes. Fairness is one of the major parts of RPR protocol. It is responsible for allocating fair bandwidth among the active nodes. A Fairness algorithm guarantees fair rate allocation among competing active nodes as fast as possible. Congestion needs to be prevented so the smooth monitoring and control of the average queue level of the congested node can ensure that the system reacts rapidly to the unpredicted traffic'change. When average queue level is too high any bursty traffic may cause queue overflow and packet loss.

1.2 Problem Identification

The goal of this thesis is to utilize control theory to investigate the behavior of the existing Fairness algorithm and introduce a new model that has more controllable behavior. The problems with the existing Fairness algorithm are as follows.

1. There is no control over the congested queue level. Queue level is just monitored and when it exceeds a low threshold, fairness is then turned on. However simulation shows that when the number of active nodes is large, fairness control cannot guarantee any bound on the queue level, which may lead to packet loss. When queue level exceeds high threshold the active node shuts down its own traffic to avoid packet loss. This behavior may lead to undesired oscillation [2].

2. The only node that takes action when congestion happens is the congested node. It tries to avoid packet loss by decreasing its own traffic. At the same time it calculates and sends the fair bandwidth to the active nodes to throttle their rate. The controlling algorithm is not efficient enough and it causes the congested node to suffer.

3. Aggressive mode in the existing Fairness algorithm is fast but leads to oscillation in throughput, especially in the case of large number of active nodes and long delays, while Conservative mode is too slow to react to bursty traffic.

4. When the ring spreads in a wider area the propagation delay of the fiber links affects the behavior of RPR fairness and causes throughput oscillation. This is because, it takes time for the fairness packets to get to the active nodes and adapt their traffic rate. This delay is called Fairness Feedback Round Trip Time and needs to be compensated by a mechanism.

1.3 Applications

The current proposal for RPR protocol incorporates an algorithm called Fairness which is responsible for allocating fair bandwidth among the competing nodes. Any implementation of RPR in a wide area network needs to compensate the effects of the link delay on the stability of the system. Our Proposed solution called "Fairness and Congestion Control Model" can be implemented in the RPR Fairness algorithm so that the system achieves Fairness in traffic rate while has a robust control over congested queue level even in Wide Area Networks and when there is a large number of active nodes.

1.4 Challenges

The major challenges in controlling Congestion and Fairness in a WAN is to overcome the link delay effects on the stability of the Fairness algorithm. This is done in our proposed solution by implementing Smith Predictor which shows a robust response. The next challenge is to analyze and overcome the nonlinearity of the system. Our queue level control can decrease the nonlinearity but it doesn't overcome this effect. The implementation is another challenge because a more complicated Fairness algorithm needs more resources such as processor time and memory.

1.5 Research Contributions

The first contribution of this thesis is introducing a feedback control model for RPR Fairness. It responds rapidly to the congestion and avoids packet loss, while fully utilizes the bandwidth. Another contribution in this thesis is the queue congestion control which keeps the queue level close to a desired reference level. By choosing the reference level sufficiently larger than zero and sufficiently smaller than the buffer size, nonlinear effects may also be avoided. Also the outgoing flow rate may be kept close to the full capacity of the link. Our contributions can be itemized as follows.

1. In our proposed solution when the average queue level is high, Fairness algorithm decreases the bandwidth utilization to bring down the queue level. Fairness increases the bandwidth utilization when queue level is low.

2. In our proposed solution all the active nodes are involved in achieving fair rate and congested queue control so the nearby active nodes help the congested node in avoiding the congestion, which prevents the congested node from shutting down its own traffic. This avoids throughput oscillation.

3. Our proposed Fairness algorithm is as fast as Aggressive mode in achieving the fair traffic rate and stable like Conservative mode. By adjusting the controllable parameters dynamically the Fairness algorithm is guaranteed to be stable.

4. Our proposed Fairness algorithm utilizes a classic controller along with a Smith Predictor to compensate the effects of long delay.

1.6 Overview

This Thesis is organized as follows. In Chapter 2, an overview of RPR is given. In Chapter 3, the literature is reviewed. In Chapter 4, some important definitions and assumptions are introduced. In Chapter 5, our Control theoretic approach to Fairness Control Model is introduced and in Chapter 6, our Fairness and Congestion Control model is introduced and its stability is analyzed. In Chapter 7, several simulations are shown and the results are analyzed also a comparison is performed. In Chapter 8, Implementation considerations are discussed and in Chapter 9, a conclusion is given and the future works are discussed.

Chapter 2

RPR Overview

2.1. Emerging RPR

One of the major challenges in today's data communication technology is the migration of packet-based technologies from Local Area Networks to Metropolitan Area Networks [1]. Rapid increase of bandwidth usage in emerging applications challenges the existing data infrastructure capacities. These infrastructures are based on SONET/SDH technology which is a circuit-oriented technology designed and optimized for voice systems. Carrying high volume of data traffic over these systems make it difficult to provision new services and increases the cost of adding capacity.

One of the solutions is utilizing Ethernet in Metropolitan networks. Ethernet is a packet based technology that is designed and optimized for data traffic. Ethernet has evolved from 10Mbps to 100Mbps to 1Gbps and now to 10Gbps. Gigabit Ethernet is capable of supporting fiber spans of more than 80 Km. Although it is suitable for point-to-point and mesh networks, it is difficult to deploy it in ring networks as a shared medium. On the other hand most of the existing metro area networks are ring based, because SONET is typically deployed over fiber rings.

SONET in ring topologies implements a rapid protection mechanism that can restore connectivity in case of fiber or equipment failure. Ethernet lacks such a protection mechanism. An ideal technology could be the one that can exploit fiber rings (ring resiliency) completely and support all the advantages of data-based transport system like Ethernet. Resilient Packet Ring (RPR) protocol promises to be the one [2].

SONET suffers from some limitations. It was designed originally for point to point, circuit switched applications like voice traffic. This fact brings some disadvantages of using it for data traffic in metropolitan rings:

1. *Fixed Circuit:* Each circuit allocates a fixed bandwidth that may be wasted if not used. This disadvantage is significant for data traffic which is bursty in nature.
2. *Bandwidth waste in Mesh topology:* Creating a logical mesh circuit in a ring topology wastes bandwidth.
3. *Multicast traffic limitation:* Each source needs to allocate a separate circuit for each destination. Also a new copy of the packet should be sent to each destination which consumes the bandwidth.
4. *Wasted protection bandwidth:* Half of the ring bandwidth is reserved in SONET for protection and is not used in the normal situation.

Ethernet if considered as another option for the Metro Rings has its limitations too. Although Ethernet is efficient for data traffic, implementing it in ring topology is complex. It is optimized for point-to-point or meshed topologies, not for ring. Ethernet uses spanning tree on the bridges to avoid loops but it is slow to converge in case of fiber cut in the ring. The slow convergence is because of the fact that the information needs to be propagated to each upstream nodes. The lack of resiliency can be compensated to some level by link aggregation, but it is still too slow. Ethernet also doesn't provide fairness.

Different companies have developed their own solutions for ring metros and Cisco SRP/DPT is one of the successful solutions. In August 2000 Cisco documented its SRP MAC layer protocol in IETF as RFC 2892 [3].

Nortel Network's OPTera Packet Edge System (OPES) was introduced as a solution for ring networks [4]. It is an interface card that inter-works with OPTera Metro 3000 Multi-service Platform series. It introduced an efficient transmission solution of packet and TDM traffic on the same physical SONET infrastructure.

In December, 2000, IEEE 802.17 working group was formed to standardize Resilient Packet Ring (RPR) protocol [2].

2.2. Industry Motivations

Optical networking has changed the network infrastructure by offering cost effective high bandwidth technologies. The current trends in the networks as follows.

1. The backbone is becoming an optical mesh with a clear focus on the maximum utilization of the installed base.
2. The Metro core is transferring from a SONET/SDH ring-based network to a pure low-cost WDM network with the main purpose of aggregating different traffic types and customers on wavelengths. Network intelligence is pushed from the metro core into the metro access.
3. The metro access is being made optical. RPR will dominant metro access and its network intelligence will be distributed among the nodes and will support functionality, enabling immediate and customer-managed provisioning of service and applications.
4. The customer drop will generally be Ethernet, allowing for up to 100Mbps and in some cases gigabit Ethernet speed will be applicable.

RPR would be the main solution for Metropolitan Area network (MAN) access networks [1].

2.3. RPR Market Overview

Market analysts believe that the network costs for service providers are rising more rapidly than their revenues. This forces Carriers to focus almost exclusively on cost of any solution. They are in need of a new way to gain revenue from differentiated data services while supporting their legacy circuit-based voice systems.

Ethernet is efficient in LAN and Point-to-Point implementation. It provides low-cost solutions for LAN and Point-to-Point. However, the majority of networks in the MAN are ring-based with rigorous Fault, Configuration, Accounting, Performance, and Security Management (FCAPS)-compliant requirements on network availability. Therefore, there is an immediate need for new technology standards that provide a low-cost solution for building highly scalable resilient ring networks, which are optimized for packet transport and able to provide deterministic traffic performance.

By implementing RPR, existing equipment can be used by merely upgrading the interface cards, so RPR Implementation offers cost saving for the service providers [5]

2.4. RPR Characteristics

The characteristics of the RPR [1, 2] could be summarized as follows:

1. *Resiliency:* RPR offers high metro access network reliability, by fiber protection and restoration capabilities (less than 50 msec) and also automatic topology discovery for ease of configuration.
2. *Fairness:* RPR offers fairness management.
3. *Scalability:* RPR is capable of scaling with the number of users and applications.
4. *Predicted performance:* High priority packets are delivered with minimal jitter and latency.
5. *Optimized for the Ring topology*
6. *Supports the legacy voice services:* RPR is compatible with SONET.
7. *Efficient Bandwidth utilization:* By offering Spatial Reuse and utilizing both rings RPR offers very efficient bandwidth utilization. Also Multicasting is very efficient because one RPR multicast packet can be transmitted around the ring and can be received by multiple nodes.
8. *Operation, Administration and Maintenance (OAM) capability*
9. *Support for various physical layers:* (Gigabit Ethernet, 10 GigE, SONET/SDH, and WDM)
10. *Interoperability:* The devices from different vendors can be configured on the same ring.
11. *Compatibility:* It is compatible with relevant portions of other Ethernet standards (e.g., 802.1d, 802.1q).

2.5. RPR Fairness

Fairness is a mechanism for calculating and reporting fair rates of a ring access for fairness eligible traffic [3]. The use of fair rates prevents one station from occupying a disproportionate share of available ringlet capacity with respect to other stations on the ringlet. Fairness eligible traffic passes through Secondary Transit Queue (STQ).

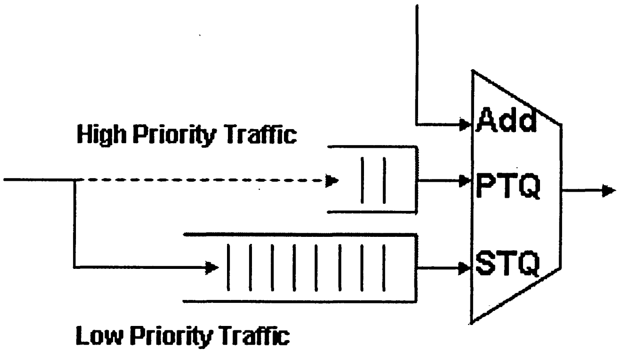


Figure 2.1: Primary and Secondary Transit Queues in RPR nodes

High Priority traffic passes through Primary Transit Queue (PTQ) and is not affected by fairness control. Fairness algorithm on the congested node computes a fair rate value. This rate provides the basis for the computation of other rates that are communicated to active nodes on the ring. Fairness information is carried by fairness frames. A fairness frame is carried on the ring, opposing that of the data flow sent. A fairness frame allows a station to advertise a fair rate value to its upstream neighbors.

A node is not permitted to use more than its fair rate of available capacity for the insertion of fairness eligible traffic when congestion has been detected on a ring. This restriction prevents a node from utilizing a disproportionate share of available capacity by virtue of its relative position on the ring. There are two modes of Fairness, Aggressive and Conservative. Conservative fairness is slow in convergence but more stable while Aggressive is fast in convergence but less stable.

Chapter 3

Literature review

3.1. Academic Research on RPR

Standardization of RPR by IEEE 802.17 has put this protocol under attention by Industry and Academic world. It is extremely attractive for industry because of its market potential and for Universities because of the challenges it faces in its promises. Some Universities have established their own RPR research groups and in some cases the researches are done by individuals.

Vienna University of Technology (Austria) was one of the first universities that worked closely with IEEE 802.17 in developing the new RPR technology [6]. Their contribution into the group in understanding the nature of this emerging ring technology was significant especially from theoretical point of view. They proposed a new solution for RPR.

Researchers in Simula Research Laboratory in University of Oslo (Norway) introduced a fairness algorithm for RPR [7]. They also developed a java simulator to simulate RPR network. Their approach is to modify the RPR fairness to fit in short distance LANs to gain a better performance because the RPR fairness generally is optimized for long distance MANs. Their approach is to implement triggers instead of times to meet the tighter latency requirements of high-speed networks.

In Munich University of Technology (Germany), a research is done on Single Failure situations in RPR rings [8]. Protection mechanism is analyzed and new approaches are introduced.

Rice University (US) RPR group has published their work recently [9] and introduced their bandwidth allocation algorithm, called Distributed Virtual-time scheduling in Rings (DVSR) and fairness algorithm RIAS to achieve less Oscillation while obtaining a complete spatial reuse. The key idea is for nodes to complete a simple lower bound of temporally and spatially aggregated virtual time using per-ingress counters of packet arrivals.

Carlton University (Canada) optical Network Laboratory research group joint with Nortel Networks conducted some researches in performance analysis of RPR [10]. They showed that the single buffer technology which is the Nortel implementation of RPR called OPTera can achieve as high as 95% utilization with very low ring access delay and is an excellent candidate for 802.17. The major advantage of this model is that, it is extremely simple to implement. The IEEE 802.17, standard RPR will support both single and double transit buffer.

Ryerson University (Canada) RPR research group has conducted several researches in RPR and developed an OPNET Model based on the RPR proposal Draft. This group also developed a novel Fairness and Congestion Control model for RPR by using Control Theory. The closed-loop nature of Fairness causes throughput instability when the ring delay and the number of nodes involved are significant. This thesis is the first attempt to approach RPR Fairness using Control Theory. This approach gives this group the chance to use powerful control theories to develop a new Fairness algorithm and analyze its behavior.

3.2. Academic Research on Congestion Control using Control Theory

Many researches have been done in the recent years to develop theoretical control models for computer networking and internet protocols. One of the most important problems in internet protocols is the congestion control. Un-controlled congestion may cause packet loss. When congestion happens there should be some mechanisms to acknowledge the sources to decrease their rate or change their other controllable parameters. The closed loop nature of these protocols can lead to oscillation or instability. By modeling these systems to classic control systems we can utilize control theory to stabilize the protocol responses. In case of Congestion, Fairness will be an important issue. Fairness is a mechanism to allocate a limited shared resource fairly among the competent sources to overcome congestion.

This thesis is the first of its kind that approaches RPR Fairness using Control Theory, but there have been some other attempts to model other congestion and fairness mechanisms using control theory that we refer to them here, as a starting point.

The first mechanisms relate to internet protocols like TCP congestion control. In [11] a new model for TCP's window control mechanism is developed. In [12] a dynamic TCP model is developed to analyze and design an Active Queue Management (AQM) control systems, using RED. The bottlenecked queue level is selected as the observation parameter. The queue level is controlled by changing the window size.

In [12] the stability of an internet-like congestion communication network is analyzed by modeling the TCP congestion control as a control system. The network consists of sources and links with heterogonous propagation delays. Each link sets a price per unit flow, based on the aggregate flow through that link. The sources set their transmission rates based on the aggregated price they see. In the absence of delays this scheme is globally stable but when delay is significant the system goes to instability. Then the stability conditions are discussed.

Many works in the area of ATM are done to theoretically model and analyze the congestion control in ATM switches. In [13] a new feedback control algorithm is introduced for Available Bit Rate (ABR) service in ATM. It is mentioned that the difficulties in providing ABR service is because of the burstiness nature of data traffic, the dynamic nature of the available bandwidth, as well as the feedback delay. Then a Control theoretical methodology is proposed for ABR congestion control algorithms by separating the problem to Rate Reference Control (RRC) and Queue Reference Control (QRC). This way the output rate of the congested switch and queue level would be controlled at the same time.

[14] introduces a dynamic Max Min Fairness for Storage Area Networks in Ring networks with spatial reuse and [15] discusses the allocation of the resources in a shared environment. In [16] a decentralized control law is developed at the end systems and routers in the internet, to provide high utilization, low queuing delay and dynamic stability and fairness. Then this control law is extended at TCP sources, provided that a bound on round trip time is known. Early Congestion Notification (ECN) marking is used to carry congestion information between Links and sources. In [17] the focus is on global stability of this controlled system in case of multiple sources and a

single link network. In [18] a classic control theory model is introduced to achieve QOS and fairness among several competent applications. The analysis is done in discrete time domain but delay is not put into consideration. A PID controller is implemented and its stability criteria are introduced. In [19] a TCP-like protocol called XCP is developed using control theory. In [20] analysis of the stability of a TCP-like congestion control is presented. In [21] TCP inefficiency and instability is addressed in case of higher delay. As a solution an internet congestion control is introduced that outperforms TCP and remains efficient, fair, scalable and stable as the bandwidth-delay product increases. This new eXplicit Control Protocol, XCP, generalizes the Explicit Congestion Notification (ECN). In addition XCP introduces the new concept of decoupling utilization control from fairness control. In [22] a rate based ABR control is introduced in ATM networks. In [23] the focus is on delayed networks. Delayed networks are fundamentally more complicated and difficult to control.

In [24] and [25] high speed ATM networks have been approached using control theory and Smith Predictor [26] is proposed to avoid oscillatory behavior. In [25] the effect of the delay on the queue level is compensated using Smith predictor. Each source has limited information about the network and needs to depend on its own assumptions,

In our approach each node creates a view of the network as if all other nodes see the same delay, have the same parameters and behave the same way it does. Based on this assumption it compensates the effect of the delay and adapts its traffic rate to achieve queue congestion control and fairness. Because all the nodes remove the delay from their Fairness Control feedback loop using Smith Predictor and adjust their controller parameters dynamically the overall adaptation is smooth and with minimum transient time. Our model can achieve Queue Congestion Control and Fairness while utilizes the maximum bandwidth and guarantees minimum packet loss.

Chapter 4

Definitions and Assumptions

Fairness algorithm is responsible for allocating fair bandwidth among the competing nodes over the shared bandwidth. To analyze the behavior of Fairness we define some terms in section 4.1.

4.1. Definitions and Assumptions

1) It is assumed that the control theoretic model of the Fairness Algorithm can be approximated by linear equations.
2) The constant time interval $[k, k+1]$ is the sampling time.
3) Node S_h is the upstream of node S_i when $h < i$. Traffic flow is from upstream to downstream.
4) c is the total traffic rate that congested node sends out. This traffic is the actual transmit rate or service rate of the congested link.
5) $c_{\max}$ is the maximum shared bandwidth.
6) Congested node is a node that detects the congestion and goes to fairness mode. In this work just single chock congestion is discussed.
7) Active node is a node in a congestion domain that contributes traffic into the congested node S_j and its traffic is adapted to the fair rate. All active nodes would be considered to be part of S_j congestion domain.
8) l is the number of active nodes. It is assumed to be fixed during the adaptation.
9) x_i is the traffic rate that active node S_i sends to the ring.
10) $x_i(k)$ is active node's traffic at sampling interval k in discrete time domain.

11) $\bar{x}(t)$ in our formulas is the average traffic rate of all the active nodes at the time of adaptation.

12) u_i is the compensation value that will adjust the queue level and x_i.

13) Semi-Active node is a node in a congestion domain that contributes traffic into the congested node S_j and its traffic is less than fair rate.

14) M is the total number of Active and Semi-Active nodes, fixed during the adaptation.

15) $M - l$ is the number of Semi-Active nodes which their traffic (r_i) is less than the fair rate.

16) r_i is the traffic rate of a Semi-Active.

17) R is the traffic of all the Semi-Active nodes received by the congested node, $R = \sum r_i$.

18) Traffic received by the congested node is $\sum x_i + R$, which in steady state is equal to c.

19) Fairness eligible traffic is the low priority traffic that is affected by fairness algorithm. The high priority traffic are not affected by fairness.

20) STQ or Secondary Transit Queue is a queue that fairness eligible traffic is buffered.

21) k_c is the controllable parameter of the Fairness Control. It shows how fast the adaptation of traffic rate is. The bigger the value of k_c, the faster the fairness adaptation.

22) k_q is the controllable parameter of the Queue Congestion Control. It shows how fast the adaptation of Queue level is.

23) Fairness Feedback Round Trip Time RTT, is assumed to be fixed during the adaptation period.

24) Each node has a static weight ω_i. This weight represents the proportion of the Bandwidth it must utilize. As number of active nodes varies we need to calculate the dynamic weight for nodes as

$$\gamma_i = \frac{\omega_i}{\sum_{n=1}^{l} \omega_n},$$

where γ_i is the dynamic weight of the node S_i and $\sum_{i=1}^{l} \gamma_i = 1$.

In case of same static weight $\omega_1 = \omega_2 = ... = \omega_i$, the dynamic weight would be

$$\gamma_i = \frac{\omega_i}{l\omega_i} = \frac{1}{l}.$$

4.1.1. Fluid Flow Model

As in many other works in the literature, the analysis in this thesis is based on a fluid approximation of the traffic flow.

Definition: Fluid flow model is a model in which the packets are assumed to be infinitely small in size so that the traffic flow sent by a node can be seen as a fluid flow. Different traffic sent to different nodes may be considered as sub-flows. This assumption is more accurate when the packet size is fairly small.

4.1.2. Max-Min Fairness

In a dynamic scenario each node exchanges data with several other nodes at the same time. We call each of them a flow. Let x_f denote the rate of flow f. Let $c_{\max}$ be the capacity of link L and let F_L be the set of all the flows using link L, and also c denotes the sum of rates of all the flows using link L.

Definition: An allocation is said to be feasible if it is r_f>=0 for any flow f and $c \le c_{\max}$ for any link L.

Definition (Max-Min Fairness): An allocation of a rate x_f for flow f is Max-Min fair if it is feasible and for each flow f, x_f cannot be increased (while maintaining feasibility) without decreasing $x_{f'}$, for some flow f' for which $x_{f'} \le x_f$ [27].

Definition (Dynamic Max-Min Fairness): An allocation of a rate x_f for flow f is Max-Min fair if it is feasible and for each flow f, x_f cannot be increased (while maintaining feasibility) without decreasing $x_{f'}$, for some flow f' for which $\frac{x_{f'}}{\gamma_{f'}} \leq \frac{x_f}{\gamma_f}$ [14].

In the discussions in this thesis it is assumed that RPR Fairness is a Dynamic Max-Min Fairness.

4.1.3. Delay

4.1.3.1 Delay Sources

The total delay of sending and receiving packets between two nodes on a link can be represented by the formula

$$T_d = T_{dp} + T_{dt} + T_{dq},$$

where T_d is the total delay. T_{dp} is the propagation delay which equals to the link distance divided by speed of the light in fiber ($0.66c$ or $198000\ km/s$). Each kilometer of fiber has 5 $\mu\sec$ Propagation delay. T_{dt} is the transmission delay which is equal to the number of bits in each packet (MTU for the worst case) divided by the link bandwidth. T_{dq} is the queuing and processing delay which equals to the number of nodes multiplied by the processing delay plus sum of the queuing delays.

Assuming that the links inside a congestion domain are not congested and that processing delay is in the order of 10^{-5} seconds we can say that T_{dq} is negligible. This assumption is based on a single chock congestion where there is just one congested node. Therefore the links inside its congestion domain are not congested. When number of bits transmitted on a link in one second is much higher than MTU (in bits), transmission delay (T_{dt}), also becomes negligible. This means that for an RPR ring, it is the propagation delay (T_{dp}) of the physical link that determines the delay of the link. In our analysis whenever we refer to delay between two nodes (T_{fw} or T_{fb}), we mean the total delay T_d. A 100km distance between two nodes creates 0.5 $m\sec$ delay.

4.1.3.1 Delay Types

The delay in the transmission of fairness packets from the congestion node to an active node is called feedback delay and is shown by T_{fb}. The traffic generated by the active node encounters with its forward delay T_{fw}.

The forward delay differs based on the distance and the situation of the nodes between source and the congested node. The overall forward delay is T_{fw}. .These delays contain a fixed component of propagation delay and packet processing, and a time varying component of queuing delay. The time that the fairness packet needs to get to the active node and change its rate (T_{fb}), plus the time that this change will need to get to the output queue of the congested node (T_{fw}), is called Fairness Feedback Round Trip Time, RTT ($RTT = T_{fw} + T_{fb}$). Each active node estimates the Fairness Feedback RTT by sending a packet to the congested node and receiving the response and calculating the Round Trip Time.

In our discussions Fairness Feedback Round Trip Time, RTT , is assumed to be fixed during the adaptation period.

4.1.4. Congestion Domain

RPR Fairness system consists of a single bottleneck node fed by multiple traffic sources in the ring [2]. In order to analyze the fairness behavior using control theory, a mapping needs to be done between RPR Fairness and the classic control system.

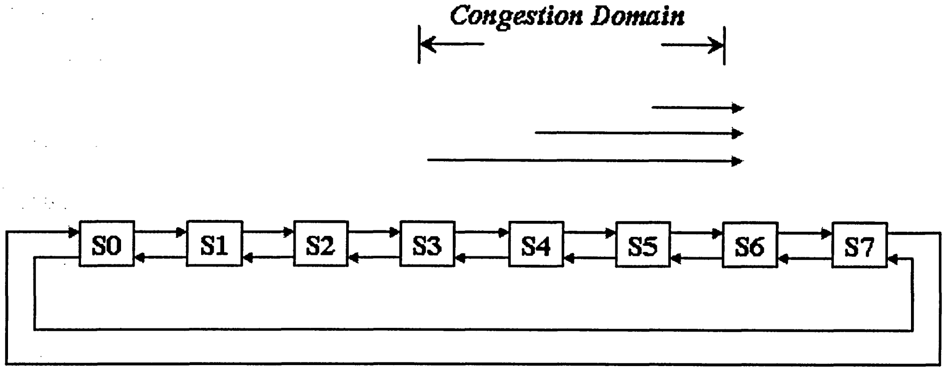


Figure 4.1: RPR Ring with S_3, S_4 and S_5 active nodes that contribute to the traffic of the congested node S_5 and their traffic need to be adapted.

At any given time a node that detects the congestion goes to Fairness mode and is called congested node. All the nodes that contribute to the traffic of the congested node and their traffic are eventually adapted to fair rate are considered as Active nodes. Congested node monitors the traffic that needs to be serviced (the received traffic and its own traffic) and if it is more than a low threshold, it sends a feedback to its active upstream nodes to inform them of its congestion.

Figure 4.1 shows a scenario that nodes S_3, S_4 and S_5 attempt to send a traffic to node S_6,that is more than the ring available bandwidth. In this situation node S_5 detects the congestion first.

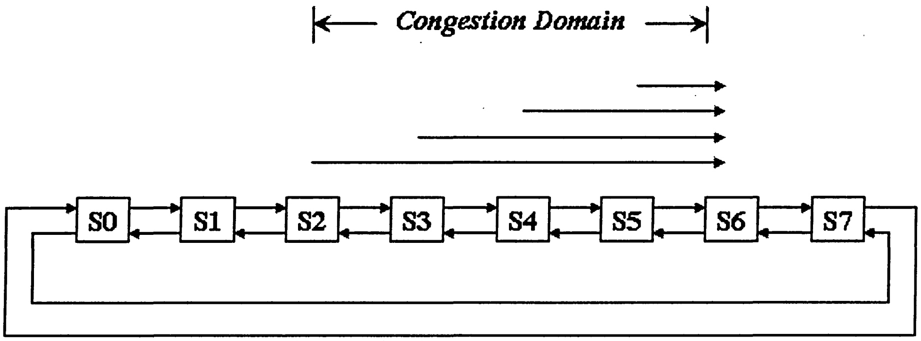


Figure 4.2: S_2 starts sending traffic and becomes a member of the S_5 congestion domain.

The congestion domain will vary based on the number of nodes that are active. In Figure 4.2 node S_2 starts sending traffic to node S_6. At this time the congestion domain changes and node S_2 becomes one of the active nodes to this new congestion domain.

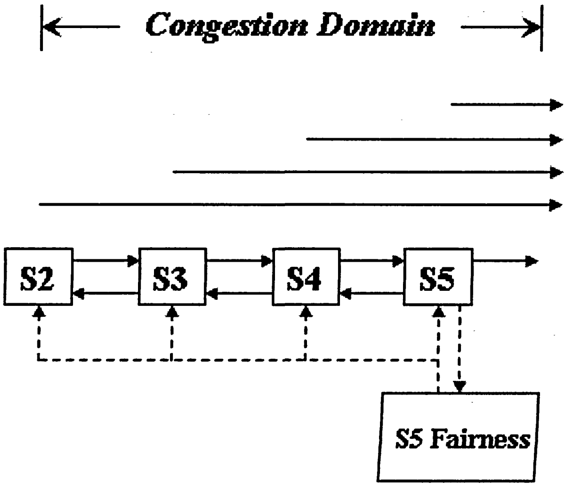


Figure 4.3: Active nodes and congested node with its feedback messages in the congestion domain of Figure 4.2.

The Fairness algorithm is dynamic, because at any given time each node may go to a different State. A node is considered as Congested when the traffic rate that it needs to service goes beyond a known low threshold. As soon as the node goes to congested state it recognizes all the upstream nodes that contribute to its traffic. All those nodes are known as Active or Semi-Active nodes for this congestion domain. Several congestion domains may overlap, and a node might be in several congested domain.

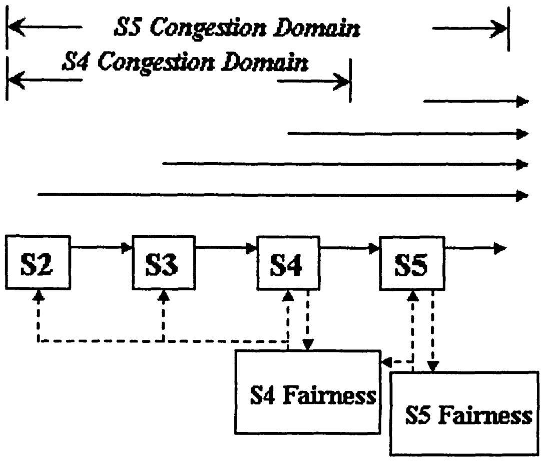


Figure 4.4: When several congestion domain are overlapped, the domain of the congested node with lower fair rate is dominant.

In Figure 4.4 each node attempts to send 250 Mbps at time t=0, so node S_4 and node S_5 will be congested and the fair rate for S_4 is $\frac{622}{3} Mbps$ and for S_5 is $\frac{622}{4} Mbps$, so Fair rate of S_5 is less than Fair rate of S_4, and therefore, it becomes dominant congestion domain. In this thesis we attempt to map the congestion domain to a control system.

Chapter 5

Control Theoretic Approach to Fairness

RPR Fairness algorithm can be considered as a dynamic distributed system. It is a distributed system because Fairness algorithm on the congested node and the active nodes run as a distributed application to achieve the fair rate. They try to control their output. This eventually decreases the rate received by the congested node, decreases its queue level and prevents it from packet loss while guarantees a fair rate allocation for active nodes. The job of the fairness algorithm is to adjust the traffic of all those nodes to achieve a smooth output. The steady State rate for all those nodes is eventually the fair rate.

5.1. Introduction to RPR Fairness

Based on the RPR draft, Conservative mode can be mapped roughly as an On-Off control. RPR congestion detection consists of two methods, first overall traffic and second STQ queue. In both of them low threshold and high threshold are used for On-Off control.

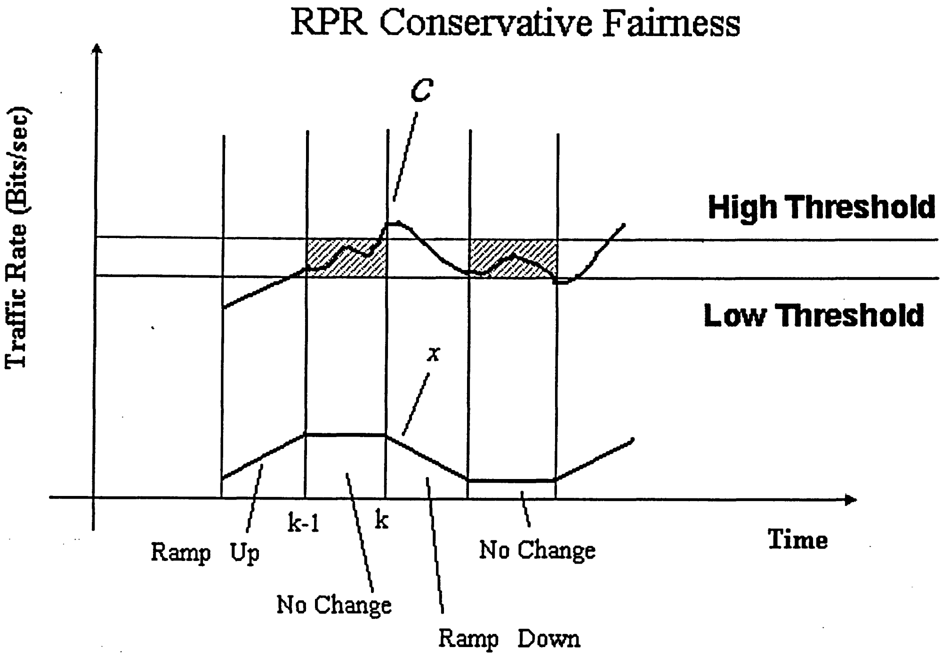


Figure 5.1: On-Off Congestion Detection.

In discrete domain and conservative mode if $c(k)$, the output rate of the congested node, exceeds the low threshold the system recognizes transitioning into congestion. When the system is congested and its $c(k)$ is higher than the high threshold it decreases its rate and throttles other active nodes to its pre-calculated fair rate. If $c(k)$ is less than low threshold, then the node increases its rate. When $c(k)$ is between low threshold and high threshold, the node does not change its rate.

$$x_i(k) = \begin{cases} x_i(k-1) - \theta_1 x_i(k-1) & c(k) > High\_Threshold\,(RampDown) \\ x_i(k-1) + \theta_1 (c_{\max} - x_i(k-1)) & c(k) < Low\_Threshold\,(RampUp) \\ x_i(k-1) \quad Low\_Threshold < c(k) < High\_Threshold\,(NoChange) \end{cases}$$

From this equation it can be seen how Conservative model adapts its rate to achieve fairness. At the same time the fair rate is calculated based on the available bandwidth and number of active nodes and is sent to the active nodes to throttle their rate to this pre-calculated rate.

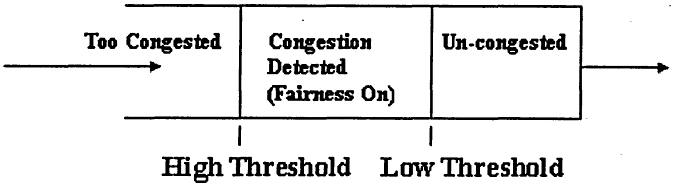


Figure 5.2: STQ Queue Congestion Detection

One of the weaknesses of the current RPR Fairness algorithm is the non-linearity, or its On-Off behavior when congestion happens. There is a parameter which works as a switch to let the node to add traffic or not at all. When congestion happens and the queue level is higher than high threshold, this parameter becomes off (False) which means the node is not allowed to add any traffic to the ring.

The following figure shows how current fairness works.

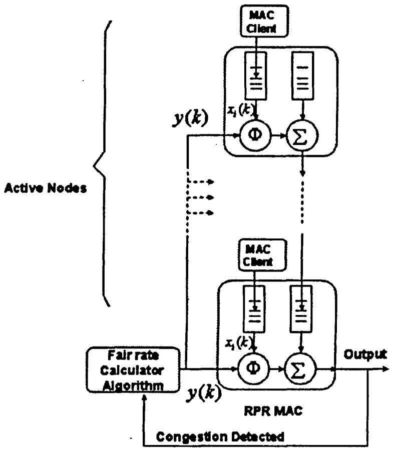


Figure 5.3: Existing RPR Fairness

$\Phi_{x_1}(x)$ is a Saturation Function and is defined as

$$\Phi_{x_1}(x) = \begin{cases} 0 & x < 0 \\ x_1 & x > x_1 \\ x & 0 \le x \le x_1 \end{cases}$$

so

$$\Phi_{y(k)}(x_i(k)) = \begin{cases} 0 & x_i(k) < 0 \\ y(k) & x_i(k) > y(k) \\ x_i(k) = r_i(k) & 0 \le x_i(k) \le y_i(k) \end{cases}$$

Which means that the RPR node S_i which attempts to add its traffic to the ring by the rate $x_i(k)$, throttles its traffic to the lower rate, $y(k)$. The nodes with traffic less than $y(k)$ won't be throttled and can send their rate $x_i(k)$ which is equal to $r_i(k)$.

5.2. Mapping RPR Fairness to a Classic Control System

In the current RPR Proposal drafts, the congested node, first decreases its rate, to prevent packet loss, then calculates the fair rate based on the information it has and advertises this rate to throttles all active nodes to the fair rate. In this approach the delay is not considered, hence in case of high delay, by the time the active node receives the fairness packet it has already sent a high number of its packets. This causes a longer time to converge and in some cases oscillation in throughput, or packet loss.

To solve this problem, in this thesis a new mechanism is proposed. In this mechanism each active node participates in the controlled loop and calculates its rate by considering its estimation of the Round Trip Time, RTT, of the fairness packet. In this approach, the congested node monitors its received traffic rate and queue level and advertises them to its active nodes. Each active node then uses these information and its estimated RTT to adjust its rate. This approach dynamically controls the rates and guarantees no packet loss by controlling the congested queue level. It speeds up the convergence and also achieves higher bandwidth utilization. To simplify the model, each node is considered as a queue serviced as FIFO, in cascade with a traffic shaper.

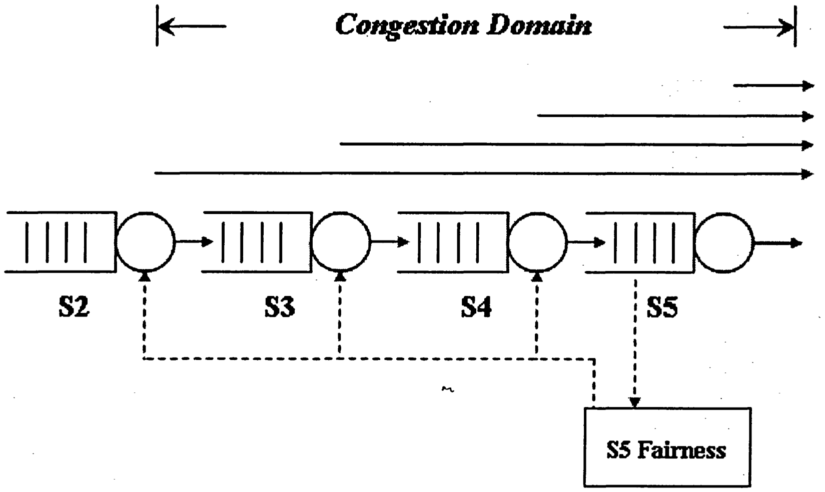


Figure 5.4: Simple RPR Queue Model

The traffic shaper is in charge of deciding how much traffic needs to be serviced and sent out to the next node. It uses a token bucket which allows burstiness while bounds it. In Figure 4.8, it is assumed that node S_5 is congested. S_5 Fairness algorithm turns on, as soon as the congestion is detected. As the active nodes are trying to send more than the fair rate, S_5 shaper will compensate the overload of the traffic of the other active nodes by decreasing its own traffic. It means S_5 always send less than the fair rate to make sure the overall traffic c is less than $c_{\max}$.

To map RPR Fairness to a control system, the similarities need to be identified. In a classic control system which is shown in Figure 5.5, there is a target system which needs to be controlled, a controller which monitors the state of the target system and compares it with the reference value.

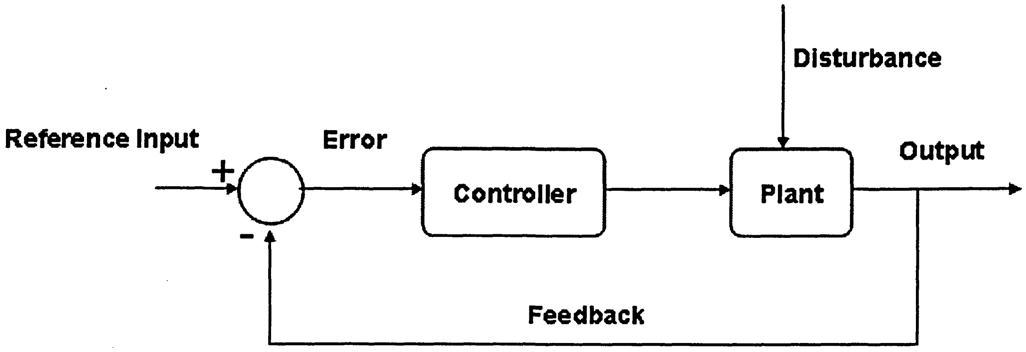
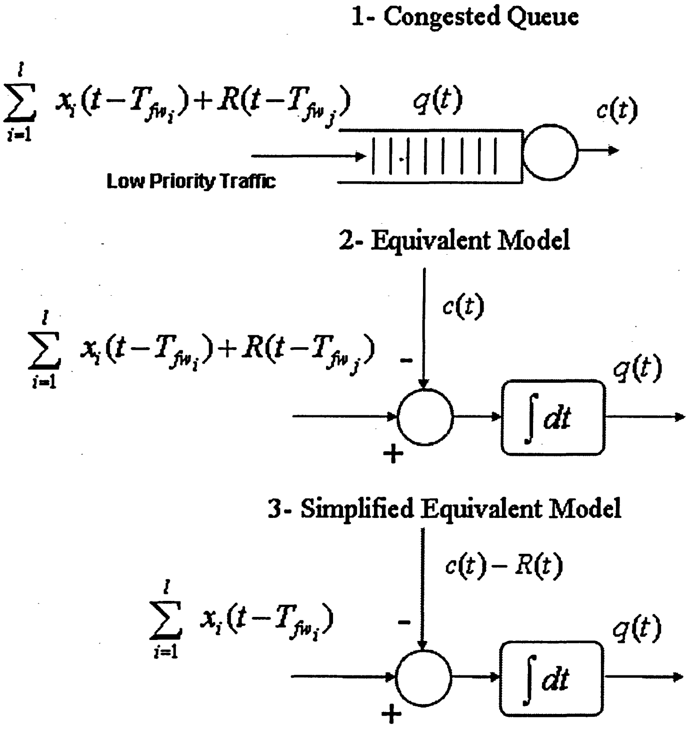


Figure 5.5: A Classic Control Model

In this thesis RPR Fairness algorithm is mapped to a classic closed loop control system. In this mapping the monitoring system is the received traffic rate and the queue level of the congested node. All the traffic is destined to the congested queue and served in a FIFO service discipline. The target system is the active node that sends traffic to the congested node. After the congestion is detected the congested node starts sending fairness packets to the active nodes.

5.3. Control Model for the Congested Node

Congested node can be simplified as a Queue and a policer that services the incoming low priority fairness eligible traffic. The high priority traffic is not affected by fairness.


1- Congested Queue

2- Equivalent Model

3- Simplified Equivalent Model

$\sum_{i=1}^{I} x_i(t-T_{fw_i})+R(t-T_{fw_j})$
$q(t)$
$c(t)$
Low Priority Traffic
$\sum_{i=1}^{I} x_i(t-T_{fw_i})+R(t-T_{fw_j})$
$c(t)$
-
+
$\int dt$
$q(t)$
$c(t)-R(t)$
$\sum_{i=1}^{I} x_i(t-T_{fw_i})$
-
+
$\int dt$
$q(t)$

Figure 5.6: Control Model for the congested node

The incoming traffic to the congested node is $\sum_{i=1}^{I} x_i(t-T_{fw_i})+R(t-T_{fw_i})$, which is the aggregated traffic of all active nodes and semi-active nodes. Figure 5.6.1 shows the equivalent of the congested node. T_{fw_i} is the forward delay that takes for data traffic to get to the congested node. $q(t)$ is the queue level of the congested nodes transit queue. $c(t)$ is the rate of serviced traffic by the congested node. It is very close to the link bandwidth. $R(t-T_{fw_j})$ is the traffic of all the semi-active nodes when they are received by active node. Since it is constant during the adaptation of the active nodes, $R(t-T_{fw_j})=R(t)$ and this can be considered as a delay independent parameter that is received by the congested node. Figure 5.6.2 can be simplified to Figure 5.6.3.

5.4. Control Model for an Active Node

As we mentioned before for conservative fairness convergence of the traffic can be mathematically shown by:

$$x_i(k) = \begin{cases} x_i(k-1) - \theta_1 x_i(k-1) & c(k) > High\_Threshold\,(RampDown) \\ x_i(k-1) + \theta_1(c_{\max} - x_i(k-1)) & c(k) < Low\_Threshold\,(RampUp) \\ x_i(k-1) \quad Low\_Threshold < c(k) < High\_Threshold\,(NoChange) & \end{cases}$$

$x_i(k)$ is active node's traffic at sampling interval k. A simple linear control equivalent of this formula is a proportional controller in cascade with an integrator.

$$x_i(k) = x_i(k-1) + k_c e(k)$$

or

$$x_i(k) - x_i(k-1) = k_c e(k)$$

or in continuous time domain

$$\frac{dx_i(t)}{dt} = k_c e(k)$$

which is shown in Figure 5.7.

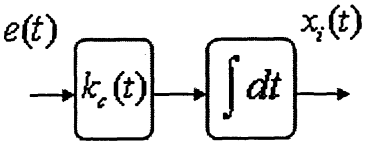


Figure 5.7: Control Model for an active node

Figure 5.7 shows the linear model for the fairness algorithm of an active node.

Chapter 6

The proposed Fairness and Congestion Control model

To guarantee that Fairness algorithm can perform a stable bandwidth allocation and no packet is lost, the received traffic rate and the queue level of the congested node need to be monitored and controlled. If the queue is empty most of the time, most likely the bandwidth is not highly utilized and if it is almost full the probability of packet loss is high. Keeping the queue level in a fixed level guarantees a high bandwidth allocation while minimizes the chance of packet loss.

RPR Fairness relies on feedback messages sent using fairness packets. When the distance is significant, the delay will play an important role in the stability of the system. The Fairness algorithm can be viewed as a delayed feedback system, where the congested node informs the active nodes about its congestion status, so that they can adjust their rate to a fair rate. In this Chapter we will map RPR to a classical control system and then analyze the effect of the delay on congested queue level and overall fairness behavior.

When Fairness is on, the simplest mechanism to control the flow of data from the active nodes to the congested node that is used in the existing RPR proposal drafts, is to command the active nodes to send data at a rate no greater than the fair rate or in other words to command them to throttle themselves to the fair rate. In steady state the output link of the congested node is highly utilized and there is no packet loss due to buffer overflow. However, when the fair rate is changed due to change of number of active nodes, the propagation delay in the link can prevent the active nodes from reacting correctly to the changes which may lead to oscillation. Even in steady state if there is an error in calculating the fair rate, the queue will either be empty or grow until it overflows.

One mechanism could be that the congested node computes a fair traffic rate. This calculated fair rate is then sent to the active nodes by fairness packet. However the link delay can prevent the active nodes from reacting quickly which generally leads to packet loss or throughput inefficiency. Furthermore, if the number of active nodes is large, the system may become more sensitive or even unstable.

It is assumed that the Fairness Feedback Round Trip Time, RTT, is constant for each active node while it adapts to the fair rate. This assumption is relatively accurate because in Wide and Metropolitan Area networks the round trip time is mostly determined by propagation delay and this delay is fixed. To take into account the jitter of round trip time due to queuing time, a model containing time varying delays could be considered. So in our analysis, during the rate adaptation time, the Fairness Round Trip Time is assumed to be constant.

6.1 Fairness Control Model

6.1.1 Analysis of the existing RPR Fairness

Fairness algorithm is responsible for monitoring and controlling its received traffic rate including its own traffic to keep it as close as its maximum service rate which is equal to the link bandwidth (c_{ref}) . It needs to monitor the received traffic rate and send it to the active nodes. Each active node then decreases its traffic rate. This mechanism involves all the active nodes in the process of fairness.

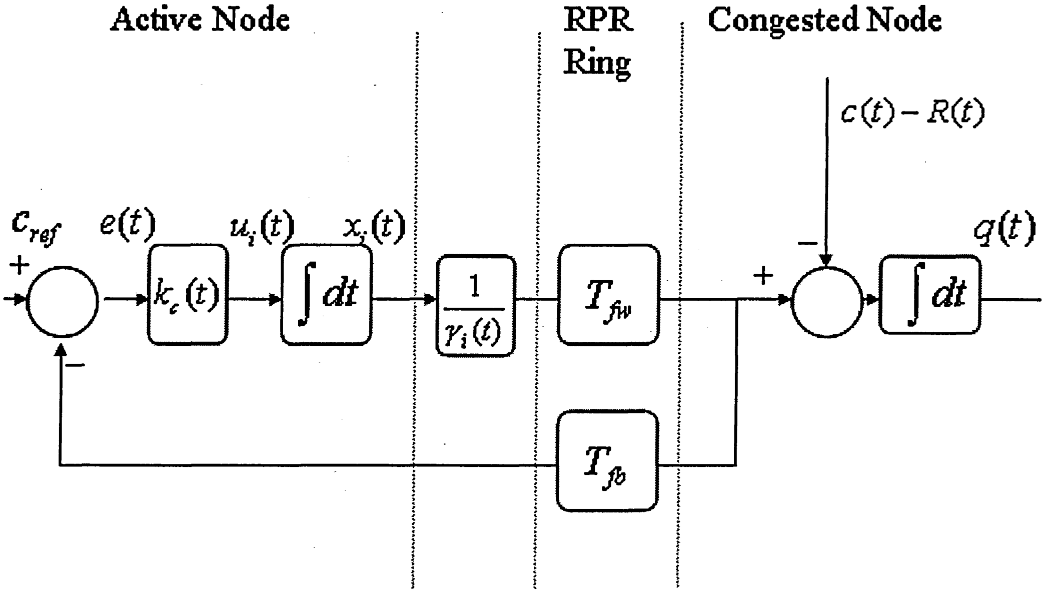


Figure 6.1: Existing RPR Fairness linear model from one of the active nodes view

Each active node has it's own view of the network based on its c_{ref}, and active weight $\gamma_i(t)$.

Major elements of this model are Congested and Active node which have been introduced in chapter 5 section 5.3 and 5.4. This model consists of the Active Node's fairness which shown simply as a proportional controller in cascade with an integrator. RPR Ring simply causes a forward delay in the path of data traffic and a feedback delay in the path of feedback information.

Congested node is considered as a queue that its overall traffic rate and its level will be monitored. k_c is the controllable parameter of the Fairness Control. It shows how fast the adaptation of traffic rate is. The bigger the value of k_c, the faster the fairness adaptation. Since each active node has limited view of the rest of active nodes, it assumes that other active nodes behave the same way as it does. So the parameter $\frac{1}{\gamma_i(t)}$ represents the rest of the network form its view. When all nodes have the same static weights, then $\frac{1}{\gamma_i(t)} = l(t)$. So simply this nodes multiply its own traffic to the number of active nodes which is constant during the adaptation to find out how much traffic needs to be sent to the congested node. The overall traffic should not be more than the available service rate of the congested node $c(t) - R(t)$.

Proposition 1: The adaptation of the rate by each active node results in fair traffic rate when forward delay is negligible.

Proof: If there are l active nodes the dynamic of the system would be

$$\frac{dq(t)}{dt} = \sum_{i=1}^{l} x_i(t - T_{fw_i}) + R(t) - c(t).$$

The congested node considers its own Transmit Rate or Add_Rate as an input traffic too. It is assumed that the number of active nodes does not change during one fairness feedback round trip time so $l(t - T_{fw}) = l(t)$ and $\gamma_i(t - T_{fw}) = \gamma_i(t) = \gamma_i$ and also $R(t - T_{fw}) = R(t)$. Replacing the aggregated traffic of active nodes by the average of active nodes traffic, times number of active nodes

$$\sum_{i=1}^{l} x_i(t - T_{fw_i}) = l(t)\bar{x}(t - T_{fw_i})$$

we have

$$\frac{dq(t)}{dt} = l(t)\bar{x}(t - T_{fw_i}) - [c(t) - R(t)]$$

or

$$\frac{dq(t)}{dt}=\frac{1}{\gamma_i(t)}\gamma_i(t)l(t)\bar{x}(t-T_{fw_i})-[c(t)-R(t)],$$

but

$$x_i(t-T_{fw_i})=\gamma_i(t)l(t)\bar{x}(t-T_{fw_i}),$$

So the dynamic of the queue can be shown as

$$\frac{dq(t)}{dt}=\frac{1}{\gamma_i(t)}x_i(t-T_{fw_i})-[c(t)-R(t)].$$

In the steady state, when the queue level is stable, then the overall traffic sent by the active nodes is equal to c_{ref} which can be the full available bandwidth or

$$\frac{1}{\gamma_i(t)}x_i(t-T_{fw_i})+R(t)=c_{ref},$$

and the traffic sent by each node can be viewed as

$$x_i(t-T_{fw_i})=\gamma_i(t)[c_{ref}-R(t)].$$

As it is shown in this equation the traffic rate sent by each node is dependent on its own forward delay. When delay is negligible the fair traffic rate is achieved.

$$x_i(t)=\gamma_i[c_{ref}-R(t)]$$

But when delay is long then the traffic rate for each active node depends on its own RTT. So the effects of RTT needs to be analyzed.

The equivalent model in Laplace domain with $x_i(s)$ can be shown as follows.

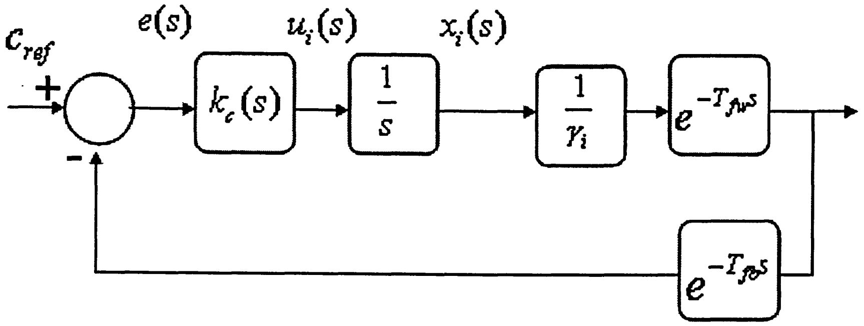


Figure 6.2: Existing RPR Fairness linear model equivalent in Laplace domain.

The open loop transfer function of the system is

$$G(s)H(s) = \frac{k_c(s)}{s} \times \frac{e^{-T_{fw}s}}{\gamma_i(s)} \times e^{-T_{fb}s},$$

or

$$G(s)H(s) = \frac{k_c(s)}{\gamma_i(s)} \times \frac{e^{-RTTs}}{s}.$$

As the active node can estimate its RTT, when $RTT_e = RTT$, the system can be simplified as

$$G(s)H(s) = \frac{k_c(s)RTT_e}{\gamma_i(s)} \times \frac{e^{-RTTs}}{RTT_e s}.$$

By replacing $s = j\omega$, $k(s) = k_c$, $\gamma_i(s) = \gamma_i$ and $RTT\omega = \theta$ then

$$G(j\omega)H(j\omega) = \frac{k_c RTT}{\gamma_i} \times \frac{e^{-j\theta}}{j\theta}.$$

6.1.2 Stability Analysis

Using the open loop transfer function, the stability of the closed loop system can be discussed using Bode-Plot.

$$|G(j\omega)H(j\omega)| = \left| \frac{k_c RTT}{\gamma_i} \times \frac{e^{-j\theta}}{j\theta} \right|$$

$$\angle G(j\omega)H(j\omega) = \angle \frac{e^{-j\theta}}{j\theta}.$$

6.1.2.1 Bode-Plot Analysis:

The goal of control system is to have an acceptable margin of safety. These margins are called stability margins. A Bode-Plot is the frequency response of the open-loop system. This figure shows the bode plot diagram of $\frac{e^{-j\omega}}{j\omega}$ which is the same for $\frac{e^{-j\theta}}{j\theta}$.

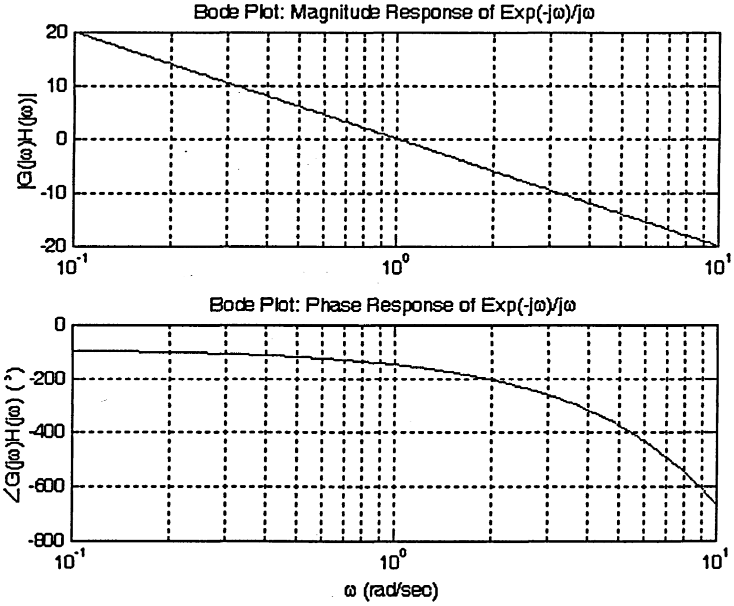


Figure 6.3: Bode-Plot $G(j\omega)H(j\omega)$

The phase margin can be interpreted as the uncertainty in the fairness feedback round trip time delay (RTT), that the system can tolerate before the active nodes rate, $x_i(t)$, becomes unstable and start oscillating. The phase margin is at the point that the gain is unity or $0\ dB$.

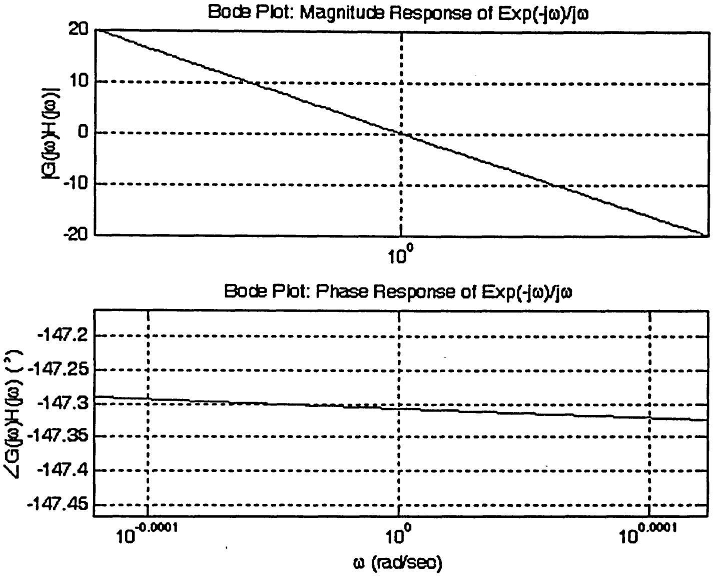


Figure 6.4: Phase Margin can be interpreted as the uncertainty in the RTT.

As it is shown in Figure 5.4 the Phase Margin is 180-147.31= 32.69 degrees. To maintain stability for any delay, it is necessary to make sure that the phase margin is independent of delay and always remains positive.

The gain margin can be viewed mostly as the uncertainty in the load level that the system can tolerate. This is represented by RTT and $\gamma_i(t)$. The gain margin of the system is at the point that the phase response is -180°. It is defined as the change in open loop gain, required to make the closed loop system unstable.

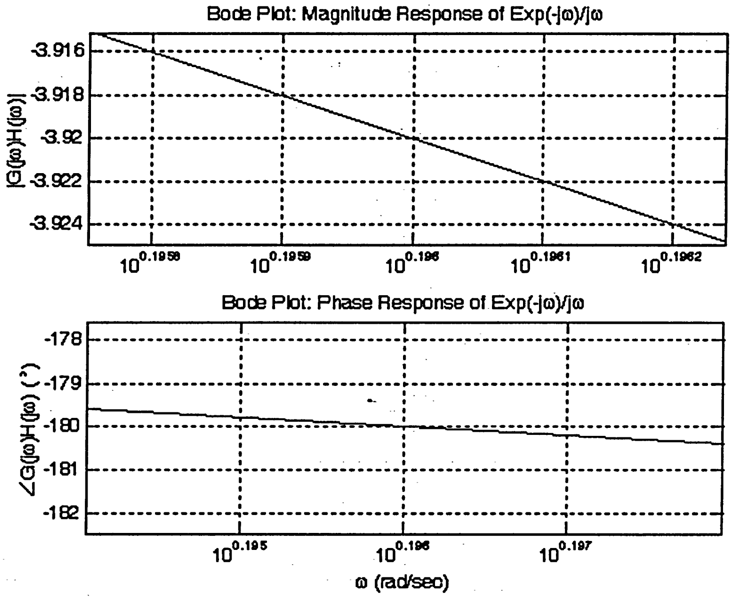


Figure 6.5: Gain Margin can be interpreted as uncertainty in the load level, which is caused by RTT and $\gamma_i(t)$.

Remarks 1: As this Figure shows the Gain Margin is -3.92 dB. For the closed loop system to be unstable, the gain margin should be unity ($0 \ \ dB$)

$$20\log(\frac{k_c RTT}{\gamma_i} \times \frac{e^{-j\theta}}{j\theta}) = 0$$

and

$$20\log\frac{k_c RTT}{\gamma_i} + 20\log\frac{e^{-j\theta}}{j\theta} = 0,$$

but the gain margin of $\frac{e^{-j\theta}}{j\theta}$ is $20\log\frac{e^{-j\theta}}{j\theta} = -3.92$ so

$$20\log\frac{k_c RTT}{\gamma_i} = +3.92$$

and

$$\frac{k_c RTT}{\gamma_i} = 10^{\frac{3.92}{20}} = 1.57,$$

then the stability criteria is

$$\frac{k_c RTT}{\gamma_i} < 1.57.$$

As the only controllable parameter is k_c, so

$$k_c < 1.57 \times \frac{\gamma_i}{RTT}.$$

For large RTT and large number of active nodes, k_c needs to be very small to keep the system in the stable status, but very small k_c means very slow response to Fairness message and rate change which is not desirable, so we need to provide a solution that compensates the effect of large RTT.

6.1.3 Fairness Control Model using Smith predictor

The Smith Predictor attempts to remove the effect of the Fairness Feedback RTT delay from the closed loop system, so that the controller can be designed as if there was no time delay present. The delay-free model is used to generate the output which existed if the RTT delay were absent. This signal is used in the feedback loop instead of the output. Smith predictor uses the estimated RTT to compensate the real round trip time delay. Each node estimates its RTT by estimating the time difference between sending a packet and receiving the response.

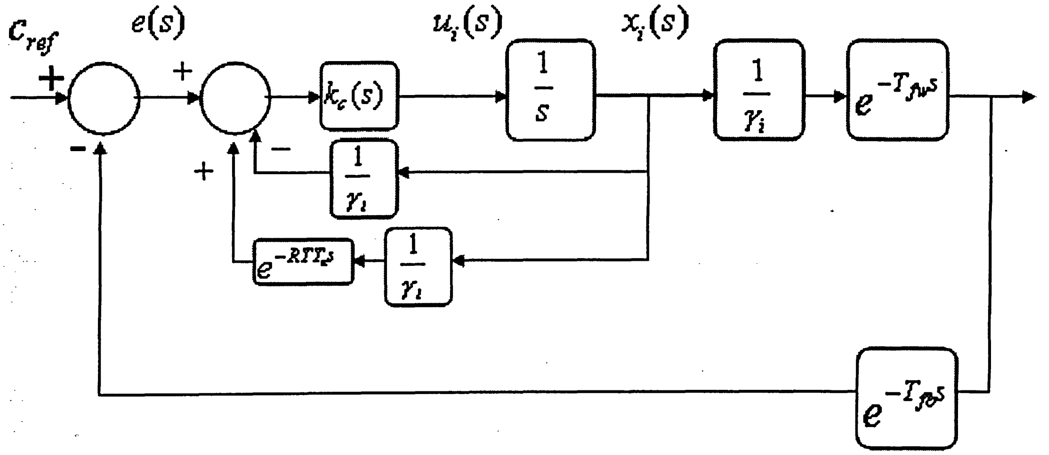


Figure 6.6: Fairness Control Model with Smith Predictor

The instability of the control system based on the effective output rate includes all the packets passing through the congested node, as well as the packets that are traveling from the active node

PROPERTY OF
RYERSON UNIVERSITY LIBRARY

to the congested node. This traffic is called "in flight" traffic. By our proposed approach, instead of stabilizing the actual output rate, the controller can stabilize the effective output rate at a reference level.

Proposition 2: Smith Predictor compensates the effect of round trip time delay from Fairness Control Model feedback loop.

Proof: The transfer function of the Smith Predictor controller is

$$\frac{x_i(s)}{e(s)} = \frac{k_c(s) \times \frac{1}{s}}{1 + \frac{k_c(s)}{s}(\frac{1 - e^{-RTT_e s}}{\gamma_i})},$$

and after some calculation [Appendix A.2], the closed loop transfer function yields as

$$\frac{\frac{k_c(s)}{\gamma_i s}}{1 + \frac{k_c(s)}{\gamma_i s}} e^{-T_{fw} s}.$$

Figure 6.7 shows how Smith Predictor compensates the delay effect.

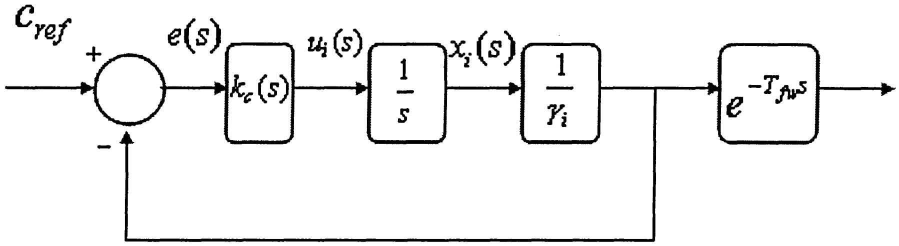


Figure 6.7: Equivalent Model with Smith Predictor in Laplace space

This model consists of a first order lag $k(s)$ and a pure delay. This completes the proof.

Remarks 2: The frequency response of the first order lag of Fairness Control equivalent model, $k(j\omega)$ can be shown as

$$k(j\omega) = \frac{1}{T_c\omega + 1},$$

where T_c is the time constant of the Queue Congestion Control and $T_c = \frac{\gamma_i}{k_c}$. The bode plot of $k(j\omega)$ shows that when $T_c\omega << 1$ or $\omega << \frac{k_c}{\gamma_i}$ the gain is unity and the phase is zero. k_c can be selected large enough for fast response to the rate change.

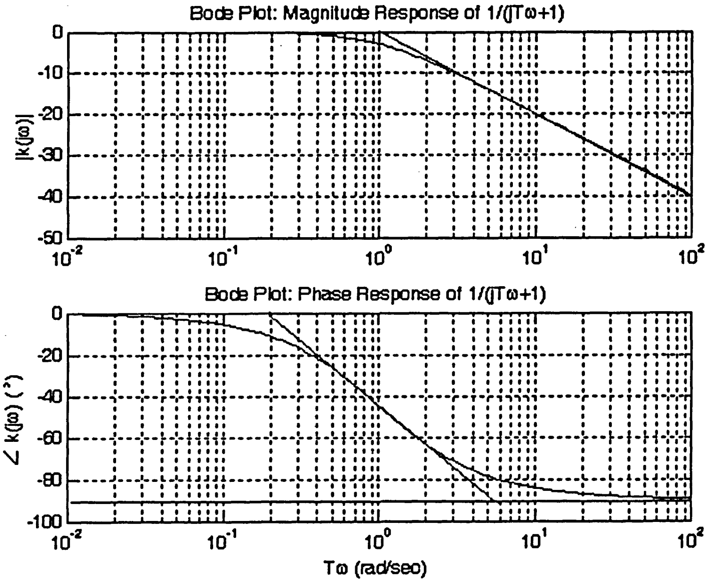


Figure 6.8: When $\omega << \frac{k_c}{\gamma_i}$ the gain is unity and the phase is zero.

Remarks 3: Another variable factor in the equivalent model is γ_i. To make sure the response is the same for any number of nodes, k_c can be adjusted as a function of γ_i, or $k_c = k_{cc} \times \gamma_i$, this way the closed loop pole would be constant $s = \frac{k_c}{\gamma_i} = k_{cc}$.

6.2 Fairness and Congestion Control Model

To create the model the focus needs to be on the Secondary Transit Queue (STQ) and a closed loop control based on the occupancy level of the output queue of the congested node S_i. STQ is used to buffer the fairness eligible traffic in an RPR node. When congestion is detected and the fairness turns on, the Fairness algorithm tries to control the level of the queue for the congested node S_i to a set point Q_{ref} by decreasing the traffic sent by the active nodes. Queue Congestion Control can be combined with Fairness Control in the Fairness algorithm.

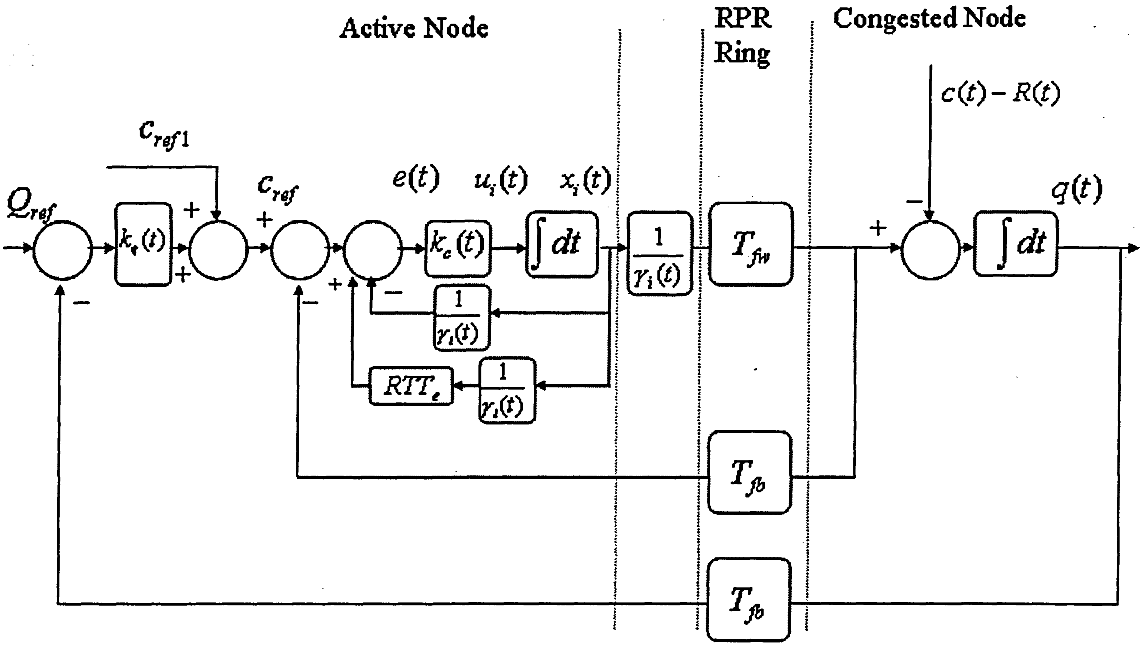


Figure 6.9: Fairness and Congestion Control model with Smith Predictor

As can be seen from the figure, to control the queue level to a reference point Q_{ref} ,the error is calculated in the active node, then it is multiplied by $k_q(t)$ and added to the fairness control reference rate c_{ref1} to make c_{ref}. This means that when the queue level is low the system increases the fairness reference rate to increase the queue level while achieving a higher bandwidth utilization and when the queue level is too high, the bandwidth utilization would be decreased to get the queue level back to the reference level. This mechanism guarantees queue congestion control and the highest bandwidth utilization possible. This model with Smith predictor shows the best stability even with large number of active nodes and long delay.

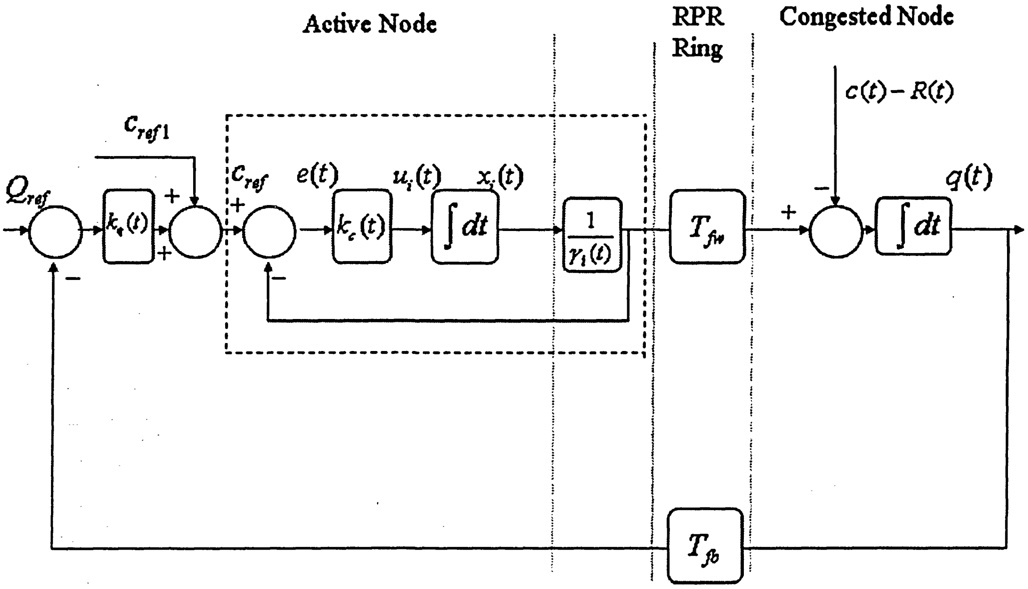


Figure 6.10: Equivalent of Fairness Control Model with Smith Predictor along with Queue Congestion Control

The model for each node in the Laplace domain would be simplified as follows.

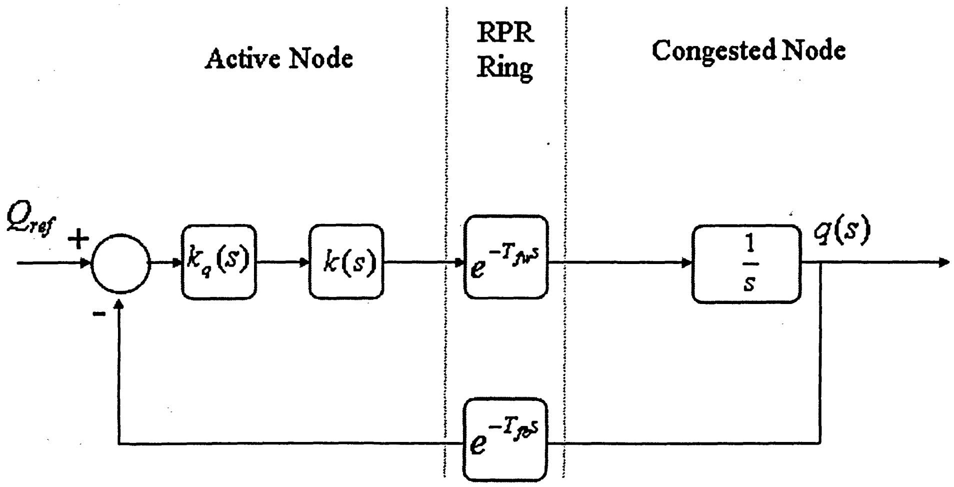


Figure 6.11: Queue Model from one of the active nodes view

where $k(s) = \dfrac{\frac{k_c}{\gamma_i}}{s + \frac{k_c}{\gamma_i}}$.

6.2.1 Stability Analysis of the system with Proportional Controller

The open loop transfer function for the simplified system for Proportional controller $k_q(s) = k_q$ is

$$G(s)H(s) = k(s) \times k_q \frac{e^{-RTTs}}{s}.$$

It is desirable that Fairness Control doesn't interfere with Queue Congestion Control.

Proposition 3: There are values of controllable parameters k_c and k_q, that Fairness Control Model has minimum impact on Queue Congestion Control.

Proof: As it is mentioned in section 5.1.2, k_c can be selected large enough so that in normal situation $\omega << \frac{k_c}{\gamma_i}$ and the gain is unity and the phase is zero or $k(j\omega) = 1^{\angle 0^o}$. For the Queue Congestion Control model $\omega = \frac{1}{T_q} = k_q$, where T_q is Queue Congestion Control time constant. So the criteria would be

$$k_q << \frac{k_c}{\gamma_i}.$$

In practice a ratio of about $\frac{1}{5}$ is good enough so

$$k_q < \frac{1}{5} \times \frac{k_c}{\gamma_i}.$$

Then the simplified transfer function when active node estimate its round trip time, would be

$$G(s)H(s)=\frac{q(s)}{e(s)}=k_q RTT_e\times\frac{e^{-RTTs}}{RTT_e s}.$$

By replacing $s=j\omega$, $RTT_e=RTT$ and $RTT\omega=\theta$ the open loop would be

$$G(j\omega)H(j\omega)=k_q RTT\times\frac{e^{-j\theta}}{j\theta}.$$

Using the open loop transfer function, the stability of the closed loop system can be discussed using Bode-Plot or Nyquist.

$$|G(j\omega)H(j\omega)|=\left|k_q RTT\times\frac{e^{-j\theta}}{j\theta}\right|$$

$$\angle G(j\omega)H(j\omega)=\angle\frac{e^{-j\theta}}{j\theta}.$$

6.2.1.1 Nyquist Analysis:

The simplified equation makes all the plots for different RTT s fall on one single curve. Note that the curve of $\frac{e^{-j\theta}}{j\theta}$ is the same as $\frac{e^{-j\omega}}{j\omega}$.

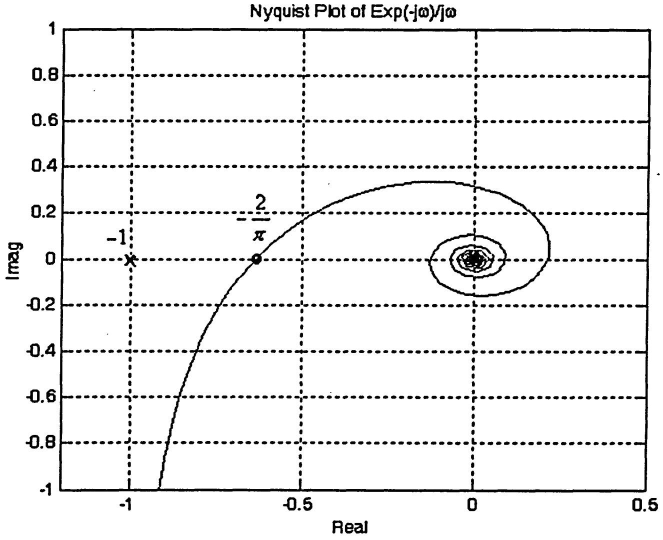


Figure 6.12: Nyquist Plot of $\frac{e^{-j\theta}}{j\theta}$ when $k_q RTT=1$

Proposition 4: To guarantee the stability of Queue Congestion Control Model it is necessary that

$$k_q < \frac{\pi}{2} \times \frac{1}{RTT}.$$

Proof: In Figure 5.12 the curve crosses the real axis at $-\frac{2}{\pi}$ when $k_q RTT = 1$, which is on the right side of the critical point, -1. So the closed loop system is stable for all RTTs that meet the following stability criteria

$$k_q RTT < \frac{\pi}{2}.$$

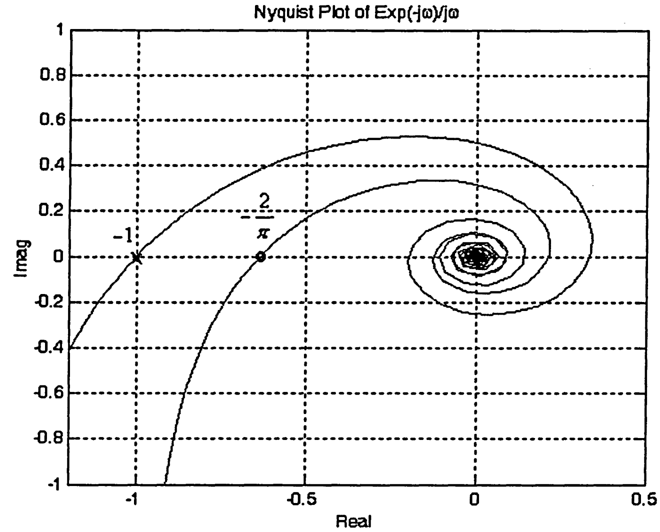


Figure 6.13: Nyquist Plot encircles the critical point -1, when $k_q RTT = \frac{\pi}{2}$

To keep the system in the stable area the active node needs to follow the stability criteria,

$$k_q < \frac{\pi}{2} \times \frac{1}{RTT}.$$

Remarks 4: For large RTT, k_q needs to be very small. Small k_q means that the system reaction to the queue is not too fast, which is desirable. So a relatively big k_c and a small k_q can guarantee a stable system although too big k_c makes the system too sensitive and too small k_q makes the system too slow in controlling the queue level.

So the estimated RTT can be used to adapt k_q for each active node. About one third is a good estimate, while it is at least five time smaller than k_c.

$$k_q = \frac{1}{4} \times \frac{\pi}{2} \times \frac{1}{RTT} \approx \frac{0.4}{RTT} \text{ while } k_q < \frac{1}{5} \times \frac{k_c}{\gamma_i},$$

or

$$k_q = Min\left\{\frac{0.4}{RTT}, \frac{k_c}{5\gamma_i}\right\}$$

which means k_q must be less or equal to the smaller one.

6.2.2 Steady State Analysis

Proposition 5: Fairness and Congestion Control algorithm guarantees the redistribution of the unutilized bandwidth.

Proof: In steady state, when $l(t)$, $M(t)$ and $R(t)$ are stable and stay at constant l_s, M_s and R_s respectively then the queue level $q(t)$ would be equal to Q_{ref}. If u_{iqs} and $\bar{x}_s$ are the steady state values then

$$\bar{x}_s = \frac{c_{ref} - R_s}{l_s},$$

and for $c = c_{ref} = c_{ref1} + u_{iqs}$

$$\bar{x}_s = \frac{c - R_s}{l_s},$$

where in ideal situation the steady state of u_{iq_s} is zero. If all the nodes involved in congestion, needed to be adapted, the steady state traffic rate sent from each node would be

$$\bar{x}_1 = \frac{c}{M_s},$$

but because some of the involved nodes have less traffic to send and only l_s nodes are adapted, $M_s - l_s$ nodes allocate less than the fair share. The difference traffic needs to be distributed to the adapted active nodes.

$$\bar{x}_s - \bar{x}_1 = \frac{c - R_s}{l_s} - \frac{c}{M_s},$$

$$\bar{x}_s - \bar{x}_1 = \frac{(M_s - l_s)\frac{c}{M_s} - R_s}{l_s},$$

and

$$\bar{x}_s = \frac{c}{M_s} + \frac{(M_s - l_s)\frac{c}{M_s} - R_s}{l_s}.$$

Also it is known that

$$(x_i)_s = \gamma_i l_s \bar{x}_s,$$

then

$$(x_i)_s = \gamma_i l_s \left(\frac{c}{M_s} + \frac{(M_s - l_s)\frac{c}{M_s} - R_s}{l_s} \right).$$

This equation shows that the unutilized bandwidth $\frac{(M_s - l_s)c}{M_s} - R_s$ is distributed among the active nodes that are adapted. Each node takes its share based on its dynamic weight.

Remarks 5: When all the involved nodes are active nodes, then $M_s = l_s$ and $R_s = 0$ so

$$(x_i)_s = \gamma_i c,$$

which is what we expect. This equation shows that the traffic added by each active node would be adapted to a fair rate, which is proportional to its dynamic weight. In case of equal static weights,

$$(x_i)_s = \frac{c}{l_s}.$$

Chapter 7
Simulations

The simulator of MATLAB, Simulink [30] is used to simulate the RPR Fairness and Congestion Control model. As in this thesis the emphasis is on the control theoretic approach to RPR, Simulink is a valuable tool to show the behavior of the system just by simulating the system with standard control theory components.

As it is mentioned in delay definition, the propagation delay is the major source of delay in Metropolitan and Wide Area Networks. As the speed of data transfer in fiber optics media is $0.66c$ or almost $200000km/s$, every kilometer of fiber represents $5\mu\sec$ forward delay and $10\mu\sec$ round trip time delay. Also the links are assumed to be $OC-12$ ($622Mbps$) and the queue size is assumed to be $300kbyte$ ($2.4Mbit$).

In the simulations in this chapter, it is assumed that the system is a linear system so the rates are not cut off at the maximum rate of the link. This may not be the same as other simulations methods but it is close enough. The nonlinearity is left as future work.

7.1 Existing RPR Fairness linear model

In this section, the focus is on the effect of the link delay and number of active nodes on the fairness. It is assumed that c_{ref} is a step function starting at $t = 0$. Also $c(t)$ is very close to the full bandwidth, $c_{\max} = 622Mbps$ for example 99%. The queue size is assumed to be $300kbyte$ $(2.4Mbit)$.

7.1.1 Scenario 1.1

In this scenario it is assumed that at time zero there are two active nodes involved in the congestion that try to send a full traffic through the congested node. Each one millisecond the number of active nodes doubles. Also it is assumed that the distance between the active node and the congested node is $4\,km$ so $T_{fw1} = T_{fb1} = 20\,\mu\sec$ (RTT_1=40 $\mu\sec$). As it is mentioned, each node creates a view of the network based on its own traffic and dynamic weight. This way each node can adjust its own traffic.

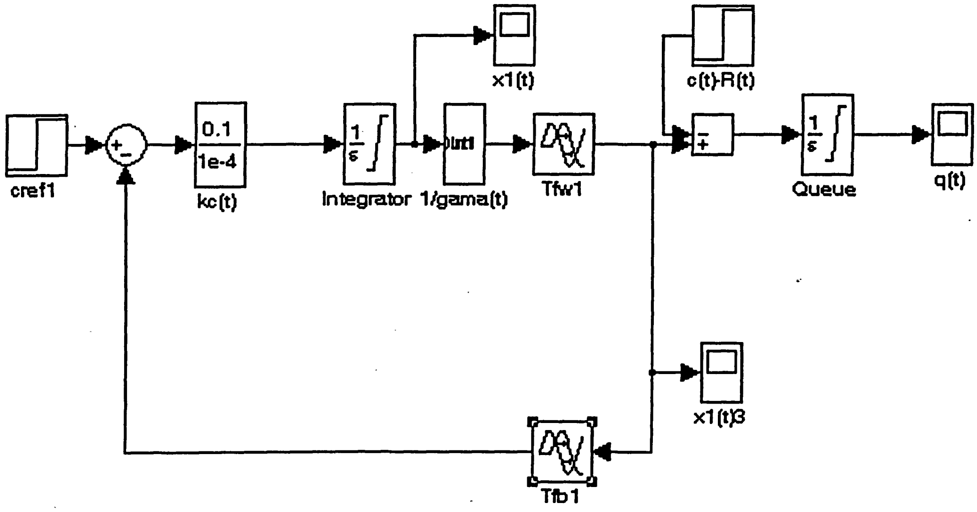


Figure 7.1: Fairness Control Model viewed by each active node

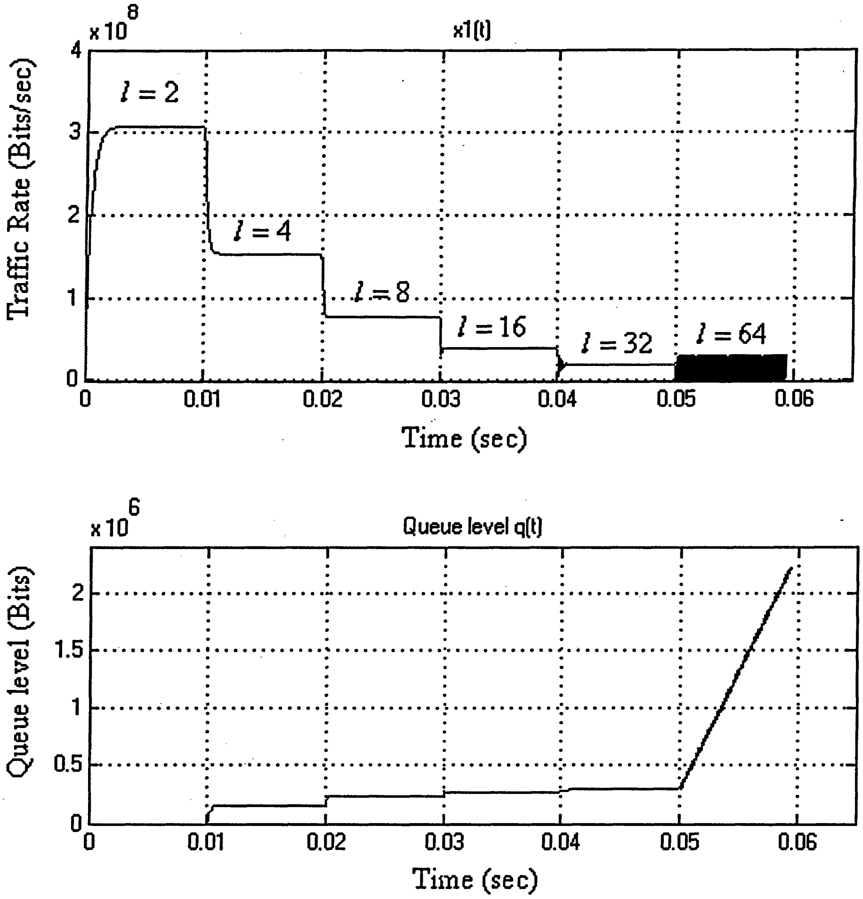


Figure 7.2: $x_1(t)$ and Queue level for Fairness Control Model in case of large number of active nodes and long delay become unstable.

Figure 7.2 shows the simulation result when number of active nodes doubles every 10 ms. For small number of active nodes the system is stable, but when number of active nodes increases the system becomes unstable. In this scenario for $l = 32$ active nodes, the throughput starts oscillating and there is no packet loss but when number of active nodes increases to $l = 64$ then the queue overflows and leads to packet loss. When delay is higher, the system becomes unstable with less active nodes. The overall throughput and fairness will suffer by throughput instability. One way to bring the node to stable status is to decrease k_c but small k_c means slow response. Smith Predictor will compensate the affect of big RTT and γ_i.

7.2 Fairness Control model using Smith predictor

7.2.1 Scenario 2.1

In this scenario it is assumed that the distance between the active node and the congested node is $400\,km$ so $T_{fw1} = T_{fb1} = 2\,m\sec$ ($RTT_1 = 4\,m\sec$).

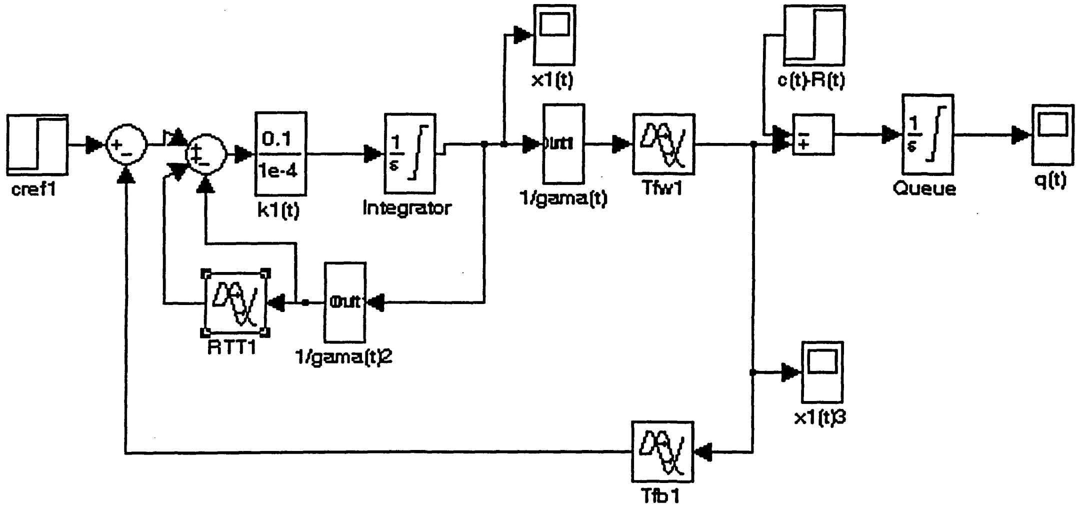


Figure 7.3: Fairness Control Model with Smith Predictor

Smith Predictor, predicts the output rate as if there was no delay. Based on this prediction each node adapts its rate.

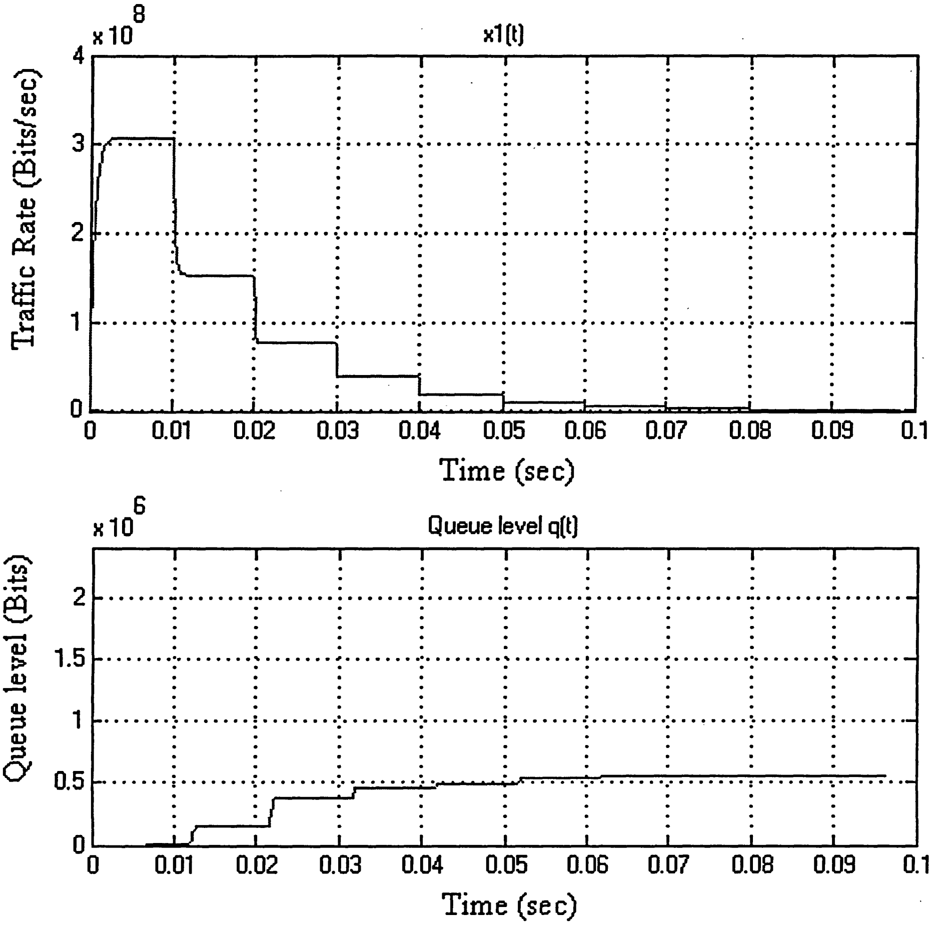


Figure 7.4: $x_1(t)$ is adjusted to the fair rate but Queue level is not controlled in the system with just Fairness Control with Smith Predictor.

As it is shown in Figure 7.4, even with 100 times more delay comparing to Scenario 2.1, the system is totally stable. With Smith Predictor the system will behave as if there was no delay in the loop. The only delay that is seen is T_{fw} which is out of the feedback loop. This delay can be seen in the queue level figure. It is a pure delay and doesn't affect the stability of the system. Although the fairness is achieved, the queue level is not under control. Any rapid change in the throughput may cause some increase in the queue level.

7.3 Fairness and Congestion Control Model

7.3.1 Scenario 3.1

In this scenario it is assumed that the distance between the active node and the congested node is $2000\,km$ so $T_{fw1} = T_{fb1} = 10\,m\sec$ (RTT_1=$20\,m\sec$). This delay causes the system without Smith predictor to be totally unstable but with Smith predictor the system is stable. It is assumed that the reference queue length Q_{ref} is one sixth of the queue size. The controllable parameters are $k_c = 1000$ and $k_q = \frac{0.4}{RTT} = 20$.

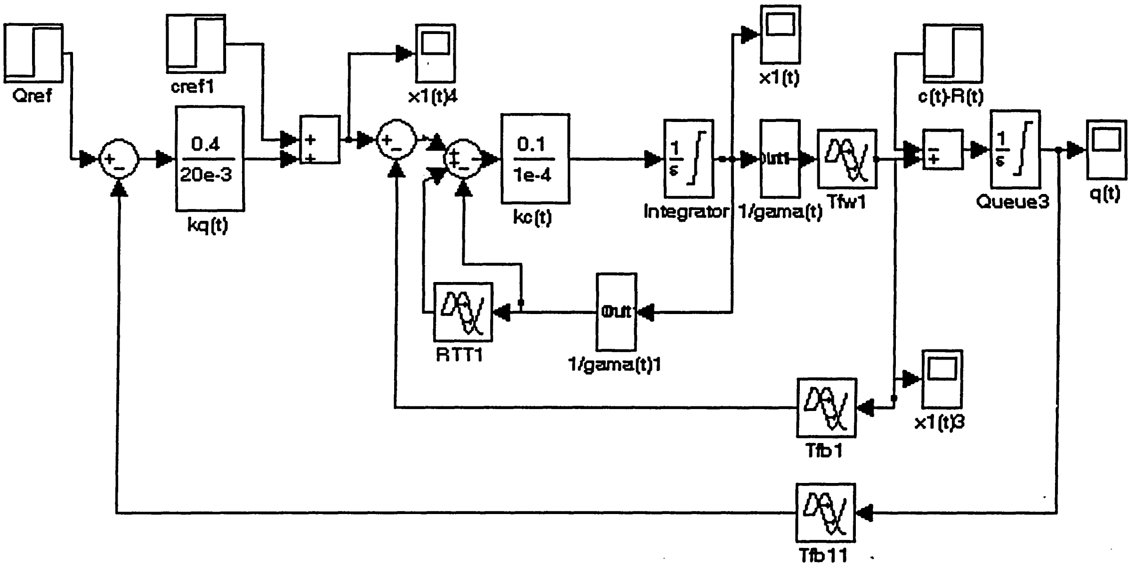


Figure 7.5: Queue Congestion Control and Fairness Control model with Smith Predictor

Figure 6.5 shows the model for this scenario. This scenario shows how queue level is controlled by adjusting c_{ref} for the Fairness Control. This means that if the queue level is too high the system decreases the overall throughput to control the queue level.

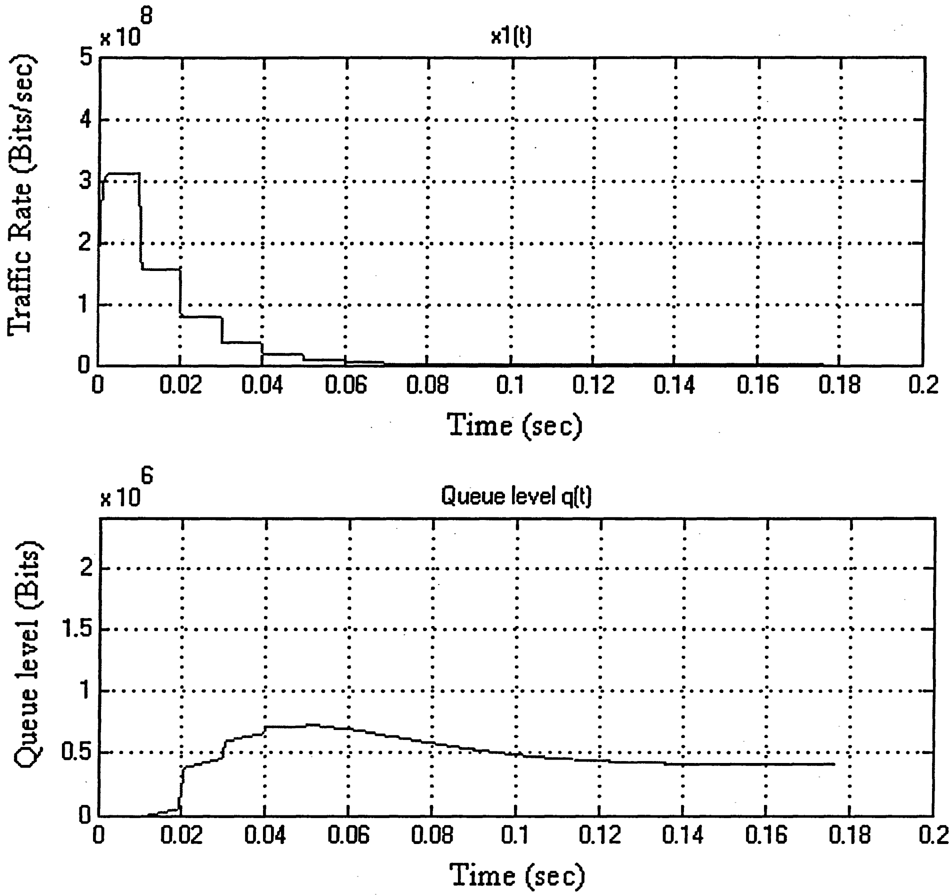


Figure 7.6: The Combined Fairness and Congestion Control with smith predictor can achieve fairness and queue control even with large delay.

As it is shown in Figure 7.6, this system can achieve a fast fairness while keeping the queue level at one sixth of the queue size. When the queue is more than the reference level, each node sends less traffic so that the overall traffic decreases and queue level goes to the desired level. Each node adapt its controllable parameters dynamically.

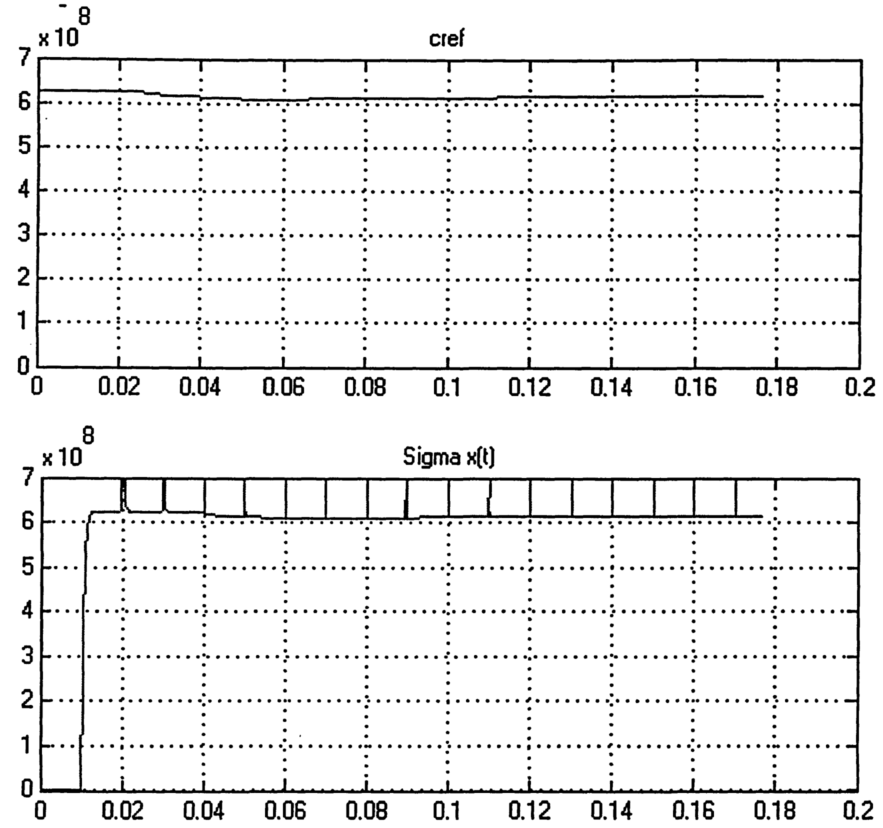


Figure 7.7: The reference rate for Fairness Control system is adjusted by Queue Congestion Control.

As it is shown in Figure 7.7, the reference rate for Fairness Control system is adjusted by Queue Congestion Control mechanism. Sigma $x(t)$ is the aggregate traffic sent by all the active nodes to the congested nodes queue. This rate is controlled by the Fairness Control system to the reference rate. If the queue is too high and the packet loss is imminent, the Queue Congestion Control mechanism decreases the reference rate of the Fairness Control.

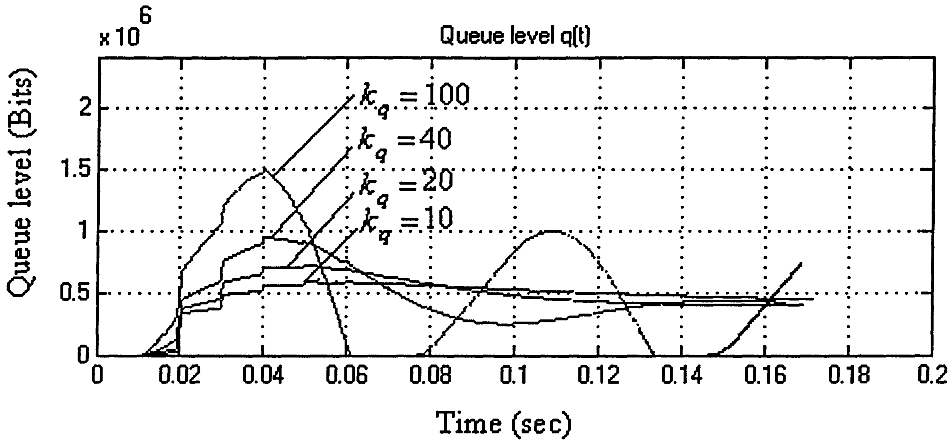


Figure 7.8: Effect of different k_q on the Queue level when RTT and k_c are constant

In Figure 7.8, the effect of changes in Queue Congestion Control parameter, k_q on the queue level is shown. When k_q is increased the system becomes more unstable. To keep the system in the stability area, k_q must meet the stability criteria.

$$k_q = Min\left\{\frac{0.4}{RTT}, \frac{k_c}{5\gamma_i}\right\}$$

which in this case,

$$k_q \leq 20 .$$

As it is shown in this figure for k_q =100, the queue level becomes unstable and oscillates. For k_q =10 the system is stable but too slow. The optimum queue control happens when k_q =20.

7.3.2 Scenario 3.2

In this scenario there are 16 nodes with no delay $T_{fwl} = T_{fbl} = 0\,m\text{sec}$ ($RTT_1 = 0\,m\text{sec}$) between the nodes. The fair rate for each of the 16 nodes would be $38\,Mbps$. $k_c = 1000$.

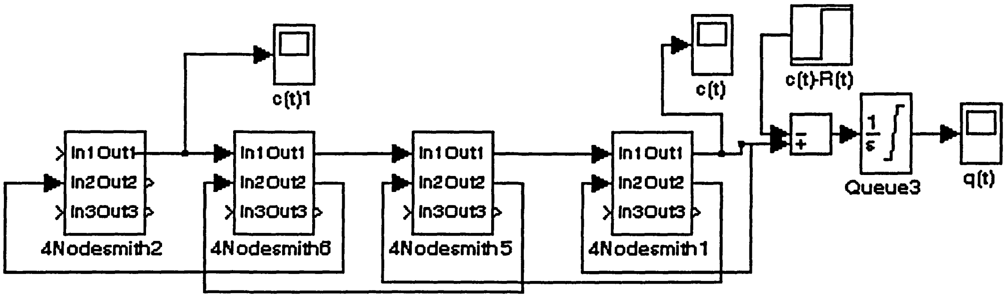


Figure 7.9: 16 active nodes with no delay

Each block in this figure consists of four nodes.

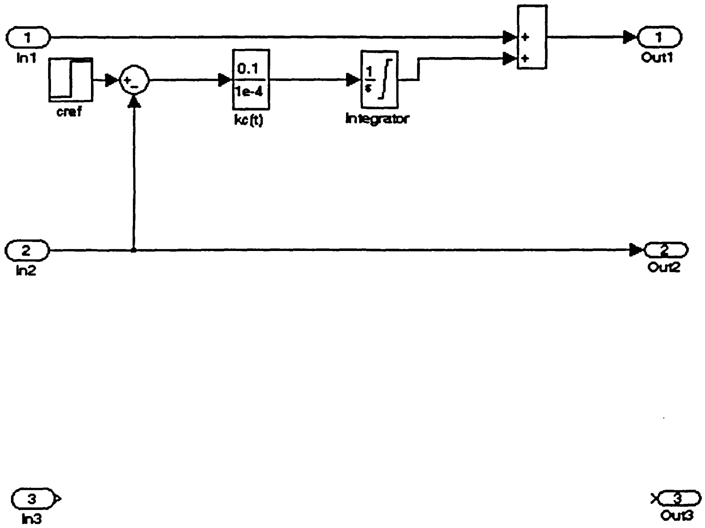


Figure 7.10: A simple node model with Fairness Control mechanism

The model shown in Figure 7.10, is very similar to the existing RPR Fairness. A smaller k_c may represent Conservative mode and very high k_c may represent Aggressive mode.

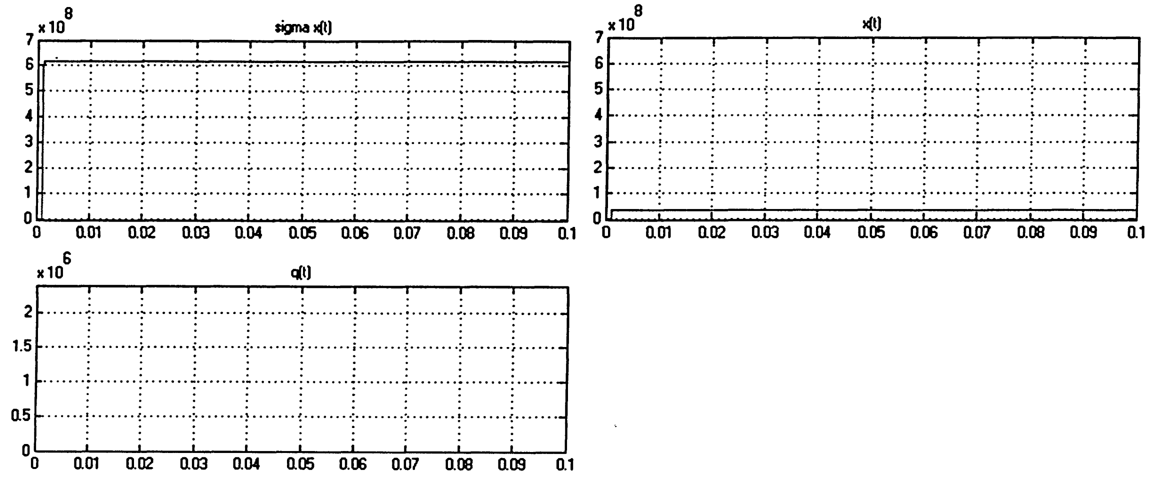


Figure 7.11: Traffic rate $x_1(t)$ of an active node, Overall traffic (Sigma $x(t)$) and Queue dynamic of the congested node in a system with no delay shows that fairness can be achieved.

Figure 7.11 shows when there is no delay fairness is achieved while queue is empty. This is an ideal situation.

7.3.3 Scenario 3.3

In this scenario there are 16 nodes and the distance between the nodes is $0.2\,km$ so for the closest node to the congested node $T_{fw1} = T_{fb1} = 1\,\mu\text{sec}$ (RTT_1=2 μsec) between each node. In this scenario the delay for the farthest node is $RTT_{16} = 16 \times RTT_1$=16×2=32 μsec. The fair rate for each of the 16 nodes would be $38\,Mbps$. The controllable parameter is $k_c = 1000$.

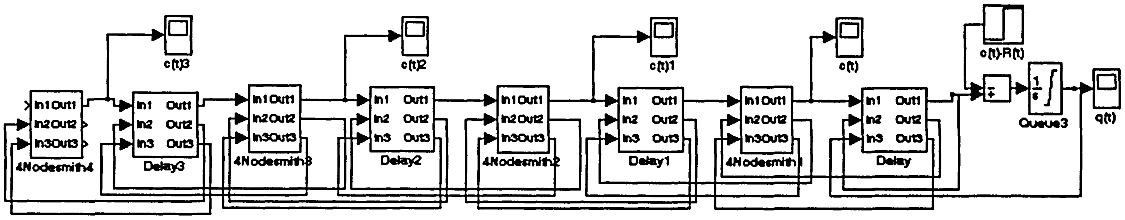


Figure 7.12: 16 active nodes with a fix delay between them

Each block in Figure 7.12 consists of four nodes with the same delay between them.

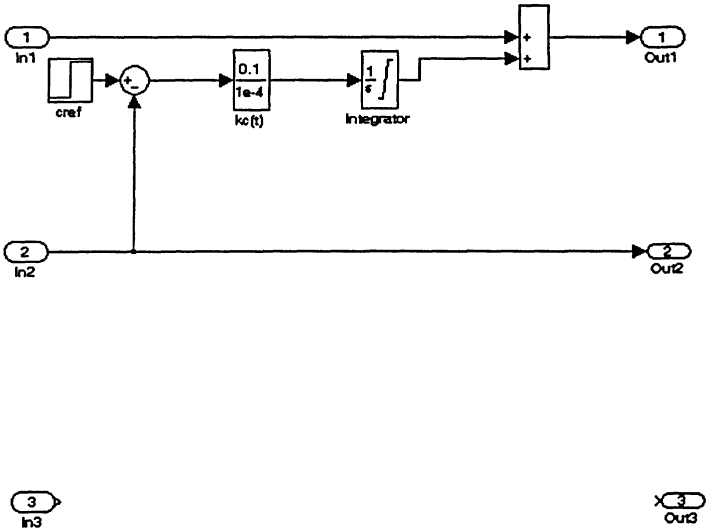


Figure 7.13: A simple node model with Fairness Control mechanism

This model shown in Figure 7.13 is very similar to RPR Conservative mode.

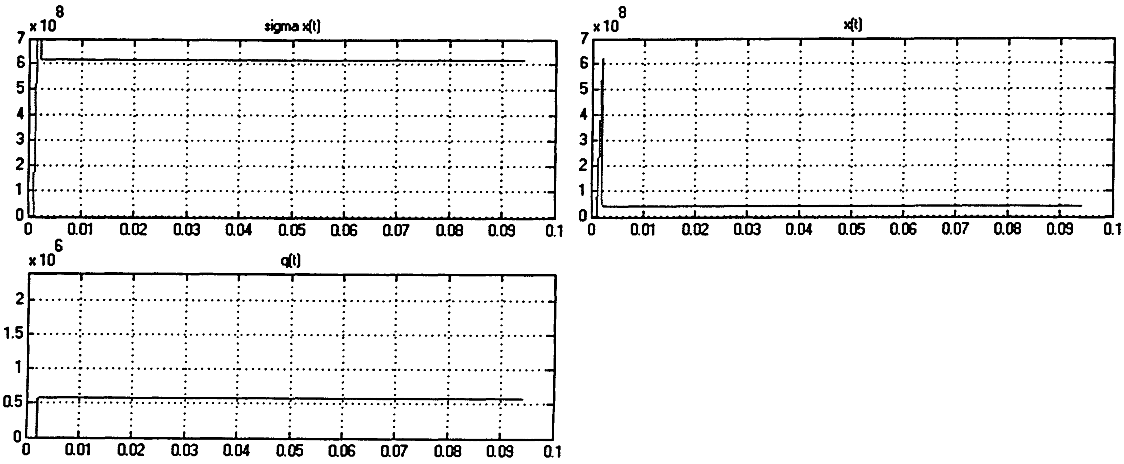


Figure 7.14: Fairness in a system with short delay can successfully be achieved but queue level increases rapidly.

Figure 7.14 shows traffic rate, $x_1(t)$ of an active node, Overall traffic (Sigma $x(t)$) and Queue dynamic of the congested node is shown. Short delay doesn't affect the throughput much but the queue level increases rapidly.

7.3.4 Scenario 3.4

In this scenario there are 16 nodes and the distance between the nodes is $1.5\,km$ so for the closest node to the congested node $T_{fw1} = T_{fb1} = 7.5\,\mu\sec$ (RTT_1=15 $\mu\sec$). The delay for the farthest node is $RTT_{16} = 16 \times RTT_1$=16× 15=240 $\mu\sec$. The fair rate for each of the 16 nodes would be $38\,Mbps$. The controllable parameter is $k_c = 1000$.

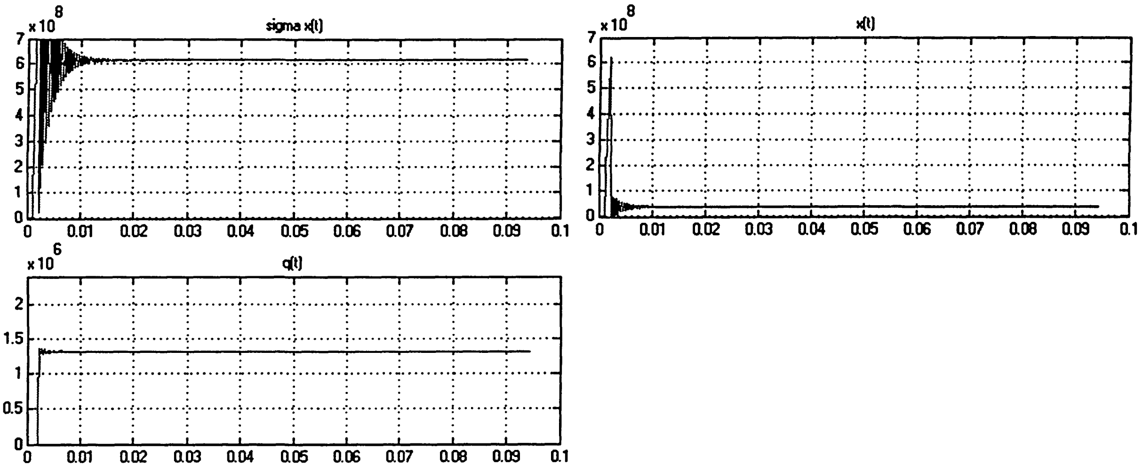


Figure 7.15: Fairness in a system with relatively longer delay is achieved but with some throughput oscillations and queue level increases rapidly.

As it is shown in Figure 7.15, longer delay causes the throughput to oscillate but fairness still converges the throughput to the fair rate. The queue level increases rapidly.

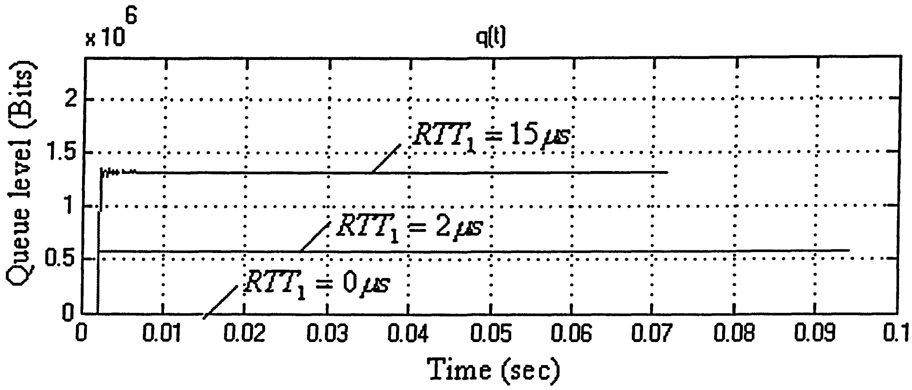


Figure 7.16: Effect of RTT_1 on the STQ Queue level

As it is shown in Figure 7.16, when delay increases the queue level is increased with no control over it. When delay is zero, the queue level is zero and the system gives service to the received packages promptly. When delay increases the queue level increases for RTT_1=2 μsec, the queue level increases to almost 600Kbit and for RTT_1=15 μsec, the queue level is almost 1.3 Mbit. For a high delay this system becomes unstable and packet may be lost. Next scenario shows such a situation.

7.3.5 Scenario 3.5

In this scenario there are 16 nodes and the distance between the nodes is $20\,km$ so $T_{fw1} = T_{fb1} = 100\,\mu\sec$ (RTT_1=200 $\mu\sec$) between each node. In this scenario the delay for the farthest node is $RTT_{16} = 16 \times RTT_1$=16×200=3200 $\mu\sec$. The fair rate for each of the 16 nodes would be $38\,Mbps$. The controllable parameter is $k_c = 1000$.

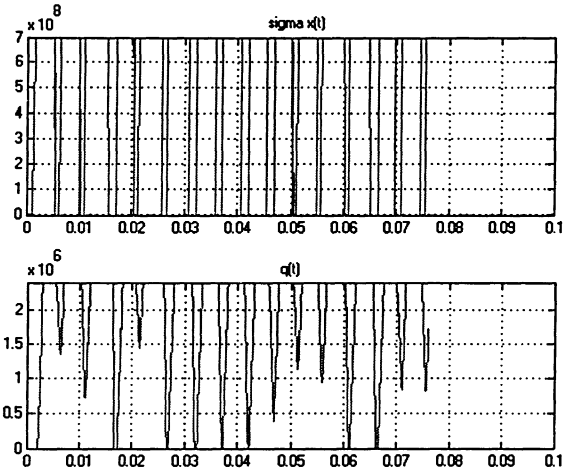


Figure 7.17: Fairness in a system with long delay cannot successfully be achieved.

As it is shown in Figure 7.17, long delay causes the throughput sent by each node to oscillate. Because of the nonlinearity of the system it is shown as periods of sending full traffic and periods of shutting down. The queue level is oscillating and packets are lost.

7.3.6 Scenario 3.6

In this scenario, it is assumed that there are 16 nodes with smith predictor and the distance between the nodes is $20\,km$ so for the closest node to the congested node $T_{fw1} = T_{fb1} = 100\,\mu\sec$ (RTT_1=200 $\mu\sec$) between the nodes. The controllable parameter is $k_c = 1000$.

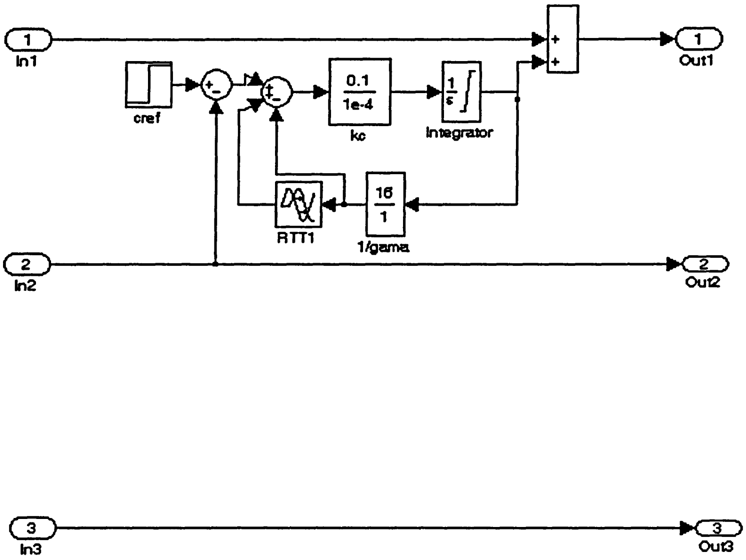


Figure 7.18: A simple node with Fairness control model and Smith Predictor

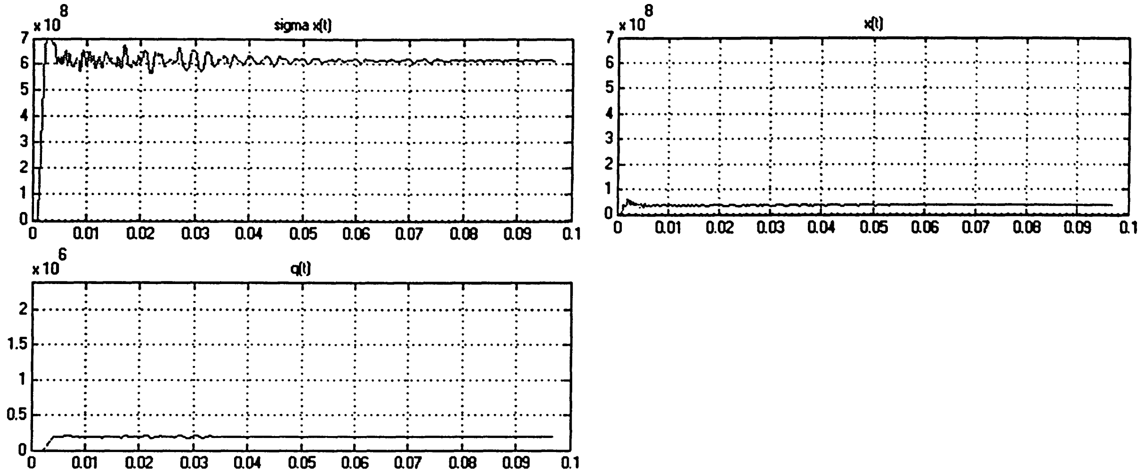


Figure 7.19: Traffic rate $x_1(t)$ of an active node, Overall traffic (Sigma $x(t)$) and Queue dynamic of the congested node when Fairness Control with Smith Predictor is implemented.

Figure 7.19 is the result of the simulation. In each node Fairness Control with Smith Predictor is implemented. The delay is large but with smith predictor, the system is stable. Although this system is stable but for faster and smoother transient behavior the Fairness Controller parameter k_c needs to be adjusted by making it proportional to γ_i to prevent from the oscillation seen in this figure.

7.3.7 Scenario 3.7

In this scenario it is assumed that there are 16 nodes that have Fairness Control with smith predictor and the distance between the nodes is $20\,km$ so for the closest node to congested node $T_{fw1} = T_{fb1} = 100\,\mu\sec$ (RTT_1=200$\,\mu\sec$). The controllable parameter is $k_c = \dfrac{0.1}{T} \times \gamma_i = \dfrac{1000}{16} = 62.5$ which T is sampling interval $\gamma_i = \dfrac{1}{16}$.

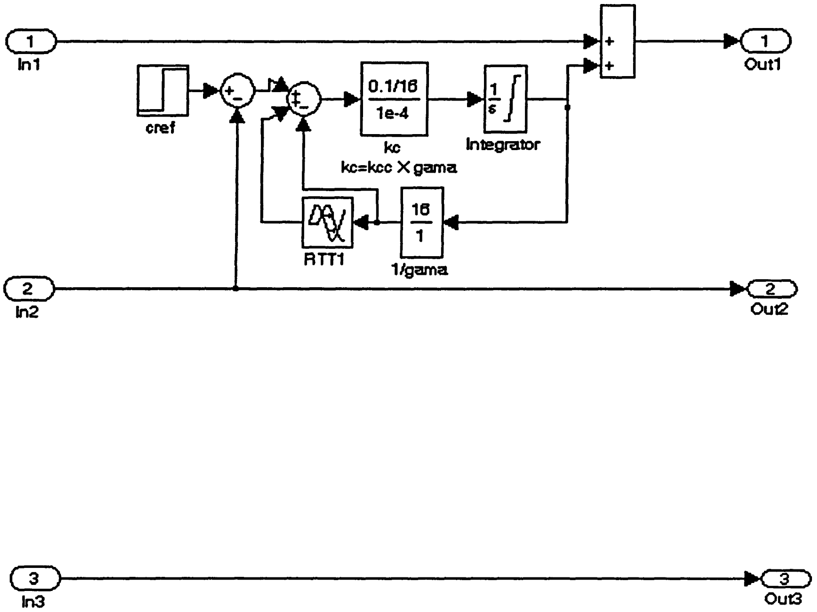


Figure 7.20: A simple node model with Fairness Control and Smith Predictor and adjusted parameter

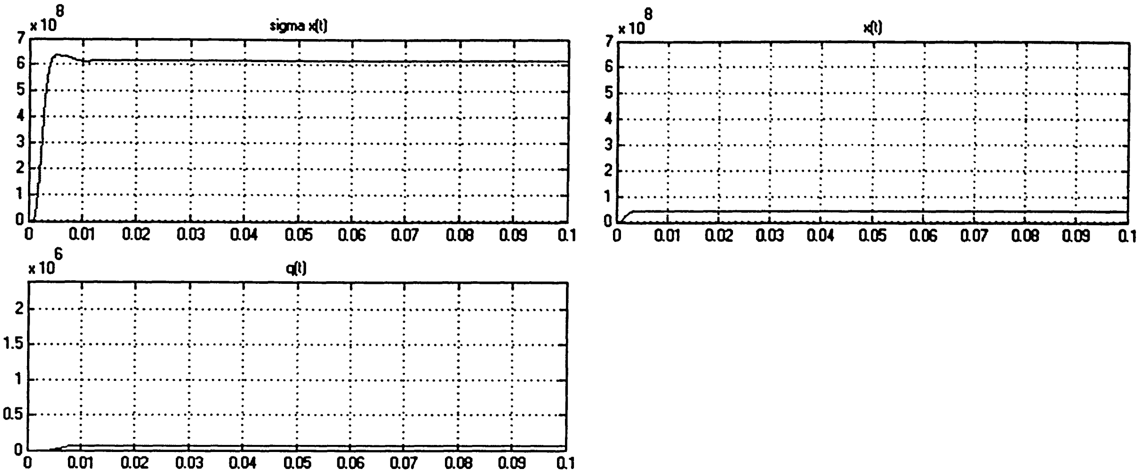


Figure 7.21: traffic rate $x_1(t)$ of an active node, Overall traffic (Sigma $x(t)$) and Queue dynamic of the congested node when the controller parameter is adjusted

Figure 7.21 shows the result of Fairness Control model. In each node Fairness Control with Smith Predictor is implemented. Also in this scenario the Fairness Controller parameter k_c is adjusted by making it proportional to γ_i. This is done to compensate the effect of the number of active nodes on the Fairness Control. This shows a very smooth transient behavior.

7.3.8 Scenario 3.8

In this scenario, 16 nodes are shown with Smith Predictor and Queue Congestion Control. The distance between the nodes is $20\,km$ so for the closest node to congested node $T_{fw1} = T_{fb1} = 100\,\mu\text{sec}$ (RTT_1=200 μsec). The controllable parameters are $k_c = \frac{0.1}{T} \times \frac{1}{16} = 62.5$ and $k_q = \frac{0.04}{RTT}$ which is adapted by each node.

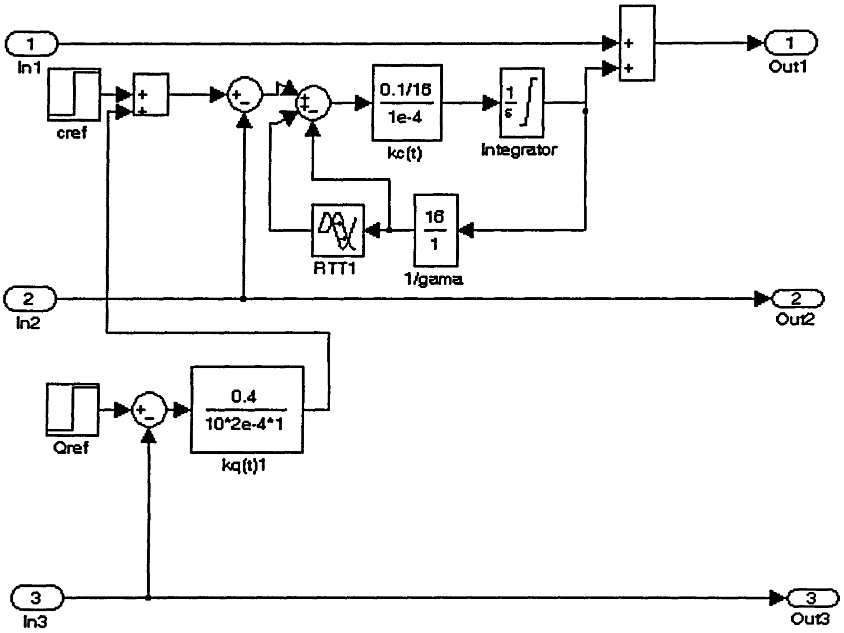


Figure 7.22: A simple node model with Fairness and Congestion Control with Smith Predictor

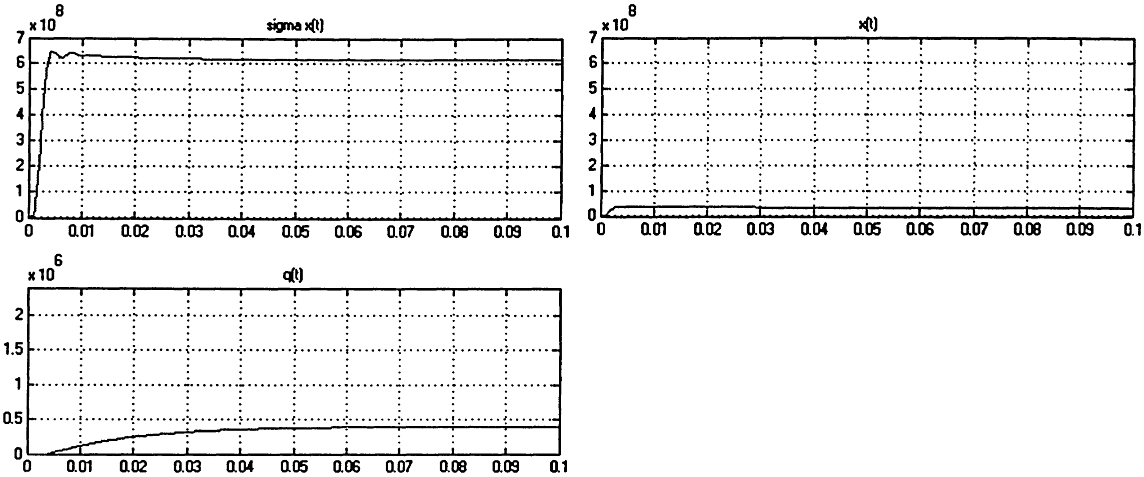


Figure 7.23: traffic rate $x_1(t)$ of an active node, Overall traffic (Sigma $x(t)$) and Queue dynamic of the congested node, show how Fairness and Queue level control are achieved when controller parameters are adjusted.

Figure 7.23 and Figure 6.24 show the result of both Fairness and Queue congestion control. In each node Fairness Control with Smith Predictor is implemented. The Controller parameter k_c is a proportional to γ_i to compensate the affect of the number of active nodes. Also for the Queue Congestion Control an adapted proportional controller is implemented and the controller parameter k_q is reversely proportional to RTT, to meet the stability criteria. The delay is large but with smith predictor the system achieves fairness and queue control.

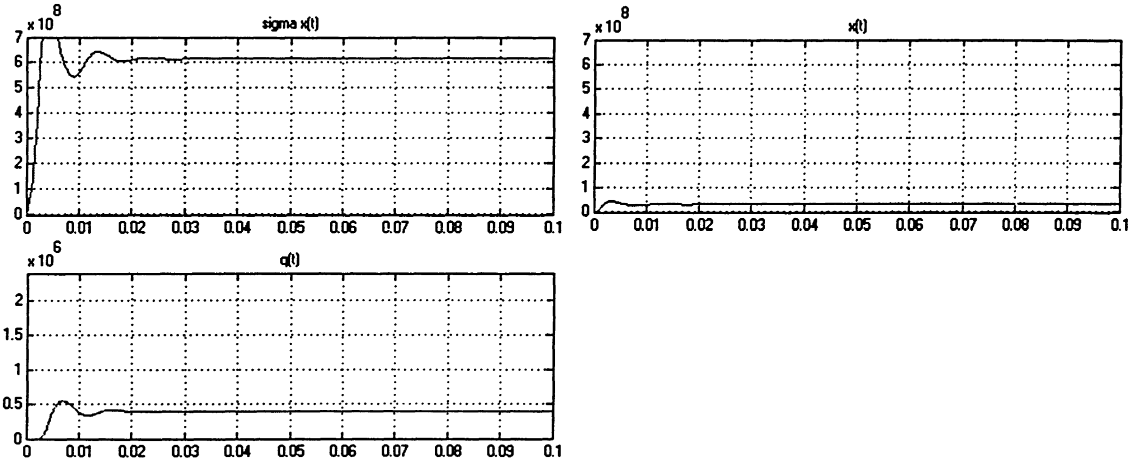


Figure 7.24: The system response when $k_q = \dfrac{0.4}{RTT}$

7.3.9 Scenario 3.9

This scenario is the same as Scenario 2.7 but it is simulated in the discrete time domain. In this scenario, 16 nodes are shown with Smith Predictor and Queue Congestion Control. The sampling time is $T = 100\mu\sec$ and the distance between the nodes is $20\,km$ so for the closest node to congested node has a delay equal to unit delay $T_{fw1} = T_{fb1} = T = 100\,\mu\sec$ (RTT_1=200$\,\mu$sec). The controllable parameters are $k_c = \frac{0.1}{T} \times \frac{1}{16} = 62.5$ and $k_q = \frac{0.4}{RTT}$ which is adapted by each node.

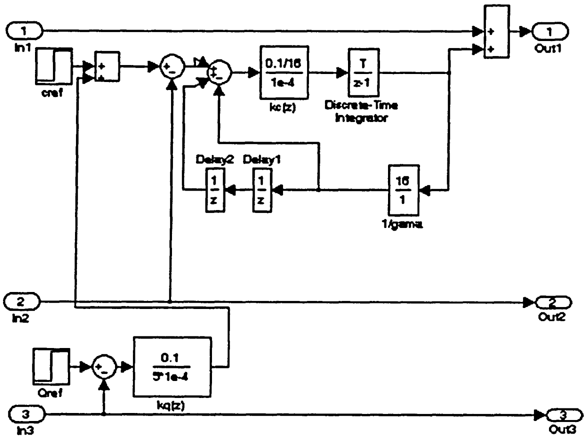


Figure 7.25: The closest node to the congested node with Fairness and Congestion Control with Smith Predictor

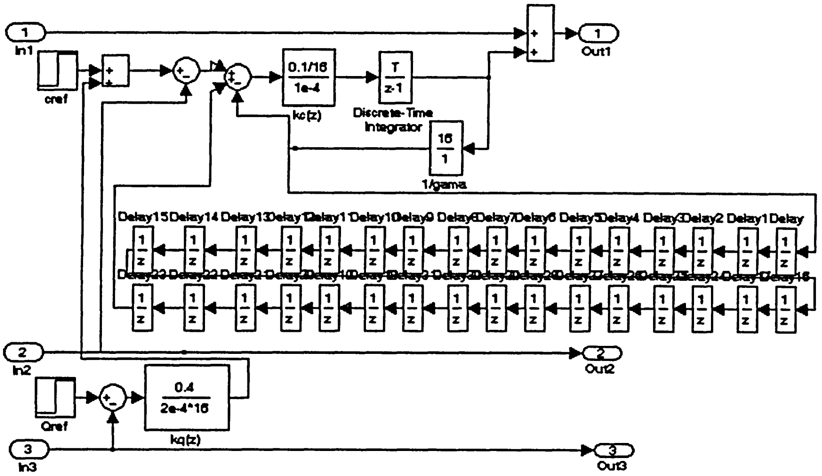

1
In1
cref
0.1/16
1e-4
kc(z)
T
z-1
Discrete-Time Integrator
16
1
1/gama
Delay15 Delay14 Delay13 Delay12 Delay11 Delay10 Delay9 Delay8 Delay7 Delay6 Delay5 Delay4 Delay3 Delay2 Delay1 Delay
1
z
1
Out1
2
In2
2
Out2
0.4
2e-4*16
kq(z)
Qref
3
In3
3
Out3

Figure 7.26: The furthest node to the congested node with Fairness and Congestion Control with Smith Predictor with $RTT_{16} = 32T$

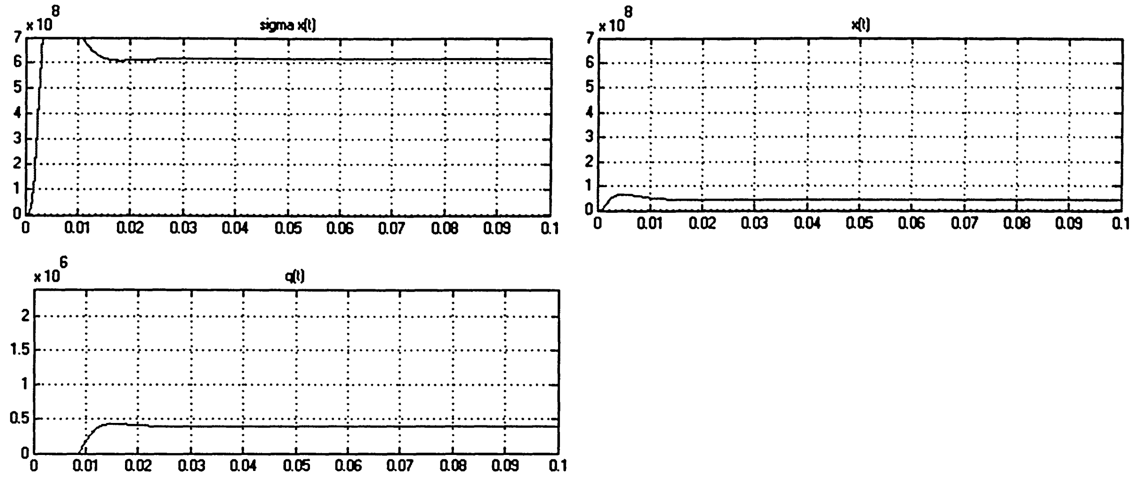


Figure 7.27: Discrete time domain simulation shows that Fairness and Queue level control are achieved.

In each node Fairness Control with Smith Predictor is implemented. The Controller parameter k_c is a proportional to γ_i to compensate the affect of the number of active nodes. Also for the Queue Congestion Control an adapted proportional controller is implemented and the controller parameter k_q for the closest 8 nodes is $\frac{k_c}{5\gamma_i}$ and for the rest is reversely proportional to RTT, to meet the stability criteria. The delay is large but with smith predictor the system achieves the fairness and queue control.

7.4 OPNET and MATLAB Simulation: Comparison

In this work a theoretical control model is developed for RPR fairness. MATLAB is used to simulate the existing and the controlled model. The model is compared with the RPR OPNET models such as Gandolf and SRP. Although OPNET models are nonlinear and more complicated while MATLAB models are linear and simple but they both show the trend of how delay and number of active nodes affect the throughput and performance. The main problem that will be addressed here is the delay. When delay is increased the throughput becomes unstable especially when the number of active nodes is large.

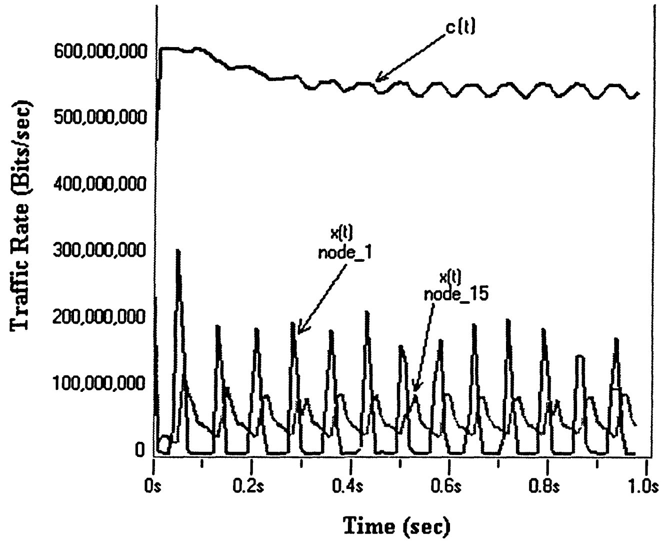


Figure 7.28: SRP OPNET result of 15 active nodes with 7ms delay between every two nodes. Throughput oscillates in RPR Conservative (SRP), when delay is high.

Figure 7.28 shows the simulation result of a scenario, in which 15 SRP nodes send traffic to node 16. As it is shown in the above figure, in SRP the overall throughput is less than the available bandwidth because the fairness algorithm is not effective enough. Also even with a conservative type of fairness, throughput oscillates for high number of nodes and long delay.

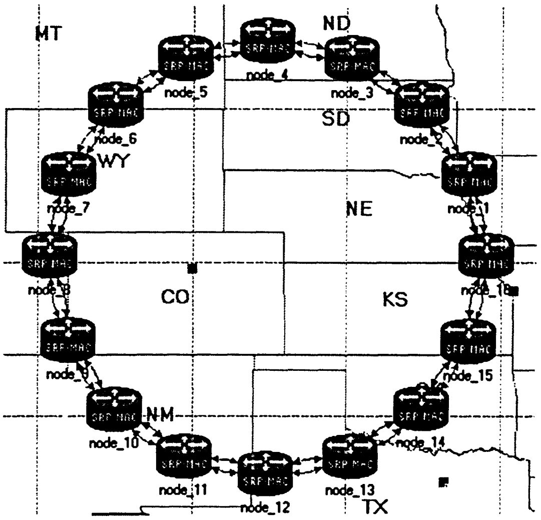


Figure 7.29: RPR Conservative (SRP) Ring in a WAN

Figure 7.29 shows that the ring spans in a WAN. In this scenario 16 SRP nodes are spread over a large ring with about $700km$ distance between any node pair. This is a sample scenario of an RPR ring as a Wide Area Network (WAN).

Here, the throughput comparison of RPR Conservative (Gandolf) and Aggressive is performed in the same scenario.

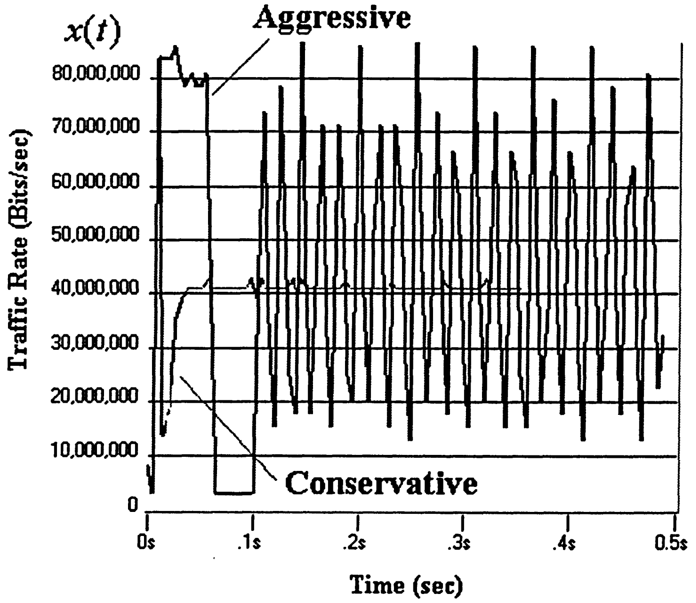


Figure 7.30: OPNET simulation shows the throughput in Conservative (Gandolf) versus Aggressive when RTT delay is zero.

Figure 7.30 shows how throughput in Aggressive mode oscillates around the fair rate even in short delay. In case of high delay the oscillation is even more. In Conservative mode throughput performance is relatively good when delay is short. When delay is increased, the throughput for conservative becomes more unstable too. In high delay scenario the throughput in both Aggressive and Conservative will oscillate.

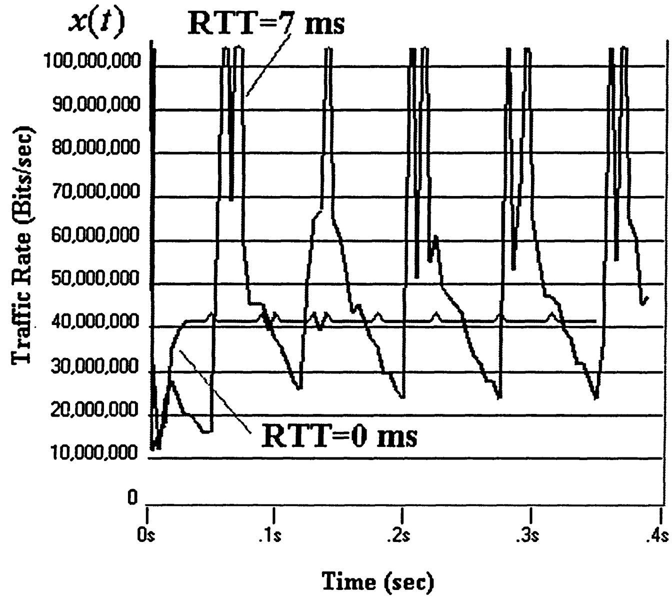


Figure 7.31: OPNET simulation shows that the throughput Oscillates in RPR Conservative (Gandolf) too, when delay is high.

Figure 7.31 shows how RPR Conservative mode becomes oscillatory when delay is increased. In this scenario the delay between any two nodes is almost 7 ms for the distance of $700km$. The Throughput oscillation for RPR Aggressive mode is more than Conservative, when delay is high.

The equivalent MATLAB simulation result is as follows.

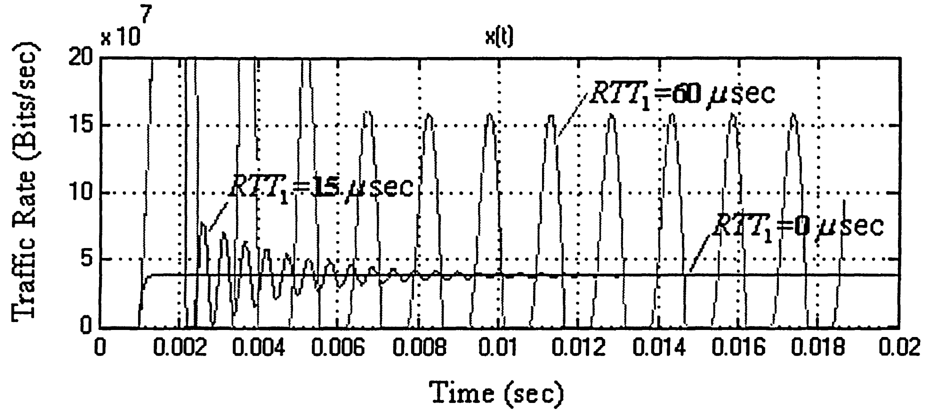


Figure 7.32: Throughput oscillates in RPR Conservative when delay is increased.(MATLAB Simulation)

Figure 7.32 shows three different situations. The RTT between each two nodes are zero, 15 μsec and 60 μsec. When delay increases from zero to 15μsec, the throughput starts to oscillate but it is temporary and the steady state is the fair rate. For a delay of 60μsec , throughput becomes unstable and it oscillates.

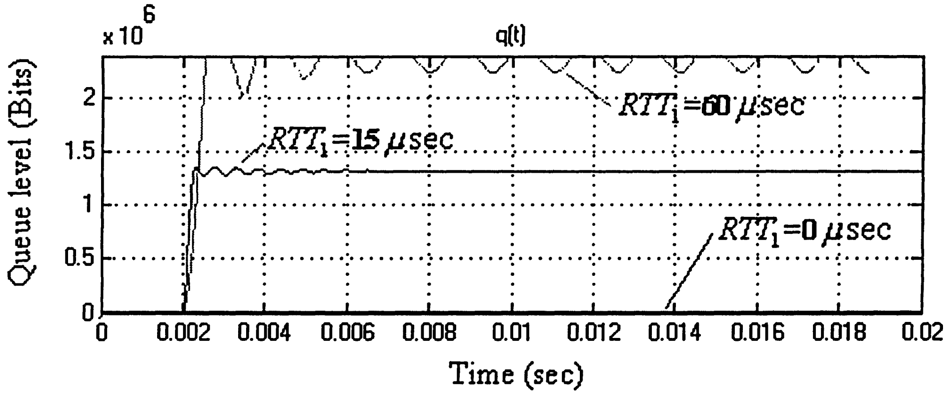


Figure 7.33: Queue level is increased and become unstable in RPR Conservative when delay is increased. (MATLAB Simulation)

As it is shown in Figure 7.33, increase of the RTT delay causes increase in queue level also oscillation in throughput for RTT delay of 60μsec leads to instability and overflowing of the congested queue which causes packet loss.

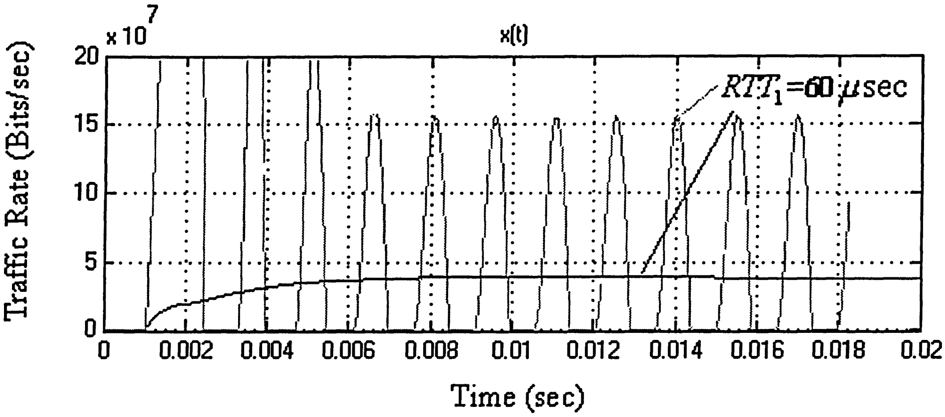


Figure 7.34: Simulation shows that RPR with Queue Congestion and Fairness Control makes the system stable so that Throughput won't oscillate even when delay is high. (MATLAB Simulation)

As it is shown in Figure 7.34 the system with RPR Queue Congestion and Fairness control becomes stable even with long delays. This effect improves the overall throughput performance by removing oscillation, when number of active nodes is large. As this system compensates the effect of the delay it is almost independent of the delay when this delay is estimated correctly.

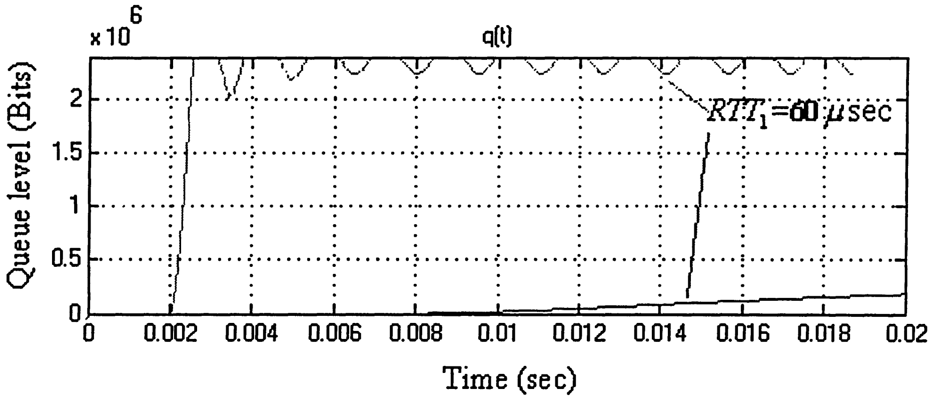


Figure 7.35: Queue level becomes stable when Queue Congestion Control is implemented. (MATLAB Simulation)

Figure 7.35 shows the queue level. The queue level can be stabilized on any predetermined level. The queue control parameter k_q needs to be assigned properly to optimize the queue control speed. To show the steady state queue level this figure needs to be shown with a new scale.

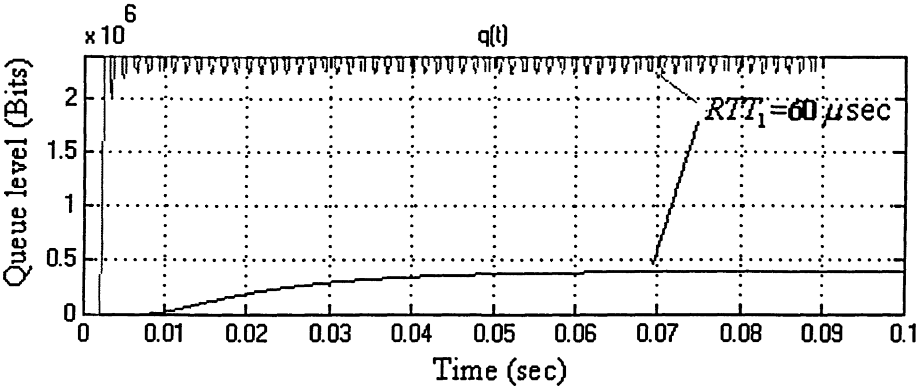


Figure 7.36: Queue level rises slowly and is controlled to a predetermined level (MATLAB Simulation)

Figure 7.36 is the same as the previous one with another scale. As it is shown the level of the queue is controlled on the predetermined level. This level is set based on the needs on the network. In RPR an acceptable STQ level is between one eight to one fourth of the STQ size.

In this scenario it is assigned as one sixth of the STQ size. By selecting an STQ size of $300KB$ this size is $50KB$ or $0.4Mbit$.

Chapter 8
Implementation

The scope of this thesis is mainly analysis the behavior of the current RPR fairness by fluid flow and linear approximations in continuous time domain and introducing our own proposed Fairness and Congestion control model but for Implementation some considerations need to be addressed.

Fluid flow, continuous time and linear approximations have their own implementation limitations so in this chapter we go over some considerations need to be done to implement such a model in the real world which is packet based, discrete and nonlinear.

When congestion is detected and the fairness turns on, two major adaptation needs to be done. First all active nodes need to adapt their traffic to a fair rate. At the same time if the level of the congested queue is too high it needs to be controlled and set to a set point Q_{ref} by decreasing the traffic sent by the active nodes.

8.1 Congested Node

To implement the model the focus needs to be on the traffic rate received by the congested node and the Secondary Transit Queue (STQ) of the congested node. STQ is used to buffer the fairness eligible traffic in an RPR node.

When congested node goes to fairness mode, the congested node sends the feedback information to the active nodes by sending fairness feedback message. Congested node is an active node itself so it starts adapting its own rate to the fair rate too. The feedback information consists of the received traffic rate, the occupancy level of its queue, the sum of the weights of the active nodes and number of active nodes. Each active node then, calculates its own target rate based on the received information.

8.2 Fairness Packet

Fairness packet contains:

1) Traffic rate received by the congested node, $\sum_{i=1}^{l} x_i(t) + R(t)$ (including its own traffic)
2) Occupancy level of the congested queue , $q(t)$
3) Number of active nodes, $l(t)$
4) Overall weight of all the active nodes, $\sum \omega_i$.
5) The fair rate calculated by the congested node.

These information then are used by the active nodes for their adaptation. When an active node receives a fairness packet it reads the information and corrects its own rate based on these information.

8.3 Fairness Traffic Adaptation

The value of $\sum_{i=1}^{l} x_i(t) + R(t)$ is sent to active nodes by the Congested node. This value, then, is compared with c_{ref} by active nodes to find the error $e(t)$. Then it is multiplied to $k_c(t) = k_c$. k_c is the controllable parameter of the Fairness Control. It shows how fast the adaptation of traffic rate is. A smaller k_c may represent more Conservative mode and very high k_c may represent less Conservative behavior.

The result is $u_i(t)$ which is used to correct $x_i(t)$. In steady state, error and $u_i(t)$ become zero and traffic rate becomes stable. In the discrete time domain the integrator can be shown as follows.

$$x_i(k) = x_i(k-1) + u_i(k),$$

or

$$x_i(k) = x_i(k-1) + k_c e(k).$$

In every sampling interval the traffic rate is corrected and in the steady state it stays in the fair rate. $\sum \omega_i$ along with $l(t)$, is used by each active node to calculate its own dynamic weight $\gamma_i = \frac{\omega_i}{\sum_{n=1}^{l} \omega_n}$. This dynamic weight is used for traffic rate adaptation.

8.3.1 Smith Predictor

Smith Predictor attempts to compensate the effect of the Fairness Feedback RTT delay from the closed loop system. Smith predictor uses the estimated RTT to compensate the real round trip time delay. Each node estimates its Fairness Feedback RTT by estimating the time difference between sending a packet and receiving the response. The active node has a table that keeps the information about how much traffic it has sent in the past. After estimation of RTT, this table is used to find the traffic sent during the last RTT.

8.4 Queue Congestion Control

The average level of the congested Queue level needs to be controlled. If the average level is too high then it must be decreased. When congestion is detected and the fairness turns on, the Fairness algorithm tries to control the level of the queue for the congested node S_i to a set point Q_{ref} by decreasing the traffic sent by the active nodes. This is done by decreasing the c_{ref} for Fairness Control. This means that traffic rate of all the active nodes are be decreased.

The average queue level needs to adapt to a predetermined level. This level is set based on the needs on the network. In RPR draft an acceptable STQ level is between one eight to one fourth of the STQ size. A practical reference level could be one sixth of the STQ size. In practice STQ size in many implementation is $300KB$ so the queue reference level would be $50KB$.

Chapter 9

Conclusions

Link delay and number of active nodes have a great impact on the stability of the RPR Fairness algorithm. The queue level occupancy may show oscillation and instability when Link delay and number of active nodes are significant but there is not enough theoretical analysis to discuss this behavior. The present work is the first theoretical analysis of the RPR Fairness algorithm using control theory. Our contribution to RPR is developing a new dynamic model for RPR Fairness and Congestion Control algorithm based on the control theory. Simulation, theoretical analysis, and comparison are employed to explore the effects of different controllers on the closed loop system behavior.

In this thesis RPR Fairness algorithm is described in the congestion domain as a distributed system which its characteristics varies from time to time because of the number of active nodes. Then it is mapped to a classic control system. A Congestion and Fairness Control Model is proposed and analyzed. Time delay analysis was used to describe how the system behaves when the delay is significant. Smith Predictor is proposed to compensate the round trip time delay.

Our model is a dynamic model which shows excellent stability even with large number of active nodes and long delays. Each active node estimates its round trip time and compensates the ring delay so that the ring delay won't affect the nodes rate. Also the active nodes will tune the controller parameters dynamically based on their received information from the congested node. By tuning these configurable parameters, the model can make the system independent to RPR internal parameters and in case of any unpredicted disturbance, the system always stays in the stable state.

Future Works

In this thesis we focused on single-chocked fairness. The next challenge would be working on the Multi-chock point problem and making its control theoretical model. Also we assumed that the system is working in the linear area. We will need to analyze the nonlinearity of the system. The model needs to be extended to the discrete time domain and implementation challenges need to be addressed. Implementing this model in OPNET simulator is another future work that would give the opportunity to compare the behavior of this model with the existing RPR models simulated with OPNET.

Appendix A

A.1. Classic Controllers

As in the real implementation, RPR Fairness algorithm works in a discrete time domain here it is shown how to convert continues classic controller to discrete form. This is the equation of a Proportional, Integrated and Derivative (PID) controller:

$$u(t) = K\left(e(t) + \frac{1}{T_i}\int e(t)dt + T_d\frac{de(t)}{dt}\right)$$

The equivalent in Laplace domain is:

$$k(s) = \frac{u(s)}{e(s)} = K\left(1 + \frac{1}{T_i s} + T_d s\right)$$

Laplace transfer function is

$$k(s) = \frac{u(s)}{e(s)} = KT_d\left(\frac{s^2 + \frac{1}{T_d}s + \frac{1}{T_d T_i}}{s}\right) = \frac{a_1 s^2 + a_2 s + a_3}{s}$$

Using the system backward difference approximation for integration and differentiation, the transfer function of a discrete PID controller can be written as:

$$K(z) = \frac{u(z)}{e(z)} = K\left(1 + \frac{Tz}{T_i(z-1)} + \frac{T_d(z-1)}{Tz}\right)$$

$$K(z) = K\left(\frac{z^2\left(1+\frac{T_d}{T}+\frac{T}{T_i}\right)-z\left(1-\frac{2T_d}{T}\right)+\frac{T_d}{T}}{z(z-1)}\right)$$

PID controller

Continues PID transfer function is

$$k(s) = \frac{a_1 s^2 + a_2 s + a_3}{s}.$$

Discrete PID transfer function is

$$K(z) = \frac{A_1 z^2 + A_2 z + A_3}{z(z-1)}.$$

Converting it into difference equation

$$u(k) = u(k-1) + A_1 e(k) + A_2 e(k-1) + A_3 e(k-2).$$

PI controller

Continues PI transfer can be obtained from the PID transfer function, when $a_1 = 0$.

$$k(s) = \frac{a_2 s + a_3}{s}$$

In case of a Discrete PI controller, we have $T_d = 0$ so $A_3 = 0$ and the transfer function would be

$$K(z) = \frac{A_1 z^2 + A_2 z}{z(z-1)} = \frac{A_1 z + A_2}{z-1}.$$

PI controller can be simulated as follows.

The difference equation is

$$u(k) = u(k-1) + A_1 e(k) + A_2 e(k-1).$$

P controller

Proportional controller is the simplest form of controller and it is used in our paper.
In continues

$$k(s) = a_2 = K.$$

In Discrete case

$$K(z) = \frac{A_1 z^2}{z(z-1)} = \frac{A_1 z}{z-1}.$$

The difference equation is

$$u(k) = u(k-1) + A_1 e(k).$$

A.2. Smith Predictor

Smith Predictor compensates the Round Trip Time delay by estimating it.

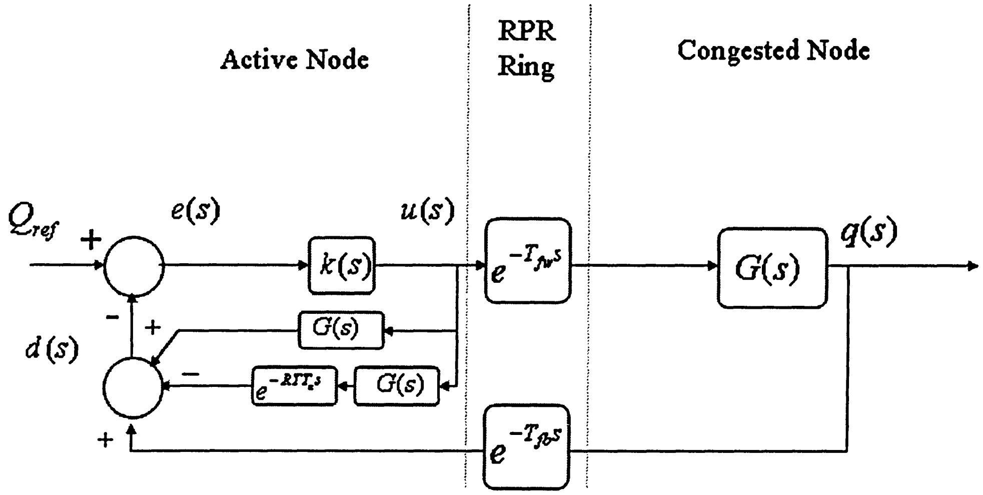


Figure A.1: Delay compensation by Smith Predictor

To compensate the delay, the active node can subtract the feedback by its estimated value. So

$$d(s) = u(s)e^{-T_{fw}s}G(s)e^{-T_{fb}s} - u(s)G(s)e^{-RTT_e s} + u(s)G(s),$$

$$d(s) = u(s)G(s)e^{-(T_{fb}+T_{fb})s} - u(s)G(s)e^{-RTT_e s} + u(s)G(s),$$

and

$$d(s) = u(s)G(s)e^{-RTTs} - u(s)G(s)e^{-RTT_e s} + u(s)G(s).$$

When the estimated Round Trip Time, RTT_e is equal to the real RTT, then

$$d(s) = u(s)G(s)(e^{-RTTs} - e^{-RTT_e s}) + u(s)G(s),$$

so

$$d(s) = u(s)G(s).$$

This way the active node removes the delay from the loop as if there were no delay. This is another view of Smith predictor:

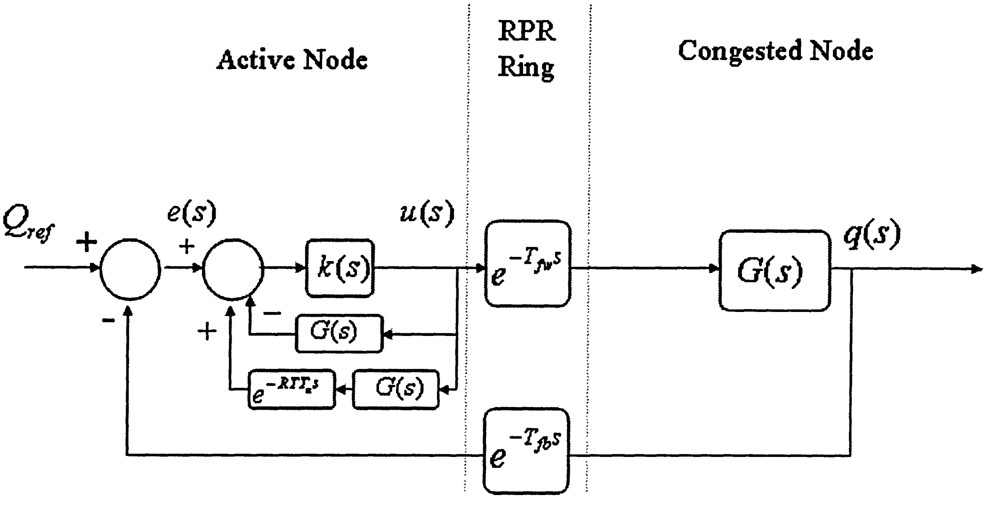


Figure A.2: Smith Predictor

The transfer function of the Smith Predictor controller is:

$$\frac{u(s)}{e(s)} = \frac{k(s)}{1 + k(s)G(s)(1 - e^{-RTT_e s})}$$

And

$$\frac{q(s)}{u(s)} = G(s)e^{-T_{fw}s}$$

So the open loop transfer function will be:

$$G'(s)=\frac{q(s)}{e(s)}=\frac{k(s)G(s)e^{-T_{fw}s}}{1+k(s)G(s)(1-e^{-RTT_e s})}$$

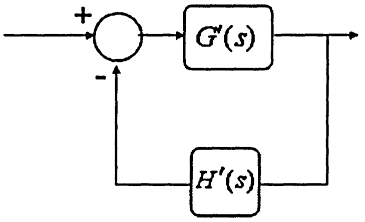


Figure A.3: The simplified model

And

$$H'(s)=e^{-T_{fb}s}$$

When the estimated Round Trip Time, RTT_e is very close to the real RTT for the node, the closed loop function is:

$$\frac{G'(s)}{1+G'(s)H'(s)}=\frac{k(s)G(s)}{1+k(s)G(s)}e^{-T_{fw}s}$$

This transfer function is equivalent to the transfer function of the following diagram:

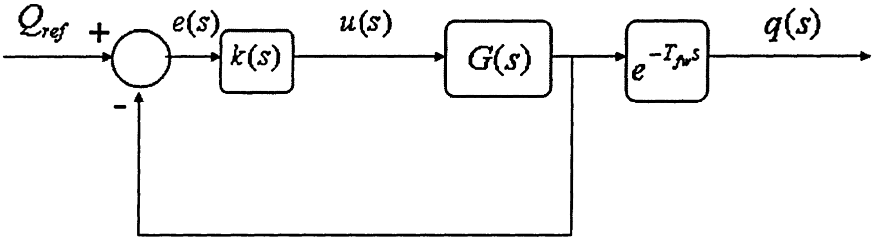


Figure A.4: Equivalent model with Smith Predictor in Laplace space

As we see the delay system is equal to a pure closed loop system, in cascade with a pure delay. This means that change in delay won't affect the stability of the system.

Appendix B

Scenarios

Here the OPNET Simulation of two major implementations of the conservative Fairness, SRP and Gandolf are compared.

SRP

This is a 4 node scenario that Node_1, 2,3 and 4 send 622 Mbps ClassC traffic to node_0 at 0.0, 0.1, 0.2 and 0.3 Sec. Packet size is 1200 with Dual Queue implementation. The traffic is sent to the Outer Ring and node_1 is the congested node

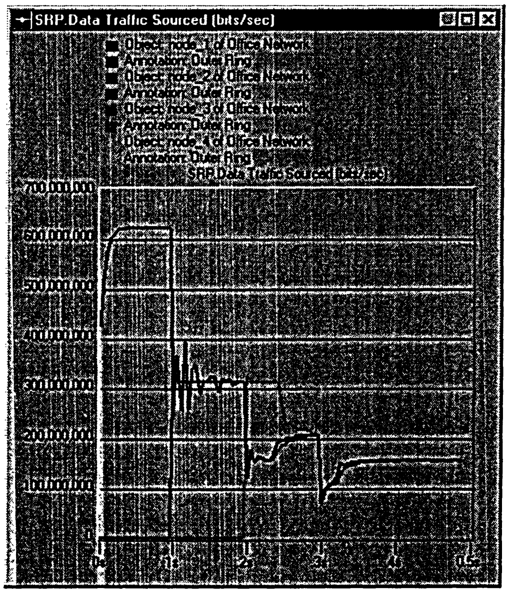


Figure B.1: SRP implementation of Fairness is fast with some transient oscillation.

SRP implementation of fairness is fast but but with some transient oscillation.

Gandolf

As we see Gandolf is more conservative than SRP, therefore, more stable with no oscillation.

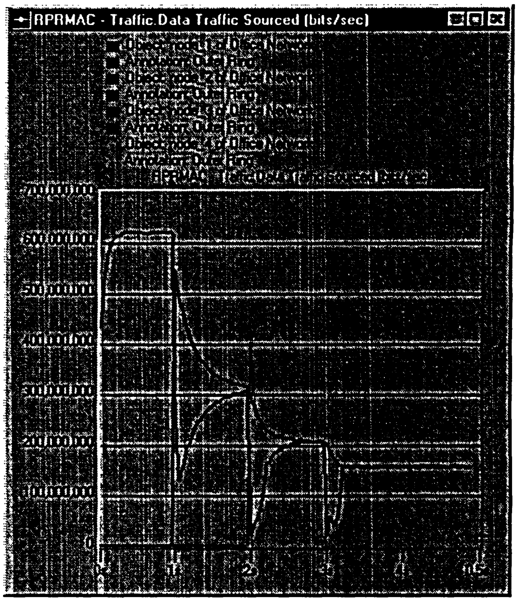


Figure B.2: Gandolf implementation of Fairness is much more conservative, stable but slow.

The conservative mode is shown in the Gandolf model which the congested node decreases its rate to a level less than the fair rate to compensate the slow response of the fairness mode. When the fairness is too slow, the active nodes cannot response to the congestion fast enough so the congested node has to decrease its rate somehow that the extra traffic is compensated. This will cause a non fair rate because the congested node will suffer. If the system is too conservative and the extra traffic is more than the congested node then, congested node will decrease to zero which is not desirable at all. Gandolf model is stable but the bandwidth utilization is poor. In cases of too many nodes the congested node may suffer.

Bibliography

[1] *An Introduction to Resilient Packet Ring Technology*, RPR Alliance, www.rpralliance.org

[2] *IEEE Draft 802.17/D2.2.*, www.grouper.ieee.org/groups/802/17/. IEEE802.17 official web site, April 9, 2003.

[3] D. Tesiang. '*The Cisco SRP MAC layer Protocol*" http://www.apps.ietf.org/rfc/rfc2892.html.

[4] OpTera Packet Edge System/RPR Technology, www.nortelnetworks.com

[5] F. Hanell, "*Global Optical Communications- Market Overview*", VP Marketing, Dynarc AB

[6] H. R. Van As, Vienna University of Technology, "*Cyclic Demand Advertisement boosts, RPR MAC Performance*", IEEE802.17 web site, www.grouper.ieee.org/groups/802/17/, 2002.

[7] S. Gjessing, Simula Research Laboratory, University of Oslo, Norway, "*A fairness Algorithm for High-speed Networks based on a Resilient Packet Ring Architecture*", in Proc. to Workshop on Communication Architecture for Clusters-CAC'02, IEEE802.7 web site.

[8] D.A. Schupke, Munich University of Technology "*A Throughput Study of RPR Networks Subject Single Failures*", Citeseer.nj.nec.com/359755.html, 2002.

[9] V. Gambiroza, Y. Liu, P. Yuan, and E. Knightly, Rice University, USA., "High *performance Fair Bandwidth Allocation for Resilient Packet Rings*", In Proc. of the 15th ITC Specialist Seminar on Internet Traffic Engineering and Traffic Management, Wurzburg, Germany, 2002.

[10] Fenjie Yuan. Carlton University, "*Performance Analysis for Resilience Packet Ring With single transit buffer*", IEEE802.17 web site, www.grouper.ieee.org/groups/802/17/.

[11] S. Mascolo, " *Congestion control in high-speed communication networks* ", Automatica, Vol 35, PP. 1921-31, March 1999.

[12] C.V. Hollot, V Misra, University of Massachusettes, Amherst, "*A Control Theoretical Analysis of RED*", Pages 1510-1519, INFOCOMM, 2001.

[13] P. Narvaez, K. Y. Siu, , d'Arbeloff Laboratories for Information Systems & Technology, MIT, "*Design of Feedback Control Algorithms for ABR Service in ATM*", Tech. Report 11/96.

[14] G. Anastasi1, L. Lenzini, M. La Porta, Y. Ofek, University of Pisa, Dept. of Information Engineering, "*Dynamic Max-Min Fairness in Ring Networks*", Cluster Computing Vol. 3, pp. 215-230, 2000.

[15] D. Rosu, K. Schwan, S. Yalamanchili, and R. Jha. "*On Adaptive Resource Allocation for Complex Real-Time Applications*". 18th IEEE Real-Time System Symposium, pp 320-329, 1997.

[16] F. Paganini, Z. Wang, J. C. Doyle and S. H. Low, UCLA, "*Congestion Control for high performance, Stability and Fairness in general networks*", In Proc. to IEEE/ACM Transaction on Networking, 2004.

[17] Z. Wang, F. Paganini, UCLA, "*Global Stability with Time-Delay in Network Congestion Control*", IEEE Conf. on Decision and Control, Las Vegas, NV, 2002.

[18] B. Li, K. Nahrtedt. University of Illinois, "*A Control Theoretical Model for Quality of Service Adaptations*", IEEE/IFIP International Workshop on Quality of Service (IWQoS), (Napa, CA, USA) May 1998.

[19] I. Papadimitriou, G. Mavromatis, Stanford University, "*Stability of Congestion Control Algorithms using Control Theory with an application of XCP*", 384Y Project, June 5,2003

[20] G. Vinnicombe, University of Cambridge, "*On Stability of Networks Operating TCP-Like Congestion Control*", In 15th IFAC World Congress on Automatic Control, Barcelona, Spain, 2002.

[21] D. Katabi, M. Handley and C. Rohrs, MIT, "*Internet Congestion Control for Future High Bandwidth-Delay Product Environments*", ACM SIGCOMM, 2002.

[22] S. Chong, R. Nagarajan, Y. Wang, Sogang University, Seoul, South Korea. "*Designing Stable ABR flow Control with rate feedback and open-loop control: first-order control case*", Performance Evaluation 34: 189-206, 1998.

[23] B. Ataslar, P.-F Quet, A. Iftar, S. Kalyanaraman, Rensselaer Polytechnic Institute, Tory, USA, "*Robust Rate-Based Flow Controllers for High-Speed Networks: The Case of Uncertain Time-Varying Multiple Time-delays*".In Proc. ACC, Pages 2804—2808, Chicago, IL, 2000.

[24] D. Cavendish, S. Mascolo and M. Gerla, "*SP-EPRCA: an ATM Rate Based Congestion Conrol Scheme based on a Smith Predictor*". Technical report UCLA, 1997.

[25] S. Mascolo, Plitecnico di Bari, Italy, "*Smith's Principle for Congestion Control in High Speed ATM Networks*", In Proc. of 36th Conference on Decision and Control, pp. 4595—4600, Dec 1997.

[26] O. Smith. "*A Controller to Overcome the Dead Time*", ISA Journal, Vol 6, No 2, Feb 1959.

[27] K. J. Astorm, "*Computer Controlled Systems*", Prantice Hall 1997.

[28] G. Franklin and J. Powell. "*Digital Control of Dynamic systems*". Addison-Welsely 1981.

[29] D. Bertsekas and R. Gallager, *Data Networks,* Prentice Hall, 1987.

[30] Mathworks, Inc., *MatLab*, R11, http://www.mathworks.com/