

TFRC Modeling and Its Applications

CHEN, Liang

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Information Engineering

July 2009

Abstract of thesis entitled:

TFRC Modeling and Its Applications

Submitted by CHEN, Liang

for the degree of Master of Philosophy

at The Chinese University of Hong Kong in July 2009

The purpose of this thesis is to demonstrate the applicability of the TCP-Friendly Rate Control (TFRC) [1] algorithm from theoretical perspective. TFRC has the same long-term throughput as TCP, while its short-time throughput fluctuation is much smaller than that of TCP. These quality of service properties make TFRC appropriate for multimedia streaming applications which require constant audio and video quality and fairness to TCP flows. While there are many work on empirically evaluating throughput and fairness of TFRC, little work has been done on examining TFRC's performance from theoretical aspects. Our work is intended as an investigation of TFRC algorithm based on mathematical modeling and analysis.

Under the utility maximization framework, we propose two TFRC models in terms of their detection approaches of loss rate. Both the dynamic systems defined by TFRC algorithm and the systems consisting of TCP and TFRC flows are proved with globally asymptotically stability. Then through the study of the convergence rate and robustness to perturbations of TFRC algorithm, we show that the smoothing factor of TFRC has great influence on these properties. In the presence of delay, we extend the TFRC models by introducing the delay factor, and derive sufficient conditions for the stability of TFRC algorithm under delay. These results imply that the long RTT delay may compromise the system stability, which verifies the empirical study of TFRC. Through simulations, we are able to validate our theoretical analysis and evaluate the relative importance of the smoothing factor and RTT delay on network stability. Therefore, based on our theoretical analysis validated by empirical results, we can confirm that the TFRC algorithm works stably and can coexist with TCP in one network.

摘要

本論文的目的是從理論角度討論 TCP-Friendly Rate Control (TFRC) [1] 算法的適用性。就穩態平均傳輸速率而言，TFRC 可同 TCP 保持一致，而它在瞬態時間範圍內吞吐率的波動相比 TCP 平滑很多。這樣的服務質量 (QoS) 特性使得 TFRC 更適合多媒體流應用程序的需求，這些應用傾向於較為平穩的音頻視頻傳輸，並且希望能與占主導地位的 TCP 數據共存。儘管已有一些工作從實驗角度對 TFRC 的吞吐率和公平性進行研究，但從理論方面分析評價 TFRC 性能的工作較少。本文的目的是基于數學建模和理論分析對 TFRC 算法進行研究。

基于 utility maximization framework，我們依據丟包率得到的方式提出兩個 TFRC 模型。然後討論并證明了由 TFRC 算法定義的動態系統以及包含 TCP 和 TFRC 流的共存系統的全局漸進穩定性。通過對 TFRC 算法的收斂速率和關於擾動的魯棒性的研究，可以看到 TFRC 的平滑因子對這些特性有很大影響。當考慮實際系統中存在的時延時，我們擴展了 TFRC 模型，加入時延對系統的影響，并通過分析得到在時延情況下能保證穩定性的充分條件。研究的結果顯示較大的 RTT 時延會影響系統的穩定性，這點通過實驗工作可得到驗證。通過仿真，我們驗證了關於 TFRC 模型的理論分析，并且考察了平滑因子和 RTT 時延對網絡穩定性的重要影響。因此，基于理論分析和對實驗結果的驗證，我們可以確定 TFRC 算法工作穩定并且可以和 TCP 共存于同一個網絡。

Acknowledgement

I wish to express my gratitude to my supervisors, Prof. Ng Wai-Yin and Prof. Chen Minghua. The life spirit and research style I learned from both of them have great influence over my life. Anywhere and anytime, I will keep these words in heart: “make things simple but non-trivial” and “don’t ever give up”.

This research was supported by the postgraduate studentship from CUHK and by the Direct Grant (Project Number 2050397) of The Chinese University of Hong Kong. I gratefully acknowledge these financial support.

This work is dedicated to my precious experience at CUHK.

Contents

Abstract	i
Acknowledgement	iii
1 Introduction	1
1.1 Problem	1
1.2 Motivation	3
1.3 Thesis Contribution and Organization	5
2 Background Study	9
2.1 TFRC	9
2.2 Related Work	11
3 Network Modeling	15
3.1 Network Utility Maximization Framework	15
3.1.1 Primal Algorithm	16
3.1.2 Dual Algorithm	17
3.2 Overview of TCP Reno Modeling	18
3.3 Modeling TFRC	19
3.3.1 TFRC Model I	20
3.3.2 TFRC Model II	21
3.4 Modeling Coexistence Case	23
4 Stability Analysis	27
4.1 TFRC Network	27
4.1.1 Global Stability	28
4.1.2 Rate of Convergence	32
4.1.3 Rate-adaptation Comparison	36
4.2 TCP Reno and TFRC Coexistence Network	40

4.2.1	Existence and Uniqueness of Equilibrium	40
4.2.2	Stability Analysis of the Coexistence Case	41
5	Delay Analysis	45
5.1	TFRC Network Model I	46
5.2	TFRC Network Model II	51
5.3	Robustness Comparison of TCP and TFRC	55
6	Simulation Results	61
6.1	Matlab Simulations	61
6.1.1	Smoothed Effects and Rate Convergence	61
6.1.2	Rate-adaptation Comparison of Two Models	64
6.1.3	Delay Instability	65
6.2	NS2 Simulations	69
6.2.1	Traffic Smoothness and Jitter Property	70
6.2.2	Necessity of Adaptive Scheme	73
7	Conclusion	77
A	Appendix	81
A.1	Delay Analysis for the Single Link Case of TFRC I	81
A.2	Delay Analysis for the Single Link Case of TFRC II	84
	Bibliography	87

List of Figures

4.1	The case of loss interval increase.	38
4.2	The case of loss interval decrease.	39
5.1	The network case.	46
5.2	Nyquist convex hull of $\frac{\pi}{2} \frac{e^{-jwT}}{jwT}$	50
6.1	Dumbbell network topology.	62
6.2	Throughput of TCP.	63
6.3	Throughput of TFRC.	63
6.4	TCP Reno coexisting with TFRC.	64
6.5	TFRC with $\alpha_1 = \frac{5}{4}\alpha$	65
6.6	TFRC with $\alpha_2 = 2\alpha$	65
6.7	TFRC model I with CBR traffic.	66
6.8	TFRC model II with CBR traffic.	66
6.9	TFRC with $T_r = 0.02s$	67
6.10	Delay instability of TFRC.	67
6.11	Delay instability of TFRC coexisting with TCP Reno.	68
6.12	TCP flows with fluctuations.	68
6.13	TFRC flow rates converge.	68
6.14	TCP flow rates converge.	69
6.15	TFRC flows with fluctuations.	69
6.16	Simulation network topology.	70
6.17	Throughput of TCP.	71
6.18	Throughput of TFRC.	71
6.19	Jitter of TCP.	71
6.20	Jitter of TFRC.	71
6.21	Simulation network topology for multiple users.	72
6.22	The throughput of multiple TCP flows.	72
6.23	The throughput of multiple TFRC flows.	73

6.24	TFRC flows coexisting with TCP Reno flows.	74
6.25	TFRC without EWMA mechanism.	75
A.1	The single link case.	82

List of Tables

1.1	Stability and fairness among traffics controlled by TCP and/or TFRC.	3
1.2	Comparison of TCP, TFRC I and TFRC II.	7

Chapter 1

Introduction

Summary

Congestion control is of major importance for computer network to operate properly, it determines the network stability. As one of the most successful man-made systems, the Internet works well with embedded Transmission Control Protocol (TCP) implementing the Additive Increase and Multiplicative Decrease (AIMD) congestion control algorithm. While TCP is appropriate for bulk-data communication, it is not suitable for real-time applications, which prefer smoothed transmission rate and continuous reception. To satisfy these requirements, TCP-Friendly Rate Control (TFRC) algorithm is proposed as a congestion control mechanism for streaming data transfer. In this thesis, we model TFRC from theoretical perspective and study this congestion control scheme to understand its network dynamics.

1.1 Problem

Peer-to-Peer (P2P) video streaming services, such as PPStream, online multimedia services, such as YouTube, and multimedia conferencing applications, such as Skype, become more and more popular currently in the Internet. Consequently, the major sources of traffic on Internet will consist of both data flows and multimedia streams. A key to enable these diverse services is to control the sending rates of these two types

of traffics in such a way that their quality of services (QoS) requirements are met, and at the same time ensures the scalable development of the Internet.

On one hand, scalable development of the Internet requires any rate control protocols must be designed in a way to prevent network instability, e.g. congestion collapse. On the other hand, these two types of traffics may have different QoS preferences, due to different applications they serve. For instance, video streaming applications that require constant video quality typically prefer small fluctuation in throughput, while bulk-file transfer applications favor high long-term throughput such that total downloading time is minimum. In addition, video streaming applications prefer to handle the error control by themselves, while bulk-file transfer applications favor delegating error-free transmission control to the underlying rate control protocol. As a result, these two types of traffics may use different protocols to control their sending rates.

Nowadays, data flows typically use TCP [2, 3] to control their sending rate. TCP's stability and fairness performance have already been extensively evaluated in both practice and theory [4, 5]. As the dominant transport protocol in the Internet, the current TCP provides an end-to-end congestion control to ensure network stability and fairness. In brief, it has been shown that in a network consisting of TCP flows, the flow rates will converge to a stable equilibrium exponentially fast, and avoid congestion collapse [5]. Moreover, the flow rates of TCP are roughly α -fair with the parameter $\alpha = 2$ [6]. Given its success, it is doubtless that TCP will continue to serve data flows in the future.

Though currently using TCP, multimedia streams are expected to switch to TCP-friendly [7] schemes since they can better satisfy the QoS preferences than TCP. TCP-friendly is a generic term describing a flow does not reduce the long-term throughput of any coexistent TCP flow more than another TCP flow on the same path would under the same network conditions [8]. TFRC is the most well-known TCP-friendly rate control scheme and is widely accepted. It is conceivable that multimedia streams will be mainly controlled by TFRC in the near future.

Two important questions to answer before deploying TFRC along with TCP in practice are the following. First, is the network consisting of only TFRC flows stable, and whether TFRC flows are fair to each other? Second, is the network stable when TFRC coexists with TCP, and whether they are fair to each other? The questions are summarized in Table 1.1. Without positive answers to these questions, it is risky and irresponsible to deploy TFRC in practice, and it can be very costly to fix its flaws, if any, in a scale as large as the Internet. On the basis of solving these problems, we intend to further study the system defined by TFRC scheme in terms of rates of

Table 1.1: Stability and fairness among traffics controlled by TCP and/or TFRC.

	TCP	TFRC
TCP	stable and fair	?
TFRC	?	?

convergence, stochastic perturbations and delay stability.

Proving TCP and TFRC coexist stably and fairly also has direct consequence on the latest wireless flow control schemes [9]. E-MULTTCP and E-MULTFRC are two latest flow control algorithms that have provable optimal performance over wireless networks. E-MULTTCP is designed for data flows, while E-MULTFRC is designed for multimedia streams. It is shown in [9] that if TCP and TFRC can coexist stably and fairly, then so does any combination of TCP, TFRC, E-MULTTCP and E-MULTFRC.

The goal of our study is to explore answers to these questions. There exists work on answering the two questions mainly based on practical experiments [1, 10], but little work has been done on the theoretical side. Theoretical evaluation has two advantages comparing to practical evaluation. On one hand, the results of theoretical evaluation are more general than that of practical evaluation, in the sense that they typically apply to a wider range of network topologies and traffic patterns than practical evaluations can possibly check. On the other hand, theoretical evaluation typically generates deeper understanding to performance of the protocols and shines the lights on possible improvement.

1.2 Motivation

Network bandwidth is a limited resource. When the demand for bandwidth exceeds available network resources, network links would get congested. Especially for the case that requires reliable transmission, lost packets need to be retransmitted further increasing the load on the network. Thus, congestion control is needed in order to allow the network to recover from congestion and operate in a relative stable state.

While TCP embedded AIMD congestion control algorithm performs well for bulk data transfer, it may not be appropriate for the delivery of streaming media. Because halving the sending rate in response to a single congestion indication is unnecessarily severe for real-time applications, in which TCP's abrupt changes in the sending rate can noticeably reduce the user-perceived quality [11]. In current network, some of these applications may adopt User Datagram Protocol (UDP) to transfer multimedia streams. However, UDP does not implement any congestion control algorithms, such

that it could cause congestion collapse and the unfairness to competing flows that are congestion-aware [7]. Thus, it can be expected that the TFRC, which is especially designed for streaming applications, could provide much lower variation of throughput as compared with TCP, and implement congestion control to protect against hampering other flows' transmission by comparison with UDP.

Since TFRC can provide better service for streaming multimedia transfer, adopting powerful methods to model and study it draws our concern. Surely, simulation tools that closely mirror software implementations of TFRC can be used to investigate the performance of TFRC. And to some extent, it may seem that we could simply observe the dynamics and operations of the protocol in a given situation without using mathematical models. However, several reasons make us move beyond the simulation approach and start using mathematical analysis to advance our understanding.

First, the sheer scale of the network system in which TFRC will operate is tremendous. As the goal of TFRC design is to support the media streams flowing smoothly in the Internet, it has to use an Internet-magnitude environment to simulate the operation of this congestion control algorithm. But evaluating the performance of TFRC in such large system would be difficult to implement. So definitely it is needed to resort to mathematical models. Second, there are many unknown environment in which TFRC may be operating. For instance, it is still not sure whether the TFRC and TCP flows can coexist in a stable and fair pattern. Also, we expect that the TFRC can be implemented in an optimized way, and the mathematical tools are effective to help to investigate it from the theoretical perspective.

Besides the advantage of mathematical analysis over the simulation approach, it is worthwhile considering the profound reasons for why we need the theoretical study on TFRC, or, in other words, why we would be so interested in modeling the congestion control algorithm of TFRC.

First of all, the mathematical models are fundamental and effective tools to capture the dynamics of congestion control algorithm. Since we intend to investigate the properties of TFRC scheme in depth in order to provide theoretical support for its implementations, the mathematical approaches enable us to observe the results without resorting to building up a real network composed of a large number of hosts.

In addition, the modeling analysis is also a persuasive way to explain how the algorithm works. We show that by adopting the powerful mathematical techniques, a compact model will help us to understand the significant aspects of this network system controlled by the TFRC algorithm, although it would be possible at the expense of stripping away some practical details.

Furthermore, beyond the understanding of algorithm, we can gain further insight into the behavior of TFRC through the use of mathematical modeling. If the mathematical analysis proves that the TFRC can meet the requirement of system, then we can apply the TFRC algorithm surely and responsibly in the Internet. Otherwise, we expect the modeling analysis could help us to find the essential problem which affects its performance. With such investigations, we expect to be capable of improving the design of algorithm to satisfy the service of network applications.

1.3 Thesis Contribution and Organization

Lacking of proper model of TFRC is the main reason of why there is little work on theoretical evaluation of TFRC. Therefore, the first step of our study is to provide a meaningful model for TFRC. In particular, we conjecture that TFRC is a dual algorithm solving the network utility maximization problem of which TCP is interpreted as a primal algorithm. In the dual algorithm, the sending rates are updated using deterministic functions, while the link feedbacks are generated according to differential equations. We will validate that such modeling is meaningful and matches TFRC's basic specification. Based on the TFRC models, we intend to propose combined models for coexistence case which involves both the primal and dual algorithms in one system.

Then, we will show that the TFRC, modeled as a dual algorithm, converges to the unique and stable optimal equilibrium of a network utility maximization problem. Also, we will investigate the fairness properties of the equilibrium. These results directly answer the first question on TFRC's stability. Moreover, we will investigate whether TCP and TFRC, modeled as primal algorithm and dual algorithm respectively, can converge to the unique equilibrium when they are coexisting. These results could answer the second question about the stability of TFRC coexisting with TCP.

To advance the investigation, we will analyze the rate of convergence, stochastic perturbations and delay stability of the system defined by the TFRC algorithm. These mathematical analysis provide theoretical support for determining the proper range of the value of TFRC's average factor and its corresponding round-trip delay in network. By discussing these aspects in depth, we gain more insight about the dynamics of TFRC. And the main contributions of this thesis are summarized in Table 1.2, where the items without reference are derived in this work.

Thus, the thesis is organized as follows. After the background study and survey on related work presented in Chapter 2, we will discuss the modeling about TFRC and its

coexistence with TCP in Chapter 3. The analysis for its stability, rates of convergence and stochastic perturbations are discussed in Chapter 4. To extend the discussion, we will explore the issue of delay stability in greater depth, study how the smoothing factor relate to the delay stability in Chapter 5 and Appendix A. And Chapter 6 illustrates the theoretical results with various simulations. Finally, we conclude the thesis with discussions in Chapter 7.

Table 1.2: Comparison of TCP, TFRC I and TFRC II.

	TCP	TFRC I	TFRC II
Global stability without delay	G.A.S. [4]	G.A.S.	G.A.S.
Local stability with delay	$\kappa_r T_r (\hat{q}_r + \sum_{j \in r} p_j \hat{y}_j) < \frac{\pi}{2}$ [12]	$\alpha T_r \left(\frac{1}{2\hat{q}_r} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2}$	$\alpha T_r \left(\frac{\hat{L}_r}{2} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2}$
Convergence rate	κ and P' [4]	α and G	α and G
Robustness to disturbance	$\frac{[\Gamma F F^T \Gamma^T]_{rs}}{\lambda_r + \lambda_s}$ [4]	$\frac{[\Gamma F F^T \Gamma^T]_{rs}}{\lambda_r + \lambda_s}$	$\frac{[\Gamma F F^T \Gamma^T]_{rs}}{\lambda_r + \lambda_s}$
Practical measure in implementation	Not applicable	Loss rate	Loss interval

□ End of chapter.

Chapter 2

Background Study

Summary

In this section, we investigate the details of TFRC congestion control algorithm to lay down a foundation for the following analysis. As the research background, the related work would be reviewed from reference literatures of congestion control algorithm and its modeling methodology. This survey makes the basis for our theoretical study.

2.1 TFRC

TFRC is a congestion control mechanism for unicast flows operating in the best-effort Internet environment [13]. As discussed in the previous chapter, the proposal of TFRC is related to the requirement of multimedia streaming applications in the Internet. These applications favor smoothed sending rate, and prefer to adopt transport protocol with congestion control to avoid harming the transmission of other flows in a network.

Since TCP is the dominant transport protocol in the Internet, any new congestion control mechanisms joining in the Internet should consider the coexistence problem with TCP. An acceptable strategy named TCP-friendly for a non-TCP flow aims to use no more bandwidth than a conformant TCP connection under the same conditions [7]. When competing for bandwidth with TCP flows, it is found that TFRC flows occupy reasonably fair resources under the steady state with much lower variation of throughput over time. This prominent advantage makes TFRC more suitable for

applications such as streaming media where a relatively smooth sending rate is of importance.

However, the tradeoff of having smoother throughput than TCP while competing fairly for bandwidth is that TFRC responds slower than TCP to the changes of available bandwidth. The investigation in the following chapters would show that its aggressiveness to available bandwidth and responsiveness to congestion indication is slower than that of TCP. Thus TFRC could be used when the application has a requirement of smoothed throughput, in particular, avoiding TCP's halving of the sending rate in response to a single packet drop [13].

Since TFRC is a proper candidate for media streams transfer, it is necessary to consider how it works. TFRC uses a model for steady state TCP throughput to limit the sending rate and assure its fair behavior against competing flows. As shown in [14], the steady state throughput of TCP flow is modeled as a function of the packet size S , steady state loss rate p , round-trip time T , and the TCP retransmit timeout value t_{RTO} :

$$x = \frac{S}{T\sqrt{\frac{2p}{3}} + t_{RTO}(3\sqrt{\frac{3p}{8}})p(1 + 32p^2)}, \quad (2.1)$$

which gives an upper bound on the sending rate of TFRC. With the deduction at steady state shown in [15], a simplified formulation of the TCP response function is given as

$$x = \frac{S \cdot c}{T\sqrt{p}}, \quad (2.2)$$

where c is a constant related to loss assumptions. This model is a simplification in that it does not take into account TCP timeout. In our study, we will adopt the equation (2.2) as the TFRC's transmission rate function. The difference between (2.1) and (2.2) becomes insignificant in the sense of theoretical modeling, especially for the case that the value of loss event rate p is small, which would not invoke the timeout of transmission. Also, the equation (2.2) is helpful for us to focus on the key parameters with dominant influence on a network system¹.

Instead of reacting to single congestion indication (in the form of packet loss) like TCP, TFRC changes its sending rate in response to the loss rate as shown in (2.1). In TFRC, the loss rate is estimated from loss intervals measured at receivers, which is defined as the number of packets between two loss events. In this way, the loss rate would decrease only in response to a new loss interval that is longer than the previous calculated one, and increase otherwise. Especially, the method of calculating

¹The equation (2.1) is adopted in the RFC documents of TFRC [13, 16] as its technical specification. From the modeling perspective, we use the equation (2.2) as TFRC's sending rate.

the loss rate is the key factor influencing the operation of TFRC, and is over a lot of discussion and evaluation. To provide an relatively smoothed loss rate, several methods are proposed including the Dynamic History Window method, the Average Loss Interval method, and the Exponentially Weighted Moving Average (EWMA) Loss Interval method.

The above three methods are all supported in NS2 code of TFRC, in which the Dynamic History Window method has two variances – RBPH and EBPH. Based on simulations, it is shown that the EWMA Loss Interval method and Average Loss Interval method perform better than the Dynamic History Window method. And on the other hand, since both the Average Loss Interval method and EWMA Loss Interval method adopt the weighted moving average idea to smooth the value of estimated loss rate in principle, we choose EWMA approach as our modeling object due to its simplicity and generality.

To sum up, the function of TFRC requires the loss rate must be calculated at the receiver with one of the moving average methods discussed above. When the sender gets the feedback information from receiver, it calculates the value of allowed transmission rate, then increases or decreases its sending rate accordingly. In order to perform the rate control, another parameter – the round-trip time – is also measured at the sender according to receiver’s feedback. Since no router support is necessary in the implementation, the TFRC can be readily deployed in today’s Internet.

2.2 Related Work

The requirement of TCP-friendly transport protocol for real-time video transmission was first indicated by Tan and Zakhor in [11]. Due to the retransmission latency of TCP and its abrupt changes in traffic pattern, it may noticeably reduce the user-perceived quality for real-time applications. As the result of real experiments in [17], it is confirmed that a smoothed rate generally is better for interactive video applications.

In order to provide proper service in a TCP-friendly manner for multimedia applications, a lot of transport mechanisms are designed, including RAP (Rate Adaption Protocol) [18], LDA+ (Loss-Delay Based Adaption Algorithm) [19], TEAR (TCP Emulation at the Receivers) [20], GAIMD (General AIMD) [21], IIAD (Inverse-Increase Additive-Decrease) [22], SIMD (Square Increase Multiplicative Decrease) [23], and TFRC etc. The survey on these TCP-friendly mechanisms and their characteristics are discussed in [8] and [24]. Advanced in [24], the authors classify eight typical TCP-friendly schemes according to their underlying policies on fairness, aggressiveness, and

responsiveness, and compare them with conclusion that TFRC has better behaviors under most scenarios than others on average by using the rate-based fairness and fixed-history responsiveness policies.

TFRC was first formally proposed in [1] by Floyd *et al.* as an equation-based congestion control scheme for unicast flows operating in the best-effort Internet environment. In its extended discussions [25, 26, 27] as well as the related RFC documents [13, 16], the technical details of this congestion control mechanism are presented. With numerous simulations and experiments, it is shown that TFRC performs well in the case that coexisting with TCP flows.

Comparing the performance of TFRC and that of standard TCP, where TCP with different parameters for AIMD's additive increase and multiplicative decrease, it is shown that TFRC changes its sending rate more smoothly than TCP over small and moderate timescales in [10]. By investigating the fairness, smoothness, responsiveness, and aggressiveness of TCP and three representative TCP-friendly congestion control protocols: GAIMD, TFRC, and TEAR, it is shown in [28] that TFRC has better fairness and smoothness performance at low loss rate than TCP and GAIMD, while TCP is the most responsive and aggressive in utilizing bandwidth. Also, for a stationary environment, it indicates that smoothness and fairness are positively correlated [28].

Although the benefit of TFRC relative to TCP is the smoothly-changing sending rate, the corresponding cost of TFRC is a much moderate response to transient changes in congestion status. Comparing the transient behaviors of various schemes including TCP and TFRC, Bansal *et al.* discussed the dynamic behavior of slowly-responsive congestion control algorithms in [29], which points out that most of the TCP-compatible algorithms, such as TFRC, appear to be safe for deployment even the more slowly responsive ones may cause high packet loss rates. With theoretical reasoning, Vojnovic and Boudec in [30] identified the conditions under which the throughput of TFRC is not larger than that of TCP's response function, and suggested that TFRC may experience a smaller loss rate than TCP in some special scenarios. In [31], Rhee and Xu discussed this problem with theoretical analysis, they provided two additional reasons for why TCP and TFRC have different average sending rates.

In practical design, to extend the TFRC to multicast applications, Widermer *et al.* proposed TCP-Friendly Multicast Congestion Control (TFMCC) [32, 33] by modifying the corresponding transmission feedback mechanism. On the other hand, the latest development of TFRC includes its implementation provided by Datagram Congestion Control Protocol (DCCP) [34], which is a transport level protocol designed for applications that don't need the data retransmission but want congestion control.

Evidently, TFRC already becomes an appropriate candidate for multimedia streams transfer.

Although there are a lot of work empirically studying TFRC, little work has been done on evaluating its performance from theoretical aspect. One goal of our study is to model the congestion control algorithm of TFRC, lay down the theoretical foundation for its deployment in the Internet. By tracking an reverse-engineering way, we intend to understand the dynamic behavior of the system defended by TFRC algorithm. More significant for the practical implementation of TFRC, we are interested in whether or not it can coexist with TCP, which is the dominant transport mechanism in networks.

Since the theoretical modeling on TCP may provide us experience on adopted mathematical tools, we first review the related theoretical study of TCP congestion control algorithm. TCP's stability and fairness performance have already been extensively evaluated in both practice and theory [4, 5, 35]. As introduced by Kelly *et al.*, congestion control schemes can be viewed as algorithms to solve an optimization problem, which maximizes the aggregate system utility function subject to link capacity constrains in a network.

These algorithms can be categorized into two classes: primal algorithms and dual algorithms. In both algorithms, there are primal variables – sending rates, and dual variables – link feedbacks (i.e. packet loss). In the primal algorithm, the primal variables are dynamically updated according to differential equations, and the link feedbacks are updated using deterministic functions [4]. In the dual algorithm, on the other hand, the primal variables are updated using deterministic functions, while the link feedbacks are generated according to differential equations [4, 36]. When both source and link updates are dynamic in the so-called primal-dual algorithm [37], stability has been studied by adopting singular perturbations method in [38]. And in [39], Wen *et al.* developed a unifying framework for network flow control by using a passivity approach which encompasses these stability results of primal-dual algorithm as special cases.

To understand the delay influence on practical networks, people studied the delay stable conditions for TCP in [40, 41]. In particular, the problem of the local asymptotical stability under propagation delays was firstly studied by Johar *et al.* in [40]. It is derived that the stability of TCP congestion control algorithm is related to the round-trip delay and its gain factor. The analysis results also suggest that TCP as usually implemented is likely to be prone to instabilities when the congestion window is small, and overly sluggish when it is large. In [41], Vinnicombe proposed a graphic method based on the generalized Nyquist criterion to investigate the delay system's

stability and get similar results. Altogether, the analysis approaches used in these discussions provide significant guidance and reference for our study about TFRC.

On the other hand, there exists work [42] studying the network system consists of heterogeneous congestion control protocols reacting to different pricing signals, such as TCP Reno in response to loss probability and TCP Vegas in response to queueing delay. The resulting equilibrium is difficult to be interpreted as a solution to the standard utility maximization problem, it is due to the Lagrange multiplier at link can not be uniquely determined by the congestion price of loss probability as used in [4]. In a recent paper [42], Tang *et al.* has proved the existence of equilibrium in general multiprotocol networks under mild assumptions. By adopting the Poincare-Hopf index theorem, it is shown that the equilibria are locally unique, finite, and odd in number, and they cannot all be locally stable unless there is a globally unique equilibrium. In addition, if the “degree of heterogeneity” is sufficiently small, the global uniqueness of network equilibrium could be guaranteed. However, Their work focuses on discussing the equilibrium existence for the network consisting of different type of TCP protocols, but does not study stability of the equilibrium. In this thesis, we study the case of TCP and TFRC coexistence. We show that this coexistence system has a unique equilibrium and prove the global stability of the equilibrium. We also study the local stability around the equilibrium when the system involves communication delay.

Chapter 3

Network Modeling

Summary

In this chapter, based on reviewing the network utility maximization framework and its TCP Reno modeling, we propose our models for the TFRC congestion control algorithm. According to the implementation of TFRC scheme, two variant versions concerning the loss rate and loss interval respectively are described for the purpose of their following theoretical analysis. Also, in order to investigate the TFRC's compatibility with TCP, we propose combined models consisting of both primal and dual algorithms for the coexistence case. These mathematical models will be given corresponding analysis in Chapter 4 and Chapter 5.

3.1 Network Utility Maximization Framework

Consider a network with a set J of *links (resources)*, and each link j ($j \in J$) has a finite capacity C_j . Let R be the set of *routes (sources)*, in which a route r ($r \in R$) is a non-empty subset of J ($r \subset J$) and consists of several connected links. By associating a route with a user, the user r is endowed with an utility $U_r(x_r)$ when its sending rate on the route r is x_r ($x_r \geq 0$). For elastic traffic [43], it can be assumed that the utility $U_r(x_r)$ is an increasingly, strictly concave and continuously differentiable function of x_r .

Based on this network setting, one resource allocation problem arises: how to allocate the limited network resources (bandwidth of links) to maximize the sum of

all users' utilities? Consider it as an optimization problem with the form:

$$\max_{x \geq 0} \sum_{r \in R} U_r(x_r) \quad (3.1)$$

subject to

$$\sum_{s: j \in s} x_s \leq C_j, \quad j \in J \quad (3.2)$$

where $\sum_{s: j \in s} x_s$ represents the aggregate rate arriving at link j . For convenience, define x to be a vector of users' sending rates, $x = (x_r, r \in R)$. In the following two sections, both primal and dual approaches are adopted to get distributed solutions for this problem.

3.1.1 Primal Algorithm

To solve the optimization problem (3.1)-(3.2) with a tractable algorithm, a penalty relaxation approach is introduced as in [4]. By associating a cost with overshooting the link capacity, the original problem transforms into a concave optimization problem maximizing the aggregate net utility:

$$\max_{x \geq 0} \sum_{r \in R} U_r(x_r) - \sum_{j \in J} P_j\left(\sum_{s: j \in s} x_s\right), \quad (3.3)$$

where the cost function

$$P_j(z) = \int_0^z p_j(y) dy, \quad (3.4)$$

and $p_j(y)$ is the price for sending traffic at rate y on the link j , which is assumed to be a non-negative and non-decreasing function, such that $P_j(z) \rightarrow \infty$ as $z \rightarrow \infty$. It states that as the load on a link increase, the penalty cost on this link does not decrease and would be non-zero for large exceeded rates.

Consider this unconstrained optimization problem (3.3), a natural candidate is the gradient ascent algorithm. Specifically, the direction of ascent can be gotten by differentiating the equation (3.3) with respect to x_r . Accordingly, the primal algorithm can take the general form:

$$\dot{x}_r = \kappa_r(x_r) \left(U'_r(x_r) - \sum_{j \in R} p_j\left(\sum_{s: j \in s} x_s\right) \right), \quad x_r \geq 0 \quad (3.5)$$

where $\kappa_r(x_r) > 0$ is a scaling parameter which controls the amount of change in the direction of the gradient.

Note that the primal algorithm (3.5) arises from the primal formulation of the utility maximization problem. It manifests that the source r updates its rate x_r by

collecting the link prices p_j along its route, where p_j is determined by the aggregate incoming rate $\sum_{s:j \in s} x_s(t)$ at the link j . That is to say, all information the source r needs to know is the sum of link prices on its route, and all that the link j needs to know is the aggregate arrival flow rates on it. Therefore, this primal algorithm provides a decentralized solution for the utility maximization problem.

3.1.2 Dual Algorithm

For the optimization problem (3.1)-(3.2), dual approach gives another form of distributed and decentralized solution [36]. Define the Lagrangian

$$L(x, p) = \sum_{r \in R} U_r(x_r) - \sum_{j \in J} p_j \left(\sum_{s:j \in s} x_s - C_j \right) \quad (3.6)$$

$$= \sum_{j \in J} p_j C_j + \sum_{r \in R} \left(U_r(x_r) - x_r \sum_{j \in r} p_j \right), \quad (3.7)$$

where p_j represents the price per unit flow as a Lagrange multiplier associated with the j th constrain (3.2), and $p = (p_j, j \in J)$ is the vector of dual variables. In practical networks, the link price p_j implies the congestion status of the link j . Then,

$$\frac{\partial L}{\partial x_r} = U'_r(x_r) - \sum_{j \in r} p_j \quad (3.8)$$

and the unique optimum to the primal problem is given by

$$x_r = U_r'^{-1} \left(\sum_{j \in r} p_j \right). \quad (3.9)$$

Furthermore, the dual problem becomes

$$\max_{p \geq 0} \sum_{r \in R} \left(x_r \sum_{j \in r} p_j - U_r(x_r) \right) - \sum_{j \in J} p_j C_j \quad (3.10)$$

where x_r takes its optimum (3.9). Consider a system with link prices vary gradually:

$$\dot{p}_j = \alpha \left(\sum_{s:j \in s} x_s(t) - \eta_j(p_j(t)) \right), \quad p_j \geq 0 \quad (3.11)$$

where $\alpha > 0$ is a gain factor affecting the update of p_j , and rates x_r given as function (3.9). Suppose that $\eta_j(p_j(t))$ is the flow through link j which generates a price of $p_j(t)$ at time t , then the right hand side of (3.11) can be described as the excess demand at prices $p_j(t)$. In [4], it is shown the dual algorithm solves the network's optimization problem.

Since the objective function in (3.1) is continuous and concave over the constraint set determined by affine functions in (3.2), there is no duality gap between the primal and dual problems. Moreover, it is shown in (3.9)-(3.11) that source r and link j update their values in a distributed manner. Thus, the dual algorithm (3.9)-(3.11) provides a decentralized solution for the optimization problem (3.1)-(3.2).

3.2 Overview of TCP Reno Modeling

The discussion in Section 3.1 studies the network utility maximization problem, and explores its solutions. In practical networks, there already exist successful algorithms work well. From the theoretical perspective, people concerns why these algorithms can operate properly, and how to improve them. This is a reverse-engineering study helping people to understand and improve the designed system. The dynamic of TCP congestion control can be understood in this way.

For TCP Reno, one specific version of TCP, Kelly has shown its primal-like algorithm in [35] as:

$$\dot{x}_r = \frac{x_r^2}{2S} \left(\frac{2S^2}{T_r^2 x_r^2} - \sum_{j \in r} p_j(y_j) \right), \quad r \in R \quad (3.12)$$

in which S is the TCP packet size, T_r is the round-trip time for the connection of user r , and $y_j(t) = \sum_{s:j \in s} x_s(t)$ represents the incoming rate on the link j . Here, the link price¹ p_j indicates the “packet loss rate” at link j :

$$p_j(y_j) = \frac{(y_j - C_j)^+}{y_j}, \quad j \in J \quad (3.13)$$

where $(\cdot)^+$ denotes

$$(f(z))^+ = \begin{cases} f(z), & \text{if } f(z) > 0; \\ 0, & \text{otherwise.} \end{cases} \quad (3.14)$$

The algorithm (3.12)-(3.13) solves the optimization problem in (3.3) with utility function $U_r(x_r) = -\frac{2S^2}{T_r^2 x_r}$. Consequently, the utility maximization problem for TCP Reno network can be written as

$$\max_{x \geq 0} \left(- \sum_{r \in R} \frac{2S^2}{T_r^2 x_r} - \sum_{j \in J} \int_0^{\sum_{s:j \in s} x_s} p_j(y) dy \right). \quad (3.15)$$

¹It can be deduced that the cost at link j is

$$P_j(z) = \int_0^z \frac{(y - C_j)^+}{y} dy = \int_{C_j}^z \left(1 - \frac{C_j}{y}\right) dy = z - C_j \left(1 + \ln \frac{z}{C_j}\right), \quad z > C_j$$

which is a convex, continuous function of z in its domain.

Since the objective function of the optimization problem (3.15) is strictly concave on $x \geq 0$ with an interior maximum, the maximizing value of x is thus unique.

From the nonlinear system viewpoint, this primal-like algorithm (3.12)-(3.13) can be viewed as a feedback control system: user r input its sending rate $x_r(t)$ into the network, and adjusts $x_r(t)$ dynamically according to the feedback $p_j(t)$. With theoretical analysis in [5], Kelly showed that the system (3.12)-(3.13) has a unique equilibrium

$$\hat{x}_r = \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} p_j(\hat{y}_j)}}, \quad r \in R \quad (3.16)$$

to which all trajectories converge. This unique stable point of the system is also the optimal solution for the optimization problem (3.15). At the equilibrium, the aggregate net utility is maximized, and all users are *weighted α -fair* [6] to each other with $\alpha = 2$. Note that the equation (3.16) is similar to the TCP steady state throughput equation as described in [14].

3.3 Modeling TFRC

Primal-like algorithm modeling TCP Reno (3.12) provides a solution for the network utility maximization problem (3.15). Concerning the same optimization problem, we model TFRC from the perspective of dual approach. In this dual-like algorithm, the primal variables – sending rates are updated using deterministic functions, while the dual variables – link congestion feedbacks are generated according to differential equations.

In the TFRC protocol, receivers are required to feedback the information about transmission loss rate it calculated; based on this reply, sender regulates its sending rate accordingly. It cannot be emphasized too strongly that the calculation of the loss rate is one of the critical parts in TFRC, and the part that has been through a lot of design iteration. High-levelly speaking, the key design purpose is to get a smoothed loss rate in order to reduce throughput fluctuation, which is regarded as the TFRC Model I in our study. While taking the protocol specification into consideration, the loss rate is calculated by converting the loss interval detected at receiver, we consider it in the TFRC Model II.

The proposal of the TFRC Model II is based on the mechanism details about detecting loss rate in the TFRC. Because the loss rate in practical may involves different methods to estimate or measure, we consider its detection by inverting loss interval in the TFRC Model II according to the TFRC protocol [16]. Our purpose is to study

whether the estimation of loss rate from loss interval could affect the operation of TFRC congestion control algorithm. To make their difference more noticeable, we can say the TFRC Model I assumes the loss rate is acquirable and focus on its dynamics, while the TFRC Model II introduces the realization details of protocol related to the loss interval.

3.3.1 TFRC Model I

Adopting the same model settings described in Section 3.1, we consider a TFRC network composed of J links (*resources*) and R routes (*sources*). Each link j has a finite link capacity C_j , and each route r is a non-empty subset of J , consisting of several connected links. Associating a route with a user, each user r sends packets in the rate $x_r \geq 0$, which is regulated according to the TFRC congestion control algorithm.

By considering the compatibility with TCP flows, TFRC adopts TCP response function as a control equation for its sending rate. Such that in the steady state, a TFRC flow uses no more bandwidth than a corresponding TCP running under comparable conditions. For user r in a TFRC network, its sending rate x_r can be expressed as follows:

$$x_r = \frac{\sqrt{2}S}{T_r\sqrt{q_r}}, \quad (3.17)$$

where S is the packet size, T_r is the round-trip time for the connection of user r , and q_r is the measure of the network congestion along the route r . It shows that the transmission rate x_r is depending on the feedback information q_r , which is received from the network regarding the traffic load on links along the route r .

The loss measurement of TFRC considers the history of loss events, the older the history, the less effects on the current loss measurement. We model it with an exponential moving average method. For user r , q_r is updated as follows:

$$\dot{q}_r = \alpha \left(\sum_{j \in r} \frac{(y_j - C_j)^+}{y_j} - q_r \right), \quad (3.18)$$

where $\alpha > 0$ is a constant, and $y_j = \sum_{s: j \in s} x_s$ is the aggregate rate passing through link j . The smoothed $q_r(t)$ is a simple weighted average of the latest observation $\sum_{j \in r} \frac{(y_j - C_j)^+}{y_j}$ and the previous smoothed result. Note that the factor α influences the smoothing issue: the greater values of α have less of smoothing effect and give greater weight to recent changes, while values of α close to zero have a greater smoothing effect and are less responsive to recent changes. Thus, α is the smoothing factor impacting the throughput fluctuation of TFRC.

From the viewpoint of nonlinear system, these two equations (3.17)-(3.18) define the dynamics of the system in which all users are running TFRC. The system state vector is (x, q) , where $x = [x_1, x_2, \dots]^T$ and $q = [q_1, q_2, \dots]^T$.

Meanwhile, the above dynamical equation of q_r (3.18) can also be interpreted as consequence of the following relation and dynamics:

$$q_r = \sum_{j \in r} p_j, \quad \dot{p}_j = \alpha \left(\frac{(y_j - C_j)^+}{y_j} - p_j \right). \quad (3.19)$$

It is shown that the source r needs to know the sum of the prices of each link on its route in order to adjust its sending rate. And the resource j calculates its price based on the aggregated rates of all sources whose flows pass through the link j . To model this feedback concisely, we introduce a 0 – 1 matrix $A = (a_{jr}, j \in J, r \in R)$ which is called the routing matrix of the network. Set $a_{jr} = 1$ if $j \in r$, that is, link j lies on route r , and set $a_{jr} = 0$ otherwise. Using the elements of the routing matrix, y_j and q_r can be written as:

$$y_j = \sum_{s: j \in s} a_{js} x_s, \quad q_r = \sum_{j \in r} a_{jr} p_j. \quad (3.20)$$

Letting y be the vector of all y_j ($j \in J$), p be the vector of all link prices, and q be the vector of all route prices, we have

$$y = Ax, \quad q = A^T p. \quad (3.21)$$

As such, dynamics of the system where all users are running TFRC can be studied by investigating a nonlinear system with system states being (x, p) and dynamics described by the following differential equations:

$$\begin{cases} \dot{x}_r = \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} p_j}}, & \forall r \in R, \\ \dot{p}_j = \alpha \left(\frac{(y_j - C_j)^+}{y_j} - p_j \right), & \forall j \in J. \end{cases} \quad (3.22)$$

This is because, given the same initial state $(x(0), q(0))$ for the original system, its dynamics must be exactly the same as some dynamics of the system in (3.22) with certain initial state $(x(0), p(0))$. Therefore, the dynamics of the original system is a subset of the dynamics of the system in (3.22). Consequently, if the system in (3.22) has unique stable equilibrium, then so does the original system. And the corresponding analysis will be discussed in Chapter 4 and Chapter 5.

3.3.2 TFRC Model II

While the TFRC Model I (3.22) considers the principle of TFRC algorithm based on the dynamic of loss rate, it does not take into consideration the specification of

the mechanism on detecting the loss rate. In practical networks, the TFRC receiver usually calculates the loss rate along one route indirectly from the observation of the detected loss interval. In general, the loss rate can be approximated by the reciprocal of loss interval, which is the number of packets between two loss events. By involving this mechanism details of loss rate conversion, we propose the TFRC Model II in terms of the loss interval.

By taking the same network settings as the TFRC Model I, we model the TFRC concerning its update of loss interval at the end user:

$$\dot{l}_r = \alpha \left(\frac{1}{\sum_{j \in r} \frac{(y_j - C_j)^+}{y_j}} - l_r \right) \quad (3.23)$$

where l_r denotes the loss interval on the route r , and α is a constant ($\alpha > 0$): the smaller the α , the greater smoothing effect but less responsiveness for l_r . Recall that $\sum_{j \in r} \frac{(y_j - C_j)^+}{y_j}$ is the loss rate along the route r , its inverse thus could indicate the route loss interval for the user r when loss occurs. Because the update of loss interval involves its history information in order to get smoothed value, we model it using an exponential weighted moving average method in (3.23).

Upon receiving the smoothed result about l_r as the feedback information from the receiver is obtained, user r updates its sending rate according to

$$x_r = \frac{\sqrt{2}S}{T_r \sqrt{\frac{1}{l_r}}} = \frac{\sqrt{2}S}{T_r} \sqrt{l_r}, \quad (3.24)$$

in which $\frac{1}{l_r}$ indicates the loss rate along the route r . Note that the estimated loss rate should decrease only in response to a new loss interval that is longer than the previously calculated average, or a sufficiently long interval since the last loss event, correspondingly the sending rate would increase.

According to the discussion in Section 3.1, it presents a dual controller to solve the network utility maximization problem. Viewed in this light, the dual-like algorithm of TFRC (3.23)-(3.24) can be regarded as a nonlinear system with system states (x, l) whose dynamic can be described by the following differential equations:

$$\begin{cases} \dot{x}_r = \frac{\sqrt{2}S}{T_r} \sqrt{l_r}, \\ \dot{l}_r = \alpha \left(\frac{1}{\sum_{j \in r} \frac{(y_j - C_j)^+}{y_j}} - l_r \right), \end{cases} \quad \forall r \in R. \quad (3.25)$$

Concerning the TFRC Model II, we intend to analyse its stability, and especially the stability condition for the practical case with transmission delay. As the comparison

with the TFRC Model I, the difference between two TFRC models would also be investigated in our study. The main questions we are interested in are whether the system defined by TFRC rate control algorithm is stable, and what is the major difference of modeling it from loss rate and loss interval perspectives. These questions would be taken up in Chapter 4 and Chapter 5.

3.4 Modeling Coexistence Case

As discussed in the previous sections, the models of TCP Reno and TFRC are presented as the primal-like algorithm and dual-like algorithm respectively. Since TCP has already been widely applied in today's networks, any new protocols participating in the network should consider its coexistence with TCP. Thus, it is of importance for practical networks to study whether TFRC can coexist with TCP. In this section, we will propose two models for this coexistence case, laying down a foundation for the following analysis.

Consider a network contains both TCP Reno and TFRC flows, we can model this system by combining the primal-like algorithm of TCP Reno and the dual-like algorithm of TFRC together. According to the two TFRC models we proposed in previous sections, the coexistence case would also have two corresponding models as shown in the following.

TCP Reno Coexisting with TFRC I

The coexistence case involves two types of flows, TCP Reno and TFRC, competing the network resources at bottleneck links. To study their dynamic behavior, we first propose a network model for the TCP Reno coexisting with the TFRC I based on their respective modeling settings.

This network can be described via J links, and R users (routes) which consists of several connected links. Each link j ($j \in J$) has a finite capacity C_j , and each route r ($r \in R$) associates a positive round trip time delay T_r . To identify the TCP Reno and TFRC flows, we adopt the notation x_{r_1} to represent the sending rate for users running TCP Reno, and x_{r_2} for users running TFRC. Due to the different feedback information about the link price considered in TCP Reno and TFRC, we use $p_{j,1}$ and $p_{j,2}$ to denote the loss rate at link j recognized by TCP Reno and TFRC users respectively.

Thus, based on the TCP Reno and TFRC modeling discussed in the previous sections, the network containing both TCP Reno and TFRC flows could be described

by the following differential equation set:

$$\begin{cases} \dot{x}_{r_1} = \frac{x_{r_1}^2}{2S} \left(\frac{2S^2}{T_{r_1}^2 x_{r_1}^2} - \sum_{j \in r_1} p_{j,1} \right), & r_1 \in R_1 \\ p_{j,1} = \frac{(y_j - C_j)^+}{y_j}, & j \in J \\ x_{r_2} = \frac{\sqrt{2S}}{T_{r_2} \sqrt{\sum_{j \in r_2} p_{j,2}}}, & r_2 \in R_2 \\ \dot{p}_{j,2} = \alpha \left(\frac{(y_j - C_j)^+}{y_j} - p_{j,2} \right), & j \in J \end{cases} \quad (3.26)$$

in which x_{r_1} adjusts its sending rate according to the congestion control algorithm of TCP Reno, and $p_{j,2}$ updates its link price following the dynamic of TFRC algorithm. With this mathematical model of two types of congestion control algorithms, we intend to investigate the stability property of the nonlinear system (3.26) in Chapter 4, in order to study whether TCP Reno flows can coexist with TFRC flows. If it can be proved the system of equations (3.26) is stable, these two kinds of flows will be capable of coexisting in one network.

Note that the aggregated rate y_j at link j may contain two types of flows. For TCP Reno, its rate update associates with the feedback link price $p_{j,1}$, which is the packet loss rate at link j . While TFRC users adopt the EWMA method to smooth its link price $p_{j,2}$, and use this value to determine the sending rate according to the TCP response function. That is the different feedbacks for two type of receiver to adjust their sending rate.

TCP Reno Coexisting with TFRC II

For the TFRC Model II (3.25), we study its coexistence case with TCP Reno as following.

Consider a network with J links, and R users (routes), in which C_j ($j \in J$) is the finite capacity of link j and T_r ($r \in R$) is the round-trip time for user r . To distinguish the TCP Reno and TFRC flows, the notations x_{r_1} and x_{r_2} are adopted to represent the sending rate for users running TCP Reno and users running TFRC respectively. Due to the different strategies that TCP Reno and TFRC uses to get the feedback information about link congestion, we use $p_{j,1}$ to denote the loss rate at link j recognized by TCP Reno, and l_{r_2} to represent the calculated loss interval along route r_2 for TFRC user.

For the system consisting of the primal-like algorithm of TCP Reno and the dual-

like algorithm of TFRC Model II, its dynamic can be described as:

$$\begin{cases} \dot{x}_{r_1} = \frac{x_{r_1}^2}{2S} \left(\frac{2S^2}{T_{r_1}^2 x_{r_1}^2} - \sum_{j \in r_1} p_j \right), & r_1 \in R_1 \\ p_j = \frac{(y_j - C_j)^+}{y_j}, & j \in J \\ x_{r_2} = \frac{\sqrt{2}S}{T_{r_2}} \sqrt{l_{r_2}}, & r_2 \in R_2 \\ \dot{l}_{r_2} = \alpha \left(\frac{1}{\sum_{j \in r_2} p_j} - l_{r_2} \right), & r_2 \in R_2 \end{cases} \quad (3.27)$$

in which user r_1 adjusts its sending rate according to the congestion control algorithm of TCP Reno, and the TFRC user r_2 calculates its sending rate in terms of a deterministic equation with dynamically updated route price l_{r_2} .

It is shown in the system (3.26) and (3.27) that TCP Reno is presented as a primal-like algorithm, and TFRC is expressed as a dual-like algorithm according to their each own dynamics. An extended problem for more general case would be: if a primal problem and its dual problem can be proved to be stable, can it be expected that their coexistence case by combining both algorithms would also work stably? However, we only focus the specific case for TCP Reno coexisting with TFRC in our study. The general one could be discussed in the future work.

From the theoretical perspective, these concise mathematical models make it possible for us to investigate the feasibility of two kinds of flows coexisting in one network. In the following analysis, we shall consider the stability of these systems representing the coexistence case, which is an essential investigation for applying TFRC in the practical networks.

Summary

Network modeling for TFRC algorithm is the primary task in this chapter. By reviewing the network utility maximization framework, we get the sense for modeling congestion control algorithms from the optimization perspective. Essentially, it could be regarded as the resource allocation problem solving through primal or dual approaches. For TFRC, we model it as a dual-like algorithm according to its implementations in protocol. Since the loss rate detection involves some realization details, we propose the TFRC Model I and TFRC Model II by concerning the mechanism principle and the protocol specification respectively.

Chapter 4

Stability Analysis

Summary

In this chapter, we will investigate the nonlinear system defined by TFRC algorithm in terms of stability, rate of convergence, and robustness of stochastic perturbations. By theoretical analysis, it shows that the TFRC congestion control algorithm is globally stable at equilibrium, and its robustness to perturbations is determined by the smoothing factor. In addition, we establish the evidence of global stability for the coexistence system consisting of both TCP Reno and TFRC flows. These investigations provide theoretical support for the implementation of TFRC in practice.

4.1 TFRC Network

The goal of stability study for TFRC is to ensure its deployment in the Internet, specifically, the available bandwidth within a network can be shared fairly and stably by users. In this section, we show that stability is established by Lyapunov functions for the dynamical systems defined by TFRC congestion control algorithm. As an engineering issue, stability analysis requires the investigations of convergence rate and stochastic effect around the stable point, in order to study the system robustness under perturbations.

4.1.1 Global Stability

For a system whose dynamic is described by differential equations, the first question to ask is whether the system has any equilibrium, and if so how many. The second question is that whether the equilibrium is stable, and further, locally or globally.

These two questions are important because they describe the behavior of the system as time evolves, predicting the system's performance in actual operation. Specifically, the existence and uniqueness of equilibrium indicates the system characteristic in steady state. And the global stability shows the system can move into these equilibrium from any initial conditions. In the following parts, we study these questions of two dynamical systems representing the TFRC algorithm, says Model I (3.22) and Model II (3.25), respectively.

TFRC I

We now show that the system (3.22) defined by TFRC algorithm is globally stable with a unique equilibrium, and the dual-like algorithm solves a concave optimization problem. For convenience, we describe the system (3.22) with the following state model:

$$x_r = \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} p_j}}, \quad \forall r \in R \quad (4.1)$$

$$\dot{p}_j = \alpha (f_j(y_j) - p_j), \quad \forall j \in J \quad (4.2)$$

where the function

$$f_j(y_j) = \frac{(y_j - C_j)^+}{y_j} = \begin{cases} 1 - \frac{C_j}{y_j}, & y_j > C_j \\ 0, & y_j \leq C_j \end{cases} \quad (4.3)$$

which is a non-negative, continuous, increasing function of y_j , and $y_j = \sum_{s:j \in s} x_s$ denotes the arriving rate at link j .

By taking the derivative of the function (4.1), we obtain

$$\dot{x}_r = -\frac{1}{2} \frac{\sqrt{2}S}{T_r \left(\sum_{j \in r} p_j \right)^{\frac{3}{2}}} \sum_{j \in r} \dot{p}_j \quad (4.4)$$

$$= \alpha \frac{T_r^2 x_r^3}{4S^2} \left(\frac{2S^2}{T_r^2 x_r^2} - \sum_{j \in r} f_j(y_j) \right), \quad (4.5)$$

which shows a differential equation possessing the similar form of formulation as (3.5). From the perspective of utility maximization framework, it is expected the dual-like

algorithm of TFRC solves the same optimization problem as the TCP Reno (3.12)-(3.13). Recall the network utility maximization problem

$$\max_{x \geq 0} \left(- \sum_{r \in R} \frac{2S^2}{T_r^2 x_r} - \sum_{j \in J} \int_0^{y_j} f(z) dz \right), \quad (4.6)$$

whose objective function is strictly concave on $x \geq 0$ due to the strictly concavity of its components $-\frac{2S^2}{T_r^2 x_r}$ and $-\int_0^{y_j} f(z)$. Thus, this problem attains its maximum value at a unique point \hat{x} , which can be determined by setting the derivative of the objective function to zero. On the other hand, according to the relationship (4.1) between \hat{x} and \hat{p} , it has $\frac{2S^2}{T_r^2 \hat{x}_r^2} = \sum_{j \in r} \hat{p}_j$ at steady state, such that

$$\hat{p}_j = f_j(\hat{y}_j) = \frac{(\sum_{s: j \in s} \hat{x}_s - C_j)^+}{\sum_{s: j \in s} \hat{x}_s}, \quad j \in J \quad (4.7)$$

which represents the equilibrium of the dynamic system in equation (4.2). Due to the mapping of \hat{x} and \hat{p} , the dynamic system has a unique equilibrium.

Given the system in (4.1)-(4.2) has the unique optimal equilibrium, we study the stability of the equilibrium through the following theorem.

Theorem 4.1.1. *The strictly concave function*

$$V(x) = - \sum_{r \in R} \frac{2S^2}{T_r^2 x_r} - \sum_{j \in J} \int_0^{y_j} f(z) dz, \quad (4.8)$$

is a Lyapunov function for the system of differential equations (4.1)-(4.2). The unique value x maximizing $V(x)$ is a stable point of the system, to which all trajectories converge.

Proof. Since that $-\frac{2S^2}{T_r^2 x_r}$ is a strictly concave function of x_r , and

$$\int_0^{y_j} f(z) dz = \int_0^{y_j} \frac{(z - C_j)^+}{z} dy = y_j - C_j \left(1 + \ln \frac{y_j}{C_j} \right) \quad (4.9)$$

is a strictly convex function of y_j , the function $V(x)$ is thus a strictly concave function of x on $x > 0$. It ensures $V(x)$ has a global maximum at the unique point \hat{x} .

Due to the property

$$\int_0^{y_j} f(z) dz \rightarrow \infty \quad \text{as } y_j \rightarrow \infty, \quad (4.10)$$

it satisfies the condition of global stability: $V(x) \rightarrow \infty$ if $|x| \rightarrow \infty$.

Observe that

$$\frac{\partial V}{\partial x_r} = \frac{2S^2}{T_r^2 x_r^2} - \sum_{j \in r} f_j(y_j), \quad (4.11)$$

by setting these derivatives to zero, we can identify the value of \hat{x} maximizing $V(x)$. Further, its time derivative satisfies

$$\dot{V} = \sum_{r \in R} \frac{\partial V}{\partial x_r} \dot{x}_r \quad (4.12)$$

$$= \alpha \sum_{r \in R} \frac{T_r^2 x_r^3}{4S^2} \left(\frac{2S^2}{T_r^2 x_r^2} - \sum_{j \in r} f_j(y_j) \right)^2 > 0, \quad (4.13)$$

which establishes that $V(x)$ is strictly increasing with time t , unless $x(t) = \hat{x}$, the unique value maximizing $V(x)$. Thus, the function (4.8) is a Lyapunov function for the system (4.1)-(4.2), which has a unique equilibrium, and is globally asymptotically stable. \square

TFRC II

By adopting the similar analysis procedure as above, we study the system (3.25) with the following state model: $\forall r \in R$

$$x_r = \frac{\sqrt{2}S}{T_r} \sqrt{l_r}, \quad (4.14)$$

$$\dot{l}_r = \alpha \left(\frac{1}{\sum_{j \in r} f(y_j)} - l_r \right), \quad (4.15)$$

where $\sum_{j \in r} f_j(y_j) = \sum_{j \in r} \frac{(y_j - C_j)^+}{y_j}$ indicates the route congestion for the user r , and its reciprocal represents the loss interval detected at the end user.

Consider the derivative of the sending rate (4.14), it has

$$\dot{x}_r = \frac{\sqrt{2}S}{2T_r \sqrt{l_r}} \dot{l}_r \quad (4.16)$$

$$= \alpha \frac{S^2}{T_r^2 x_r} \left(\frac{1}{\sum_{j \in r} f(y_j)} - \frac{T_r^2 x_r^2}{2S^2} \right). \quad (4.17)$$

Recall that TFRC solves the same optimization problem as TCP Reno, which is

$$\max_{x \geq 0} \left(- \sum_{r \in R} \frac{2S^2}{T_r^2 x_r} - \sum_{j \in J} \int_0^{y_j} f(z) dz \right). \quad (4.18)$$

Since the objective function is strictly concave on $x \geq 0$, there exist the unique equilibrium \hat{x} and corresponding \hat{l} for the system (4.14)-(4.15), as follows

$$\hat{l}_r = \frac{1}{\sum_{j \in r} \frac{(\hat{y}_j - C_j)^+}{\hat{y}_j}}, \quad (4.19)$$

in which $\hat{y}_j = \sum_{s:j \in s} \hat{x}_s$.

To investigate the global stability of the equilibrium, we consider the following theorem.

Theorem 4.1.2. *The strictly concave function*

$$V = - \sum_{r \in R} \frac{2S^2}{T_r^2 x_r} - \sum_{j \in J} \int_0^{y_j} f(z) dz \quad (4.20)$$

is a Lyapunov function for the system of differential equations (4.14)-(4.15), and hence the unique value x maximizing $V(x)$ is a stable point of the system, to which all trajectories converge.

Proof. Due to the strictly concavity of $V(x)$, it would have a global maximum at a unique point. The assumptions ensures that $V(x)$ is strictly concave on $x \geq 0$ with an interior maximum; the maximizing value of x is thus unique.

Observe that

$$\frac{\partial V}{\partial x_r} = \frac{2S^2}{T_r^2 x_r^2} - \sum_{j \in r} f_j(y_j), \quad (4.21)$$

setting these derivatives to zero identifies the maximum, and

$$\dot{V} = \sum_{r \in R} \frac{\partial V}{\partial x_r} \dot{x}_r \quad (4.22)$$

$$= -\alpha \sum_{r \in R} \frac{S^2}{T_r^2 x_r} \left(\frac{2S^2}{T_r^2 x_r^2} - \sum_{j \in r} f(y_j) \right) \left(\frac{T_r^2 x_r^2}{2S^2} - \frac{1}{\sum_{j \in r} f(y_j)} \right). \quad (4.23)$$

The last two brackets items can be taken as a general form:

$$(h(\xi) - h(\eta))(\xi - \eta), \quad (4.24)$$

which takes negative value by the reason that $h(\xi) = \frac{1}{\xi}$ is a decreasing function on $\xi \geq 0$. Thus, it establishes that $V(x)$ is strictly increasing with t , unless $x(t) = \hat{x}$, the unique value maximizing $V(x)$.

Therefore, the system (4.14)-(4.15) is globally asymptotically stable with the Lyapunov function (4.20). All trajectories converge to its unique equilibrium point. \square

The global stability analysis about two TFRC models indicates the available bandwidth within a network could be shared by TFRC sources. Refer to the question how it should be shared, we shall pay attention to the utility function TFRC users are applied. Since the dual-like algorithm of TFRC solves the same network utility maximization problem as TCP Reno, the utility function of all user r is given by

$$U_r(x_r) = - \frac{2S^2}{T_r^2 x_r}, \quad r \in R \quad (4.25)$$

Thus, the network's resource allocation is weighted minimum potential delay fair, which is a special case of the general class of utility functions according to [6]. Since TFRC is endowed with the same utility as TCP Reno, it can be deduced that all sources including TFRC and TCP are weighted α -fair to each other with $\alpha = 2$. It implies that these two types of flows can share the network resources fairly.

4.1.2 Rate of Convergence

The study of the global stability for TFRC networks exhibits all trajectories of the dynamic system converge to the unique equilibrium. But it does not show how fast the system converges and what the rate of convergence depends on. The analysis on the rate of convergence gives insights to these questions, and also provides hints for designing the improved protocols. In this section, we determine the rate of convergence for the proposed two TFRC models.

TFRC Model I

It is shown in Theorem 4.1.1 that the system (4.1)-(4.2) of TFRC model I is globally stable with a unique stable point. We now investigate its rate of convergence, by adopting the linearization approach¹ around its stable equilibrium.

Note that the non-differentiability of the function $f_j(y_j)$ in (4.2) hinders the study of the convergence rate. To carry out the analysis, we first approximate this non-differentiable function with a differentiable one, in order to generate an approximate system to the original system. Let

$$f_j(y_j(t)) \approx \frac{1}{\beta} \ln \left(1 + e^{\beta \frac{y_j(t) - C_j}{y_j(t)}} \right) \triangleq g_j(y_j(t)), \quad (4.26)$$

where $\beta > 0$, and it has

$$g_j(y_j(t)) \rightarrow f_j(y_j(t)), \text{ as } \beta \rightarrow \infty \quad (4.27)$$

due to the differentiability of $g_j(y_j(t))$, we get

$$g'_j(y_j(t)) = \frac{C_j}{\left(1 + e^{-\beta \frac{y_j(t) - C_j}{y_j(t)}} \right) y_j^2(t)} > 0. \quad (4.28)$$

¹In the study of dynamical systems, linearization is a method for assessing the local stability of an equilibrium point of a system with nonlinear differential equations.

Thus, the approximate system is

$$x_r(t) = \frac{\sqrt{2S}}{T_r \sqrt{\sum_{j \in r} p_j(t)}}, \quad (4.29)$$

$$\frac{d}{dt} p_j(t) = \alpha \left(g_j(y_j(t)) - p_j(t) \right). \quad (4.30)$$

which approaches the original system in (4.1)-(4.2) as $\beta \rightarrow \infty$. Therefore, the convergence analysis for the approximate system also corresponds to the results for the original system.

Now that \hat{x} is the unique vector maximizing Lyapunov function (4.8), we have $\hat{x}_r = \frac{\sqrt{2S}}{T_r \sqrt{\sum_{j \in r} \hat{p}_j}}$ and $g_j(\hat{y}_j) = \hat{p}_j$ at the unique equilibrium. Let $p_j(t) = \hat{p}_j + \delta p_j(t)$.

Then, the linearization of $x_r(t)$ about the unique stable point \hat{x}_r is

$$x_r(t) = \hat{x}_r - \frac{\sqrt{2S}}{2T_r \left(\sum_{j \in r} \hat{p}_j \right)^{\frac{3}{2}}} \sum_{j \in r} \left(p_j(t) - \hat{p}_j \right) \quad (4.31)$$

$$= \hat{x}_r - \frac{T_r^2 \hat{x}_r^3}{4S^2} \sum_{j \in r} \delta p_j(t). \quad (4.32)$$

By linearizing the dynamic system $p_j(t)$ about \hat{p} , we obtain:

$$\frac{d}{dt} \delta p_j(t) = \alpha \left(g_j(y_j(t)) - p_j(t) \right) \quad (4.33)$$

$$= \alpha \left(g_j(\hat{y}_j) + g'_j \sum_{s: j \in s} \left(x_s(t) - \hat{x}_s \right) - \hat{p}_j - \delta p_j(t) \right) \quad (4.34)$$

$$= -\alpha \left(g'_j \sum_{s: j \in s} \frac{T_s^2 \hat{x}_s^3}{4S^2} \sum_{j \in s} \delta p_j(t) + \delta p_j(t) \right). \quad (4.35)$$

We may write this in matrix form as

$$\frac{d}{dt} \delta p(t) = -\alpha \left((GA)(KX^3A^T) + I \right) \delta p(t) \quad (4.36)$$

$$= -\alpha G \left((AX)(KX)(AX)^T + G^{-1} \right) \delta p(t), \quad (4.37)$$

where $X = \text{diag}(\hat{x}_r, r \in R)$, $K = \text{diag}(\frac{T_r^2}{4S^2}, r \in R)$ and $G = \text{diag}(g'_j(\hat{y}_j), j \in J)$. Let $M_1 = GA$, $M_2 = KX^3A^T$, the matrix form can be written as

$$\frac{d}{dt} \delta p(t) = -\alpha (M_1 M_2 + I) \delta p(t). \quad (4.38)$$

Let $\Gamma^T \Lambda \Gamma = M_1 M_2 + I$, where Γ is an orthogonal matrix, $\Gamma^T \Gamma = I$, and $\Lambda = \text{diag}(\lambda_j, j \in J)$ is the matrix of eigenvalues, necessarily positive, of the real, symmetric, positive definite matrix. Then

$$\frac{d}{dt} \delta p(t) = -\alpha \Gamma^T \Lambda \Gamma \delta p(t). \quad (4.39)$$

Thus the rate of convergence to the stable point is determined by the smallest eigenvalue, $\lambda_j, j \in J$, of the matrix $\Gamma^T \Lambda \Gamma$. And the speed of convergence is affected by the smoothing parameter α and the magnitude of the derivatives G .

TFRC Model II

For the system (4.14)-(4.15), we use the following system to approximate the original one by replacing $f_j(\cdot)$ with the $g_j(\cdot)$,

$$x_r(t) = \frac{\sqrt{2}S}{T_r} \sqrt{l_r(t)}, \quad (4.40)$$

$$\frac{d}{dt} l_r(t) = \alpha \left(\frac{1}{\sum_{j \in r} g_j(y_j(t))} - l_r(t) \right), \quad (4.41)$$

in which $y_j(t) = \sum_{s:j \in s} x_s(t)$. In Theorem 4.1.2, we have shown that it converges to a unique stable point. We now investigate its convergence rate by linearizing about the stable point. Let \hat{l}_r identify the unique equilibrium point, there has $\hat{x}_r = \frac{\sqrt{2}S}{T_r} \sqrt{\hat{l}_r}$, and $\frac{1}{\left(\sum_{j \in r} g_j(\hat{y}_j)\right)} = \hat{l}_r$ at equilibrium. Let $l_r(t) = \hat{l}_r + \delta l_r(t)$, the linearization of $x_r(t)$ at the stable point \hat{x}_r is:

$$x_r(t) = \hat{x}_r + \frac{1}{2} \frac{\sqrt{2}S}{T_r \sqrt{\hat{l}_r}} (l_r(t) - \hat{l}_r) \quad (4.42)$$

$$= \hat{x}_r + \frac{S^2}{T_r^2 \hat{x}_r} \delta l_r(t). \quad (4.43)$$

Let $b_r(y_j(t)) = \frac{1}{\sum_{j \in r} g_j(y_j(t))}$, and it has $b_r(\hat{y}_j) = \hat{l}_r$ from the previous discussion. At equilibrium, it has

$$b'_r(\hat{y}_j) = -\hat{l}_r^2 \sum_{j \in r} g'_j(\hat{y}_j). \quad (4.44)$$

By linearizing the system about \hat{l}_r , we obtain

$$\frac{d}{dt} \delta l_r(t) = \alpha \left(b_r(y_j(t)) - l_r(t) \right) \quad (4.45)$$

$$= \alpha \left(b_r(\hat{y}_j) + b'_r \sum_{s:j \in s} (x_s(t) - \hat{x}_s) - \hat{l}_r - \delta l_r(t) \right) \quad (4.46)$$

$$= -\alpha \left(\delta l_r(t) + \frac{T_r^4 x_r^4}{4S^2} \sum_{j \in r} g'_j \sum_{s:j \in s} \frac{1}{T_s^2 \hat{x}_s} \delta l_s(t) \right) \quad (4.47)$$

It can be written in matrix form as

$$\frac{d}{dt} \delta l(t) = -\alpha \left(I + K T^2 X^4 A^T G A T^{-2} X^{-1} \right) \delta l(t) \quad (4.48)$$

$$= -\alpha \left(I + (X T^2) (K X^3 A^T) (G A) (X T^2)^{-1} \right) \delta l(t) \quad (4.49)$$

where $X = \text{diag}(\hat{x}_r, r \in R)$, $T = \text{diag}(T_r, r \in R)$, $K = \text{diag}(\frac{T_r^2}{4S^2}, r \in R)$ and $G = \text{diag}(g'_j, j \in J)$. Let $M_1 = GA$, $M_2 = KX^3A^T$, $M_3 = XT^2$, we can have the matrix form

$$\frac{d}{dt}\delta l(t) = -\alpha(I + M_3M_2M_1M_3^{-1})\delta l(t) \quad (4.50)$$

$$= -\alpha M_3(I + M_2M_1)M_3^{-1}\delta l(t) \quad (4.51)$$

Let $\Theta^T\Phi\Theta = I + M_3M_2M_1M_3^{-1}$, where Θ is an orthogonal matrix, $\Theta^T\Theta = I$, and $\Phi = \text{diag}(\phi_r)$ is the matrix of eigenvalues, necessarily positive, of the real, symmetric, positive definite matrix. Then

$$\frac{d}{dt}\delta l(t) = -\alpha\Theta^T\Phi\Theta\delta l(t). \quad (4.52)$$

Thus the rate of convergence to the stable point is determined by the smallest eigenvalue ϕ_r of the matrix $\Theta^T\Phi\Theta$. And the speed of convergence is affected by the smoothing parameter α and the magnitude of the derivatives G .

Stochastic Analysis

In practical, the implementation of TFRC congestion control algorithm depends on accurate measure about the loss rate. However, the real network environment always introduces noise to the measurements. By modeling the effects of all these disturbances as Brownian motion perturbations on the system as in [4], the stochastic analysis can help to understand robustness of the algorithm to these inevitable disturbances.

We consider a stochastic perturbation of the linearized equation (4.39). Let

$$d\delta p(t) = -\alpha(\Gamma^T\Lambda\Gamma\delta p(t) + FdB(t)) \quad (4.53)$$

where F is an arbitrary $|J| \times |J|$ matrix and $B(t) = (B_j(t), j \in J)$ is a collection of independent standard Brownian motions, extended to $-\infty < t < \infty$.

Following the similar procedure for the stochastic analysis as adopted in [4], we conclude that the stationary solution to the system (4.1)-(4.2) has a multivariate normal distribution, $\delta p(t) \sim N(0, \Sigma)$, where

$$\Sigma = E[\delta p(t)\delta p(t)^T] = \alpha\Gamma^T[\Gamma F; \Lambda]\Gamma, \quad (4.54)$$

where the symmetric matrix $[\Gamma F; \Lambda]$ is given by

$$[\Gamma F; \Lambda]_{rs} = \int_{-\infty}^0 e^{\tau\lambda_r}[\Gamma F F^T \Gamma^T]_{rs} e^{\tau\lambda_s} d\tau = \frac{[\Gamma F F^T \Gamma^T]_{rs}}{\lambda_r + \lambda_s}. \quad (4.55)$$

Note that the covariance matrix (4.54) increases linearly with the smoothing parameter α ; as α increases, the faster convergence to equilibrium described by relation (4.39) is at the cost of a greater spread at equilibrium. On the other hand, as G increases, not only is convergence to equilibrium faster, but also the spread at equilibrium decreases. However, we will see in Chapter 5 that, in the presence of delay, increasing G may compromise stability.

Since the linearization of TFRC model II has similar simplified matrix as in (4.52), the stochastic analysis for TFRC model II would obtain the same results.

4.1.3 Rate-adaptation Comparison

We have shown the convergence rate study for the two TFRC models in the above discussion. One natural question raises: what is the difference between TFRC model I and TFRC model II? Since both systems are proved with global stability, can their convergence rate around equilibrium indicate some difference?

An interesting observation is that, for TFRC model I, the linearized matrix of dynamic system is

$$\Gamma^T \Lambda \Gamma = M_1 M_2 + I = M_2^{-1} (M_2 M_1 + I) M_2, \quad (4.56)$$

and for TFRC model II, its matrix form of linearization result is

$$\Theta^T \Phi \Theta = I + M_3 M_2 M_1 M_3^{-1} = M_3 (I + M_2 M_1) M_3^{-1}, \quad (4.57)$$

such that it can be found that $\Gamma^T \Lambda \Gamma$ is similar to $M_2 M_1 + I$, and $M_2 M_1 + I$ is similar to $\Theta^T \Phi \Theta$. Thus, the matrix $\Gamma^T \Lambda \Gamma$ and $\Theta^T \Phi \Theta$ have the same eigenvalues.

Theorem 4.1.3. *The rate of convergence around the equilibrium point for two TFRC models (3.22) and (3.25) are the same.*

Proof. Due to

$$\Theta^T \Phi \Theta = M_3 (I + M_2 M_1) M_3^{-1}, \quad (4.58)$$

the $\Theta^T \Phi \Theta$ and $I + M_2 M_1$ are similar matrices. And consider that

$$I + M_2 M_1 = M_2 (I + M_1 M_2) M_2^{-1}, \quad (4.59)$$

it shows the matrix $I + M_2 M_1$ is similar to $M_1 M_2 + I$, which equals to $\Gamma^T \Lambda \Gamma$. Hence that it has

$$\Theta^T \Phi \Theta \sim \Gamma^T \Lambda \Gamma, \quad (4.60)$$

so the linearized matrix about equilibrium for two TFRC models have the same eigenvalues. Recall that the convergence rate of two systems is determined by the smallest eigenvalue of $\Theta^T \Phi \Theta$ and $\Gamma^T \Lambda \Gamma$ respectively, these two TFRC systems converge to their respective stable points with the same convergence rate. \square

Although the study on rate convergence could not reveal their distinction, we indeed observe different dynamics of these two kinds of TFRC models. This observation is focus on the dynamic update of the state $x_r(t)$. For TFRC Model I, it has

$$\frac{d}{dt}x_r(t) = -\frac{1}{2} \frac{\sqrt{2}S}{T_r \left(\sum_{j \in r} p_j(t) \right)^{\frac{3}{2}}} \sum_{j \in r} \frac{d}{dt}p_j(t) \quad (4.61)$$

$$= -\frac{\alpha}{2} x_r(t) \left(\frac{T_r^2 x_r^2(t)}{2S^2} \sum_{j \in r} f_j(y_j(t)) - 1 \right), \quad (4.62)$$

while for TFRC Model II,

$$\frac{d}{dt}x_r(t) = \frac{1}{2} \frac{\sqrt{2}S}{T_r \sqrt{l_r(t)}} \frac{d}{dt}l_r(t) \quad (4.63)$$

$$= \frac{\alpha}{2} x_r(t) \left(\frac{2S^2}{T_r^2 x_r^2(t)} \frac{1}{\sum_{j \in r} f_j(y_j(t))} - 1 \right). \quad (4.64)$$

Define a notation

$$w_r(t) = \frac{\sum_{j \in r} f_j(y_j(t))}{\sum_{j \in r} p_j(t)}, \quad (4.65)$$

where $\sum_{j \in r} f_j(y_j(t))$ is the loss rate along route r , and $\sum_{j \in r} p_j(t) = \frac{2S^2}{T_r^2 x_r^2(t)}$ represents the smoothed loss rate on route r calculated by TFRC source r .

Thus, for TFRC Model I, it has

$$\frac{d}{dt}x_r(t) = -\frac{\alpha}{2} x_r(t) \left(\frac{w_r(t) - 1}{w_r(t)} \right) w_r(t), \quad (4.66)$$

and for TFRC Model II, it gets

$$\frac{d}{dt}x_r(t) = -\frac{\alpha}{2} x_r(t) \left(\frac{w_r(t) - 1}{w_r(t)} \right). \quad (4.67)$$

The factor $w_r(t)$ indicates the relative ratio of actual loss rate and averaged loss rate. When the route loss rate increase, the smoothed result comprise the increment after averaging, says $\sum_{j \in r} p_j(t) < \sum_{j \in r} f_j(y_j(t))$. Hence TFRC Model I react faster than TFRC Model II, which indicates TFRC Model I has prompt responsiveness to network congestion. On the other hand, in the case that more bandwidth becomes

available accompanied by the decrease of route loss rate, TFRC Model II responds more promptly than TFRC Model I in the light of $\sum_{j \in r} p_j(t) > \sum_{j \in r} f_j(y_j(t))$ at this situation. We may, therefore, reasonably conclude that, compared with TFRC Model I, the flow rate of TFRC Model II increase faster when more bandwidth are available, and decrease slower when congestion become severe.

The intuitive reason behind this result can be understood with the following example. For the reason that the loss rate is calculated indirectly from the inverse of loss interval, it has the relation $p = \frac{1}{l}$ as shown in Figure 4.1. Note that the discrete

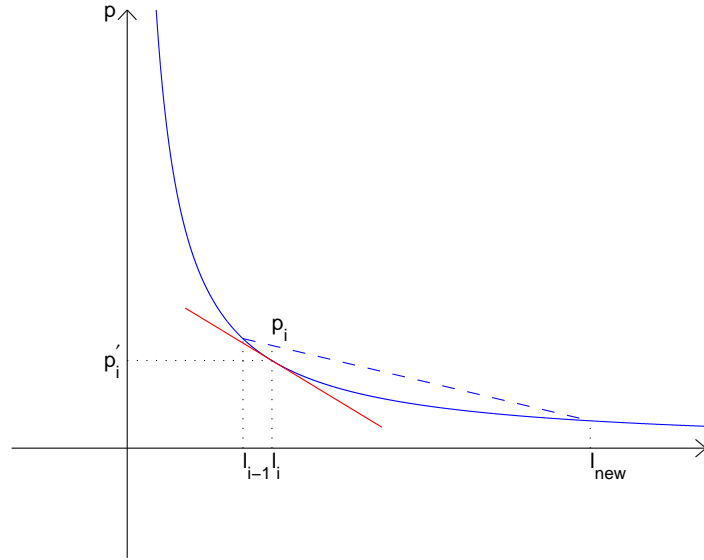


Figure 4.1: The case of loss interval increase.

version of (4.2) and (4.15) could be written as

$$p_i = (1 - \alpha)p_{i-1} + \alpha p_{new} \quad \text{and} \quad l_i = (1 - \alpha)l_{i-1} + \alpha l_{new}, \quad (4.68)$$

where p_{new} and l_{new} represent the current loss rate and loss interval, p_i and l_i denote the averaged result at the state i .

Generally speaking, these two TFRC models discussed above correspond to two kinds of swapped calculation sequence. For Model I, it is equivalent to get p_{new} with $p = \frac{1}{l}$ firstly, then calculates p_i using the Exponentially Weighted Moving Average (EWMA) method. While for Model II, it first adopts EWMA method to get the smoothed l_i , after that, p_i is calculated according to the relation $p = \frac{1}{l}$. In the

following, we first consider the case that loss interval becomes larger due to reduced congestion, then study the case that congestion becomes severe indicated by decreased loss interval.

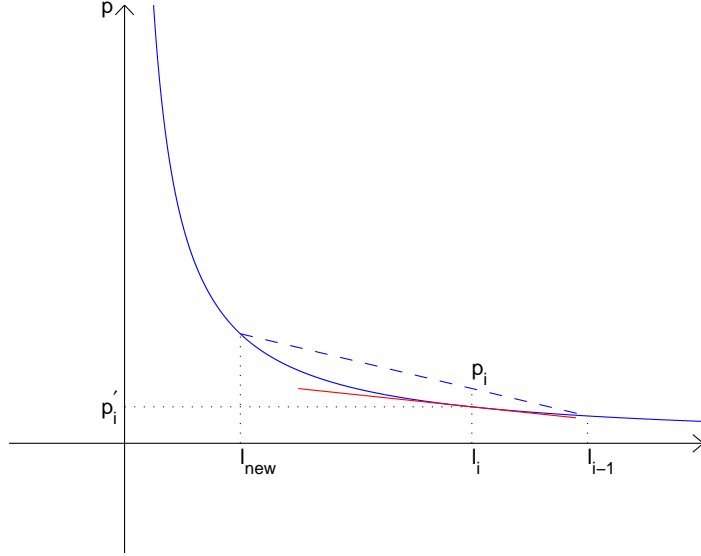


Figure 4.2: The case of loss interval decrease.

Let p_i and p'_i denote the loss rate calculated by TFRC Model I and TFRC Model II respectively. Because the small value of α makes the averaged l_i towards its last value l_{i-1} , the corresponding calculated p'_i is near to its last value p'_{i-1} when the new loss interval l_{new} is larger than the previous one l_{i-1} . Evidently shown in Figure 4.1, the slope of reciprocal function at point (l_i, p'_i) is larger than that of the line on which p_i lies. That indicates the flow of TFRC Model II reacts faster than that of TFRC Model I as loss rate is decreased.

Respect to the case that the loss rate is increased, it is shown in Figure 4.2 that the slope of $\frac{1}{l}$ at point (l_i, p'_i) is smaller than that of the line on which p_i lies. Thus it shows the flow of TFRC Model I reacts more promptly than that of TFRC Model II in this situation. To sum up, this simple example shows TFRC Model I responses timely to increased network congestion, and TFRC Model II reacts fast to decreased congestion.

4.2 TCP Reno and TFRC Coexistence Network

In Chapter 3, we propose models for the case that TFRC flows coexisting with TCP Reno flows. We now establish the global stability for their coexistence case. Since the dynamical system (3.27) extends the design detail of TFRC modeled as in (3.26), we will focus on the analysis about system (3.26), and the similar result of (3.27) follows.

4.2.1 Existence and Uniqueness of Equilibrium

To investigate the system convergence, the equilibrium of the dynamic system (3.26) should be considered firstly. In this section, we show that the system in equations (3.26) has only one unique equilibrium, and the unique equilibrium solves a concave optimization problem.

Theorem 4.2.1. *The system in equation (3.26) has a unique equilibrium, which could be denoted by (\hat{x}, \hat{p}) , given by*

$$\begin{aligned}\hat{x}_r &= \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} \hat{p}_j}}, \quad r \in R \\ \hat{p}_j &= f_j(\hat{y}_j), \quad j \in J\end{aligned}\tag{4.69}$$

Further, this unique equilibrium solves the following concave optimization problem

$$\max_x \sum_{r_1 \in R_1} U_{r_1}(x_{r_1}) + \sum_{r_2 \in R_2} U_{r_2}(x_{r_2}) - \sum_{j \in J} \int_0^{y_j} f_j(z) dz.\tag{4.70}$$

with U_{r_1} and U_{r_2} under the regularity conditions.

Proof. By setting the derivative functions of (3.26) to zero, we can get the equilibrium of the system in equation (3.26):

$$\begin{aligned}\hat{x}_{r_1} &= \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} \hat{p}_{j,1}}}, \quad \hat{x}_{r_2} = \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} \hat{p}_{j,2}}} \\ \hat{p}_{j,1} &= f_j(\hat{y}_j), \quad \hat{p}_{j,2} = f_j(\hat{y}_j)\end{aligned}\tag{4.71}$$

in which the \hat{x}_{r_1} and \hat{x}_{r_2} are also the solution for the optimization problem in equation (4.70). Evidently, there has $\hat{p}_{j,1} = \hat{p}_{j,2}$. Since \hat{x}_{r_1} and \hat{x}_{r_2} have the same form on $\hat{p}_{j,1}$ and $\hat{p}_{j,2}$ respectively, the equilibrium of this system could be rewritten as

$$\begin{aligned}\hat{x}_r &= \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} \hat{p}_j}}, \quad r \in R \\ \hat{p}_j &= f_j(\hat{y}_j), \quad j \in J\end{aligned}\tag{4.72}$$

According to the above definition, the utility functions $U_{r_1}(x_{r_1})$ and $U_{r_2}(x_{r_2})$ are both strictly concave. And moreover, an integral of the increasing function $f_j(y_j)$ is a convex function. Consequently, it is easy to see that the objective function in equation (4.70) is a concave function. Note that the constraint set of the optimization problem (4.70) is $x \geq 0$, which is a convex set, so that the optimization problem is in fact a concave optimization problem. Therefore, it has a unique solution, which is the \hat{x}_r .

Since $y_j = \sum_{s:j \in s} x_s$, it can be shown that the unique \hat{x}_r leads to a unique \hat{y}_j . By its definition of, $f_j(y_j)$ is a continuous, increasing function of y_j , which indicates that there is a one-to-one mapping between f_j and y_j . Hence it can be seen that the unique \hat{y}_j leads to a unique \hat{p}_j . Therefore, there exists a unique equilibrium, which is denoted by (\hat{x}, \hat{p}) .

□

With the same analysis procedure as above, it can be shown the system (3.27) can get similar results.

4.2.2 Stability Analysis of the Coexistence Case

For the system modeling the TCP Reno flows coexisting with TFRC flows in equations (3.26), the $(\dot{x}_{r_1}, p_{j,1})$ and $(x_{r_2}, \dot{p}_{j,2})$ indicate the dynamic of TCP Reno and TFRC flows respectively.

As discussed in Chapter 3, the utility function for TCP Reno flow is $U_r(x_r) = -\frac{2S^2}{T_r^2 x_r}$. Since the TFRC flow has the same transmission rate as TCP Reno flow at the steady state, it can be expected that the TFRC flow may have the same utility function as TCP Reno. Under the conditions about the function f_j discussed above, the expression

$$V(x) = - \sum_{r_1 \in R_1} \frac{2S^2}{T_r^2 x_{r_1}} - \sum_{r_2 \in R_2} \frac{2S^2}{T_r^2 x_{r_2}} - \sum_{j \in J} \int_0^{y_j} f_j(z) dz \quad (4.73)$$

provides a Lyapunov function for the system of differential equations (3.26), and we deduce that the vector x maximizing $V(x)$ is a stable point of the system, to which all trajectories converge.

Theorem 4.2.2. *The strictly concave function $V(x)$ in (4.73) is a Lyapunov function for the system of differential equations (3.26). The unique equilibrium of the system in equations (3.26) is a stable point of the system, to which all trajectories converge.*

Proof. The assumptions on $f_j(y_j)$, $j \in J$, ensures that $V(x)$ is strictly concave on $x \geq 0$, so that the maximizing value of x is a unique one. Observe that

$$\frac{\partial V}{\partial x_{r_1}} = \frac{2S^2}{T_r^2 x_{r_1}^2} - \sum_{j \in r_1} f_j(y_j) \quad (4.74)$$

and

$$\frac{\partial V}{\partial x_{r_2}} = \frac{2S^2}{T_r^2 x_{r_2}^2} - \sum_{j \in r_2} f_j(y_j) \quad (4.75)$$

setting these derivatives to zero identifies the maximum. From the equation $x_{r_2} = \frac{\sqrt{2S}}{T_r \sqrt{\sum_{j \in r_2} p_{j,2}}}$, we can get

$$\sum_{j \in r_2} p_{j,2} = \frac{2S^2}{T_r^2 x_{r_2}^2}, \quad (4.76)$$

and take the derivation of x_{r_2} , which is

$$\dot{x}_{r_2} = -\frac{1}{2} \cdot \frac{\sqrt{2S}}{T_r (\sum_{j \in r_2} p_{j,2})^{3/2}} \sum_{j \in r_2} \dot{p}_{j,2} \quad (4.77)$$

$$= \frac{\alpha x_{r_2}}{2 \sum_{j \in r_2} p_{j,2}} \cdot \left(\frac{2S^2}{T_r^2 x_{r_2}^2} - \sum_{j \in r_2} f_j(y_j) \right) \quad (4.78)$$

Further, it can be seen that

$$\dot{V} = \sum_{r_1 \in R_1} \frac{\partial V}{\partial x_{r_1}} \cdot \dot{x}_{r_1} + \sum_{r_2 \in R_2} \frac{\partial V}{\partial x_{r_2}} \cdot \dot{x}_{r_2} \quad (4.79)$$

$$= \sum_{r_1 \in R_1} \frac{x_{r_1}^2}{2S} \left(\frac{2S^2}{T_r^2 x_{r_1}^2} - \sum_{j \in r_1} f_j(y_j) \right)^2 \quad (4.80)$$

$$+ \sum_{r_2 \in R_2} \frac{\alpha x_{r_2}}{2 \sum_{j \in r_2} p_{j,2}} \left(\frac{2S^2}{T_r^2 x_{r_2}^2} - \sum_{j \in r_2} f_j(y_j) \right)^2 \quad (4.81)$$

$$= \sum_{r_1 \in R_1} \frac{x_{r_1}^2}{2S} \left(\frac{2S^2}{T_r^2 x_{r_1}^2} - \sum_{j \in r_1} f_j(y_j) \right)^2 \quad (4.82)$$

$$+ \sum_{r_2 \in R_2} \alpha \frac{T_r^2 x_{r_2}^3}{4S^2} \left(\frac{2S^2}{T_r^2 x_{r_2}^2} - \sum_{j \in r_2} f_j(y_j) \right)^2 \quad (4.83)$$

Since the user's sending rate x_{r_2} is non-negative, the above result establishes that $V(x)$ is strictly increasing with t , unless $x = \hat{x}$, the unique x maximizing $V(x)$. The function $V(x)$ is thus a Lyapunov function for the system (3.26). \square

For the system (3.27) representing the coexistence case of TCP and TFRC Model II, we can adopt the same Lyapunov function (4.73) to get similar results. In this case,

it has

$$\dot{V} = \sum_{r_1 \in R_1} \frac{\partial V}{\partial x_{r_1}} \cdot \dot{x}_{r_1} + \sum_{r_2 \in R_2} \frac{\partial V}{\partial x_{r_2}} \cdot \dot{x}_{r_2} \quad (4.84)$$

$$= \sum_{r_1 \in R_1} \frac{x_{r_1}^2}{2S} \left(\frac{2S^2}{T_r^2 x_{r_1}^2} - \sum_{j \in r_1} f_j(y_j) \right)^2 \quad (4.85)$$

$$- \sum_{r_2 \in R_2} \alpha \frac{S^2}{T_r^2 x_{r_2}} \left(\frac{2S^2}{T_r^2 x_{r_2}^2} - \sum_{j \in r_2} f(y_j) \right) \left(\frac{T_r^2 x_{r_2}^2}{2S^2} - \frac{1}{\sum_{j \in r_2} f(y_j)} \right) \quad (4.86)$$

which shows $\dot{V} \geq 0$ due to the function property of two items. Hence $V(x)$ is strictly increasing with t , unless $x = \hat{x}$, the unique value maximizing $V(x)$. So the system stability is established by its Lyapunov function (4.73).

Summary

This chapter establishes the stability evidence for TFRC algorithm in Theorem 4.1.1 and Theorem 4.1.2. Especially, the global stability for the coexistence case of TCP and TFRC is studied in Theorem 4.2.2. Based on these results, it could be ensured that the TFRC algorithm can work properly in the practical networks even when it coexists with TCP. In Section 4.1.3, we compare these two TFRC models in terms of their convergence rates and difference dynamics. It shows that TFRC Model I responses more promptly than TFRC Model II when network congestion becomes severe, while TFRC Model II reacts faster than TFRC Model I as route loss rate decreases. Also, we investigate the robustness under perturbations for these two models.

Chapter 5

Delay Analysis

Summary

In the previous chapter, we investigate the global stability of the system defined by TFRC algorithm in the absence of delay. However, communication delays exist in practical networks inevitably, which may compromise the system stability. Our goal in this chapter is to derive the condition that the TFRC network remain stable under communication delays. We first study the case that a single link accessed by one source in Section A.1, and extend these results to the case of general topology networks, accessed by a set of sources.

The delay analysis in this chapter aims to provide a sufficient condition under which the operation of practical networks implementing the TFRC algorithm is stable. For TCP, the problem of the local asymptotical stability under communication delays has drawn much attention from researchers. In [40], Johari *et al.* discussed the effect of propagation delays on the stability of the TCP primal algorithm, and gave sufficient conditions for the local stability of networks with the same round-trip delays. Then Vinnicombe investigated the continuous-time analogue of the same problem by adopting the generalized Nyquist criterion in [41], which derives the similar result for the primal algorithm with delays. By using similar analysis approaches in this chapter, we study the stability condition for TFRC networks in the presence of delay.

5.1 TFRC Network Model I

The goal in this section is to derive sufficient conditions for the system (3.22) to keep stability under the impact of delays. The investigation on the single link case is presented in Appendix A.1. And we focus on the network case for the TFRC model I firstly.

Consider the system (3.22) with delay

$$x_r(t) = \frac{\sqrt{2S}}{T_r \sqrt{q_r(t)}}, \quad (5.1)$$

where the evolution of $q_r(t)$ is given by

$$\frac{d}{dt}q_r(t) = \alpha \left(\sum_{j \in r} f_j(y_j(t - \tau_{jr})) - q_r(t - T_r) \right). \quad (5.2)$$

For each source r , as shown in Figure 5.1, $T_r = \tau_{rj} + \tau_{jr}$ is the round trip time along its route, where τ_{rj} and τ_{jr} represent the forward delay and feedback delay respectively.

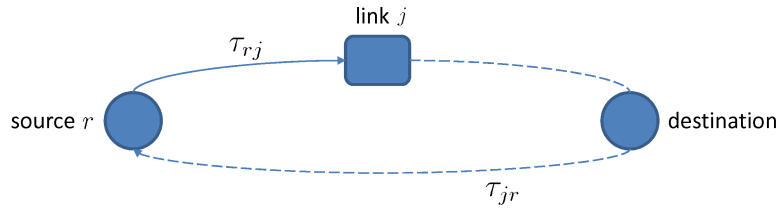


Figure 5.1: The network case.

By taking the derivative of $x_r(t)$, it has

$$\frac{d}{dt}x_r(t) = \frac{dx_r}{dq_r} \frac{dq_r}{dt} \quad (5.3)$$

$$= -\frac{\alpha x_r(t)}{2 q_r(t)} \left(\sum_{j \in r} f_j(y_j(t - \tau_{jr})) - q_r(t - T_r) \right) \quad (5.4)$$

$$= -\frac{\alpha T_r^2 x_r^3(t)}{2 S^2} \left(\sum_{j \in r} f_j(y_j(t - \tau_{jr})) - q_r(t - T_r) \right). \quad (5.5)$$

To approximate the function $f_j(\cdot)$ with a differentiable function

$$g_j(t) \triangleq \frac{1}{\beta} \ln \left(1 + e^{\beta \frac{\sum_{s:j \in s} x_s(t) - C_j}{\sum_{s:j \in s} x_s(t)}} \right) \approx f_j(t), \quad (5.6)$$

which has $g_j(t) \rightarrow f_j(t)$ as $\beta \rightarrow \infty$. Let $x_r(t) = \hat{x}_r + \delta x_r(t)$, $q_r(t) = \hat{q}_r + \delta q_r(t)$ and $y_j(t) = \hat{y}_j + \delta y_j(t)$. By linearizing the delay system (5.1)-(5.2), we get

$$\begin{aligned} \frac{d}{dt} \delta x_r(t) &= -\frac{\alpha T_r^2 x_r^3(t)}{2 \cdot 2S^2} \left(\sum_{j \in r} f_j(y_j(t - \tau_{jr})) - q_r(t - T_r) \right) \\ &\approx -\frac{\alpha T_r^2 x_r^3(t)}{2 \cdot 2S^2} \left(\sum_{j \in r} \left(\hat{g}_j + g'_j \cdot (y_j(t - \tau_{jr}) - \hat{y}_j) \right) - (\hat{q}_r + \delta q_r(t - T_r)) \right) \\ &\approx -\frac{\alpha T_r^2 \hat{x}_r^3}{4S^2} \left(\sum_{j \in r} g'_j \cdot \delta y_j(t - \tau_{jr}) - \delta q_r(t - T_r) \right). \end{aligned}$$

Taking the Laplace transform, it gets

$$s \delta x_r(s) - \delta x_r(0) = -\frac{\alpha \hat{x}_r}{2 \hat{q}_r} \left(e^{-sT_r} \sum_{j \in r} g'_j e^{s\tau_{rj}} \delta y_j(s) - e^{-sT_r} \delta q_r(s) \right), \quad (5.7)$$

Let $D(s)$ denote the diagonal matrix of RTTs in the Laplace domain, i.e. $D(s) = \text{diag}\left(\frac{e^{-sT_r}}{sT_r}\right)$. And let $R(s)$ denote an $|J| \times |R|$ Laplace domain routing matrix that includes both routing and delay information, whose $(j, r)^{th}$ entry is defined as follows:

$$R_{jr} = \begin{cases} e^{-s\tau_{rj}}, & \text{if } j \in r \\ 0, & \text{otherwise} \end{cases} \quad (5.8)$$

For other notations, $T = \text{diag}(T_r)$, $X = \text{diag}(\hat{x}_r)$, $Q = \text{diag}(\hat{q}_r)$ and $G = \text{diag}(g'_j)$. Then the above Laplace transform equation can be written as

$$s \delta X(s) - \delta X(0) = -\frac{\alpha}{2} X Q^{-1} \left(s T D(s) R^T(-s) G \delta Y(s) - s T D(s) \delta Q(s) \right) \quad (5.9)$$

Note that

$$y_j(t) = \sum_{r:j \in r} x_r(t - \tau_{rj}) \quad (5.10)$$

$$= \sum_{r:j \in r} \hat{x}_r + \sum_{r:j \in r} \delta x_r(t - \tau_{rj}) \quad (5.11)$$

$$= \hat{y}_j + \delta y_j(t) \quad (5.12)$$

whose Laplace transform has

$$\delta y_j(s) = \sum_{r:j \in r} e^{-s\tau_{rj}} \delta x_r(s), \quad (5.13)$$

and its matrix form is $\delta Y(s) = R(s) \delta X(s)$.

Consider the linearization of x about its equilibrium

$$x_r(t) = x_r(q_r(t)) \quad (5.14)$$

$$= \hat{x}_r + x'_r \cdot (q_r(t) - \hat{q}_r) \quad (5.15)$$

$$= \hat{x}_r - \frac{\hat{x}_r}{2\hat{q}_r} \delta q_r(t), \quad (5.16)$$

and recall that $x_r(t) = \hat{x}_r + \delta x_r(t)$, the Laplace transform gets the relation

$$\delta q_r(s) = -\frac{2\hat{q}_r}{\hat{x}_r} \delta x_r(s) \quad (5.17)$$

whose matrix form is $\delta Q(s) = -2QX^{-1}\delta X(s)$.

By substituting these results into (5.9), we obtain

$$s \left(I + \frac{\alpha T}{2} XQ^{-1} \left(R^T(-s)GR(s) + 2QX^{-1} \right) D(s) \right) \delta X(s) = \delta X(0) \quad (5.18)$$

Define $G(s) = \frac{\alpha T}{2} XQ^{-1} \left(R^T(-s)GR(s) + 2QX^{-1} \right) D(s)$, it would be

$$s(I + G(s)) \delta X(s) = \delta X(0) \quad (5.19)$$

Therefore, it is equivalently check if all roots of

$$\det(I + G(s)) = 0 \quad (5.20)$$

have negative real parts. From the multivariable Nyquist criterion, the stability condition is equivalent to the following statement: the eigenvalues of $G(jw)$, denoted by $\sigma(G(jw))$, should not encircle the point -1. Let the notation $K = \frac{1}{2}\alpha T$ for simplification.

It has

$$\begin{aligned} \sigma(G(jw)) &= \sigma \left(KXQ^{-1} \left(R^T(-jw)GR(jw) + 2QX^{-1} \right) D(jw) \right) \\ &= \sigma \left(\frac{2}{\pi} \sqrt{KXQ^{-1}} \left(R^T(-jw)GR(jw) + 2QX^{-1} \right) \sqrt{KXQ^{-1}} \cdot \frac{\pi}{2} D(jw) \right) \\ &= \sigma(M(jw) \cdot N(jw)) \end{aligned}$$

where $M(jw) = \frac{2}{\pi} \sqrt{KXQ^{-1}} \left(R^T(-jw)GR(jw) + 2QX^{-1} \right) \sqrt{KXQ^{-1}}$ and $N(jw) = \frac{\pi}{2} D(jw) = \text{diag} \left(\frac{\pi}{2} \frac{e^{-jwT_r}}{jwT_r} \right)$.

Note that $\sqrt{KXQ^{-1}} R^T(-jw)GR(jw) \sqrt{KXQ^{-1}}$ is positive semi-definite, and the matrix $\sqrt{KXQ^{-1}} 2QX^{-1} \sqrt{KXQ^{-1}}$ is positive definite, thus we obtain $M(jw)$ is a positive definite matrix. Due to that, the eigenvalues of matrix M are real and positive, which would have the following Lemma.

Lemma 5.1.1. *Let $M = M^* > 0$ and $N = \text{diag}(n_i)$, $n_i \in \mathbb{C}$, $\forall i$ be given. Then*

$$\sigma(MN) \subset \rho(M)\text{conv}(0 \cup \{n_i\}), \quad (5.21)$$

in which $\sigma(MN)$ and $\rho(M)$ denote the spectrum of the square matrix MN and the spectral radius of the matrix M respectively, and $\text{conv}(0 \cup \{n_i\})$ represents the convex hull of the set of points $\{n_i\}$ and 0.

Proof. Let v be a normalized eigenvector of MN , corresponding to an eigenvalue λ then

$$MNv = \lambda v \quad (5.22)$$

and so

$$Nv = \lambda M^{-1}v \implies \lambda = \frac{v^* Nv}{v^* M^{-1}v} = \rho(M) \left(\left(1 - \frac{1}{k}\right) \cdot 0 + \sum_i \frac{|v_i|^2}{k} n_i \right), \quad (5.23)$$

where $k = \rho(M)(v^* M^{-1}v) \geq 1$ (symbol * denotes conjugate transpose). Let $\theta_0 = 1 - \frac{1}{k}$, $\theta_i = \frac{|v_i|^2}{k}$, ($i = 1, 2, \dots, r$), it has $\sum_{i=0}^r \theta_i = 1$ due to $v^* v = |v_1|^2 + |v_2|^2 + \dots + |v_r|^2 = 1$. Hence, it has

$$\text{conv}(0 \cup \{n_i\}) = \left\{ \theta_0 \cdot 0 + \sum_{i=1}^r \theta_i n_i \right\}. \quad (5.24)$$

For the eigenvalue λ of MN , it satisfies $\lambda \in \rho(M)\text{conv}(0 \cup \{n_i\})$, such that

$$\sigma(MN) \subset \rho(M)\text{conv}(0 \cup \{n_i\}). \quad (5.25)$$

This result shows that the spectrum of MN is included in the convex hull of the set $(0 \cup \{n_i\})$ magnified by $\rho(M)$. \square

Note that the convex hull of all points $\{n_i\} = \left\{ \frac{\pi e^{-jwT_i}}{2 jwT_i} \right\}$ includes the point -1 on its boundary. If $\rho(M) < 1$, it would have

$$-1 \notin \rho(M)\text{conv}(0 \cup \{n_i\}) \quad (5.26)$$

and hence that the eigenloci of MN cannot enclose the point -1. Let's deduce the condition for $\rho(M) < 1$ in the following theorem.

Theorem 5.1.1. *If the sufficient condition*

$$\alpha T_r \left(\frac{1}{2\hat{q}_r} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2} \quad (5.27)$$

is satisfied, then the system (5.1)-(5.2) is locally asymptotically stable.

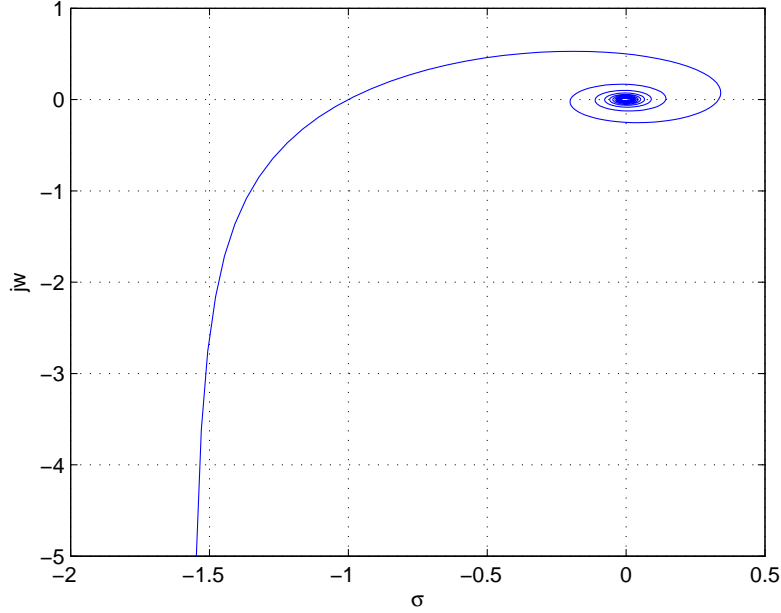


Figure 5.2: Nyquist convex hull of $\frac{\pi}{2} \frac{e^{-jwT}}{jwT}$.

Proof. For the matrix

$$M(jw) = \frac{2}{\pi} \sqrt{KXQ^{-1}} \left(R^T(-jw)GR(jw) + 2QX^{-1} \right) \sqrt{KXQ^{-1}} \quad (5.28)$$

$$= \frac{2}{\pi} \alpha T \left(\frac{1}{2} R^T(-jw)GR(jw)XQ^{-1} + I \right) \quad (5.29)$$

it has

$$\rho(M) \leq \|M\|_{\infty} = \max_{1 \leq i \leq m} \sum_{j=1}^n |M_{ij}| \quad (5.30)$$

according to the property of matrix norm in the matrix analysis.

The above result indicates that the spectral radius of any matrix is bounded by its maximum absolute row sum. If it is satisfied that

$$\frac{2}{\pi} \alpha T_r \left(\frac{1}{2\hat{q}_r} \sum_{j \in r} g'_j \sum_{s: j \in s} \hat{x}_s + 1 \right) < 1, \quad (5.31)$$

the $\rho(M) < 1$ would be guaranteed. That is to say under the condition

$$\alpha T_r \left(\frac{1}{2\hat{q}_r} \sum_{j \in r} g'_j \sum_{s: j \in s} \hat{x}_s + 1 \right) < \frac{\pi}{2}, \quad (5.32)$$

the eigenloci of MN will not enclose the point -1, which means the $\sigma(M(jw))$ does not encircle the point -1. Thus the system (5.1)-(5.2) is locally asymptotically stable.

By simplification with

$$g'_j = \frac{C_j}{\left(1 + e^{-\beta \frac{y_j - C_j}{y_j}}\right) \hat{y}_j^2} = \frac{C_j}{\hat{y}_j^2} \quad \text{as } \beta \rightarrow \infty, \quad (5.33)$$

where $\hat{y}_j = \sum_{s:j \in s} \hat{x}_s$, it gets

$$\alpha T_r \left(\frac{1}{2 \hat{q}_r} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2}, \quad (5.34)$$

which is the sufficient condition for the local stability of the system (5.1)-(5.2). \square

This result has difference with that for the single link case in Theorem A.1.1. The sufficient condition shown in Theorem 5.1.1 also implies the inverse relation of the smoothing factor α and round trip delay T_r , but it involves the consideration for the general network case.

5.2 TFRC Network Model II

For the TFRC model II, its single link case with delay is discussed in the Appendix A.2, which gets the same result with that of the TFRC model I. As regards the network case, the system model is presented as follows by introducing the delay factor into the equation set (3.25):

$$x_r(t) = \frac{\sqrt{2}S}{T_r} \sqrt{l_r(t)}, \quad (5.35)$$

$$\frac{d}{dt} l_r(t) = \alpha \left(\frac{1}{\sum_{j \in r} f_j(y_j(t - \tau_{jr}))} - l_r(t - T_r) \right), \quad (5.36)$$

where notations τ_{rj} , τ_{jr} and T_r take the same meanings as in Section 5.1.

Take the derivative of $x_r(t)$, it has

$$\frac{d}{dt} x_r(t) = \frac{dx_r}{dl_r} \frac{dl_r}{dt} \quad (5.37)$$

$$= \frac{\alpha x_r(t)}{2 l_r(t)} \left(\frac{1}{\sum_{j \in r} f_j(y_j(t - \tau_{jr}))} - l_r(t - T_r) \right) \quad (5.38)$$

$$= \frac{\alpha S^2}{T_r^2 x_r(t)} \left(\frac{1}{\sum_{j \in r} f_j(y_j(t - \tau_{jr}))} - l_r(t - T_r) \right) \quad (5.39)$$

Let $x_r(t) = \hat{x}_r + \delta x_r(t)$ and $l_r(t) = \hat{l}_r + \delta l_r(t)$. Replace the function $f_j(t)$ with $g_j(t)$ as approximation, which takes the same form as in the previous discussion. At equilibrium, it has $\frac{1}{\sum_{j \in r} \hat{g}_j} \approx \frac{1}{\sum_{j \in r} \hat{f}_j} = \hat{l}_r$.

By linearizing the system about \hat{x}_r , we obtain

$$\frac{d}{dt}\delta l_r(t) \approx \frac{\alpha S^2}{T_r^2 x_r(t)} \left(\frac{1}{\sum_{j \in r} g_j(y_j(t - \tau_{jr}))} - l_r(t - T_r) \right) \quad (5.40)$$

$$\approx \frac{\alpha S^2}{T_r^2 \hat{x}_r(t)} \left(\hat{l}_r - \hat{l}_r^2 \sum_{j \in r} g'_j \cdot (y_j(t - \tau_{jr}) - \hat{y}_j) - l_r(t - T_r) \right) \quad (5.41)$$

$$= -\frac{\alpha S^2}{T_r^2 \hat{x}_r(t)} \left(\hat{l}_r^2 \sum_{j \in r} g'_j \delta y_j(t - \tau_{jr}) + \delta l_r(t - T_r) \right) \quad (5.42)$$

whose Laplace transform is

$$s\delta l_r(s) - \delta l_r(0) = -\frac{\alpha \hat{x}_r}{2 \hat{l}_r} \left(\hat{l}_r^2 e^{-sT} \sum_{j \in r} g'_j e^{s\tau_{rj}} \delta y_j(s) + e^{-sT_r} \delta l_r(s) \right). \quad (5.43)$$

Let $R(s)$ denote an $|J| \times |R|$ Laplace domain routing matrix that includes both routing and delay information, and its $(j, r)^{th}$ entry is defined as follows:

$$R_{jr} = \begin{cases} e^{-s\tau_{rj}}, & \text{if } j \in r \\ 0, & \text{otherwise} \end{cases} \quad (5.44)$$

Let $T = \text{diag}(T_r)$, $X = \text{diag}(x_r)$, $L = \text{diag}(l_r)$, $G = \text{diag}(g'_j)$, and $D(s) = \text{diag}\left(\frac{e^{-sT_r}}{sT_r}\right)$, the above Laplace transform has its corresponding matrix form:

$$s\delta L(s) - \delta L(0) = -\frac{\alpha}{2} X L^{-1} \left(s T L^2 D(s) R^T(-s) G \delta Y(s) + s T D(s) \delta L(s) \right) \quad (5.45)$$

Note that

$$y_j(t) = \sum_{r:j \in r} x_r(t - \tau_{rj}) \quad (5.46)$$

$$= \sum_{r:j \in r} \hat{x}_r + \sum_{r:j \in r} \delta x_r(t - \tau_{rj}) \quad (5.47)$$

$$= \hat{y}_j + \delta y_j(t) \quad (5.48)$$

whose Laplace transform has

$$\delta y_j(s) = \sum_{r:j \in r} e^{-s\tau_{rj}} \delta x_r(s), \quad (5.49)$$

and its matrix form is $\delta Y(s) = R(s) \delta X(s)$.

Consider the linearization of $x_r(t)$ about its equilibrium

$$x_r(t) = x_r(l_r(t)) \quad (5.50)$$

$$= \hat{x}_r + x'_r(l_r(t) - \hat{l}_r) \quad (5.51)$$

$$= \hat{x}_r + \frac{\hat{x}_r}{2\hat{l}_r} \delta l_r(t), \quad (5.52)$$

and $x_r(t) = \hat{x}_r + \delta x_r(t)$, the Laplace transform of $\delta x_r(s)$ is

$$\delta l_r(s) = \frac{2\hat{l}_r}{\hat{x}_r} \delta x_r(t), \quad (5.53)$$

whose matrix form takes $\delta L(s) = 2LX^{-1}\delta X(s)$.

By substituting these results into (5.45), we obtain

$$s \left(I + \frac{\alpha T}{2} XL \left(R^T(-s)GR(s) + 2L^{-1}X^{-1} \right) D(s) \right) \delta X(s) = \delta X(0). \quad (5.54)$$

Define $G(s) = \frac{\alpha T}{2} XL \left(R^T(-s)GR(s) + 2L^{-1}X^{-1} \right)$, it becomes

$$s(I + G(s))\delta X(s) = \delta X(0) \quad (5.55)$$

Therefore, we can equivalently check if all roots of

$$\det(I + G(s)) = 0 \quad (5.56)$$

have negative real parts. According to the multivariable Nyquist criterion, the stability condition is equivalent to the following statement: the eigenvalues of $G(jw)$, denoted by $\sigma(G(jw))$, should not encircle the point -1 .

Consider that

$$\begin{aligned} \sigma(G(jw)) &= \sigma\left(KXL(R^T(-jw)GR(jw) + 2L^{-1}X^{-1})D(jw)\right) \\ &= \sigma\left(\frac{2}{\pi}\sqrt{KXL}(R^T(-jw)GR(jw) + 2L^{-1}X^{-1})\sqrt{KXL} \cdot \frac{\pi}{2}D(jw)\right) \\ &= \sigma(M(jw) \cdot N(jw)) \end{aligned}$$

where $K = \frac{\alpha T}{2}$, $M(jw) = \frac{2}{\pi}\sqrt{KXL}(R^T(-jw)GR(jw) + 2L^{-1}X^{-1})\sqrt{KXL}$, and $N(jw) = \frac{\pi}{2}D(jw)$.

Note that $\sqrt{KXL}R^T(-jw)GR(jw)\sqrt{KXL}$ is positive semi-definite, and the matrix $\sqrt{KXQL}2QX^{-1}\sqrt{KXQL}$ is positive definite, thus we obtain $M(jw)$ is a positive definite matrix. Thus, the eigenvalues of matrix M are real and positive. According to the Lemma 5.1.1, it has

$$\sigma(MN) \subset \rho(M)\text{conv}(0 \cup \{n_i\}), \quad (5.57)$$

which indicates that if $\rho(M) < 1$, the eigenloci of MN will not encircle the point $(-1, j0)$. We will derive a sufficient condition for $\rho(M) < 1$ in the following Theorem.

Theorem 5.2.1. *If it is satisfied the condition*

$$\alpha T_r \left(\frac{\hat{l}_r}{2} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2}, \quad (5.58)$$

then the system (5.1)-(5.2) is locally asymptotically stable.

Proof. For the matrix

$$M(jw) = \frac{2}{\pi} \sqrt{KXL} (R^T(-jw)GR(jw) + 2L^{-1}X^{-1}) \sqrt{KXL}, \quad (5.59)$$

according to the property of matrix norm in matrix analysis

$$\rho(M) \leq \|M\|_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^n |M_{ij}| \quad (5.60)$$

which indicates the spectral radius of any matrix is bounded by its maximum absolute row sum. If it is satisfied that

$$\frac{2}{\pi} \alpha T_r \left(\frac{\hat{l}_r}{2} \sum_{j \in r} g'_j \hat{y}_j + 1 \right) < 1, \quad (5.61)$$

which guarantees $\rho(Q) < 1$. Recall that $g'_j = \frac{C_j}{\hat{y}_j^2}$, it gets

$$\alpha T_r \left(\frac{\hat{l}_r}{2} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2}, \quad (5.62)$$

under which the eigenloci of MN cannot enclose the point -1 , which means the $\sigma(G(jw))$ does not encircle the point -1 . Therefore, when satisfying the sufficient condition as above, the system (5.35)-(5.36) is locally asymptotically stable. \square

Both the Theorem 5.2.1 and 5.1.1 show the same result on the system stability of the TFRC network with delay, if we take the relation between \hat{q}_r and \hat{l}_r into consideration. In other words, the detection of loss rate with the inverse of loss interval would not affect the system stability. On the other hand, since the condition for these two models of TFRC network with delay keeping stability is the same, it can be sure that both versions can model the TFRC rate control algorithm, which can operate stably under the derived sufficient conditions.

Because the stability analysis is based on the linearized system, the Theorem 5.2.1 only gives the conditions for local stability. For the study about its globally asymptotically stability, we intend to investigate it as the future work.

5.3 Robustness Comparison of TCP and TFRC

For Kelly's primal algorithm of TCP, existing works in [40, 41] have discussed its stability in the presence of propagation delay. These discussions present sufficient condition on the bound of delay for keeping the network stability. It was learned that the relation between delay and the gain factor of TCP affects the design of this kind of rate control algorithm.

As regards the TFRC network with delay, we have studied its stability condition for the Model I and Model II in Theorem 5.1.1 and Theorem 5.2.1 respectively. These results indicate that the system stability is influenced by the delay and the smoothing factor of TFRC. It is shown that the large value of both of them may compromise the stability of practical network system.

In this section, We are interested in the delay robustness comparison of TCP and TFRC. Since it is found that almost existing works discussing the TCP network with delay focus on the primal algorithm proposed by Kelly *et al.* in [4]:

$$\frac{d}{dt}x_r(t) = \kappa_r (w_r - x_r(t)q_r(t)), \quad (5.63)$$

$$q_r(t) = \sum_{j \in r} p_j \left(\sum_{s: j \in s} x_s(t) \right), \quad (5.64)$$

while little work has been done on the delay analysis of the TCP network model proposed in [5] and [35]:

$$\frac{d}{dt}x_r(t) = \frac{1}{2S} \left(\frac{2S^2}{T_r^2} - x_r^2(t) \cdot q_r(t) \right), \quad (5.65)$$

$$q_r(t) = \sum_{j \in r} f_j(y_j(t)), \quad (5.66)$$

thus we will first study the sufficient stability condition for this TCP network in the presence of delay. Then compare it with the previous results on the TFRC networks.

Consider the system defined by TCP algorithm with delay:

$$\frac{d}{dt}x_r(t) = \frac{1}{2S} \left(\frac{2S^2}{T_r^2} - x_r^2(t - T_r) \cdot q_r(t) \right) \quad (5.67)$$

$$q_r(t) = \sum_{j \in r} f_j(y_j(t - \tau_{jr})) \quad (5.68)$$

For TCP, sources are indicated by $r \in R$, which is the set of all users who run TCP as the rate control algorithm. Along route r , its round trip delay time is $T_r = \tau_{rj} + \tau_{jr}$, where τ_{rj} and τ_{jr} represent the forward delay to link j and the feedback delay to user r respectively.

As introduced in the previous discussion, we adopt the differentiable function $g(\cdot)$ to approximate $f(\cdot)$, and adopt the linearization approach to study the local stability for this delay system. Let $x_r(t) = \hat{x}_r + \delta x_r(t)$ and $q_r(t) = \hat{q}_r + \delta q_r(t)$, note that it has $\frac{2S^2}{T_r^2} = \hat{x}_r^2 \hat{q}_r$ and $\sum_{j \in r} \hat{g}_j \approx \sum_{j \in r} \hat{f}_j = \hat{q}_r$ at equilibrium. Due to

$$\frac{d}{dt} \delta x_r(t) = \frac{1}{2S} \left(\frac{2S^2}{T_r^2} - (\hat{x}_r + \delta x_r(t - T_r))^2 (\hat{q}_r + \delta q_r(t)) \right) \quad (5.69)$$

$$\approx \frac{1}{2S} \left(\frac{2S^2}{T_r^2} - (\hat{x}_r^2 \hat{q}_r + \hat{x}_r^2 \delta q_r(t) + 2\hat{x}_r \hat{q}_r \delta x_r(t - T_r)) \right) \quad (5.70)$$

$$= -\frac{1}{2S} (\hat{x}_r^2 \delta q_r(t) + 2\hat{x}_r \hat{q}_r \delta x_r(t - T_r)), \quad (5.71)$$

By taking the Laplace transform of the above result, it yields

$$s \delta x_r(s) - \delta x_r(0) = -\frac{1}{2S} (\hat{x}_r^2 \delta q_r(s) + 2\hat{x}_r \hat{q}_r e^{-sT_r} \delta x_r(s)). \quad (5.72)$$

Let $D(s)$ denote the diagonal matrix of RTTs of TCP Reno sources in the Laplace domain, i.e. $D(s) = \text{diag} \left(\frac{e^{-sT_r}}{sT_r} \right)$. And let $X = \text{diag}(\hat{x}_r)$, $T = \text{diag}(T_r)$, $Q = \text{diag}(\hat{q}_r)$, then the above Laplace transform equation can be written as

$$s \delta X(s) + \frac{1}{2S} (X^2 \delta Q(s) + 2XQsTD(s) \delta X(s)) = \delta X(0), \quad (5.73)$$

where $X(0)$ is the vector of initial states.

For the link j , let $y_j(t) = \hat{y}_j + \delta y_j(t)$ and $f_j(t) = \hat{f}_j + \delta f_j(t)$. Linearize and the route price function

$$q_r(t) = \sum_{j \in r} f_j(y_j(t - \tau_{jr})) \quad (5.74)$$

$$= \sum_{j \in r} (\hat{f}_j + g'_j \cdot (y_j(t - \tau_{jr}) - \hat{y}_j)) \quad (5.75)$$

$$= \sum_{j \in r} \hat{f}_j + \sum_{j \in r} g'_j \cdot \delta y_j(t - \tau_{jr}) \quad (5.76)$$

$$= \hat{q}_r + \delta q_r(t), \quad (5.77)$$

By taking the Laplace transform, it yields

$$\delta q_r(s) = \sum_{j \in r} g'_j e^{-s\tau_{jr}} \delta y_j(s) = sT_r \cdot \frac{e^{-sT_r}}{sT_r} \sum_{j \in r} g'_j e^{s\tau_{jr}} \delta y_j(s). \quad (5.78)$$

Let $R(s)$ denote an $|J| \times |R|$ Laplace domain routing matrix that includes both routing and delay information, whose $(j, r)^{th}$ entry is defined as follows:

$$R_{jr} = \begin{cases} e^{-s\tau_{rj}}, & \text{if } j \in r \\ 0, & \text{otherwise} \end{cases} \quad (5.79)$$

and we get

$$\delta Q(s) = sTD(s)R^T(-s)G\delta Y(s), \quad (5.80)$$

where $G = \text{diag}(g'_j)$.

Note that

$$y_j(t) = \sum_{r:j \in r} x_r(t - \tau_{rj}) \quad (5.81)$$

$$= \sum_{r:j \in r} (\hat{x}_r + \delta x_r(t - \tau_{rj})) \quad (5.82)$$

$$= \sum_{r:j \in r} \hat{x}_r + \sum_{r:j \in r} \delta x_r(t - \tau_{rj}) \quad (5.83)$$

$$= \hat{y}_j + \delta y_j(t), \quad (5.84)$$

we get $\delta y_j(t) = \sum_{r:j \in r} \delta x_r(t - \tau_{rj})$, which takes the Laplace transform into

$$\delta y_j(s) = \sum_{r:j \in r} e^{-s\tau_{rj}} \delta x_r(s), \quad (5.85)$$

whose matrix form is

$$\delta Y(s) = R(s)\delta X(s). \quad (5.86)$$

By substituting these results into (5.73), we obtain

$$s \left(I + \frac{TX^2}{2S} \left(R^T(-s)GR(s) + 2X^{-1}Q \right) D(s) \right) \delta X(s) = \delta X(0) \quad (5.87)$$

Define $G(s) = \frac{TX^2}{2S} \left(R^T(-s)GR(s) + 2X^{-1}Q \right) D(s)$. Therefore, we can equivalently check if all roots of

$$\det(I + G(s)) = 0 \quad (5.88)$$

have negative real parts. From the multivariable Nyquist criterion, the stability condition is equivalent to the following statement: the eigenvalues of $G(jw)$, denoted by $\sigma(G(jw))$, should not encircle the point -1 . Note that

$$\begin{aligned} \sigma(G(jw)) &= \sigma \left(KTX^2 \left(R^T(-jw)GR(jw) + 2X^{-1}Q \right) D(jw) \right) \\ &= \sigma \left(\frac{2}{\pi} \sqrt{KTX} \left(R^T(-jw)GR(jw) + 2X^{-1}Q \right) \sqrt{KTX} \cdot \frac{\pi}{2} D(jw) \right) \\ &= \sigma(M(jw) \cdot N(jw)), \end{aligned}$$

where $K = \frac{1}{2S}$, and $M(jw) = \frac{2}{\pi} \sqrt{KTX} \left(R^T(-jw)GR(jw) + 2X^{-1}Q \right) \sqrt{KTX}$ and $N(jw) = \frac{\pi}{2} D(jw)$.

Consider that $\sqrt{KTX}R^T(-jw)GR(jw)\sqrt{KTX}$ is positive semi-definite matrix, and the matrix $\sqrt{KTX}(2X^{-1}Q)\sqrt{KTX}$ is positive definite, consequently $M(jw)$ is a positive definite matrix. Therefore, the eigenvalues of M are real and positive.

Note that this result form is similar to the previous analysis on TFRC models. According to the Lemma 5.1.1, the sufficient condition for the delay stability of this system (5.67)-(5.68) is

$$\rho(M) < 1. \quad (5.89)$$

Let's derive the condition for $\rho(M) < 1$ through the following theorem.

Theorem 5.3.1. *If it is satisfied the sufficient condition*

$$\frac{T_r \hat{x}_r}{2S} \left(\sum_{j \in r} g'_j \hat{y}_j + 2\hat{q}_r \right) < \frac{\pi}{2}, \quad (5.90)$$

then the system (5.67)-(5.68) is locally asymptotically stable.

Proof. For TCP sources, the matrix $M(jw)$ is

$$M(jw) = \frac{2}{\pi} \sqrt{KTX} \left(R^T(-jw)GR(jw) + 2X^{-1}Q \right) \sqrt{KTX} \quad (5.91)$$

$$= \frac{2}{\pi} \frac{TX^2}{2S} \left(R^T(-jw)GR(jw) + 2X^{-1}Q \right). \quad (5.92)$$

According to the property of matrix norm

$$\rho(M) \leq \|M\|_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^n |M_{ij}| \quad (5.93)$$

which indicates the spectral radius of any matrix is bounded by its maximum absolute row sum. If it is satisfied that

$$\frac{T_r \hat{x}_r}{2S} \left(\sum_{j \in r} g'_j \hat{y}_j + 2\hat{q}_r \right) < \frac{\pi}{2}, \quad (5.94)$$

the sufficient condition $\rho(M_1) < 1$ is guaranteed.

Under this condition, it can be shown the $\sigma(M(jw))$ does not encircle the point $(-1, j0)$. Thus the system (5.67)-(5.68) is locally asymptotically stable. \square

So far, we have got a sufficient stability condition for the system (5.67)-(5.68) defined by the TCP algorithm with delay. It implies that large product of \hat{x}_r and T_r may compromise the system stability.

Note the fact that for TCP, it has $\frac{2S^2}{\hat{x}_r^2 T_r^2} = \hat{q}_r$ and $g'_j = \frac{C_j}{\hat{y}_j^2}$ as $\beta \rightarrow \infty$, we can obtain a variant of the above condition:

$$\sqrt{2q_r} \left(\frac{1}{2\hat{q}_r} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2}. \quad (5.95)$$

Recall that the sufficient condition for TFRC networks with delay shown in Theorem 5.1.1 is:

$$\alpha T_r \left(\frac{1}{2\hat{q}_r} \sum_{j \in r} \frac{C_j}{\hat{y}_j} + 1 \right) < \frac{\pi}{2}. \quad (5.96)$$

By comparing these conditions, we can get following results.

For two networks with the same topology, let all users (*resources*) in one network adopt TCP algorithm to control their sending rate, and all users in another network use TFRC algorithm. At the situation that

$$\alpha T_r < \sqrt{2q_r}, \quad (5.97)$$

the sufficient condition for TFRC's stability under delay is easier to be satisfied than that of TCP. This indicates TFRC maybe more robust in the presence of delay, when the value of $\sqrt{2q_r}$ is bigger than the product of the smoothing factor α of TFRC and its round trip delay T_r . In other words, this situation will happen when route packet loss occurs frequently.

On the other hand, for the case that the value of loss rate is very small, the two networks may have

$$\alpha T_r > \sqrt{2q_r}, \quad (5.98)$$

under which the sufficient condition of TCP's stability is easier to be satisfied than that of TFRC. In reality, the situation of this case may exist in the network consisting of bottleneck links with very large capacity.

In practical networks, the relation can be checked by introducing the values of real parameters. For example, a classical TFRC network may have average round trip delay $T_r = 100ms$, and a TCP's route loss rate takes $q_r = 0.01$. At this situation, even if the smoothing factor of TFRC takes its largest value $\alpha = 1$, the condition $\alpha T_r < \sqrt{2q_r}$ can still be guaranteed. If we let the route loss rate reduce to $q_r = 0.001$, a classical value of the smoothing factor $\alpha = 0.25$ also can satisfy the above relation. In other words, the above relation $\alpha T_r < \sqrt{2q_r}$ can be established by tuning the value of α . However, when the parameter takes the value $q_r = 10^{-8}$, the product of α and T_r may have chance to be larger than the value of $\sqrt{2q_r}$. For the case that route loss rate is very small, the relation $\alpha T_r > \sqrt{2q_r}$ may happen.

Summary

In this chapter, we investigate the delay stability condition for TFRC Model I and Model II respectively. By linearizing the systems and introducing Nyquist approach for analysis, we get sufficient conditions for their local stability. To compare the robustness of TCP and TFRC algorithm in the presence of delay, we deduce the corresponding sufficient condition for the stability of TCP network with delay. The results reveal relations on comparing the robustness of TFRC and TCP under delay.

□ **End of chapter.**

Chapter 6

Simulation Results

Summary

In this chapter, we carry out Matlab and NS2 simulations to investigate the TFRC algorithm and validate the preceding results of theoretical analysis. Compared with TCP, it shows TFRC has better smoothed effects on the throughput and better continuity on the receiving, which fit in with the requirement of streaming data transfer. Based on simple TFRC networks, we study the issues of stability, convergence property, rate-adaptation and delay influence through simulations.

6.1 Matlab Simulations

In principle, the performance of TFRC algorithm can be evaluated by Matlab simulations, since we already present its model in the dynamic equations (3.22). Based on the following Matlab simulations, we intend to investigate the effectiveness and differences of TFRC models, the traffic property of TFRC flows, and the delay influence on its behavior.

6.1.1 Smoothed Effects and Rate Convergence

To show the principle of TFRC algorithm and its rate dynamic, we carry out the first simulation in a simple network topology. Consider a dumbbell network of six nodes shown in Figure 6.1, on which two TFRC sources deliver packets along route 1 and

route 2 respectively: user 1 transmits data from node 1 to node 5, and user 2 transmits data from node 2 to node 6. Assume that congestion only occurs at the resource shared by both users, thus the link between node 3 and node 4 is the only bottleneck link in this network¹.

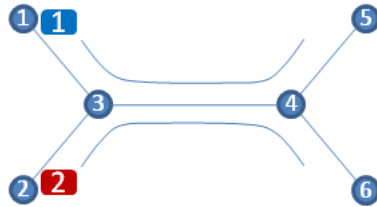


Figure 6.1: Dumbbell network topology.

In comparison, we implement both congestion control algorithms of TFRC and TCP Reno to show their traffic property difference. In the first scenario, we set the network only contains one type of flow, TCP Reno or TFRC. And then, in the second scenario, these two kinds of flows are coexisting in one network and competing at bottleneck link. In both simulations, the network parameters for two sources are chosen as follows: the packet size $S = 1.5\text{KB}$, the round-trip time $T_r = 40\text{ms}$, and the capacity of bottleneck link $C = 1\text{MB}$. In addition, we set the initial sending rates of two users as $x_1 = 300\text{KB/s}$ and $x_2 = 170\text{KB/s}$, and ignore the slow-start phase in both TCP and TFRC simulations.

Based on above settings, the simulation results for the first scenario are shown in Figure 6.2 and Figure 6.3. For two users adopting TCP Reno (3.12)-(3.13) to control their sending rates, it is shown in Figure 6.2 that their throughput have obvious fluctuations along with frequently occurred congestions at the bottleneck link. While considering two sources use TFRC (3.22) to regulate their transmission rates, we can see in Figure 6.3 that the TFRC flows have more smoothed transmission rates when reacting to packet loss. This smoothed effect originates from the exponential moving average of link loss rate p modeled in the previous discussion.

As predicted in the analysis in Section 4.1.1, the dynamical system defined by TFRC algorithm is stable with unique equilibrium, to which all trajectories converge. This characteristic can also be shown in Figure 6.3, in which two users' sending rates with different initial values converge into the unique equilibrium as time evolves. However, it also indicates that the cost of smoothed effects of TFRC is the slower response to the changes of network congestion status. Compared with the trajectories of TCP Reno in Figure 6.2, the sending rate of TFRC shown in Figure 6.3 increases mildly

¹Due to the capacity of the bottleneck link is severely limited resource compared with other links.

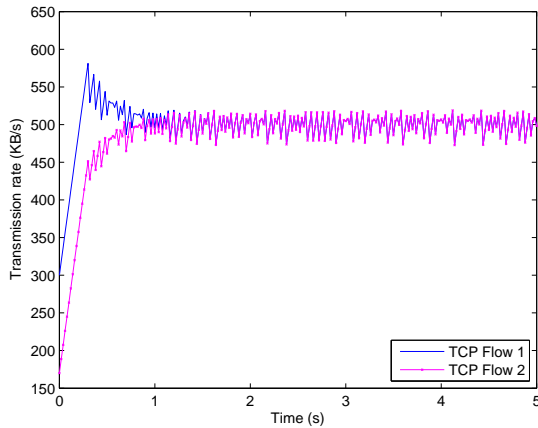


Figure 6.2: Throughput of TCP.

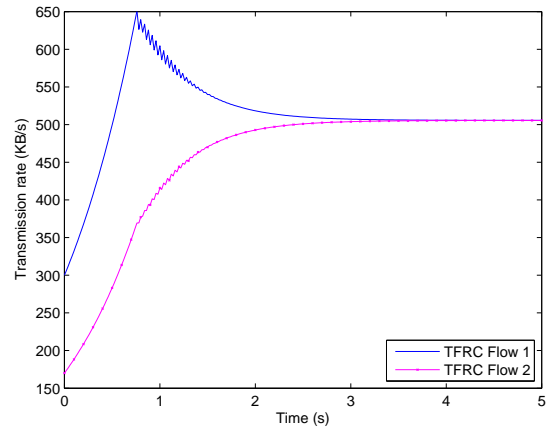


Figure 6.3: Throughput of TFRC.

as more bandwidth is available, and decrease moderately when congestion occurs. It avoids wild fluctuations on throughput, but reacts slower to the dynamical changes of network environment, which also leads to the relative slower convergency of TFRC.

Another important factor we need to consider is the coexistence compatibility of TFRC when deploying it in the TCP networks. When competing bandwidth at the bottleneck link, it is hoped that TFRC flows could share the resource fairly and stably with TCP flows. With the same network settings as above, suppose the flow 1 and flow 2 adopt TCP Reno and TFRC respectively to regulate their sending rates, it is shown in Figure 6.4 that the trajectories of TCP Reno and TFRC converge into the unique equilibrium as time evolves. This result conforms to the theoretical analysis in the previous Section 4.2. Moreover, in this simulation, it also indicates that the dynamical changes of TFRC throughput is smoother than that of TCP Reno, while the increasing and decreasing of sending rate for TFRC is slower than that for TCP Reno. Thus, we can see there is a trade-off between smoothness and responsiveness for congestion control algorithms.

Nevertheless, the Matlab simulation about the TFRC theoretical model also shows that its smoothness and responsiveness is affected by the exponential moving average factor α . With greater value of α , the sending rate will have more obvious oscillations around equilibrium at steady state. As shown in Figure 6.5 and Figure 6.6, the TFRC algorithm is set with $\alpha_1 = 1.25\alpha$ and $\alpha_2 = 2\alpha$ respectively, where α is the value taken in the previous simulation. Compared with the throughput curves depicted in Figure 6.3 with α , the sending rates of users in Figure 6.5 and Figure 6.6 have more evident vibrations as α increases. Meanwhile, it also shows the trajectories converge faster

than that of smaller α case.

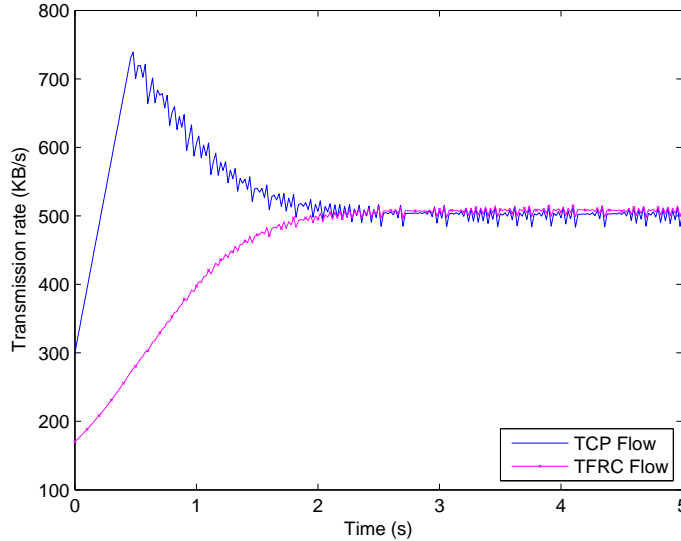


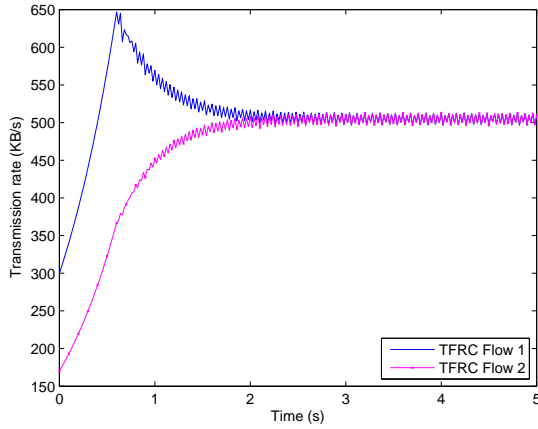
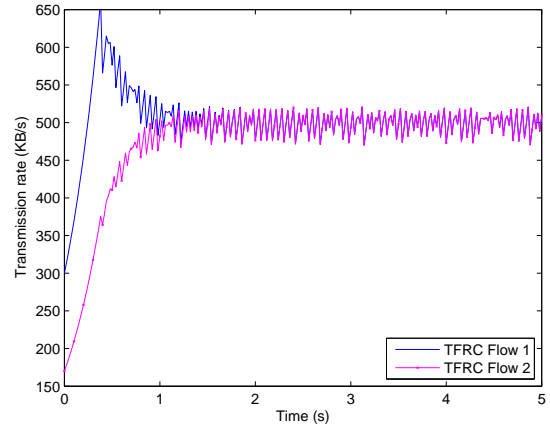
Figure 6.4: TCP Reno coexisting with TFRC.

This is because the sending rate of TFRC user is determined by the loss rate p_j according to its response function $x_r = \frac{\sqrt{2}S}{T_r \sqrt{\sum_{j \in r} p_j}}$. With a larger value of α , there will have greater changes of p_j , which results in more obvious oscillations of the calculated x_r . Thus, the value of exponential moving average factor α should not be too large to compromise the network stability.

6.1.2 Rate-adaptation Comparison of Two Models

The purpose of Matlab simulations in this section is to show the differences between two TFRC models studied in Section 4.1.3. The theoretical analysis results tell us that two models have the same convergence rate around the equilibrium at steady state, but may have different dynamic changes when confront the changes of network environment.

Based on the dumbbell network topology shown in Figure 6.1, we choose the same network parameters as in the previous discussion. To show the difference of TFRC Model I and Model II, we let the TFRC flow competes bottleneck bandwidth with a CBR flow, and they are denoted with flow 1 and flow 2 respectively. In addition, set the initial transmission rate of flow 1 is $x_1 = 300\text{KB/s}$, the cross traffic on flow 2 occurs at the time $t = 4\text{s}$ with 8s duration.

Figure 6.5: TFRC with $\alpha_1 = \frac{5}{4}\alpha$.Figure 6.6: TFRC with $\alpha_2 = 2\alpha$.

The simulation results for TFRC Model I (3.22) and TFRC Model II (3.25) are shown in Figure 6.7 and Figure 6.8 respectively. By comparing the results shown in these two figures, it indicates that the throughput of TFRC Model I decreases faster but increases slower than that of TFRC Model II. Specifically, when congestion occurs at the time $t = 4s$, the decreasing transition time Δt_1 in Figure 6.7 is smaller than the corresponding Δt_1 shown in Figure 6.8. Once there are more available bandwidth due to the vanishing of cross traffic at the bottleneck link, it is shown the duration of increasing transition Δt_2 in Figure 6.8 is shorter than the corresponding Δt_2 in Figure 6.8.

These simulation results are in accordance with the theoretical analysis discussed in Section 4.1.3. It illustrates TFRC Model I reacts promptly to the occurrence of congestion, while TFRC Model II responds rapidly to the detection of available bandwidth.

6.1.3 Delay Instability

As discussed in Chapter 5, delay affects the stability of the system defined by TFRC algorithm, and the theoretical analysis already shows that large delay may compromise the system stability. In this section, we use the Matlab simulation to illustrate the instability case under the delay influence.

The simulation scenario is based on the same network topology shown in Figure 6.1. As initialization, two TFRC flows is set with sending rates $x_1 = 300\text{KB/s}$ and $x_2 = 170\text{KB/s}$ respectively, and they interact with each other when competing the bandwidth of bottleneck link. For network setting, we choose $S = 1.5\text{KB/s}$, $C = 1\text{MB}$,

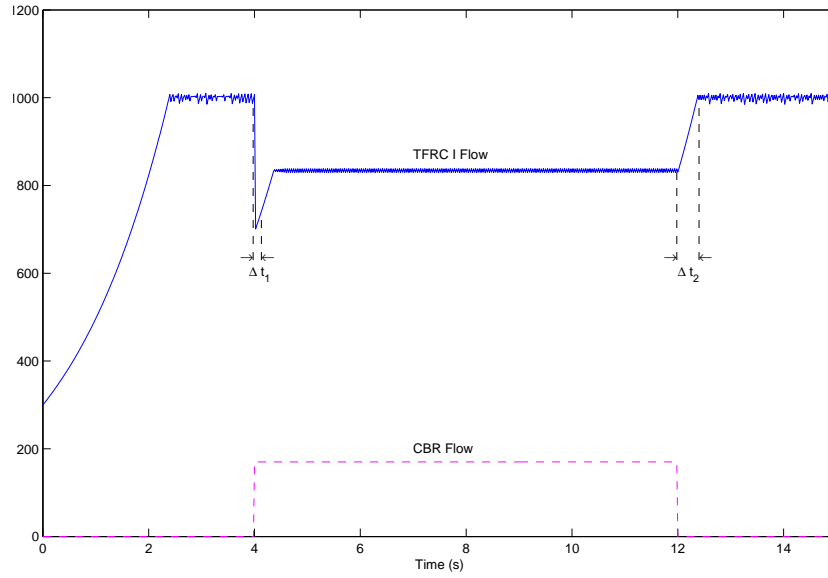


Figure 6.7: TFRC model I with CBR traffic.

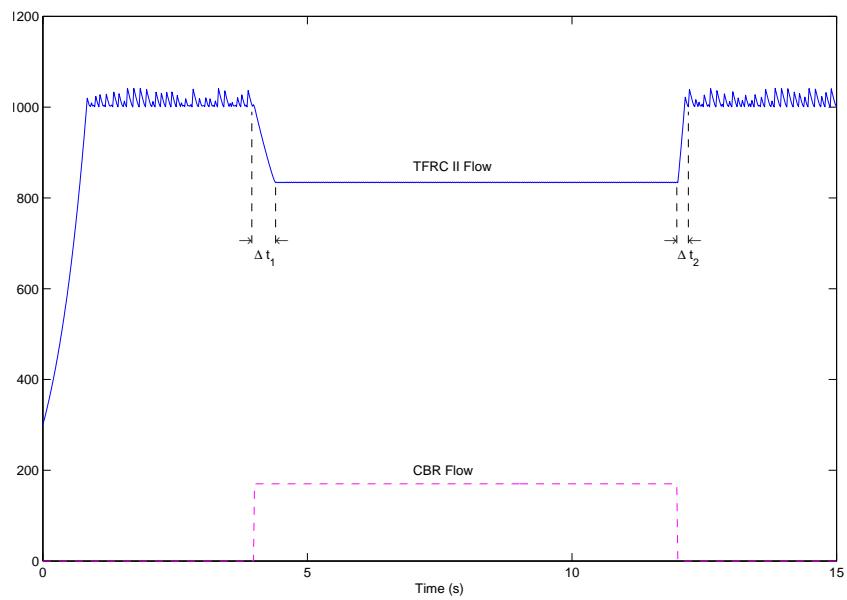


Figure 6.8: TFRC model II with CBR traffic.

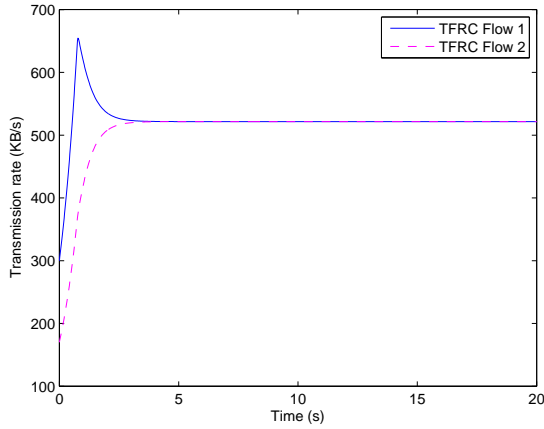
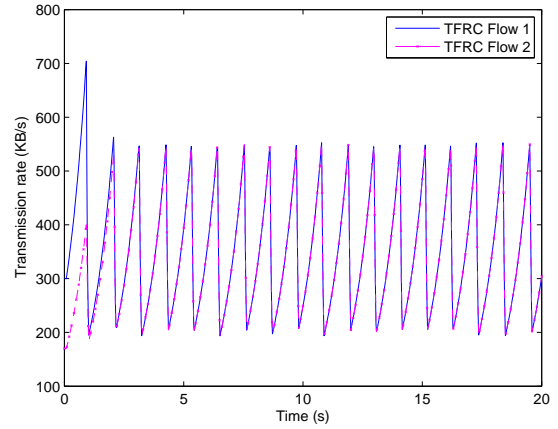
Figure 6.9: TFRC with $T_r = 0.02s$.

Figure 6.10: Delay instability of TFRC.

$T_r = 40ms$ and at equilibrium it has $\hat{x}_r = 500KB/s$. By substituting these values into the delay stability criterion deduced previously

$$\alpha < \frac{\pi}{2} \left(\frac{1}{T} - \frac{1}{\sqrt{T^2 + \frac{8S^2}{C^2}}} \right)$$

we could get the condition that the system would be stable if $\alpha < 0.219$.

Actually, the delay stability condition shown in Theorem 5.1.1 and Theorem A.1.1 is a sufficient condition. It is interesting to note that the instable phenomenon is only observed for much longer round-trip delays. For example, with shorter round-trip time $T_r = 20ms$, it is calculated the sufficient condition $\alpha < 1.71$ can always be satisfied, especially when we choose $\alpha = 0.04$ the result is shown in Figure 6.9 that all trajectories converge and the system is stable. However, when introducing larger round trip delay $T_r = 100ms$ where the forward delay $\tau_{rj} = 40ms$ and feedback delay $\tau_{jr} = 60ms$, and setting $\alpha = 0.04$ to violate its sufficient stability condition $\alpha < 0.014$ calculated correspondingly, the system instability can be observed in Figure 6.10. Thus, it can get the sense that the large round-trip delay may invoke instability of the system defined by TFRC algorithm.

Accordingly, the delay instability for coexistence case is shown in Figure 6.11, where the network parameters are chosen with the same settings as above. The initial sending rates are set with $x_1 = 300KB/s$ and $x_2 = 170KB/s$ for two types of flows, TCP Reno and TFRC respectively. It is shown that the trajectories of two flow rates oscillate with large diversity and could not converge to equilibrium, when we tune the round trip delay to $T_r = 100ms$. At the situation under large value of delay, we

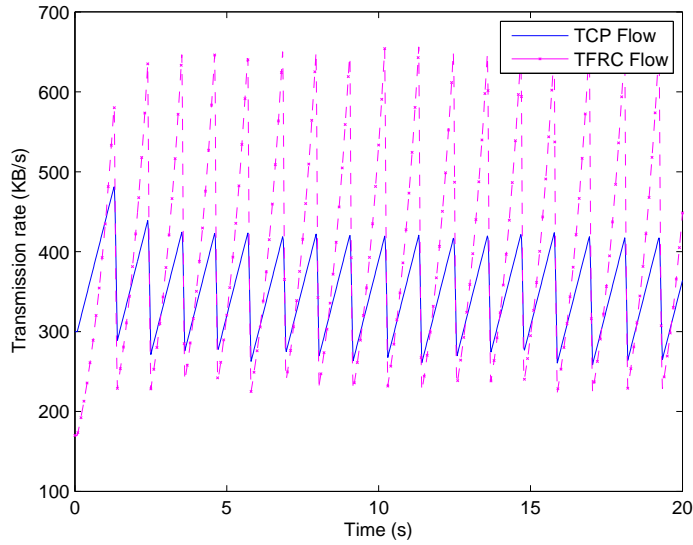


Figure 6.11: Delay instability of TFRC coexisting with TCP Reno.

observe that TFRC has much strong oscillation compared with TCP when the system is instable, it is because that the increasing composition of TCP is inverse proportional to T_r^2 , while the sending rate of TFRC is calculated according to the inverse of T_r . Thus, when the bandwidth is available, the larger the value of round-trip delay T_r , the slower the increasing of TCP than that of TFRC.

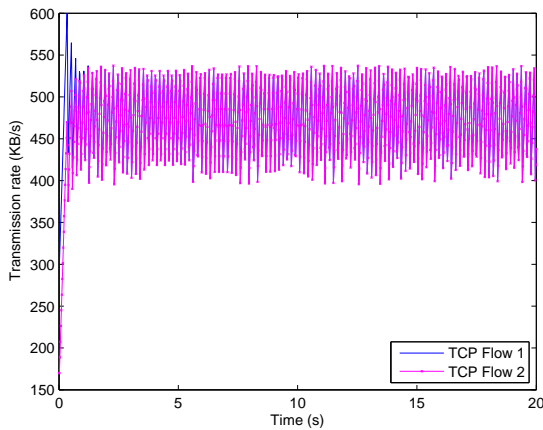


Figure 6.12: TCP flows with fluctuations.

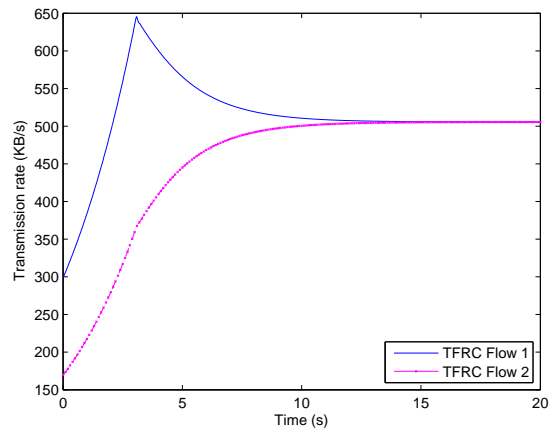


Figure 6.13: TFRC flow rates converge.

In the light of theoretical analysis in Section 5.3, we get the comparison condition for TCP and TFRC algorithms in terms of robustness under delay. Their cor-

responding simulation results would be presented here. For two networks with the same network topology shown in Figure 6.1, where users run TCP in one network and run TFRC in the other, we set two flows in each network with initial sending rates $x_1 = 300\text{KB/s}$ and $x_2 = 170\text{KB/s}$ respectively.

In the first scenario, the bottleneck link in two networks is capable of $C = 1000\text{KB/s}$ bandwidth, and the RTT delay in both networks is $T_r = 40\text{ms}$. For TFRC network, it can be calculated that its sufficient condition for delay stability is $\alpha < 0.219$. To get comparative results, the value of smoothing factor in TFRC network is set to $\alpha = 0.01$ in order to satisfy the relation $\alpha T_r < \sqrt{2q_r}$. With above settings, the simulation result in Figure 6.12 shows the obvious fluctuations of TCP flows, while the simulation result in Figure 6.13 shows the convergence of TFRC flow rates under delay.

In the second scenario, the capacity of bottleneck link in two networks is increased to $C = 10000\text{KB/s}$, and the RTT delay is $T_r = 20\text{ms}$. Due to the large capacity, the value of loss rate q_r is very small. Such that the relation establishes $\alpha T_r > \sqrt{2q_r}$ when we set the smoothing factor of TFRC with $\alpha = 0.1$. The simulation results are shown in Figure 6.14 and Figure 6.15. At this situation with above settings, it shows that TFRC flows exhibit severe fluctuations, while TCP flow rates can converge in the presence of delay.

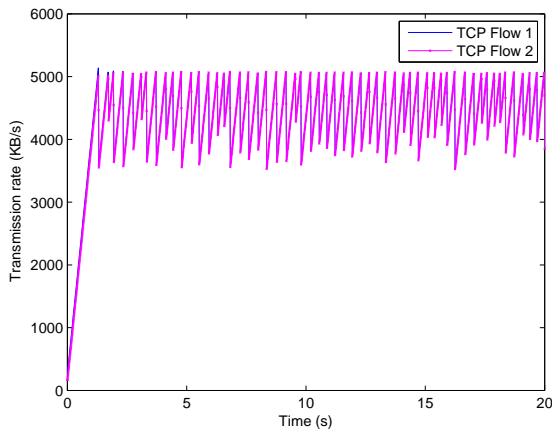


Figure 6.14: TCP flow rates converge.

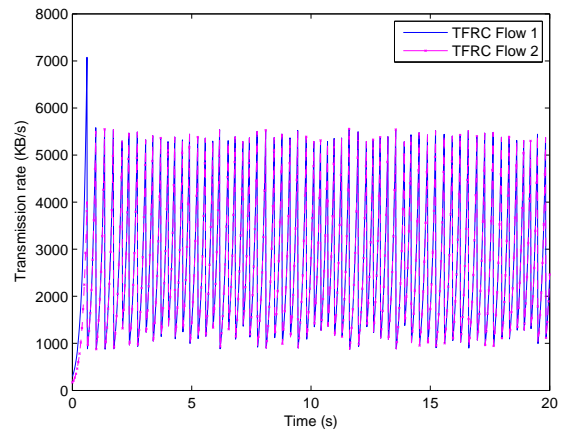


Figure 6.15: TFRC flows with fluctuations.

6.2 NS2 Simulations

To explore the performance of TFRC, a lot of experiments and simulations have been studied in [1] and [10]. These results show that the transmission rate of the TFRC

flow is smoothed with a lower variance; in contrast, the bandwidth used by each TCP flow varies strongly even over relatively short time periods.

Compared with Matlab simulation, NS2 simulation can present more details of transport protocols in the realistic network environment. In this section, we investigate the traffic property of TFRC with NS2 simulations. To validate the theoretical results studied in the previous chapter, we consider the effects of communication delay in TFRC network, and identify the reason for the necessary of exponential moving average on the loss rate p .

6.2.1 Traffic Smoothness and Jitter Property

Consider the simulation scenario as shown in Figure 6.16, which is a simple network topology consisting of four nodes. The links between node 1 and 3, node 2 and 3, node 4 and 5, node 4 and 6 have 10Mb/s of bandwidth and 1ms of delay, while the bottleneck link between node 3 and 4 has 1.7Mb/s of bandwidth and 20ms of delay. At the bottleneck link, DropTail strategy is adopted for exceeded incoming packets to simulate the packet loss scheme modeled as equation (3.13). In the following simulations, we use TCP NewReno implementation for TCP protocol.

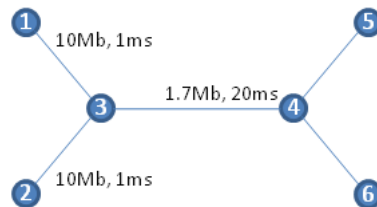


Figure 6.16: Simulation network topology.

In this simulation, we set a CBR flow delivering from node 2 to node 6 as background traffic, and let a TCP flow be transmitted from node 1 to node 5, which starts at time $t = 1s$ and stops at $t = 20s$. The simulation result of TCP throughput is shown in Figure 6.17. With the same settings but replacing the TFRC flow along the route from node 1 to node 5, we get the corresponding simulation result of TFRC throughput depicted in Figure 6.18. These two pictures show the smoothed effects of TFRC on delivering traffic compared with TCP whose throughput has wild fluctuations. The smoothly changing throughput makes TFRC more suitable for streaming multimedia transfer than TCP.

An interesting observation related to the delay variance property of TCP and TFRC is shown in Figure 6.19 and Figure 6.20. With the same network settings described as

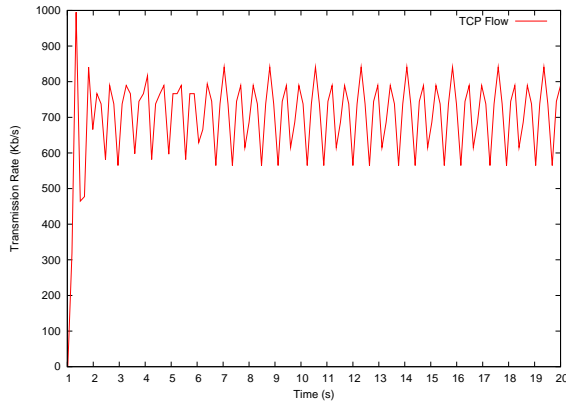


Figure 6.17: Throughput of TCP.

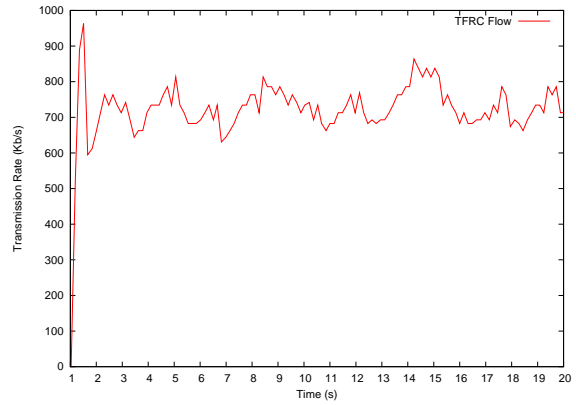


Figure 6.18: Throughput of TFRC.

above, we investigate the dynamics of delay variance at receiver in NS2 simulations. Compared with the jitter property of TCP shown in 6.19, TFRC's jitter is fluctuated more smoothly, which indicates that TFRC receiver possesses the continuous receiving property. As result, it shows that TCP has obvious jitter at receiver compared with that of TFRC. This property makes TFRC more suitable for the transmission of streaming media which favors the continuous reception.

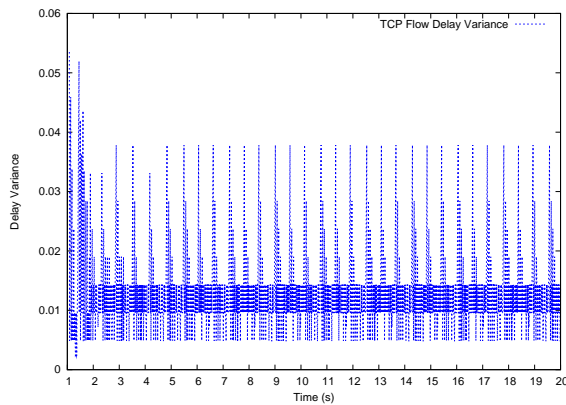


Figure 6.19: Jitter of TCP.

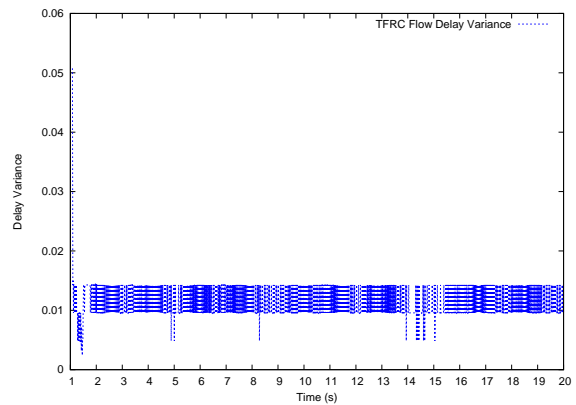


Figure 6.20: Jitter of TFRC.

To show the TFRC traffic property in more realistic network environment, we consider the following network topology which simulates multiple users sharing one bottleneck link. There are four users in this network, using the connections $1 \rightarrow 7$, $2 \rightarrow 8$, $3 \rightarrow 9$ and $4 \rightarrow 10$ respectively. The bottleneck link between node 5 and node 6 has the bandwidth of 1.7Mb/s and the delay of 20ms. Other links in this network have 10Mb/s bandwidth and 20ms delay. The simulation results are presented in Figure

6.22 and Figure 6.23.

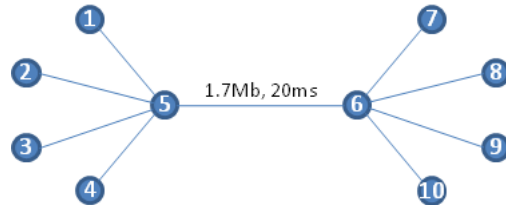


Figure 6.21: Simulation network topology for multiple users.

In Figure 6.22, it exhibits the dynamical behavior of four TCP flows sharing one bottleneck link. The four flows are set with similar parameters, but start to transmit packets one after the other. By taking the 20s duration sample to investigate the property of traffic, we can see TCP flows have drastically fluctuated throughput, while they are capable of fast responsiveness on the bandwidth variation.

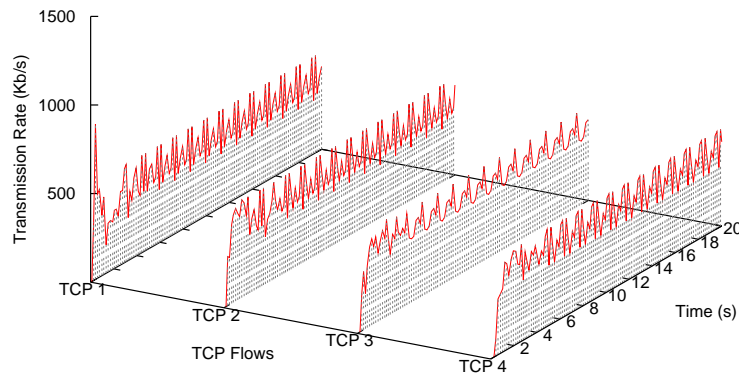


Figure 6.22: The throughput of multiple TCP flows.

Compared with TCP, four TFRC flows sharing the bottleneck link is depicted in Figure 6.23. In this simulation, four users adopt TFRC to control sending rate instead of TCP. The starting time of transmission is the same to the previous simulation of four TCP users. It is shown in Figure 6.23 that TFRC flows can share the bottleneck link with each other and have more smoothed traffic for their data transmission.

Another important problem as indicated in the previous analysis is the coexistence

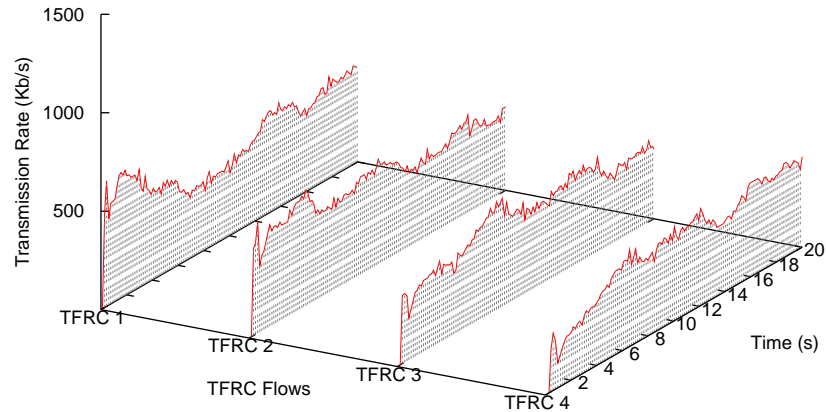


Figure 6.23: The throughput of multiple TFRC flows.

of two different type of flows. This simulation is carried out with results depicted in Figure 6.24. In this simulation, two TCP flows and two TFRC flows start to send packets alternately one after the other, says the TCP1, TCP2, TFRC1, and TFRC2 begin their transmission in succession. As result, it shows that TFRC flows can coexist with TCP Reno flows. Since the trajectories of their flow rates can converge, we can validate the stability of the coexistence system. Also, as shown in the simulation, the traffic of TFRC has less fluctuations than that of TCP.

6.2.2 Necessity of Adaptive Scheme

In dual algorithm, the dual variable update its value dynamically. For TFRC, it uses exponential moving average method to smooth the value of loss rate. If there is no smoothing effect on the loss rate, can it still work properly? This is the problem we intend to study in this section.

By using the NS2 simulation, we change its codes of TFRC in the part of link price adaptation. We modify that the route price p directly equal to the loss rate, instead of adopting EWMA method to smooth and update its value. The purpose of this setting is to study the necessity of dynamic update of p . In the simulation based on the network topology shown in Figure 6.16, we adopt CBR with rate 1Mb/s on the

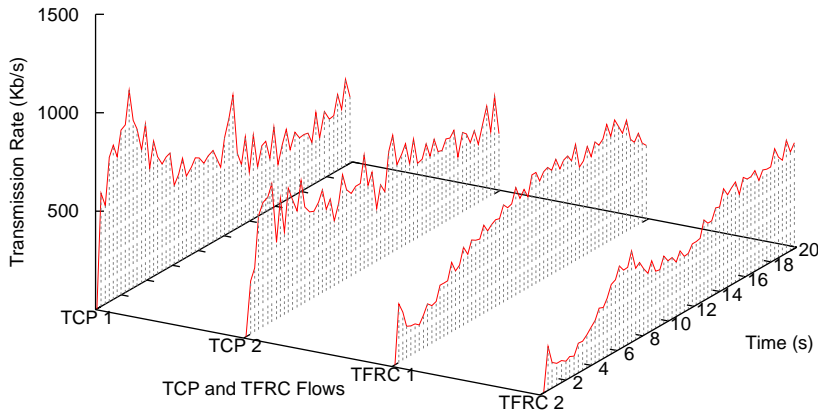


Figure 6.24: TFRC flows coexisting with TCP Reno flows.

flow 2 as the background traffic, and intend to observe the dynamic behavior of TFRC on the flow 1.

As shown in Figure 6.25, the modified system is instable, because the trajectory of the sending rate can not converge. The drastic fluctuations shown in transmission rate make the TFRC without EWMA smoothing effect not suitable for data transmission. In another word, it is necessary for dual algorithm to introduce the dynamic updates on its dual variable.

The reason for oscillations happened in this situation can be explained in this intuition way. Since the transmission rate of TFRC is obtained by the deterministic equation

$$x_r = \frac{\sqrt{2}S}{T_r} \sqrt{l_r},$$

when there is no average strategy on the update of l_r , the sending rate x_r will vary drastically according to the relatively rapid changes of route loss interval l_r . Thus, the adaptiveness of dual variable is necessary for TFRC algorithm.

□ **End of chapter.**

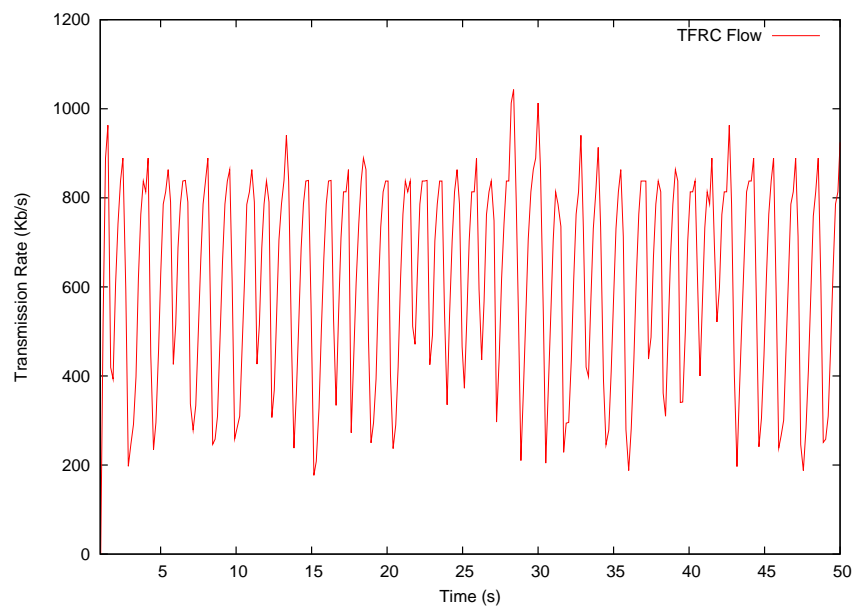


Figure 6.25: TFRC without EWMA mechanism.

Chapter 7

Conclusion

Summary

We will conclude our results in this chapter. As an investigation from the theoretical analysis perspective, the stability of the system defined by TFRC algorithm is studied in this thesis. To verify these results, we adopt Matlab and NS2 simulations to provide evidence for our theoretical study. Analysis and simulations show that TFRC is an appropriate scheme for supporting the real-time streaming multimedia transmission in practical networks.

In this thesis, we have studied the TFRC algorithm from a theoretical perspective. As a protocol designed for streaming multimedia transfer, TFRC guarantees the small fluctuations in throughput which satisfies the QoS requirement of streaming traffics. According to our analysis, the network resources can operate properly by adopting TFRC as their rate control algorithm. When competing bandwidth resources, it is proved that the system defined by TFRC algorithm is globally asymptotically stable. As seen in the utility maximization framework, the resource allocation implemented by TFRC algorithm achieves a weighted minimum potential delay fairness at equilibrium. When introducing the delay factor into network models, we investigate the local stability for these systems.

These analyses reveal the properties of TFRC algorithm besides its stability and fairness. Compared with TCP, TFRC is less aggressive to available bandwidth and slower responsive to congestion occurrence. Although it indicates the weakness of TFRC in the immediacy of utilizing resources, TFRC algorithm is compensated by its advantage on the smoothness and robustness as the trade-off. As shown in our

analysis, the throughput of TFRC is smoother than that of TCP, which makes TFRC appropriate for streaming data transfer. Also, it is exhibited that TFRC maybe more robust under delay when the product of smoothing factor and the RTT delay is very small.

Our contribution in this work is that we propose two TFRC models from theoretical perspective. As a rate control algorithm, TFRC can be taken as a dual algorithm solving a utility maximization problem. Then focusing on the dynamic system defined by TFRC algorithm, we investigate and prove its global asymptotic stability, and study its convergence rate and stochastic perturbations around equilibrium. It is shown the smoothing factor has great influence on these properties. In particular, we examine the global asymptotic stability of the coexistence case of TCP and TFRC. To further the in-depth study, we introduce delay in the TFRC models and derive sufficient conditions for these systems work stably under delay. Therefore, our work lays down a theoretical foundation to obtain a better understanding of the dynamics of TFRC networks. Based on the theoretical analysis, we may conclude that the TFRC algorithm is appropriate to be employed by a large number of network applications which require relatively smoothed transmission rate for multimedia streaming transfer.

For the future work, we have seen in the discussion of coexistence case that if a primal algorithm and its dual algorithm can be proved to be stable, can it be expected that their coexistence case by combining both algorithms would also work stably? As a general problem, it requires the investigation on the combination system consisting of two correlated algorithms. In our work, we present the result for its special case, while the general problem may involve heterogeneous algorithms with multiple equilibrium interact with each other. Another open problem concerns the delay stability of coexistence case. In Chapter 5, we investigate the sufficient conditions for TFRC algorithm and compare it with the corresponding result of TCP on their robustness under delay. But the condition for coexistence case to keep stability is remained as the future study, since the traditional approaches can not be adopted directly to deal with its analysis. The third open problem concerns the discrete-time dynamics of TFRC and its convergence properties. Since our work focuses on continuous-time dynamics, it would be of great interest to extend the analysis to the discrete ones, which would be taken as a future study.

As the result of theoretical study, it could be concluded, from what has been discussed in previous chapters, that TFRC is applicable to be deployed in practical networks. Specifically, the analysis suggests that network stability can be guaranteed by implementation of simple, decentralized conditions on end users and resources. As

network expand to the size and complexity of the Internet, we may safely conclude these theoretical predictions will become increasingly important to guarantee robust behavior of end user.

□ **End of chapter.**

Appendix A

Appendix

A.1 Delay Analysis for the Single Link Case of TFRC I

The network with single link is the simplest case for our study. It is non-trivial because the delay analysis for the single link case may give insightful implications for the further complex investigation. In Chapter 5, the study on the network case is presented.

Consider the system consisting of one source $x(t)$ and one link whose congestion is indicated by $p(t)$:

$$x(t) = \frac{\sqrt{2}S}{T\sqrt{p(t - \tau_2)}} \quad (\text{A.1})$$

$$\frac{d}{dt}p(t) = \alpha(f(t - \tau_1) - p(t - T)) \quad (\text{A.2})$$

in which the round-trip delay T consists of forward delay τ_1 and feedback delay τ_2 , says $T = \tau_1 + \tau_2$ as shown in Figure A.1. Note that $f(t) = \frac{(x(t)-C)^+}{x(t)}$ is a continuous, non-negative and non-decreasing function, but its non-differentiability hinders the analysis. Thus we approximate $f(t)$ with a differentiable function $g(t)$:

$$g(t) \triangleq \frac{1}{\beta} \ln \left(1 + e^{\beta \frac{x(t)-C}{x(t)}} \right) \approx f(t), \quad (\text{A.3})$$

where $\beta > 0$ and it has $g(t) \rightarrow f(t)$ as $\beta \rightarrow \infty$, with its derivative

$$g' = \frac{dg}{dx} = \frac{C}{\left(1 + e^{-\beta \frac{x(t)-C}{x(t)}} \right) x(t)^2} > 0. \quad (\text{A.4})$$

The problem of deriving conditions for the stability of this delay system (A.1)-(A.2) is not straightforward. We will linearize the system around its equilibrium

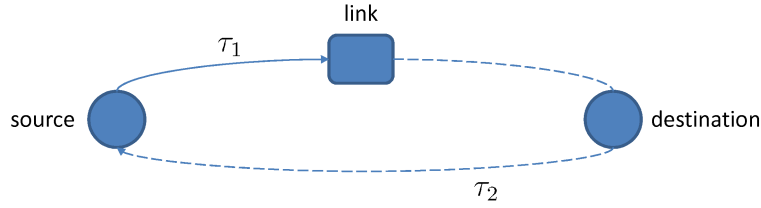


Figure A.1: The single link case.

point and obtain conditions for its local asymptotic stability. Let $x(t) = \hat{x} + \delta x(t)$, $p(t) = \hat{p} + \delta p(t)$ and notice that $\frac{d}{dt}x(t) = \frac{dx}{dp} \frac{dp}{dt}$. Then, by linearizing the controller about the equilibrium, it yields

$$\frac{d}{dt}\delta x(t) = \frac{dx}{dp} \frac{dp}{dt} \quad (\text{A.5})$$

$$= -\frac{\alpha}{2} \frac{x(t)}{p(t - \tau_2)} (f(t - \tau_2 - \tau_1) - p(t - \tau_2 - T)) \quad (\text{A.6})$$

$$\approx -\frac{\alpha T^2 x^3(t)}{4S^2} (\hat{g} + g' \cdot (x(t - T) - \hat{x}) - (\hat{p} + \delta p(t - \tau_2 - T))) \quad (\text{A.7})$$

$$= -\frac{\alpha T^2 (\hat{x} + \delta x(t))^3}{4S^2} (g' \delta x(t - T) - \delta p(t - \tau_2 - T)) \quad (\text{A.8})$$

$$\approx -\frac{\alpha T^2 \hat{x}^3}{4S^2} (g' \delta x(t - T) - \delta p(t - \tau_2 - T)). \quad (\text{A.9})$$

Taking the Laplace transform of both sides, it has

$$s\delta x(s) - \delta x(0) = -\frac{\alpha T^2 \hat{x}^3}{4S^2} (g' e^{-sT} \delta x(s) - e^{-sT} e^{-s\tau_2} \delta p(s)) \quad (\text{A.10})$$

Consider the linearization of x about its equilibrium

$$x(t) = x(p(t - \tau_2)) \quad (\text{A.11})$$

$$= \hat{x} + x' \cdot (p(t - \tau_2) - \hat{p}) \quad (\text{A.12})$$

$$= \hat{x} - \frac{\hat{x}}{2\hat{p}} \delta p(t - \tau_2) \quad (\text{A.13})$$

where x' takes the derivative value of x at the point \hat{p} . Recall that $x(t) = \hat{x} + \delta x(t)$, we hence have $\delta x(t) = -\frac{\hat{x}}{2\hat{p}} \delta p(t - \tau_2)$, whose Laplace transform is $\delta x(s) = -\frac{\hat{x}}{2\hat{p}} e^{-s\tau_2} \delta p(s)$. By substituting this result into (A.10), we get

$$\left(s + \frac{\alpha T^2 \hat{x}^3}{4S^2} \left(g' + \frac{2\hat{p}}{\hat{x}} \right) e^{-sT} \right) \delta x(s) = \delta x(0) \quad (\text{A.14})$$

If all roots of the characteristic equation $s + \frac{\alpha T^2 \hat{x}^3}{4S^2} \left(g' + \frac{2\hat{p}}{\hat{x}} \right) e^{-sT}$ lie in the complex left-half plane, the system will be stable. Thus, we are interested in the conditions

under which the characteristic equation has no roots located at the complex right-half plane. Note that $s = 0$ cannot be a solution to the characteristic equation, such that it can be rewritten as

$$1 + G(s) = 0 \quad (\text{A.15})$$

where

$$G(s) = \frac{\alpha T^3 \hat{x}^3}{4S^2} \left(g' + \frac{2\hat{p}}{\hat{x}} \right) \frac{e^{-sT}}{sT}. \quad (\text{A.16})$$

By simplification with $\hat{x} = \frac{\sqrt{2S}}{T\sqrt{\hat{p}}}$, we have

$$G(s) = \alpha T \left(\frac{\hat{x}}{2\hat{p}} g' + 1 \right) \frac{e^{-sT}}{sT}. \quad (\text{A.17})$$

We intend to discuss the system stability by introducing the Nyquist criterion, which is used for studying the stability of linear systems with pure time delay. For

$$G(jw) = \alpha T \left(\frac{\hat{x}}{2\hat{p}} g' + 1 \right) \frac{e^{-jwT}}{jwT}, \quad (\text{A.18})$$

it is needed to verify whether the plot of $G(jw)$ encircles the point $(-1, j0)$ as w is varied from $-\infty$ to $+\infty$.

Theorem A.1.1. *The system (A.1)-(A.2) of the single link network with delay is locally stable if*

$$\alpha < \frac{\pi}{2} \left(\frac{1}{T} - \frac{1}{\sqrt{T^2 + \frac{8S^2}{C^2}}} \right). \quad (\text{A.19})$$

Proof. Since it is

$$G(jw) = \frac{2}{\pi} \alpha T \left(\frac{\hat{x}}{2\hat{p}} g' + 1 \right) \cdot \frac{\pi e^{-jwT}}{2 jwT}, \quad (\text{A.20})$$

note that the Nyquist contour of $\frac{\pi e^{-jwT}}{2 jwT}$ passes through the $(-1, j0)$ point exactly. If it is satisfied the condition

$$\frac{2}{\pi} \alpha T \left(\frac{\hat{x}}{2\hat{p}} g' + 1 \right) < 1, \quad (\text{A.21})$$

the plot of $G(jw)$ will not encircle the point $(-1, j0)$ as w varying from $-\infty$ to $+\infty$. According to the Nyquist criterion, the number of poles of the linear system in the complex right-half plane is 0, which means the system (A.1)-(A.2) is locally stable around the equilibrium point under this condition.

At the steady state, it has already been shown that

$$\hat{x} = \frac{\sqrt{2}S}{T\sqrt{\hat{p}}}, \quad (\text{A.22})$$

$$\hat{p} = \frac{\hat{x} - C}{\hat{x}} = 1 - \frac{C}{\hat{x}}, \quad (\text{A.23})$$

$$g' = \frac{C}{\hat{x}^2}. \quad (\text{A.24})$$

By solving the \hat{x} and \hat{p} , and taking these results into the obtained condition

$$\alpha T \left(\frac{\hat{x}}{2\hat{p}} g' + 1 \right) < \frac{\pi}{2}, \quad (\text{A.25})$$

we can get the local stability condition for the system (A.1)-(A.2), which is

$$\alpha < \frac{\pi}{2} \left(\frac{1}{T} - \frac{1}{\sqrt{T^2 + \frac{8S^2}{C^2}}} \right). \quad (\text{A.26})$$

□

Note that this condition gives an upper bound for smoothing factor α in the TFRC algorithm, which depends on the round trip delay T . It shows that the smoothing factor is inversely related to the round trip delay if $\frac{\sqrt{2}S}{T} \gg \frac{C}{2}$. When one connection has a large value of T , it is required that α is small enough in order to keep the stability of the delay system. In other words, the longer the delay is, the smoother effect on traffic the system requires.

This result is reasonable when we consider it in the real network operation. Shorter delay allows the system to take much more of the newest updates into consideration, while longer delay limits the fluctuations of user's transmission rate.

Since this condition is deducted from the linearized system, it builds the local stability result for the single link network with delay. In addition, because this result is derived from the Nyquist criterion, it is a sufficient condition for the local stability, while how to get its necessary condition is still an open question.

A.2 Delay Analysis for the Single Link Case of TFRC II

Recall the nonlinear system (3.25) for the TFRC model II, we study its single link case with delay:

$$x(t) = \frac{\sqrt{2}S}{T} \sqrt{l(t - \tau_2)}, \quad (\text{A.27})$$

$$\frac{d}{dt}l(t) = \alpha \left(\frac{1}{f(t - \tau_1)} - l(t - T) \right), \quad (\text{A.28})$$

where τ_1 represents the delay from source to link, and τ_2 is the time for delivering feedback from link to source, thus the round trip time $T = \tau_1 + \tau_2$. Let $x(t) = \hat{x} + \delta x(t)$ and $l(t) = \hat{l} + \delta l(t)$. By adopting $g(\cdot)$ to approximate $f(\cdot)$ as in Section A.1, the linearization of the approximated system is

$$\frac{d}{dt}\delta l(t) = \alpha\left(\frac{1}{g(t-\tau_1)} - l(t-T)\right) \quad (\text{A.29})$$

$$= \alpha\left(\hat{l} - \hat{l}^2 g' \cdot (x(t-\tau_1) - \hat{x}) - (\hat{l} + \delta l(t-T))\right) \quad (\text{A.30})$$

$$= -\alpha\left(\hat{l}^2 g' \delta x(t-\tau_1) + \delta l(t-T)\right), \quad (\text{A.31})$$

where g' denotes the derivative of g at the point \hat{x} . Taking the Laplace transform of both sides, it has

$$s\delta l(s) - \delta l(0) = -\alpha\left(\hat{l}^2 g' e^{-s\tau_1} \delta x(s) + e^{-sT} \delta l(s)\right). \quad (\text{A.32})$$

Consider the linearization of x about its equilibrium,

$$x(t) = x(l(t-\tau_2)) \quad (\text{A.33})$$

$$= \hat{x} + \frac{S^2}{T^2 \hat{x}} \delta l(t-\tau_2) \quad (\text{A.34})$$

Recall that $x(t) = \hat{x} + \delta x(t)$, we have $\delta x(t) = \frac{S^2}{T^2 \hat{x}} \delta l(t-\tau_2)$, whose Laplace transform is $\delta x(s) = \frac{S^2}{T^2 \hat{x}} e^{-s\tau_2} \delta l(s)$. By substituting this into the (A.32), we get

$$\left(s + \alpha\left(\frac{\hat{l}^2 S^2}{T^2 \hat{x}} g' + 1\right) e^{-sT}\right) \delta l(s) = \delta l(0). \quad (\text{A.35})$$

If all roots of the characteristic equation $s + \alpha\left(\frac{\hat{l}^2 S^2}{T^2 \hat{x}} g' + 1\right) e^{-sT}$ lie in the complex left-half plane, the system will be stable. Note that $s = 0$ cannot be a solution to the characteristic equation, such that it can be rewritten as

$$1 + G(s) = 0 \quad (\text{A.36})$$

where

$$G(s) = \alpha T \left(\frac{\hat{l}^2 S^2}{T^2 \hat{x}} g' + 1\right) \frac{e^{-sT}}{sT}. \quad (\text{A.37})$$

By substituting that $\hat{x}^2 = \frac{2S^2}{T^2} \hat{l}$, it becomes

$$G(s) = \alpha T \left(\frac{\hat{x} \hat{l}}{2} g' + 1\right) \frac{e^{-sT}}{sT}. \quad (\text{A.38})$$

Note that the equation (A.38) is same to the characteristic equation (A.17) discussed in Section A.1. Therefore, Theorem A.1.1 establishes the sufficient condition for the TFRC model II in the single link case as well.

□ **End of chapter.**

Bibliography

- [1] S. Floyd, M. Handley, J. Padhye, and J. Widmer, “Equation-based congestion control for unicast applications,” in *SIGCOMM '00: Proceedings of the conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*. New York, NY, USA: ACM, 2000, pp. 43–56.
- [2] J. Postel, “Transmission Control Protocol,” RFC 793 (Standard), Sep. 1981, updated by RFC 3168. [Online]. Available: <http://www.ietf.org/rfc/rfc793.txt>
- [3] V. Jacobson, “Congestion avoidance and control,” in *Proc. ACM SIGCOMM*, Stanford, CA, Aug. 1988, pp. 314–329.
- [4] F. Kelly, A. Maulloo, and D. Tan, “Rate control for communication networks: shadow prices, proportional fairness and stability,” *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252, 1998.
- [5] F. Kelly, “Fairness and stability of end-to-end congestion control,” *European Journal of Control*, vol. 9, no. 2-3, pp. 159–176, 2003.
- [6] J. Mo and J. Walrand, “Fair end-to-end window-based congestion control,” *IEEE/ACM Transactions on Networking (TON)*, vol. 8, no. 5, pp. 556–567, 2000.
- [7] S. Floyd and K. Fall, “Promoting the use of end-to-end congestion control in the Internet,” *IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 4, pp. 458–472, 1999.
- [8] J. Widmer, R. Denda, and M. Mauve, “A survey on TCP-friendly congestion control,” *Network, IEEE*, vol. 15, no. 3, pp. 28–37, 2001.
- [9] M. Chen, “A general framework for flow control in wireless networks,” Ph.D. dissertation, EECS Department, University of California, Berkeley, Dec 2006.

- [10] S. Floyd, M. Handley, and J. Padhye, “A Comparison of Equation-Based and AIMD Congestion Control,” ICSI, URL <http://www.icir.org/tfrc>, Tech. Rep., 2000.
- [11] W. Tan and A. Zakhor, “Real-time Internet video using error resilient scalable compression and TCP-friendly transport protocol,” *Multimedia, IEEE Transactions on*, vol. 1, no. 2, pp. 172–186, 1999.
- [12] R. Srikant, *The mathematics of Internet congestion control*. Birkhauser, 2004.
- [13] M. Handley, S. Floyd, J. Padhye, and J. Widmer, “TCP Friendly Rate Control (TFRC): Protocol Specification,” RFC 3448 (Proposed Standard), Jan. 2003, obsoleted by RFC 5348. [Online]. Available: <http://www.ietf.org/rfc/rfc3448.txt>
- [14] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, “Modeling TCP throughput: a simple model and its empirical validation,” *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 4, pp. 303–314, 1998.
- [15] M. Mathis, J. Semke, and J. Mahdavi, “The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm,” *SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 3, pp. 67–82, 1997.
- [16] S. Floyd, M. Handley, J. Padhye, and J. Widmer, “TCP Friendly Rate Control (TFRC): Protocol Specification,” RFC 5348 (Proposed Standard), Sep. 2008. [Online]. Available: <http://www.ietf.org/rfc/rfc5348.txt>
- [17] M. Welzl and W. Stadler, “User-Centric Evaluation of TCP-friendly Congestion Control for Real-Time Video Transmission,” *Elektrotechnik und Informationstechnik*, vol. 6, 2005.
- [18] R. Rejaie, M. Handley, and D. Estrin, “RAP: An end-to-end rate-based congestion control mechanism for real-time streams in the Internet,” in *INFOCOM’99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 1999.
- [19] D. Sisalem and A. Wolisz, “LDA+ TCP-Friendly Adaptation: A Measurement and Comparison Study,” in *Proc. International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, 2000.
- [20] I. Rhee, V. Ozdemir, and Y. Yi, “TEAR: TCP emulation at receivers-flow control for multimedia streaming,” *Department of Computer Science, NCSU, Technical report, April*, 2000.

- [21] Y. Yang and S. Lam, "General AIMD Congestion Control," in *Proc. IEEE ICNP*, 2000, pp. 187–198.
- [22] D. Bansal and H. Balakrishnan, "Binomial congestion control algorithms," in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2, 2001.
- [23] S. Jin, L. Guo, I. Matta, and A. Bestavros, "A spectrum of TCP-friendly window-based congestion control algorithms," *IEEE/ACM Transactions on Networking (TON)*, vol. 11, no. 3, pp. 341–355, 2003.
- [24] S. Tsao, Y. Lai, and Y. Lin, "Taxonomy and Evaluation of TCP-Friendly Congestion-Control Schemes on Fairness, Aggressiveness, and Responsiveness," *Network, IEEE*, vol. 21, no. 6, pp. 6–15, 2007.
- [25] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-based congestion control for unicast applications: the extended version," ICSI Technical Report TR-00-03, URL <http://www.icir.org/tfrc>, Tech. Rep., 2000.
- [26] J. Padhye, "Towards a comprehensive congestion control framework for continuous media flows in best effort networks," Ph.D. dissertation, University of Massachusetts Amherst, March 2000.
- [27] J. Widmer, "Equation-based congestion control," Master's thesis, University of Mannheim / AT&T Center for Internet Research at ICSI (ACIRI), Feb. 2000.
- [28] Y. Yang, M. Kim, and S. Lam, "Transient behaviors of TCP-friendly congestion control protocols," *Computer Networks*, vol. 41, no. 2, pp. 193–210, 2003.
- [29] D. Bansal, H. Balakrishnan, S. Floyd, and S. Shenker, "Dynamic behavior of slowly-responsive congestion control algorithms," in *Proceedings of the 2001 SIGCOMM conference*, vol. 31, no. 4. ACM New York, NY, USA, 2001, pp. 263–274.
- [30] M. Vojnovic and J. Le Boudec, "On the long-run behavior of equation-based rate control," *IEEE/ACM Transactions on Networking*, vol. 13, no. 3, pp. 568–581, 2005.
- [31] I. Rhee and L. Xu, "Limitations of equation-based congestion control," *IEEE/ACM Transactions on Networking (TON)*, vol. 15, no. 4, pp. 852–865, 2007.

- [32] J. Widmer, “Equation-based congestion control for unicast and multicast data streams,” Ph.D. dissertation, University of Mannheim, May 2003.
- [33] J. Widmer and M. Handley, “Extending equation-based congestion control to multicast applications,” in *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM New York, NY, USA, 2001, pp. 275–285.
- [34] E. Kohler, M. Handley, and S. Floyd, “Designing DCCP: congestion control without reliability,” in *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM New York, NY, USA, 2006, pp. 27–38.
- [35] F. Kelly, “Mathematical modelling of the Internet,” *ICIAM 99: Proceedings of the Fourth International Congress on Industrial & Applied Mathematics, Edinburgh*, 2000.
- [36] S. Low and D. Lapsley, “Optimization flow control. i. basic algorithm and convergence,” *Networking, IEEE/ACM Transactions on*, vol. 7, no. 6, pp. 861–874, Dec. 1999.
- [37] S. Liu, T. Basar, and R. Srikant, “Controlling the Internet: a survey and some new results,” in *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, vol. 3, 2003.
- [38] S. Kunniyur and R. Srikant, “A time scale decomposition approach to adaptive ECN marking,” in *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2001.
- [39] J. Wen and M. Arcaç, “A unifying passivity framework for network flow control,” *Automatic Control, IEEE Transactions on*, vol. 49, no. 2, pp. 162–174, Feb. 2004.
- [40] R. Johari and D. Tan, “End-to-end congestion control for the Internet: delays and stability,” *Networking, IEEE/ACM Transactions on*, vol. 9, no. 6, pp. 818–832, 2001.
- [41] G. Vinnicombe, “On the stability of networks operating TCP-like congestion control,” in *Proceedings of the IFAC World Congress*, 2002.
- [42] A. Tang, J. Wang, S. H. Low, and M. Chiang, “Equilibrium of heterogeneous congestion control: existence and uniqueness,” *IEEE/ACM Trans. Netw.*, vol. 15, no. 4, pp. 824–837, 2007.

- [43] S. Shenker, “Fundamental design issues for the future internet,” *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 7, pp. 1176–1188, Sep 1995.