Cooperative End-to-end Congestion Control in Heterogeneous Wireless Networks

by

Neda Mohammadizadeh

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2013

Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Sharing the resources of multiple wireless networks with overlapped coverage areas has a potential of improving the transmission throughput. However, in the existing frameworks, the improvement cannot be achieved in congestion scenarios because of independent congestion control procedures among the end-to-end paths. Although various network characteristics make the congestion control complex, this variety can be useful in congestion avoidance if the networks cooperate with each other. When congestion happens in an end-to-end path, it is inevitable to have a packet transmission rate less than the minimum requested rate due to congestion window size adjustments.

Cooperation among networks can help to avoid this problem for better service quality. When congestion is predicted for one path, some of the on-going packets can be sent over other paths instead of the congested path. In this way, the traffic can be shifted from a congested network to others, and the overall transmission throughput does not degrade in a congestion scenario. However, cooperation is not always advantageous since the throughput of cooperative transmission in an uncongested scenario can be less than that of non-cooperative transmission due to cooperation costs such as cooperation setup time, additional signalling for cooperation, and out-of-order packet reception. In other words, a trade-off exists between congestion avoidance and cooperation cost. Thus, cooperation should be triggered only when it is beneficial according to congestion level measurements.

In this research, our aim is to develop an efficient cooperative congestion control scheme for a heterogeneous wireless environment. To this end, a cooperative congestion control algorithm is proposed, in which the state of an end-to-end path is provided at the destination terminal by measuring the queuing delay and estimating the congestion level. The decision on when to start/stop cooperation is made based on the network characteristics, instantaneous traffic condition, and the requested quality of service (QoS). Simulation results demonstrate the throughput improvement of the proposed scheme over non-cooperative congestion control.

Acknowledgements

This thesis would not be accomplished without the support of many people. I would like to express my deep gratitude to my supervisor, Prof. Weihua Zhuang, who presented great help and guidance throughout my studies. Her wide scope, clear vision and time devotion are things I have always admired. I also would like to thank Prof. Sherman Shen for his comments, and advises.

I would like to thank my committee, Prof. Zhou Wang and Prof. Liang-Liang Xie, for their valuable feedback. I highly appreciate their time and effort in reading my thesis.

Gratitude is due to the National Science and Engineering Research Council (NSERC) of Canada for a research grant that supported me during my study. I also would like to thank Dr. Bruno Preiss from Blackberry Ltd., for his valuable discussions and comments on my research work. I would like to thank Prof. Wei Song for her valuable help. I would like to extend my gratitude to all members of the Broadband Communication Research (BBCR) group for their discussions and suggestions with special thanks to Dr. Muhammad Ismail for his discussions and help.

I wish I can express my deepest gratitude for my husband, Ehsan Haj Mirza Alian for his valuable discussions and comments. I will not forget his stand beside me, his endless love, and encouragement. I hope I can show my gratitude for my parents, who were and still are the reason behind my achievements. Their support, prayers, love, encouragement and efforts are things I can never pay back.

Dedication

I dedicate my dissertation

to my mother for her love and all that she did and continues to do for me;

to my father for his continued support and encouragement;

and

to my best friend and my loving husband, Ehsan, for standing by me through thick and thin.

Contents

List of Figures v
List of Abbreviations vi
List of Notations vii

1 Introduction 1
1.1 Network Congestion 1
1.2 Heterogeneous Wireless Environment 3
1.3 Motivations and Objectives 6
1.4 Outline of the Thesis 8

2 Congestion Control in Wireless Networks 9
2.1 Congestion Control 9
2.1.1 Congestion Control Categories 12
2.2 Congestion Control Challenges in a Wireless Environment 14
2.3 Congestion Control Protocols for Wireless Networks 15
2.4 Summary 20

3 Overview of Multi-path Congestion Control 21
3.1 Congestion Control Challenges for Multi-path Transmission 21
3.1.1 Concurrent Multi-path Transmission 22
3.1.2 Packet Reordering 22
3.1.3 Service Fairness 24
3.2 Multi-path Congestion Control Protocols 25
3.2.1 SCTP-Based Solutions 26
3.2.2 Multi-path TCP-based Solutions 27
3.2.3 Limitations of Multi-path Congestion Control Protocols 29
3.3 Summary 30

4 System Model 31
4.1 System Model Components 32
4.1.1 Heterogeneous Network Configuration 32
4.1.2 Traffic Model 34
4.1.3 End-to-end Congestion Control 34
4.2 Interaction Model 35
4.3 Problem Definition 37
4.4 Summary 39

5 Cooperative Congestion Control 40
5.1 Cooperation Requirements 40
5.1.1 Congestion Control Core 41
5.1.2 End-terminal Responsibilities 42
5.1.3 Congestion Control Markov Model 45
5.2 Cooperation in Congestion Control 46
5.2.1 Cooperation Initiation 47
5.2.2 The Markov Model with Cooperation Between Networks 48
5.3 When to start/stop cooperation 49
5.3.1 Flow-Level Cooperation Thresholds 49
5.3.2 Packet-Level Cooperation Threshold 53
5.4 Simulation Results 56
5.4.1 Different Packet Arrival Rates 57
5.4.2 Multiple End-to-end Terminals 62
5.4.3 Packet Loss 64
5.5 Summary 65

6 Conclusion and Future Work 66

Appendix 69

A Packet-level congestion control primary equations 69

Bibliography 71

List of Figures

1.1 A heterogeneous wireless environment containing cellular and WiMax networks 5

2.1 End-to-end data and ACK transmission 11

4.1 End-to-end paths between two end terminals in a heterogeneous wireless environment 33

4.2 The interaction model of congestion control in a heterogeneous wireless scenario 36

5.1 Block diagrams of the source and destination tasks in a heterogeneous environment with two wireless networks 44

5.2 Congestion state diagram with two end-to-end paths 47

5.3 Congestion state diagram with network cooperation 48

5.4 The number of successfully-transmitted packets versus time with mean arrival rate of 500 packets/s 58

5.5 Transmission throughput versus arrival rate 59

5.6 End-to-end delay for different arrival rates 61

5.7 Throughput of congestion control schemes over time with the arrival rate of 400 pkts/s 62

5.8 Throughput of congestion control schemes over time with the arrival rate of 600 pkts/s 63

5.9 Throughput of congestion control in a lossy wireless environment 64

List of Abbreviations

AIMD	Additive Increase-Multiplicative Decrease
CN	Core Network
CW	Congestion Window
ECT	End-Cooperation Threshold
EWMA	Exponentially Weighted Moving Average
IETF	Internet Engineering Task Force
IP	Internet protocol
MH - C	Multi-homed Cooperative Congestion Control
MH - NC	Multi-homed Non-cooperative Congestion Control
PIF	Packet In Flight
QoS	Quality of Service
RTT	Round Trip Time
SCT	Start-Cooperation Threshold
SCTP	Stream Control Transmission Protocol
TCP	Transmission Control Protocol
TD	Transmission Duration
WiMax	Worldwide Interpretability for Microwave Access

List of Notations

$A_p(u)$	Arrival time of packet u
$B_p(u)$	Backlog when the last transmitted packet is u
c	End-to-end path bottleneck capacity
d	Propagation delay
$\mathbb{K}$	Set of wireless networks available to the end terminal
K	Number of wireless networks available to the end terminal
k	Network index or corresponding path index
$l(t)$	Queue length at time t
$\mathbb{M}$	Set of end terminals
M	Number of available end terminals in a heterogeneous area
$\mathbb{N}$	Set of wireless access networks in an area
N	Number of available wireless networks in a heterogeneous area
P	Probability function
$q(t)$	Queuing delay of the path at time t
r_{min}	Minimum requested packet transmission rate
$S_p(u)$	Sending time of packet u
t	measurement time
T	Packet TD Time
T_{min}	TD time when there is no queuing delay in the transmission

$T_{max}(t)$	Maximum measured TD before time t
$x(t)$	Packet transmission rate at time t
u	Packet index
$\alpha(t)$	End-cooperation threshold (ECT) at time t
$\beta(t)$	Start-cooperation threshold (SCT) at time t
$\tau(t)$	Round Trip Time of the packet at time t
τ_{min}	Measured RTT when there is no queuing delay in the transmission
$\psi(u)$	Number of PIFs when the last sent packet is u
$\omega(t)$	Congestion window at time t
$\vert.\vert$	Absolute value
$\Vert.\Vert$	Size of a set

Chapter 1

Introduction

The global Internet Protocol (IP) traffic in the year 2005 was two exabytes per month. According to a Cisco forecast, the busy-hour of global IP traffic will increase fivefold by 2015 as compared to 2010, while the average traffic will increase fourfold [1]. As a result, it is expected that network congestion will become a more critical problem in the near future. Cisco also predicts that the traffic from wireless devices will exceed traffic from wired devices by 2016. Wi-Fi and mobile devices will account for 61% of IP traffic. In addition to the growth of wireless traffic volume, more strict and specialized quality-of-service (QoS) is required for new applications. In other words, high-speed, always-connected, and everywhere-available Internet access with much restrictive failure tolerance will become necessary in the near future. In order to satisfy the requested QoS, congestion control should be studied for the future Internet.

1.1 Network Congestion

Congestion, in data networking, refers to a situation in which the quantity of data packets sent through network paths is more than what the network can accommodate. In this situation, any increase in the source data transmission rate results in no improvement, or even reduction in transmission throughput. In a data network, congestion is inevitable, because network resources are shared among users to achieve efficiency and scalability, and different users may simultaneously request a high rate or their peak rate values. When a network becomes congested, the queuing length of the server becomes very large in a short period of time and buffers overflow, leading to packet loss or delay. In order to deal with this problem, many techniques have been proposed, generally known as congestion control techniques.

Congestion control is a strategy to manage network input traffic, in order to avoid congestion, and to ensure network stability, throughput efficiency, and service fairness among the end users. These goals can be achieved by end-to-end congestion control protocols such as transmission control protocol (TCP) [2]. End-to-end congestion control is established at the transport layer that is responsible for controlling the input traffic via the source and/or destination. In other words, ensuring a reliable end-to-end connection while the delay, congestion, and flow are sufficiently under control to satisfy the required QoS is the transport layer's responsibility. Moreover, modifications in an end-to-end congestion control protocol need to be applied only at the two end terminals, and the settings in the entire network including routers remain unchanged. Therefore, end-to-end congestion control has played an important role in the Internet traffic control. In this research, we focus on end-to-end congestion control from the transport layer perspective.

The most well-known end-to-end congestion control protocol in the Internet is the TCP, where a source uses a congestion window (CW) to control the input traffic without employing any explicit information about the internal structure of the network and other network terminals. It uses the feedback information sent from the destination to the source. A feedback notifies the sender by an acknowledgement (ACK) packet whether or not the transmitted packet[1] has reached the destination. One important factor leading to the Internet success in the past decades is the TCP performance improvement that has mades TCP popular for

[1] The smallest unit of data at the transport layer is called segment, which includes a transport layer header attached to a network layer data packet. Here, the term "packet" always refers to segment or TCP packet for simplicity.

various Internet services. Even for real-time services such as video streaming, the TCP is widely used for more than 50% of commercial streaming recently instead of the user datagram protocol (UDP) [3, 4]. For example, popular applications (such as Skype) use the TCP as their transport layer protocol to pass through network address translators and firewalls that block UDP traffic. However, since both congestion avoidance and real-time transmission are required, developing QoS solutions for congestion control algorithms is very important for the future Internet.

1.2 Heterogeneous Wireless Environment

The need to facilitate the Internet access from mobile users requires Internet compatibility in the wireless infrastructure. Currently, LTE and WiMax are all-IP packet switched networks and can be connected to the Internet backbone. However, there are many technical challenges in mobile wireless networks that do not exist in the wired Internet and, therefore, an always-connected, everywhere-available, and high-speed Internet access is not fully supported yet in wireless networks. Specifically, wireless transmission performance degrades due to propagation impairments and user mobility. Although the performance of TCP in the wired Internet is acceptable, it is shown that TCP performance degrades in wireless networks [4, 5]. Hence, different congestion control protocols have been proposed for a wireless scenario [6, 7, 8], and QoS provisioning in wireless networks has been a focus of some recent research works [9, 10, 11]. Especially for multimedia services, various applications have different QoS requirements and traffic characteristics, which lead to more complicated congestion control problems [12].

Each wireless network is designed to satisfy service requests in a specific application environment. Consequently, coverage area overlapping of multiple wireless networks has become more probable. An environment in which the coverage areas of different wireless networks overlap is referred to as a heterogeneous wireless environment [13]. Integrating various wireless access networks can be deployed through IP-based core networks (CNs) or the Internet backbone. Using an IP platform, different access networks with different technologies and infrastructures can be gathered into a single IP-based infrastructure. Therefore, it is possible to deliver various services to the end terminals using different networks, independent of their technology differences. As networking technologies advance, it is expected that more and more end terminals will be equipped with multiple wireless interfaces for Internet access. Therefore, multiple paths can be established between end terminals and the Internet backbone. Multi-path transmission uses multiple paths from different networks in a heterogeneous environment to connect a source to a destination. The connection via multiple paths is called an association [14], which has a potential of increasing network resource utilization and improving service quality [8]. It can accommodate different service requests through multiple networks, or transmit data simultaneously through different paths to improve end-to-end packet delivery [15]. As various network resources are available in a region, end hosts can enjoy increased network capacity via simultaneous multi-path utilization [16].

Figure 1.1 illustrates a simple heterogeneous environment in which WiMax and cellular networks are available simultaneously for a number of terminals. Each terminal is located in the coverage area of a cellular network. Moreover, some terminals are also connected to the WiMax network. Such terminals can establish simultaneous transmissions through the WiMax and cellular networks. Therefore, a complete data transmission from a specified source terminal to a destination terminal is provided through a WiMax/cellular network, the Internet backbone, and then another WiMax/cellular network, respectively.

The TCP performs well in a primary wired Internet scenario, in which only one path is exploited, and delay and congestion losses are tolerable or avoidable. However, using the traditional congestion control protocol for each path of an association in multi-path transmission makes the protocol unfair. For example, in a scenario of two paths with similar round trip times (RTTs), if we run the regular TCP on both paths, the multi-path flow would obtain twice as much throughput as the single path flow, which is not fair. A straightforward

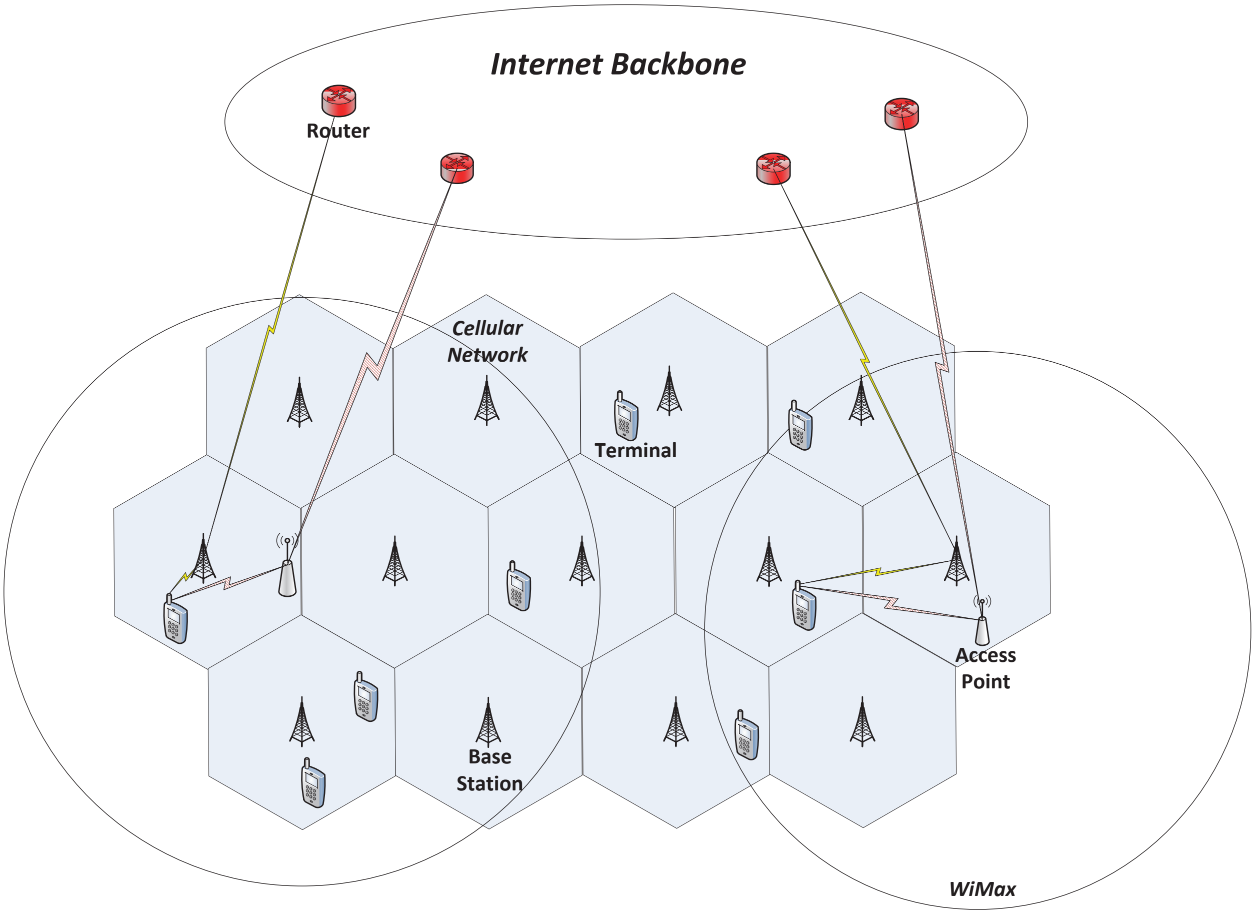


Figure 1.1: A heterogeneous wireless environment containing cellular and WiMax networks

solution is to run weighted TCP on each sub-flow based on the available bandwidth, as discussed in [17]. However, for the paths with heterogeneous characteristics, this approach can degrade the transmission throughput [18]. To increase the multi-path transmission throughput from the transport layer perspective, different methods have been proposed in the literature [13, 14, 18, 19, 20, 21, 22]. However, existing protocols suffer from some limitations which have become our motivation for this thesis as discussed in the following.

1.3 Motivations and Objectives

Multi-path transmission is to achieve reliable end-to-end transmissions. That is, when a path fails, the connection interruption can be avoided by switching from that path to another one. However, switching – or handoff in a mobile wireless environment – between paths may cause intolerable delay and/or packet loss. Therefore, a well-designed mechanism for multi-path transmission is needed to provide required reliability [18].

As an alternative approach, flexibility in sharing resources in different networks is achievable through multi-path transmission by splitting the traffic into independent paths [15]. A mobile host can have access to the Internet via multiple access links such as Bluetooth, WiFi, and cellular network modems, each of which has its corresponding service provider. The choice of a network for a specific data transmission should be made based on factors such as bandwidth, throughput, latency, jitter, QoS requirements, cost, power consumption, interference, and traffic patterns [18]. The simultaneous use of multiple paths in data transmission to enhance the overall bandwidth available to a wireless node is an advantage of multi-path transmission in terms of high transmission rate, low packet loss rate, and low transmission delay. As a result, it is desirable to stream data packets across all interfaces simultaneously whenever necessary.

The issues associated with multi-path transmission can be addressed from an end-to-end transport layer point of view. Dealing with multi-path transmission at the transport layer has become attractive recently, as the lower layers cannot completely remove all differences in delay, packet reordering, and losses to make multiple paths seem like a single path from the transport layer viewpoint. Moreover, transport layer design has some unique advantages. Below the transport layer, shifts of traffic between paths cannot be controlled as the information is too coarse [23]. Therefore, the transport layer design is necessary for an efficient implementation of multi-path transmission.

Various proposals have been addressed in the literature over the past decade in the area of multi-path transport layer protocols. However, existing protocols suffer from difficulties in guaranteeing QoS, while seeking optimal resource allocation, managing delays, and having scalable additive increase-multiplicative decrease (AIMD) congestion control. The existing congestion control protocols for the multi-path transmission mainly focus on wired networks [24, 25] without addressing time-variant delay of each path, different RTTs of multiple paths, packet loss due to wireless channel fading and other heterogeneous wireless network characteristics. Moreover, the existing protocols have independent congestion control for each path, in which the overall transmission rate may decrease in a congestion scenario, even if some networks are under-utilized. The current transport layer protocols make it possible to transmit packets through multiple paths simultaneously, but they do not achieve maximal congestion avoidance. Therefore, a new congestion control protocol is required, which should be scalable, reliable, and stable for the future Internet and be suitable for a heterogeneous wireless environment. Although it is shown in [26] that the variety in network characteristics makes congestion control complicated, this variety is expected to be useful in congestion avoidance if other available networks are not congested and can cooperate with the congested network. More importantly, cooperation for congestion control among available networks can be useful not only in packet retransmission [21], but also in the actual packet transmission. The latter, which is the focus of this work, refers to sending some of the on-going packets through an uncongested network instead of the congested one in a congestion period, without decreasing the overall transmission rate or degrading QoS. Therefore, cooperation among available networks is expected to improve data transmission throughput, and to utilize resources more efficiently in each network.

In this research, our aim is to develop a cooperative congestion control algorithm for multi-path transmission of video streaming in a heterogeneous wireless environment. By means of dependent but separate congestion control procedures, unused resources of heterogeneous networks can be used for congestion avoidance. When congestion is predicted for one path, other paths will be notified to help the congested path by moving the traffic load away from the congested path. Hence, the CW reduction of a congested path can be compensated by the increased CW of other paths. In this way, the overall packet transmission rate does not decrease, packet loss is reduced, and satisfactory QoS can be achieved. Different from prior works that focus on wired networks with a constant RTT [18, 19, 21, 22] or similar RTTs for different paths [19, 21], this research focuses on heterogeneous wireless networks where different RTTs are expected for different paths.

For the cooperative congestion control, it is necessary to find proper time instances to initiate and terminate cooperation. Since a trade-off exists between congestion avoidance and cooperation cost, the congestion levels of the end-to-end paths should be estimated and compared to the calculated cooperation thresholds to decide whether or not to start/stop cooperation.

1.4 Outline of the Thesis

In this thesis, in Chapter 2, congestion control and its challenges in a wireless environment are summarizes. In Chapter 3, we provide an overview of the state-of-the-art of multi-path transmission techniques for QoS provisioning in wireless Internet access from an end-to-end transport layer perspective. The system model for the heterogeneous wireless environment under consideration is presented in Chapter 4. Cooperative congestion control algorithm development is studied in Chapter 5, where the problem of cooperation start/stop times is formulated and the proposed congestion control scheme is evaluated. Chapter 6 concludes the research and identifies the future research directions.

Chapter 2

Congestion Control in Wireless Networks

2.1 Congestion Control

Congestion can be expressed as an overload state in a network. Obviously, this definition cannot describe when exactly congestion happens and how long the network remains in the congestion state. Various descriptions in the literature specify the congestion state. The most useful descriptions are as follows:

- **Queuing theory definition** - Congestion happens when the arrival rate exceeds the service rate.

- **Networking definition** - Congestion is defined as running out of buffer space, at which point packet dropping starts. This definition agrees exactly with the TCP definition for congestion.

- **Practical data-base definition** - The load on a network over a certain period of time shows congestion happening. Thus, the mean link utilization describes the congestion level.

- **General definition** - If increasing the use of a service that is shared among a group of people imposes a cost on the existing users, this service is congested.

Each of the preceding definitions has its own strength and weakness. The queuing theory definition can be treated as a prediction of packet loss. However, in some rare scenarios, the arrival rate becomes larger than the service rate for a short period in which the state is falsely predicted as congestion. Therefore, the congestion prediction based on the queuing definition is not error-free. On the other hand, the networking definition is more realistic, but congestion cannot be prevented before it happens. The practical definition gives an insight into traffic and channel conditions. However, it is based on the average load, which cannot describe momentary congestion or a spike in data packet losses. The last definition seems to be too general, but it can be useful in cooperation towards congestion control, and is the only definition that takes other user conditions into account. In the following, the queuing theory definition is considered because the queuing model can describe packet transmission in heterogeneous networks. Moreover, the queuing theory definition can predict congestion and can be used in a congestion avoidance scheme. However, the other definitions are also incorporated in our congestion control proposal.

Figure 2.1 depicts a one-way data transmission with feedback for congestion control between two end terminals. An end-to-end path from the source to the destination, which is used for sending data packets, is called the forward path, while an end-to-end path from the destination to the source, which is used for sending ACK packets, is called the backward path. Using the received ACK information, the number of successfully received packets is estimated at the source. Then, the CW size is set based on the congestion control strategy.

The most popular congestion control strategy in the Internet is the original version of TCP with four functionality phases. One phase is chosen at a time based on the received feedback from the destination and some pre-defined parameters. The CW size is the most important parameter that shows the maximum number of packets that can be sent without being congested. Each sending piece of information is labeled by a sequence number. To

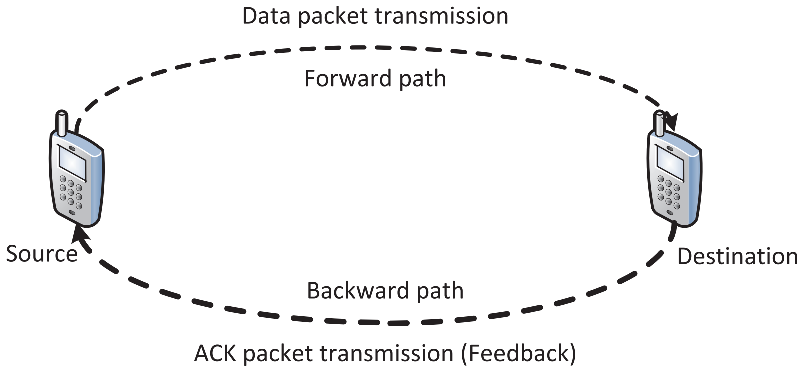


Figure 2.1: End-to-end data and ACK transmission

notify the sender that the data is correctly received, TCP has a simple but useful ACK mechanism. An ACK packet includes the sequence number of the last successfully-received packet. If an out-of-order packet arrives at the receiver, a duplicate ACK is generated. Three duplicate ACKs or no ACK during a timeout period are/is an indication of a packet loss.

Based on the provided parameters and TCP ACK strategy, TCP phases are performed. The two main congestion control phases are slow start and congestion avoidance. In the slow start phase, the initial CW size is set to one maximum segment size (MMS) and is incremented by one MMS on each new ACK. When the CW reaches the preset slow start threshold, the procedure enters the congestion avoidance phase which increases CW linearly. If timeout happens, the slow start threshold is set to the half of current transmission window size and the CW size is reduced to one MMS. Then, the slow start phase starts again. In the case of three duplicate ACKs, TCP performs fast retransmission phase. This mechanism allows TCP to avoid a lengthy timeout during which no data is transferred. If the ACK is received in about one RTT after the missing packet is retransmitted, fast recovery phase is entered. The CW size is set to slow start threshold and the congestion avoidance phase starts, because the feedback shows that the slow start is not needed anymore.

2.1.1 Congestion Control Categories

In order to study congestion control protocols, it is necessary to understand different types of congestion control protocols and to apply the most relevant and useful category for a target application. In the following, different congestion control categories are reviewed and compared.

- **Loss-based vs. Delay-based:** Most congestion control protocols can be categorized into loss-based or delay-based schemes. The congestion control in the current mainstream TCP (TCP-Reno) is loss-based, meaning that it reacts to packet loss occurrences indicated by ACKs from the destination. The congestion measure is the detected packet loss from the feedback for a loss-based scheme, and the queuing delay for delay-based congestion control such as Fast-TCP. Delay-based congestion control schemes have been shown to outperform loss-based approaches at higher transmission rates [27]. However, for a large CW, the queuing delay is not an accurate predictor of congestion level. In a network with a large bandwidth-delay product, using a delay-based protocol to augment the basic AIMD of TCP is not a proper approach. Instead, a fully delay-based protocol can be useful [27], where congestion loss rarely happens, and the queuing delay can be estimated more accurately. Another advantage of delay-based design in wireless networks is the ability to distinguish random loss due to dispersive fading channels from the one due to congestion loss.

- **Packet-level vs. Flow-level:** Congestion control can also be classified into packet-level and flow-level approaches. The flow-level perspective has a macroscopic view of the congestion control. It aims at achieving high resource utilization, low queuing delay and loss, proper fairness, and stability. The packet-level design implements the flow-level goals within the constraints imposed by end-to-end control. Historically, in congestion control protocol development, such as in TCP-Reno, the packet-level control is first developed, and then the flow-level control is added to achieve the required stability and fairness. In more recent protocols such as Fast-TCP [28] and stream control transmission protocol (SCTP) [14], the packet-level design is considered after the flow-level design [27, 29].

- **Window-based vs. Rate-based:** Congestion control protocols can also be categorized from another viewpoint into window-based or rate-based protocols [29]. Window-based congestion control protocols are mainly based on the generic AIMD algorithm or other approaches with linear CW growth. All strategies in window-based protocols are to find the best increment and decrement steps given the CW size. On the other hand, rate-based congestion control is equation-based, which finds the maximum acceptable data rate according to a recent loss rate. Thus, the sender updates its transmission rate based on the control equation. Generally speaking, developing a rate-based congestion control algorithm is more complicated than developing a window-based one [27].

- **Multicast vs. Unicast:** Based on the application, a congestion control protocol needs to be unicast for a one-source to one-destination flow, or multicast for one-source to many-destination flows in the Internet. Multicast congestion control is more challenging than unicast as traffic should be distributed along many paths to different destinations. Thus, a multicast scenario is similar to the multipath TCP in terms of dealing with more than one path, but is completely different from multi-path TCP in terms of the number of destinations.

In this work, a window-based unicast scheme that uses queuing delay as the congestion measure is considered. Both packet-level and flow-level perspectives are used for cooperation-time calculation as both need to be studied for every congestion control proposal to control packet transmission in one RTT and to manage the packets in a flow, simultaneously.

2.2 Congestion Control Challenges in a Wireless Environment

Generally speaking, performance of the popular congestion control protocol for the Internet, i.e. TCP, degrades in mobile wireless networks. The degradation is because, in a wireless network, transmission errors or packet losses happen not only due to congestion, but also due to other error sources such as channel fading, mobility, and channel contention nature of wireless networks, which are mostly unavoidable. However, the TCP cannot distinguish among different error sources and all transmission errors are attributed to network congestion. Thus, TCP decreases its CW size in response to all kinds of packet loss unnecessarily, leading to a performance degradation. Therefore, in order to develop a congestion control protocol for a wireless environment, different challenges such as single and burst packet loss and power limitation issues should be addressed.

In wireless networks, a fading dispersive channel can cause a high bit error rate. When a data packet is lost, a timeout happens or duplicate ACKs are generated. Then, TCP decreases its CW unnecessarily. Moreover, a large number of retransmissions are scheduled in this case, which consumes power and increases the link traffic. Also, in a wireless network, a connection should be kept alive as a user roams around. In an infra-structured network, when a mobile host wants to leave the coverage of a base station or access point and connect to another one, some packets can be lost during the handoff. In general, due to limited radio coverage and user mobility, frequent handoffs occur. Thus, the link experiences a short disconnection during a communication session. It is shown that a short disconnection can stall the TCP during a long period even more than the disconnection time [30].

Another characteristic of wireless networks which may lead to packet loss is the broadcasting nature of wireless signals which can lead to channel contention. Because of broadcasting, signals may interfere with each other. Hence, a collision is sensed and transmission may fail. This problem is more common in TDMA-based multi-hop wireless networks.

Another challenge of TCP is the burst packet loss. In channel deep fading, more than one packet can be dropped. It is shown that more than one packet loss in one round trip time can degrade TCP performance significantly. Moreover, mobility can cause burst packet loss. In an infra-structured network such as a cellular network, a mobile host may leave one cell and enter another one. Thus, it should be disconnected from previous base station and then connected to the new one. Some packets may be lost in the handoff process. In ad-hoc networks, network partitions and any change in routing can cause packet loss for a short duration, which leads to a burst loss.

Another wireless network property is the limited power and energy. It has been shown in [30] that the total consumed-energy for TCP decreases by increasing its goodput. Therefore, to conserve energy, it is very crucial to minimize the number of transmissions and perform the needed operations in an efficient manner.

TCP performance degradation is different among various wireless networks. In infra-structured networks, a high bit error rate is not negligible. Moreover, frequent handoffs by mobile hosts cause burst losses in wireless networks. In satellite networks, the link is capable of sending at a high data rate. Therefore, the slow start phase in TCP is not efficient for this type of networks, as it takes too much time to reach the high bit rate of satellite networks. In addition, there are long delays in satellite networks because of long distances between a source and a destination. In ad-hoc networks, along with a high bit error rate, network partitioning and changes in routing affect TCP goodput. Accordingly, many network-based congestion control protocols have been proposed in the literature [31, 32, 33].

2.3 Congestion Control Protocols for Wireless Networks

In order to improve the TCP performance in mobile wireless networks, several solutions have been proposed in the literature [34, 35]. There are also non-TCP based approaches such as wireless application protocol (WAP 1.0 or WAP 2.0 [36]), that are designed basically for flow control in mobile wireless networks. However, TCP-based protocols are more popular as they act fairly when there are coexisting TCP flows in the wired Internet [37].

TCP-based congestion detection approaches differentiate the random loss and burst loss from congestion loss, and can be categorized into reactive and proactive schemes [35]. The reactive congestion detection approaches use the feedback provided by ACKs to calculate the TCP parameters. Then, based on the current TCP parameters, the CW size is updated. Proactive congestion detection, on the other hand, uses the network condition to estimate the path capacity, end-to-end delay and other parameters. Then, the flow rate is managed based on the TCP state. All these approaches improve the congestion control performance in wireless scenarios, but do not completely eliminate the performance degradation in the wireless domain. Among reactive congestion control approaches, TCP-Probing modifies TCP at the source terminal. A significant shortcoming in the TCP protocol is entering the retransmission phase unnecessarily when three duplicate ACKs are received or a time-out occurs. To address this problem in TCP-probing, instead of entering the retransmission, two probing packets called prob1 and prob2 are sent through the link to ensure about the source of packet loss. Using the prob1 and prob2, RTT1 and RTT2 are calculated and compared to the original RTT. If both RTTs are less than the original RTT, the channel condition is acceptable (congestion is not probable), and thus TCP continues to control flow rate and ignores time-out or duplicate ACKs. If RTT2 or both RTTs are greater than the original one, fast retransmission should be started and CW should be decrease [38]. In this method, probing increases the traffic in the link.

State suspension approaches follow another TCP-based congestion control strategy for wireless networks in which congestion control is suspended for a specific time. TCP-Delayed Congestion Response (TCP-DCR) is a state suspension approach that improves the TCP robustness to non-congestive events. In this method, the receiver waits for an RTT after the first duplicate ACK is sent. Thus, it is more likely that out-of-order packets are received at the receiver in this waiting time due to different packet delays in the wireless link. In this way, unnecessary retransmissions due to reordered packets at the destination are avoided. Simulation results show the improvement of performance using this method as compared to previous ones in delayed wireless networks [39]. It is worth nothing that adaptive waiting-time can improve TCP-DCR performance in different networks especially in ad-hoc networks.

A successful proactive approach is the TCP-Westwood which can be used in both wired and wireless networks [40]. TCP-Westwood monitors the rate of acknowledged data to control the CW size. When a new ACK is received, the amount of acknowledged data is used to estimate the bandwidth of the link. The estimated bandwidth of the connection is obtained by applying a discrete-time low-pass filter. Then, the slow start threshold is calculated as the product of the estimated available bandwidth and the minimum RTT sampled throughout the duration of the connection divided by the packet size. This method improves the performance of congestion control in the presence of non-congestion losses.

As mentioned before, congestion control performance degradation varies among different wireless networks, and some congestion control protocols are proposed for a specific network. For example, the TCP-Peach algorithm is proposed to compensate for TCP performance degradation in satellite networks due to long path delays and a high bit error rate [31]. In TCP-Peach, two phases of regular TCP are changed. The slow start phase is replaced with fast start, and fast recovery with rapid recovery. Dummy packets are added to get more information about available network flow rate. To this end, a low-priority dummy packet is sent along with data packets. If congestion happens, the dummy packet will be dropped first. Therefore, receiving ACK for the dummy packet ensures TCP about more available bandwidth. If the TCP phase is fast start, the transmission window will be increased rapidly due to correctly transmitted dummy packets. Hence, the high data rate of satellite networks can be achieved in a short time. The same situation holds in the rapid recovery in which the ACKs of dummy packets allow TCP to increment the CW size rapidly and continue sending data. This method is reactive, and congestion happens before the CW modification.

Furthermore, dummy packets cause more frequent congestions. TCP-Peach also increases the traffic of the link without sending more information. Therefore, the goodput is not improved as much as expected.

TCP-Veno is an strategy that deals with random loss in infra-structured networks. The backlog accumulated along the communication path is estimated first [32]. If it is less than a threshold, a non-congested link can be assumed. If extra packets are more than the threshold, congestion is assumed. This backlog is estimated as the product of minimum measured RTT and the differences between the expected and actual rates. The expected rate is the CW size divided by the minimum measured RTT and the actual rate is the CW size divided by the smoothed measured RTT. As a drawback, TCP-Veno resumes decreasing the CW size in burst loss, which degrades performance for that case.

TCP-Feedback uses routers' assistance to improve the TCP performance in ad-hoc networks due to route failures. In ad-hoc networks, there are some intermediate mobile hosts between the source and destination. A node, which becomes aware of the next mobile host disruption, transmits a route failure notification to the source. Every node that becomes aware of this notification, stops packets from being forwarded in this route and uses an alternate route if available. The TCP discards all timing variables such as time-out counter and stops sending packets until it assures about an alternate route. This happens when the failure node sends another notification about another available route. Then, the TCP restarts and all unacknowledged packets are retransmitted first [33]. Unlike previous methods, TCP-Feedback can handle any wireless route disruption along the transmission path. However, it creates burst traffic immediately after the connection reestablishment, which can cause packet congestion. It is worth noting that all state suspension approaches suffer from time varying behaviours of wireless networks. This is because they suspend TCP and continue transmission after reconnection using the previous link flow rate and RTT, which likely are not valid for the current link.

Receiver-Assisted Congestion Control (RACC) has been proposed to improve the throughput of congestion control in high-speed lossy wireless networks by integrating the loss-based and delay-based schemes [41]. In this protocol, the source terminal sets the CW size based on the AIMD scheme. At the same time, the destination helps to choose a proper CW size by providing a delay-based bandwidth estimation. Simulation results show that RACC outperforms TCP in high-speed wireless networks.

In some congestion control protocols for wireless networks, routers are employed to distinguish random loss from congestion loss. TCP-Jersey is a congestion detection approach that uses routers to implement the approach [42]. It uses the same idea as TCP-Westwood to estimate the bandwidth of connection, but the estimation is calculated in a simpler way. TCP-Jersey uses congestion warning in congestion-notification routers to distinguish random errors from congestion. The router marks all congested bits in the IP header when the average queue length exceeds a given threshold. It is shown that TCP-Jersey performs friendly to TCP. Simulation results show that TCP-Jersey works better than all previous discussed methods in wireless networks even for high bit error rates. However, it has poor performance in the presence of burst loss. TCP-Casablanca applies a biased queue management to de-randomize congestion loss, such that it can be distinguishable from random loss based on their different distributions [43]. The streams of data which are ready to be sent to the destination are labeled by “in” and “out” marks. Every transmitted packet is marked “out” and so are retransmission packets. When a router experiences congestion, it drops out-marked packets. In this way, most of the packet losses due to congestion are out-marked packets. But in a wireless network, the packet loss distribution is random. So, the congestion can be distinguished from random loss and rate decreasing is avoided in random loss. Simulation results show that this method can distinguish congestion loss from random loss with 95 percent of accuracy. However, TCP-Casablanca similar to TCP-Jersey needs routers to participate accurately in this strategy.

Mobile-host-centric transport protocol (MCP) has been proposed for a scenario where one end terminal is mobile [44]. This includes both source and destination terminals. The congestion control is managed from the mobile terminal using the local available information about the packet loss. Therefore, the CW size is set at the source or destination based on the mobile node. This protocol is mainly defined for the scenario that the mobile user is connected to the wireless network and the other end is a server in a wired network.

As discussed in the description of wireless congestion control protocols, every protocol focuses on providing a solution for a specific challenge in a wireless network. The discussed congestion control protocols and their differences are summarizes in Table 2.1.

Table 2.1: Congestion control proposals in the literature for a wireless network

Protocol Name	Modifid node	Application	Year
TCP-Peach [31]	Router & Source & destination	Satellite	2001
TCP-Probing [38]	Source	Wired/Wireless Networks	2000
TCP-Veno [32]	Source	Wired/Wireless Networks	2003
TCP-Westwood [40]	Source	Wired/Wireless Networks	2002
TCP-Jersey [42]	Router & Source	Wired/Wireless Networks	2004
TCP-Casablanca [43]	Router & Source & destination	Wired/Wireless Networks	2005
TCP-Feedback [33]	Router	Ad-hoc	2001
TCP-DCR [39]	Destination	Delayed wireless networks	2005
MCP [44]	Source & destination	Wireless/Wired Networks	2007
RACC [41]	Source & destination	High-speed wireless networks	2010

2.4 Summary

In this chapter, we first introduce congestion control definition and categories. Then, we discuss congestion control challenges in a wireless network and review some protocols proposed for congestion control in a wireless scenario with significant improvements. Every proposed congestion control protocol improves the throughput from an aspect which may not address other challenges of wireless networks. Although the throughput of congestion control in wireless networks is improved using the proposed protocols, proposing a TCP-friendly congestion control protocol that works efficiently in wired/wireless scenarios is still an open research problem.

Chapter 3

Overview of Multi-path Congestion Control

This chapter summarizes the existing works on multi-path congestion control and their advantages and limitations. Major challenges in end-to-end congestion control for a heterogeneous wireless environment are also discussed. Gathering this information from the literature gives an insight into important problems that need further research.

3.1 Congestion Control Challenges for Multi-path Transmission

In order to establish a reliable end-to-end multi-path connection, some requirements should be satisfied at the transport layer. First of all, as the TCP does not support multiple interfaces with multiple IP addresses, a simultaneous connection to multiple paths needs to be provided. Then, concurrent packet transmissions through those paths are requested. Second, in multi-path transmission, the probability that the packets reach the destination out-of-order is significant. Third, fairness is not often achieved in the coexistence of flows in multi-path and single-path transmissions. These challenges are discussed in the following.

3.1.1 Concurrent Multi-path Transmission

For a multi-path transmission, the multi-homing is a feature that enables a transport layer's association with multiple IP addresses at each end of the association. This binding allows a source to transmit data to a multi-homed receiver through different destination addresses [45]. Thus, multi-homing is essential for simultaneous connections in heterogeneous networks. The first reliable transport layer standard to support multi-homing is the SCTP [46]. The SCTP is standardized by the Internet engineering task force (IETF) as a reliable transport protocol [14]. It has many important features of the TCP such as window-based congestion control, error detection, and retransmission along with multi-homing and multi-streaming. In the SCTP, multi-homing is enabled by letting two endpoints set up a connection with multiple IP addresses (an association) for each endpoint. One of those addresses is labelled as the primary and the others are as backup addresses. This capability enables the SCTP to communicate between two endpoints using multiple links. However, the SCTP does not support simultaneous transmission through the multiple paths and, therefore, cannot fully benefit from multiplicity of the available networks.

Utilizing the available paths for simultaneous transmission of data packets can be achieved at the transport layer through concurrent multi-path transmission [21], which is a concurrent transfer of new data from a source to a destination via two or more independent paths. The idea of concurrent multi-path transmissions is to use the multi-homing feature to distribute data across multiple end-to-end paths. It is used in recent multi-path congestion control protocols such as [21, 47, 48].

3.1.2 Packet Reordering

Another challenge for congestion control protocols in mobile wireless networks is packet reordering or out-of-order packet reception at the destination. Many congestion control protocols (such as the TCP) require a strict byte-order delivery that cannot tolerate various path delays. In heterogeneous wireless networks, packets experience different delays because of the disparity in network infrastructures. Also, propagation attenuation, shadowing and fading can result in different transmission delays in wireless networks. When packets are delivered with different delays, one packet may arrive later than its subsequent packets. This phenomenon is called out-of-order delivery and causes the congestion control protocol to reduce the transmission rate mistakenly. Moreover, as the packets received at the destination earlier should wait for the delayed packets to resume the original order, a long delay is expected in packet transmission. As a result, for multi-path transmission in heterogeneous networks, modifications are required to minimize packet reordering or its effect in the congestion control. To overcome the reordering problem in multi-path transmission, the adaptive load balancing algorithm (ALBAM) is proposed where the priority of choosing a path for packet transmission is given to the lowest-delay path. In this way, reordering is eliminated in multi-path transmission [49]. However, the throughput performance of packet transmission degrades due to this priority condition.

In a single-path transmission, out-of-order packet reception is mainly due to router activity pauses, parallelism in high speed routers to provide packet stripping, and link-layer retransmissions to recover losses in wireless networks [50, 51]. However, the packet reordering is ignored in most single-path congestion control proposals [52]. In our system model, packet reordering is more significant since different delays in various end-to-end paths are inevitable.

Packet reordering has major negative effects on congestion control throughput and end-to-end transmission delay. First, a delayed packet may be treated as a lost packet and cause the congestion control to retransmit the packets unnecessarily. Second, packet reordering results in incorrect congestion detection which is followed by decreasing the CW size, and leads to throughput degradation. Third, as an ACK packet is only sent for an in-order received packet, acknowledging several packets after receiving the delayed packet leads to a huge increase of the CW size. The latter causes the source transmitting a burst of packets which may lead to congestion in the network. Finally, the waiting time for a delayed packet to resume the original order of packets may violate QoS requirements of real-time services. In our system model, we focus on video streaming, unnecessary packet retransmission does not happen. However, CW size reduction, burst of packets transmission, and unacceptable delay need to be considered in cooperative congestion control.

Although the existing reordering reduction schemes improve the congestion control throughput significantly, their performance in multi-path scenarios is degraded due to the time-variant end-to-end delay of each path [50]. Therefore, the existing reordering reduction schemes are not applicable to heterogeneous wireless networks.

3.1.3 Service Fairness

Service fairness is essential for every congestion control protocol. Fairness from an end-to-end viewpoint is achieved if, at the equilibrium, the considered network resource such as available bandwidth is shared fairly among the sources, using only the information available to the end hosts and without any help from intermediate nodes or routers [53]. Different definitions have been proposed for the fairness in congestion control protocols [9, 53, 54, 55].

The possibility of a congestion control protocol to be fair is studied in [56]. For single-path transmission, max-min flow rate fairness is proposed in [54] and widely used in the networking research community. Later, a new definition of fairness is proposed based on a Nash arbitration scheme which resulted in proportional fairness [57]. However, it is also a flow rate allocation scheme. Weighted proportional fairness is then proposed in [55]. In this method, although the fairness can be expressed as flow rate allocation, the fairness definition is to share the congestion cost among the bits instead of flows. Fairness for the TCP and other single-path protocols are mostly considered based on the fair flow rate schemes [58, 59]. However, in most congestion control protocols, fairness can be achieved at the cost of an end-to-end delay increase [60]. In another fairness definition, a source-destination pair can have a non-TCP end-to-end congestion control protocol in the Internet, if it acts fairly to the TCP. This kind of fairness is called TCP-friendliness and means that a protocol should behave the same way as the TCP from the traffic viewpoint, such that the average throughput of a non-TCP supported flow remains around the average throughput of a TCP flow [61]. Therefore, the protocols can be deployed in the Internet without much concern about fairness to other traffic [62].

Although several congestion control protocols are fair in a single-path transmission, they are not fair in a multi-path scenario with independent congestion control for each path. Unfair resource allocation happens when a multi-path flow co-exists with a single-path one. In such a scenario, multi-path association with K paths takes the bandwidth, which is K times of a single-path bandwidth.

3.2 Multi-path Congestion Control Protocols

Recent transport layer protocols for congestion control have made it possible to transmit packets through multiple paths simultaneously. All multi-path congestion control protocols have the minimum requirement of multi-homing capability. The proposed protocols can be divided into SCTP-based and TCP-based solutions. Some of the most important protocols are summarized in Table 3.1 and are discussed in the following.

Table 3.1: Multi-path transport layer proposals in the literature

Protocol Name	Characteristics	Year
Multi-homed TCP [63]	Enabled multi-homing for TCP	2003
LS-SCTP [19]	Load Sharing	2004
pTCP [18]	Two parallel paths	2005
CMT-SCTP [21]	Concurrent multi-path transmission	2006
COUPLED [64]	Rate-based multi-path	2007
EWTCP [65]	Weighted TCP for each path	2009
CMT/RP [62]	Concurrent multi-path and resource poolings	2010
WM2-SCTP [66]	Concurrent multi-path transmission with parallel subflows	2010
MP-TCP [26]	Heterogeneous RTTs	2011

3.2.1 SCTP-Based Solutions

The SCTP standardization with the multi-homing capability has been followed by various proposals for SCTP-based multi-path congestion control. In multi-homed algorithms for the transport layer, concurrent multi-path transmission cannot be performed efficiently by itself because of the significant packet reordering observed at the destination. Also, since window-based congestion control increases the CW for a path only when the sequence number of an incoming ACK is greater than that of the previous ACK, the CW grows too slowly. To resolve the inefficient load sharing problem, different approaches have been suggested.

Load sharing SCTP (LS-SCTP), an SCTP-based load sharing technique, is proposed in [19], in which the congestion control is performed on a path basis, while the flow control is on an association basis. Thus, both source and destination endpoints use their association buffers to hold the data packets regardless of their transmission paths. As congestion control is performed on a per path basis, the source has a separate congestion control for each path. This setup provides the sender endpoint with a virtual CW size equal to the aggregate of the CWs of all the paths within the association. The LS-SCTP has separate congestion control for each path, while taking into account the overall CW size of all paths. The sum of CWs over all the available paths provides some information about the environment, but the congested paths cannot be distinguished using this information.

The CMT-SCTP, on the other hand, distributes data packets across multiple end-to-end paths [21]. It consists of three sub-algorithms to overcome the packet reordering side-effects. The CMT schedules new data packets to different paths as bandwidth becomes available on the corresponding paths, i.e., if the corresponding CWs allow it to do so. When a CW space is available simultaneously for two or more destinations, data packets are sent to them in an arbitrary order. Using the full bandwidth of a path before using other paths is just to reduce reordering. The CMT-SCTP also facilitates concurrent multi-path transmission by distributing packets based on CWs of the paths.

A resource pooling scheme, i.e. distribution of traffic along available paths, for CMT-

SCTP is the concurrent multi-path transmission/resource pooling (CMT/RP) [62]. In this protocol, three main objectives are set for congestion control based on CMT-SCTP and the idea of resource pooling [47]. First, a CMT/RP flow can have a throughput gain over a single-homed flow. Thus, it should get at least as much bandwidth via the best path as a single-homed flow. Second, a CMT/RP protocol should be fair. It should not take more bandwidth on a shared bottleneck path than a single-homed flow via the same bottleneck. Third, resource pooling should be carried out in such a way that a CMT/RP flow should balance congestion on all of its paths. Aiming at these goals, the slow start thresholds are employed as a useful metric for the available bandwidth of paths. The CW is increased based on the normalized slow start threshold for similar link characteristics.

Although the SCTP was initially designed as a transport protocol for wired networks, there are many research activities in the application of SCTP to wireless mobile networks. The SCTP suffers from random loss in wireless networks, just as the TCP does. Thus, TCP-based strategies for wireless networks may be applicable for SCTP-based protocols. An extension of the SCTP for concurrent multi-path transmission with parallel subflows is wireless multi-path multi-flow SCTP (WM2-SCTP) [66], which allows the streams to be grouped in subflows based on the required QoS. In this approach, both flow control and congestion control are performed based on subflows instead of association. Thus, a separate source buffer is assigned to each subflow to make it independent from other subflows. Moreover, in order to support mobility, a performance improvement method is proposed for mobile SCTP in integrated heterogeneous networks [24].

3.2.2 Multi-path TCP-based Solutions

Multi-path transmission through TCP-based protocols allows one data stream to split over multiple paths in transmission, which improves reliability, such that a connection can be maintained even if some of the paths fail. In the following, we discuss some algorithms proposed for TCP-based multi-path transmission.

The Parallel TCP (pTCP) [18] is a bandwidth aggregation scheme that strips data over multiple paths at the transport layer, regardless of previous processes in the link and application layers. The pTCP is composed of a stripped connection manager (SM) and TCP-virtual (TCP-v). The TCP-v controls one path, independent of other paths, by probing the path, detecting loss, and carrying out loss recovery. The SM manages independent TCP-v's. These two functions lead to intelligent congestion control for each path. However, flow control and congestion control are managed by a centralized algorithm, which makes the method complicated. Furthermore, high resource usage due to implementation of one TCP for each path makes pTCP impractical.

In [48], the concurrent multi-path transmission is extended to the TCP by optimizing cost and performance to dynamically distribute data packets into multiple paths. The idea is to use concurrent multi-path transmission in the existing Internet transport layer protocols, mainly the TCP.

Another technique is equally-weighted TCP (EWTCP) [65], where multi-path congestion control is managed for each path separately. That is, the CW increases proportionally to a weighting parameter under the assumption that the RTTs of all paths are similar. However, in reality, different paths have different RTTs due to different routings (different path lengths) and/or different technologies in heterogeneous networks. The COUPLED algorithm in [64] is a window-based TCP, derived from a rate-based multi-path version of Scalable-TCP [67]. It follows the idea that a multi-path flow should shift all its traffic onto the least congested path. It is shown in [65] that the goal can be achieved in theory without any need to separate measurement of congestion on each path. Thus, the CW size for each path changes based on the overall CW size. However, similar to EWTCP, the COUPLED algorithm suffers from differences in RTTs.

Finally, multi-path TCP (MPTCP), proposed in [22], is a protocol based on a more realistic multi-path congestion control algorithm for the Internet according to the CWs and RTTs of all paths. An end-to-end algorithm for sharing capacity is proposed with some modification to the TCP. It is assumed that the TCP controls the traffic to be sent on each path, but does not perform resource allocation to specify the paths. By studying the COUPLED and EWTCP algorithms, it is concluded that the least congested path should be used, while keeping sufficient traffic on the other paths. In TCP, insufficient traffic means insufficient feedback. Thus, the SEMICOUPLED algorithm [22] is proposed based on the two congestion control requirements that the CW should be increased based on the overall CW with some weightings and decreased based on its own path's CW for each received ACK.

3.2.3 Limitations of Multi-path Congestion Control Protocols

The existing multi-path congestion control protocols aim to use the excessive resources available in multiple networks to increase the transmission rate. However, they do not employ these given excessive resources for congestion avoidance. All the congestion control protocols for multi-path transmission apply independent congestion control to each path in the sense that congestion may happen in each path independent of other paths, as in a single path transmission. Furthermore, the existing multi-path congestion control protocols have two other limitations: sensitivity to the transmission parameters of the paths, and weak recovery from congestion.

Each transmission path between a sender and a receiver in a multi-path scenario has various parameters such as delay, which are not necessarily the same among all the available paths. However, most of the proposed congestion control protocols for multi-path networks assume that the delay, for example, is approximately the same in all paths. Unfortunately, the performance of the protocols degrades when there is a small difference in the delays. The RTT is another path parameter which is assumed to be the same for different paths in many previous works [18, 21, 63]. However, the assumption is not true generally in a heterogeneous wireless environment. Only the MPTCP protocol deals with different RTTs, at a cost of increased complexity. Therefore, how to deal with the differences in the parameters of heterogeneous networks needs to be further studied.

Another limitation in previously proposed protocols for multi-path transmission is the weak recovery from congestion. As the proposed methods so far perform congestion control independently on each path, recovery from a congestion scenario takes a long time similar to recovery from a congestion scenario in a single-path transmission. However, it is expected that a more intelligent algorithm can benefit from excessive resources given in a heterogeneous environment such that the congested network can recover from a congestion scenario sooner with less throughput performance degradation.

3.3 Summary

This chapter reviews various existing methods of multi-path transmission from an end-to-end transport layer viewpoint and for wireless Internet access. Multi-homing as a necessary condition for multi-path transmission and concurrent multi-path as the best strategy to improve performance have been considered. Furthermore, congestion control methods have been reviewed, including both SCTP-based and multi-path TCP-based protocols for simultaneous data transmission. However, the existing methods suffer from serious limitations. First, the excessive resources of multiple networks are not used for congestion avoidance. Second, they are very sensitive to disparate networking conditions in a heterogeneous scenario which is the case for multi-path transmission. Third, congestion happens in the Internet access, and the recovery from the congestion state requires network resources.

In particular, cooperation in end-to-end congestion control can be helpful in various ways. Here, we consider the cooperation as a congestion prevention method without losing service quality. Almost all the existing multi-path congestion protocols treat the available paths as independent paths with separate congestion control decisions. As different paths have different characteristics, it is possible to make use of this diversity in a cooperative way. Therefore, the traffic can be moved from congested paths to non-congested ones based on the congestion level.

Chapter 4

System Model

The system model under consideration is composed of three components, namely heterogeneous networks, traffic model, and end-to-end congestion control, as introduced in Subsection 4.1. First, in the heterogeneous network configuration, an end-to-end connection between a source and a destination in a heterogeneous wireless environment is specified, in which the wireless networks are connected to the Internet backbone. Second, the considered traffic for communication between a source terminal[1] and a destination terminal is described. Third, for an end-to-end path in the heterogeneous networking environment, the parameters and characteristics of our assumed congestion control are described. The interaction among the three system model components is described in Subsection 4.2. The interaction model captures the effect of multiple heterogeneous networks and the network traffic on an end-to-end transmission. This effect is modelled as an end-to-end queuing delay in the congestion control algorithm.

[1] Here, the terminal is any device used by an end user to communicate with another end user through the existing network(s).

4.1 System Model Components

In the system model, an end-to-end connection is assumed to have been established between any pair of source and destination terminals, which are connected to heterogeneous wireless networks. This end-to-end connection in the heterogeneous environment, the type of service that is considered for communication between the end terminals, and the congestion control model are described in the following.

4.1.1 Heterogeneous Network Configuration

Consider a heterogeneous environment consisting of a set $\mathbb{N} = \{1, 2, ..., N\}$ of heterogeneous IP-based wireless access networks, which are connected to the Internet backbone. The wireless access networks have different wireless technologies and infra-structures. A set $\mathbb{M} = \{1, 2, ..., M\}$ of terminals are assumed to be connected to these networks. Each terminal is able to identify the available networks at its current location, i.e., the terminal is located in the coverage area of one or more than one network. Therefore, a set $\mathbb{K}_m \subseteq \mathbb{N}$, $\|\mathbb{K}_m\| = K_m$, of access networks are available to the mth ($m \in \mathbb{M}$) terminal, where $\|.\|$ denotes the size of a set. The terminal connects to all K_m access networks, simultaneously. The capability is provided by the multi-homing feature of the terminals. Thus, each terminal can transmit data through one or more networks in parallel.

In this work, it is assumed that two wireless networks are available to the source, i.e. $\mathbb{K} = \{1, 2\}$, and another two wireless networks are available to the destination, i.e. $\mathbb{K}' = \{1', 2'\}$. The best combination of networks at the two end terminals is assumed to be provided by the network selection strategies [68], and is available in our system model. Therefore, for two available networks to the source and two available networks to the destination, two forward paths and two backward paths can be established between the source and the destination. Each path consists of three segments: a wireless network connection to the source from the set $\{1, 2\}$, the Internet backbone, and a wireless network connection at the other end from

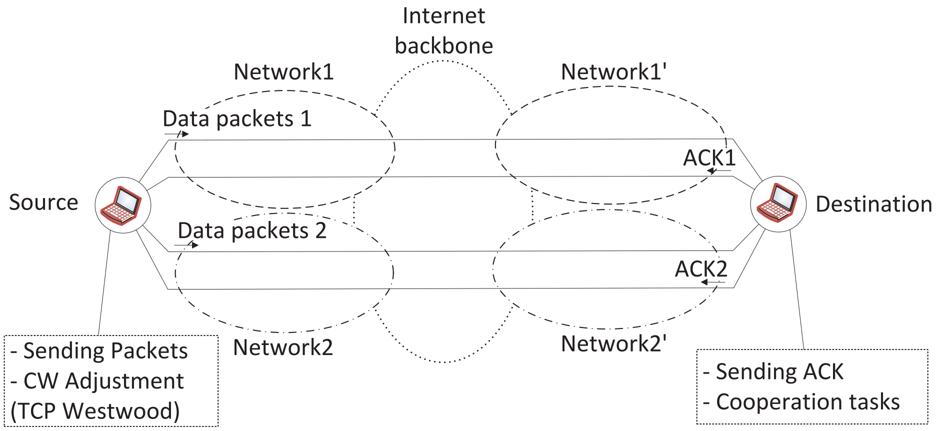


Figure 4.1: End-to-end paths between two end terminals in a heterogeneous wireless environment

set $\{1', 2'\}$, as shown in Figure 4.1. For any successfully transmitted packet, one ACK is sent to the source from the destination through the backward path of the same networks. Moreover, with the multi-homing capability, concurrent multi-path transmission is provided for the end users, such that a flow stream can be divided into two sub-flows transmitted through the two end-to-end paths. The concurrent multi-path transmission can make use of cooperation as described in the following.

For the kth end-to-end path, the aggregate delay is assumed to consist of two types of delays: queuing delay and propagation delay, denoted by q_k and d_k, respectively[2]. Queuing delay, q, is the delay due to the waiting time in the network buffers. All other types of delay in the end-to-end path, due to the wireless and wired networks, is considered as a propagation delay over a physical distance.

In addition, a single bottleneck is considered for an end-to-end path which is assumed to be in the wireless domain. The capacity of an end-to-end path bottleneck, which is the transmission rate limitation in the path, is denoted by c, and assumed to be known to the end terminals. Therefore, in terms of end-to-end congestion control, each end-to-end path is described by a single wireless network bottleneck and is modelled by a queuing system.

[2] Index k, whenever specified, denotes the considered path k.

Therefore, K parallel queues describe an association of K end-to-end paths for a source-destination pair.

4.1.2 Traffic Model

Consider video streaming applications, which constitute a large portion of recent Internet services [69, 70]. A streaming service has a variable bit rate (VBR) and a long-lived flow. A minimum packet transmission rate, denoted by r_{min}, needs to be satisfied for each stream as a QoS requirement for streaming services. To satisfy the requested QoS, a reliable end-to-end connection is provided through a congestion control protocol.

Since successive video packets are highly correlated in video streaming, the packet arrival cannot be modelled by a Markov process. Moreover, the video streaming duration in a wireless network can be modelled by a Pareto distribution [71], which has the heavy-tailed distribution characteristics [72]. Therefore, considering the packet arrival process with a general distribution, a Pareto service time for video streaming in a wireless network, and a single server to serve the packets in each path, the network queuing model is a G/G/1 queue. In the network queue, $l(t)$ denotes the queue length which is the number of packets that have entered the network bottleneck and have not departed from it yet at time t.

4.1.3 End-to-end Congestion Control

Based on the heterogeneous wireless networking model introduced in Subsection 4.1.1, the end-to-end association for the source-destination terminals has K end-to-end forward paths for video packet transmission. It is assumed that congestion control is applied to each end-to-end path separately but not independently. The congestion control protocol under consideration is window-based, in which the CW size, denoted by ω, is updated at the source based on the received ACK from the destination. For every end-to-end path, the ACK is transmitted through a backward path in the same network as the forward-path.

Based on the ACK received at the source, the RTT, denoted by τ, which is the sum of the forward and backward path transmission durations, is measured at the source for a successfully transmitted packet. Moreover, the minimum RTT, denoted by τ_{min}, is set for the minimum measured RTT of the packets from the same flow. Let $T(t)$ denote the forward path transmission duration (TD) which is measured at the destination at time t. In addition, let $T_{min}(t)$ and $T_{max}(t)$ denote the minimum and the maximum measured TD until time t, respectively.

In order to analyze the congestion control algorithm, both flow-level viewpoint and packet-level viewpoint are considered. From the flow-level viewpoint, the behaviour of a set of flows is investigated on a continues-time basis, in which the measurement time is denoted by t and the packet transmission rate at time t is denoted by $x(t)$. From the packet-level perspective, the behaviour of a set of packets in their RTTs is studied, in which the index of a packet is donated by u [28]. In the latter, the sending time of packet u from the source and the receiving time at the destination are denoted by $S_p(u)$ and $A_p(u)$, respectively. Moreover, the number of sent packets that have not reached the destination yet, i.e. the number of packets in flight (PIF), is measured for CW size adjustment. Let $\psi(u)$ denote the number of PIFs when the index of last sent packet is u. The number of packets in a queue of a path is called backlog if the arrival rate equals the bottleneck rate of the path [28] and is denoted by $B_p(u)$ when the last transmitted packet is u. More information on the primary relation among the parameters from the packet-level perspective is given in Appendix A.

4.2 Interaction Model

In this subsection, the interaction between the heterogeneous wireless networks and congestion control is considered. A realistic model of congestion control in a heterogeneous wireless environment requires all instantaneous status of the associated networks, service requests, and traffic from all users to be taken into account, which leads to a very complicated con-

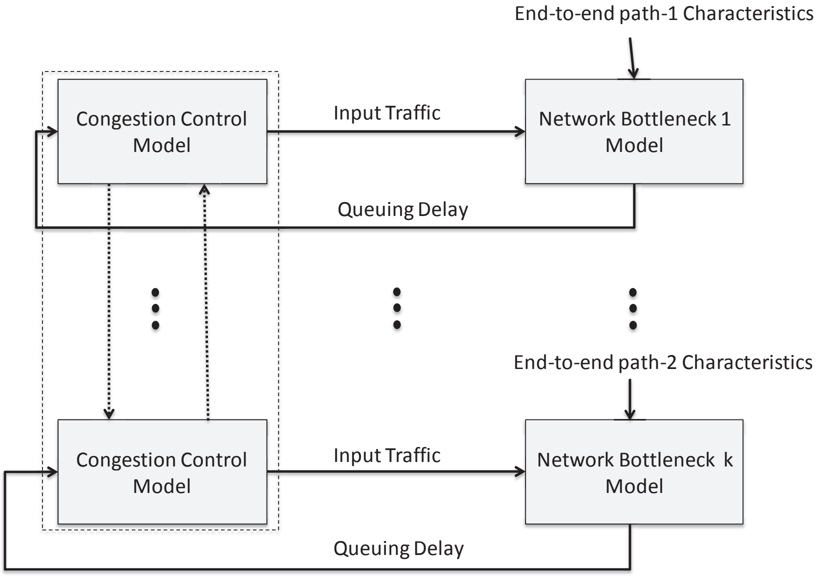

End-to-end path-1 Characteristics
Congestion Control Model
Input Traffic
Network Bottleneck 1 Model
Queuing Delay
End-to-end path-2 Characteristics
Congestion Control Model
Input Traffic
Network Bottleneck k Model
Queuing Delay

Figure 4.2: The interaction model of congestion control in a heterogeneous wireless scenario

gestion control analysis. Therefore, we consider a simplified model, while keeping a detailed description of aspects that are expected to have a major impact on the congestion control performance. To this end, we use a strategy called fixed-point method [73, 74], which has been proposed for a homogeneous network with loss-based feedback [75]. In this method, the system model is divided into two parts: congestion control model and network (network's bottleneck) model. We extend the same technique to the heterogeneous environment with K network bottlenecks (the bottlenecks of K end-to-end paths) with delay-based congestion control, as shown in Figure 4.2, and call it the *interaction model*.

For delay-based congestion control in one end-to-end path, the congestion control model and the network model interact with each other based on the input traffic and queuing delay. In the congestion control model for an end terminal, the congestion state of an existing end-to-end path between the source-destination terminals is calculated based on the queuing delay of the end-to-end path. Based on the congestion state, the congestion control protocol changes the CW size. The input of congestion control block is the queuing delay of the end-to-end path which is derived from the corresponding network model. The queuing delay represents the current condition of the end-to-end path and existing traffic flows from all other users in the wireless/wired networks. The outputs of congestion control block are the packets that are sent through the end-to-end path. Thus, the input traffic of the network model is controlled by the congestion control model. The detailed description of congestion control model can be defined or modified without changing the system model. For a cooperative congestion control solution, the detailed congestion control model is described in Chapter 5.

In the network model, for the bottleneck of a single end-to-end path, the G/G/1 queuing model is considered. Heterogeneous network characteristics (such as propagation delay) are the input parameters of the network model, and the queuing delay of the end-to-end path, which reflects the behaviour of all exiting traffic in the network and the end user requests, is the network model output.

For the heterogeneous wireless networks, as shown in Figure 4.2, the two-block (one congestion control block and one network block) model is extended to that with K congestion control blocks and K network blocks. As the heterogeneous networks in different end-to-end paths have different characteristics, they should be described in separate network blocks. Therefore, K parallel blocks represent the K end-to-end paths of an association, and each end-to-end path has separate congestion control. Note that, in our cooperative algorithm, the congestion control decisions are based on the status of all the paths. Thus, with one congestion control block associated with one end-to-end path, K separate but dependent congestion control blocks are needed.

4.3 Problem Definition

The objective of this research is to develop a cooperative transport layer algorithm that provides a reliable video streaming transmission through heterogeneous wireless networks.

By means of cooperation among available networks, this transport layer algorithm should enhance the QoS for the users. In order to develop a cooperative end-to-end congestion control algorithm for a heterogeneous wireless environment, with a focus on cooperation, we first select an available congestion control protocol from the literature as a congestion control protocol core. This protocol should be developed for wireless/wired scenarios and the CW adjustment strategy should be compatible with cooperation among heterogeneous wireless networks. Second, some information about the heterogeneous wireless networks needs to be provided to the end terminals, so that the network characteristics should be sent to the end terminals with minimum signalling overhead, such as network path propagation delay, end-to-end bottleneck capacity, and updated information about the current traffic state of the end-to-end paths. Third, a cooperation strategy should be established among available networks with minimum information exchange between the heterogeneous wireless networks. Finally, the cooperation strategy needs to be integrated with the chosen congestion control protocol, such that the uncongested paths are notified when one path is congested. The CW size of the uncongested paths should be increased, while the CW size of the congested path is decreased. A cooperative congestion control algorithm is proposed and discussed in more details in Chapter 5.

In cooperative congestion control, when congestion happens in one or more available paths, some of the on-going packets are sent over uncongested paths instead of a congested one, in order to improve the transmission throughput. Nevertheless, the throughput of cooperative transmission in an uncongested scenario can be less than that of non-cooperative transmission due to cooperation costs such as cooperation setup time, additional signalling for cooperation, and out-of-order packet reception [65]. In other words, a trade-off exists between congestion avoidance and cooperation cost and, thus, cooperation should be triggered only when it is beneficial according to congestion level measurements. This trade-off has led us to define a problem to find the best times to start cooperation and to stop it. As a result, cooperation starts only when it is necessary. This starting point can be after congestion has happened, after increased packet loss, after a significant delay or an unsatisfactory requested rate. This problem is considered in this work and a solution is provided in Chapter 5.

4.4 Summary

Development of a cooperative congestion control algorithm for heterogeneous wireless networks with the Internet backbone is our objective in this research. To this end, in this chapter, the system model under consideration is described, including heterogeneous networks, traffic model, and end-to-end congestion control. A single bottleneck of an end-to-end path in the heterogeneous network environment with the streaming services is modelled by a G/G/1 queue and the relation between the heterogeneous networking environment and congestion control is represented based on the interaction model.

Chapter 5

Cooperative Congestion Control

In this work, the objective is to develop a cooperative end-to-end congestion control algorithm for a heterogeneous wireless environment. Here, cooperation among networks is used to minimize the overall congestion over the whole association, while the end terminals are provided with satisfactory QoS.

In this chapter, a cooperative end-to-end congestion control algorithm for a heterogeneous wireless environment is presented in three steps. In Section 5.1, the cooperation requirements are discussed. Then, integration of a cooperation strategy with the selected congestion control protocol is considered in Section 5.2. Finally, finding the best times to start and stop cooperation is presented in Section 5.3. To demonstrate the performance of the proposed cooperative congestion control, simulation results are presented in Section 5.4.

5.1 Cooperation Requirements

For cooperative congestion control in the heterogeneous environment as illustrated in Figure 4.1, the congestion level of an end-to-end path for every source-destination pair is estimated by measuring the queuing delay (or queue length) of the end-to-end path and comparing it to cooperation thresholds. If an end-to-end path of a source-destination pair is predicted as a congested path and the other end-to-end path of the source-destination pair is uncongested, the corresponding networks – namely the congested network and the cooperator network – start cooperation.

In order to start cooperation, i.e. sending some of the on-going packets through the cooperator network instead of the congested one, the congestion control of the uncongested end-to-end path needs to be notified to increase its CW. In our cooperative algorithm, cooperation notification is sent to the source without modifying the congestion control protocol of the end-to-end paths. This is performed via sending an ACK, i.e., the ACK for a successfully transmitted packet in the congested network is sent through the backward path of both networks. The ACK is sent through the congested network as a regular feedback for packet reception, and is sent through the other network as a cooperation notifier, which leads to an automatic increment in the CW size.

For a cooperative congestion control algorithm, an existing congestion control protocol has been chosen as the congestion control core and, then, a cooperative algorithm is developed for a congestion scenario. Therefore, in the following, the congestion control protocol under consideration as a cooperative congestion control core is discussed. Afterwards, the responsibility of the two end terminals in cooperative congestion control and a congestion control Markov model are discussed.

5.1.1 Congestion Control Core

Among existing congestion control protocols in the literature, TCP Westwood [40] is chosen as the congestion control core in this work. This protocol improves the transmission throughput in wireless networks by changing the traditional CW adjustment of the TCP. In the TCP Westwood, instead of updating the CW size based on the number of lost packets, the received ACKs are monitored and the instantaneous achieved data rate for an end-to-end connection is estimated. Both the estimation and CW adjustment are performed at the source terminal.

Similar to other TCP-based protocols, the AIMD algorithm is used in the TCP Westwood

CW adjustment. Therefore, the TCP Westwood follows the linear increase of CW until reaching a threshold called slow-start threshold. The CW remains at this threshold unless congestion is detected, i.e. the ACK of a packet is not received in three RTTs or a time-out is reached [40]. In this scenario, the CW size is not halved based on TCP. Instead, the CW size of TCP Westwood is set as

$$\omega(t) = R(t) \cdot \tau(t) \tag{5.1}$$

where $R(t)$ denotes the estimation of the achieved data rate at time t. This rate is estimated by calculating the ratio of the number of acknowledged packets over the acknowledgement period that is after the previous estimation time and before the current time t. The slow-start threshold is also updated based on the $R(t)$ estimation. Therefore, TCP Westwood avoids unnecessary reduction in the CW size and slow start threshold, and improves the transmission throughput specially in a wireless environment.

The CW size of the TCP Westwood is proportional to the number of received ACKs such that by sending an ACK through the cooperator path as a cooperation notification, the CW size of the cooperator path increases and cooperation is established. The exact number of required ACKs to be sent through the cooperator path needs further investigation.

5.1.2 End-terminal Responsibilities

As the end-to-end congestion control procedure should be managed by the two end terminals, any necessary control to avoid congestion should be managed by the source and/or destination. Therefore, the source can control the CW of each path based on (5.1). Meanwhile, cooperation-related measurements and decisions on whether or not to start cooperation should be made at one of the two end terminals. The destination terminal is chosen for the following reasons.

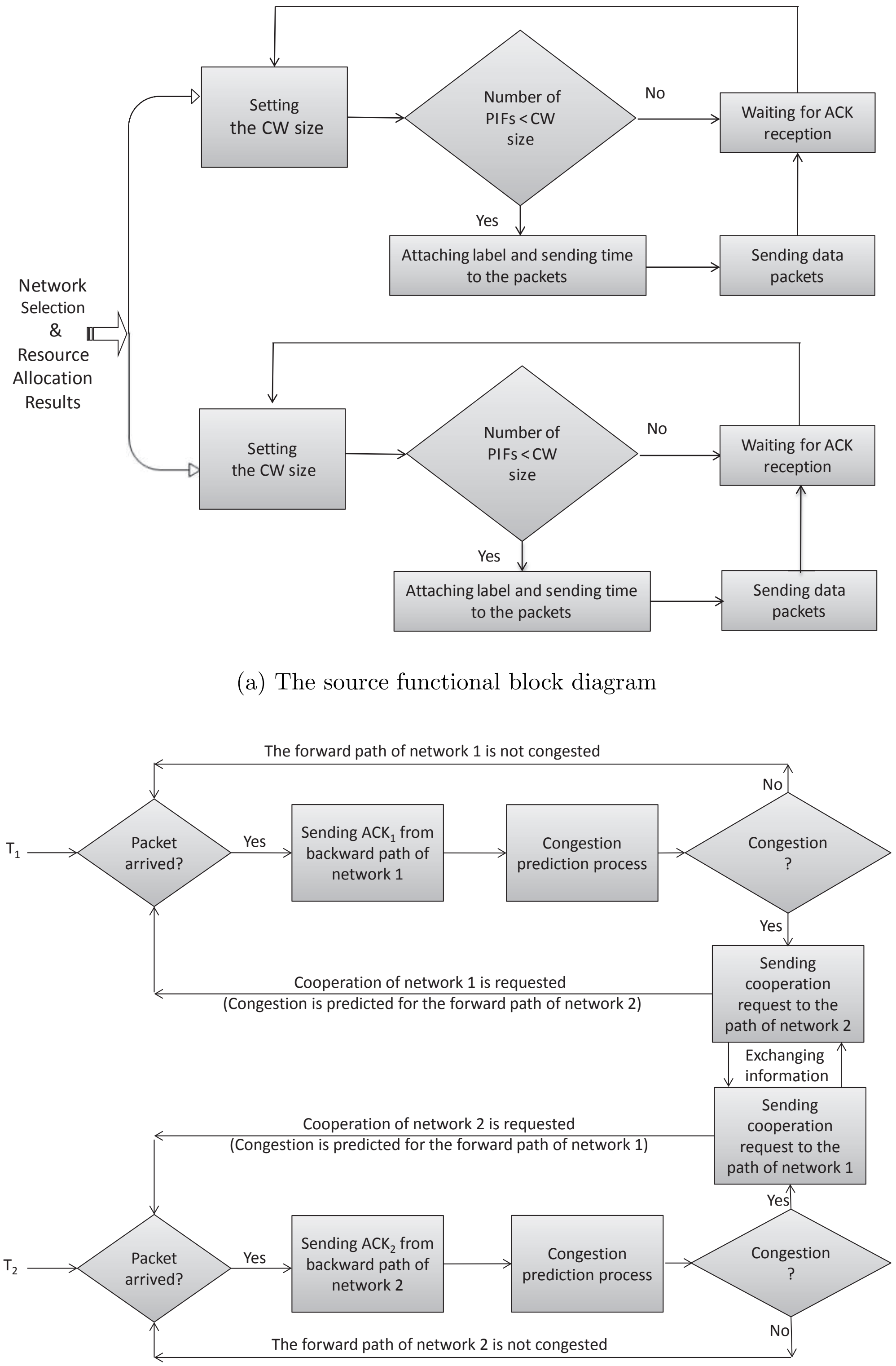
- The forward-path delay of the end-to-end path needs to be measured in order to estimate its congestion. In congestion control protocols, the forward path delay is usually calculated as half of the RTT measured at the source [76]. However, in wireless networks, the forward and backward paths may have different delays. Moreover, the processing time to send ACK after receiving a packet at the destination cannot always be neglected in the above forward-path-delay calculation [32]. In contrast, the destination is able to measure solely the delay of the forward path, i.e. TD. Therefore, the destination can predict congestion on the forward path more accurately, which will make cooperation more efficient;

- In the TCP Westwood protocol, the source updates the CW size. Devolving the cooperation decision process to the other end can decrease the processing time of the congestion control scheme. Further, the information reaches the destination first and then an ACK is sent to the source. Therefore, the destination has more up-to-date information for a better decision;

- After sending a packet, the source waits for an ACK from the destination. If the ACK is lost due to random loss or congestion loss in the backward wireless path, the source may incorrectly trigger cooperation. Note that the ACK loss cannot be ignored in wireless networks due to the wireless channel impairments.

Therefore, as shown in Figure 4.1, setting the CW (as in the regular congestion control protocol) is managed by the source, while cooperation decision is made at the destination. Figure 5.1a shows a block diagram of the source functions and Figure 5.1b shows that of the destination for the cooperative end-to-end congestion control. As can be seen in Figure 5.1a, the source should manage two separate congestion control tasks for the two paths. As illustrated in Figure 5.1b, the destination responsibilities are packet reception, sending ACKs, and cooperation decision making.

The congestion level is estimated at the destination terminal by measuring the queuing delay (or queue length) of each path. Therefore, the destination terminal needs to update information about the queuing delay (queue length) of both end-to-end paths. To this end,



(a) The source functional block diagram

(b) The destination functional block diagram

Figure 5.1: Block diagrams of the source and destination tasks in a heterogeneous environment with two wireless networks

at the source terminal, each packet is labeled based on its forward path and includes the information of its sending time. Once a packet is received at the destination terminal, the TD time is measured and the packet ACK is sent to the source through the same networks as a feedback. The values of three parameters are drawn from the received packet at the destination. First, the packet TD time of the forward path is accurately measured at the destination. Second, the number of acknowledged packets in a specific period is determined for each path. Finally, the minimum TD time experienced in each path is updated at the destination. These measurements are used to estimate the congestion level of each path and to decide on whether or not cooperation is required. Every destination needs to be aware of the congestion level of its own paths. No addition signalling and state information exchange are required between different destination terminals.

5.1.3 Congestion Control Markov Model

To model the association of the end-to-end paths, three congestion levels are considered for each path as indicated in Table 5.1: The path has no traffic when there is no packet to be sent through the path; it is uncongested when the average arrival rate of packets is less than the average service rate; it is congested if the average arrival rate is not less than the average service rate. Note that the average arrival rate and average service rate are calculated using the exponentially weighted moving average (EWMA) with a weight 0.002 to prevent the effect of sudden changes in traffic [77]. The congestion level is denoted by s_k, taking a value from $\{0, 1, 2\}$, as listed in Table 5.1.

Table 5.1: End-to-end path congestion levels

Congestion level	Traffic load condition
0	No traffic in the path
1	Uncongested: arrival rate $<$ service rate
2	Congested: arrival rate $\geq$ service rate

A pair of end terminals can determine the congestion level of the two associated end-to-end paths, denoted by state $\vec{S} = (s_1, s_2)$, in which s_k is the congestion status of path $k, k = 1, 2$. The congestion level over time can be modelled by a discrete-time Markov chain with state $\vec{S} = (s_1, s_2)$ as, given the current state, the future and past states are independent. Therefore, the probability of transition to state j can be described as follows:

$$P_s(\vec{S}_u = j | \vec{S}_{u-1} = i_1, \vec{S}_{u-2} = i_2, ..., \vec{S}_1 = i_{u-1}) = P_s(\vec{S}_u = j | \vec{S}_{u-1} = i_1), \tag{5.2}$$

where $i_v, j \in \{0, 1, 2\}^2,\ 1 \leq v \leq u - 1$, and the sampling time (time step) $u \in \{1, 2, ...\}$ is the index of the last received packet at the destination. The discrete-time Markov chain is achieved by sampling the continuous-time Markov chain with the same states as stated in Table 5.1 using the uniformization method in which the inter-arrival time of the time steps is assumed to be exponentially distributed [78].

Without cooperation between the two networks, the state diagram of end-to-end paths from the transport layer viewpoint is illustrated in Figure 5.2, which is a finite-state discrete-time irreducible Markov chain with transmission probability from state (i, j) to state (x, y) denoted by $P_{ij,xy}$. For the two end-to-end paths, there are nine states (s_1, s_2) with $s_1, s_2 \in \{0, 1, 2\}$.

5.2 Cooperation in Congestion Control

When congestion is predicted for one path, cooperation is initiated, and then some of the ongoing packets are sent over the cooperator path instead of the congested path. However, as stated in Section 4.3, a trade-off exists between congestion avoidance and cooperation cost. Thus, cooperation should be triggered only when it is beneficial according to congestion level measurements. In this section, cooperation initiation is discussed, cooperation thresholds are introduced, and the Markov model in Subsection 5.1.3 is revised for the scenario of cooperative networks.

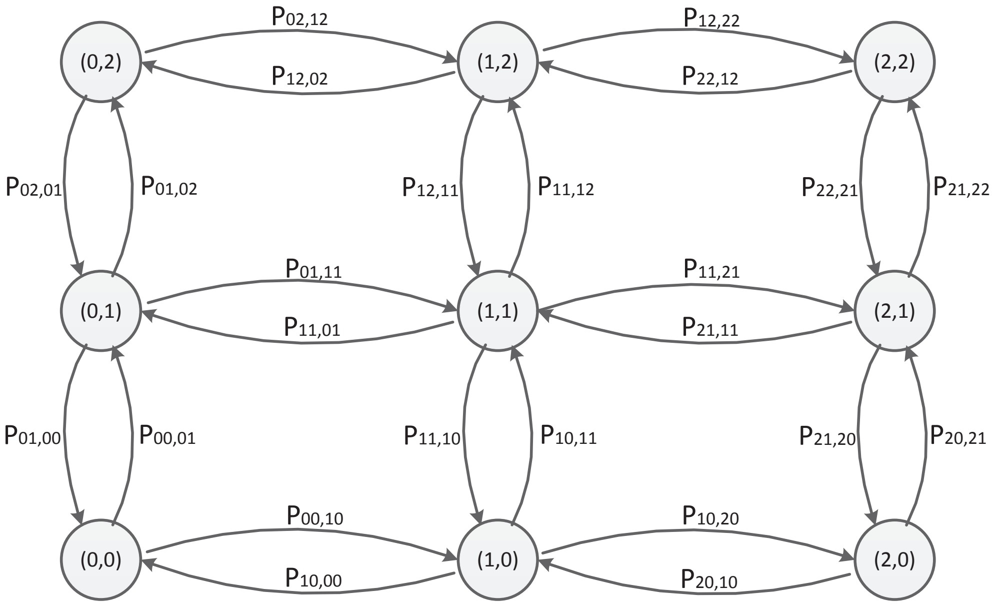


Figure 5.2: Congestion state diagram with two end-to-end paths

5.2.1 Cooperation Initiation

Three steps should be taken to initiate a cooperation. First, the destination asks the cooperator network to activate its path with the corresponding destination-to-source connection. Then, the ACK of the last received packet from the congested path is sent through both the cooperator network path and the congested path. When this ACK reaches the source terminal, the CW of the cooperator path will increase to let a number of packets be sent through the cooperator path instead of the congested one. Then, the source provides an activation request signal for the cooperator network to activate the source-to-destination path. As a result, cooperation is initiated and both congested and cooperator networks provide simultaneous packet deliveries to the end terminal.

Moreover, to address the trade-off between congestion avoidance and cooperation cost, we define two parameters for the cooperation as the start-cooperation threshold (SCT) and end-cooperation threshold (ECT), denoted by β and α, respectively. The cooperation thresholds

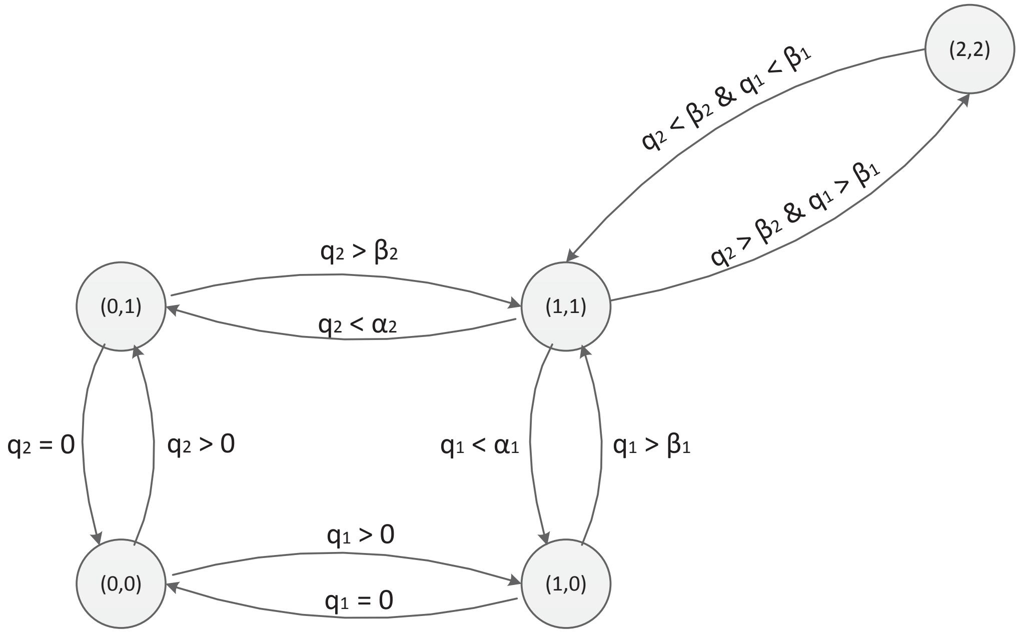


Figure 5.3: Congestion state diagram with network cooperation

are calculated based on the characteristics of available networks and current path condition. Therefore, different paths can have different threshold values. We consider the end-to-end queuing delay as a measure for the congestion level of a path. That is, if the queuing delay of a path becomes more than SCT, congestion is predicted for near future. Therefore, cooperation of other available networks can be used to avoid congestion. On the other hand, if the queuing delay becomes less than ECT, cooperation is not worth anymore.

5.2.2 The Markov Model with Cooperation Between Networks

Cooperation in the heterogeneous networks should provide maximal congestion avoidance. That is, congestion level 2 should be avoided whenever possible via cooperation between the two networks. Therefore, cooperative congestion control aims to achieve

$$P_s(\vec{S_u} = (x,y)|\vec{S_{u-1}} = (i,j)) = P_{ij,xy} = 0, \text{ if } x = 2 \text{ or } y = 2. \tag{5.3}$$

Based on the cooperation thresholds and (5.3), the Markov chain in Figure 5.2 is reduced to that shown in Figure 5.3, which does not include transient states. Our cooperative congestion control strategy to achieve (5.3) is as follows:

- When $q_k < \alpha_k$, the path over network k and network k' is not congested and, therefore does not require cooperation from the other network;

- When $q_k > \beta_k$ and the other end-to-end path is not congested, congestion is predicted for the path over network k and network k'. Therefore, the other end-to-end path starts helping to deliver some of the packets in order to achieve maximal congestion avoidance in end-to-end path k;

- When $q_1 > \beta_1$ and $q_2 > \beta_2$, congestion happens at both paths, and cannot be avoided.

In the steady state, the cooperative congestion control strategy reduces the number of states in the Markov chain from 9 (without cooperation in Figure 5.2) to 5, by achieving (5.3), and the congestion scenario of a single end-to-end path is avoided by the cooperation.

5.3 When to start/stop cooperation

For the cooperation strategy, we need to find the best cooperation time in which the objective given in (5.3) is achieved and the minimum rate requirement is satisfied. In this section, we derive proper cooperation thresholds, SCT and ECT, considering both flow-level and packet-level perspectives with macroscopic and microscopic views of congestion control, respectively.

5.3.1 Flow-Level Cooperation Thresholds

We derive the cooperation thresholds in order to satisfy the required QoS. From the flow-level viewpoint, two flows are transmitted through the two end-to-end paths. With the minimum requested rate, r_{min}, the cooperation is initiated when this rate cannot be provided in a single end-to-end path for a predicted congestion scenario. On the other hand, cooperation is stopped when cooperation is not beneficial anymore.

The RTT of a path is measured at the source in every transport layer protocol. Hence, the minimum RTT, $\tau_{min}(t)$, which is considered as the minimum measured RTT until time t and the RTT of the last acknowledged packet at t, $\tau(t)$, have been measured and are available after every ACK reception at the same node. Based on the RTT information, the queue length of the end-to-end path k at time t can be calculated according to the actual and expected rates by [79]

$$l_k(t) = \left[\left(\frac{\omega_k(t)}{\tau_k(t)} \right)_e - \left(\frac{\omega_k(t)}{\tau_k(t)} \right)_a \right] \cdot \tau_{e,k}(t) \tag{5.4}$$

$$= \frac{\tau_k(t) - \tau_{min,k}(t)}{\tau_k(t)} \cdot \omega_k(t), \; k = 1, 2. \tag{5.5}$$

where the expected rate, $(\frac{\omega_k(t)}{\tau_k(t)})_e$, is the congestion-free rate in which all the transmitted packets are received successfully with minimum RTT and the actual rate, $(\frac{\omega_k(t)}{\tau_k(t)})_a$, is the ratio of the current CW size to the current RTT. Moreover, $\tau_{e,k}(t)$ denotes the expected RTT which is the minimum RTT at time t for the end-to-end path k. The destination can directly measure the total time taken by a packet to reach the destination (TD time) that is in general more accurate than $\tau(t)/2$ in wireless networks. Therefore, by replacing $\tau_k(t)/2$ with $T_k(t)$, the queue length can be estimated more accurately as

$$l_k(t) = \frac{T_k(t) - T_{min,k}(t)}{T_k(t)} \cdot \omega_k(t), \; k = 1, 2. \tag{5.6}$$

The packet transmission rate is calculated in [40] as $x_k(t) = \frac{\omega_k(t)}{2T_k(t)}$. Thus, (5.6) can be written as

$$\omega_k(t) - l_k(t) = 2x_k(t) \cdot T_{min,k}(t), \; k = 1, 2. \tag{5.7}$$

To satisfy the minimum rate requirement, we have

$$x_1(t) + x_2(t) \geq r_{min}. \tag{5.8}$$

Multiplying (5.8) by $2T_{min,1}(t)$ and considering (5.7) for $k = 1$ without loss of generality, we have

$$\omega_1(t) - l_1(t) + 2T_{min,1}(t) \cdot x_2(t) \geq 2T_{min,1}(t) \cdot r_{min}. \tag{5.9}$$

According to Figure 5.3, before starting cooperation where $x_2(t) = 0$ as $s_2 = 0$, the rate requirement can be satisfied only if

$$l_1(t) \leq \omega_1(t) - 2r_{min} \cdot T_{min,1}(t). \tag{5.10}$$

The same inequalities as (5.9) and (5.10) hold for $k = 2$. Therefore,

$$l_k(t) \leq \omega_k(t) - 2r_{min} \cdot T_{min,k}(t),\ k = 1, 2. \tag{5.11}$$

If the queue length violates the bound in (5.11), a single path transmission cannot satisfy the QoS requirement and hence cooperation should be exploited. In order to achieve the SCT threshold, we use the Little's law [80, 81] with permissible initial and final queues over a finite time period. As stated in [81], for a queuing system observed over a finite period $[I, J]$, the little's law holds for the measurements in that period, i.e.,

$$\overline{l}_{[I,J]} = \overline{\lambda}_{[I,J]} \cdot \overline{q}_{[I,J]} \tag{5.12}$$

where $\overline{\lambda}_{[I,J]}$ is the ratio of the cumulative number of packets in the system in the time period $[I, J]$ over the period length. Note that the cumulative number of packets includes the arrival packets in $[I, J]$ as well as the packets that are in the system at time I. Moreover, $\overline{l}_{[I,J]}$ and $\overline{q}_{[I,J]}$ are the average queue length and average queuing delay experienced only inside the period $[I, J]$, respectively.

In order to apply the little's law over a finite time period to find the SCT from estimating the queue length in our system, we define an observation period as $[I, J] = [t - \epsilon/2, t + \epsilon/2]$, where ϵ is chosen small enough such that $l(t) \approx \overline{l}_{[t-\epsilon/2, t+\epsilon/2]}$ and $q_k(t) \approx \overline{q}_{[t-\epsilon/2, t+\epsilon/2]}$ over the observation period. Therefore, for the end-to-end path k, we have

$$l_k(t) = \overline{\lambda}_\epsilon \cdot q_k(t), \ k = 1, 2 \tag{5.13}$$

where $\overline{\lambda}_\epsilon$ is the ratio of the cumulative number of packets in the system in time period $[t - \epsilon/2, t + \epsilon/2]$ over ϵ.

It is assumed in this work that the considered duration for queue length estimation is sufficient. However, choosing a larger time period can reduce possible errors in estimating queuing parameters. In order to find the best time to start and stop cooperation efficiently, the effect of a larger time period and the system overhead due to cooperation measurements should be considered in further investigation.

Using (5.11) and (5.13), the queuing delay inequalities are

$$q_k(t) \leq \frac{1}{\overline{\lambda}_\epsilon} \cdot [\omega_k(t) - 2r_{min} \cdot T_{min,k}(t)], \ k = 1, 2. \tag{5.14}$$

Therefore, the SCT needs to be set to

$$\beta_k(t) = \frac{1}{\overline{\lambda}_\epsilon} \cdot [\omega_k(t) - 2r_{min} \cdot T_{min,k}(t)], \ k = 1, 2. \tag{5.15}$$

However, according to (5.13), in order to determine the start-cooperation time, instead of checking $q_k(t) \leq \beta_k(t)$, it is sufficient to check (5.11).

After calculating the start cooperation point, we need to find out when to stop it. Note that cooperation will not be beneficial anymore if a single path has the potential to transmit all the packets from the source. It means that the number, $\omega_1(t) + \omega_2(t)$, of packets can be transmitted in the TD time, $T_k(t)$. Thus, the actual rate in (5.4) becomes $\frac{\omega_1(t)+\omega_2(t)}{T_k(t)}$. Moreover, the maximum possible rate for a single path, k, is $\frac{\omega_{max,k}}{T_{min,k}(t)}$, in which ω_{max} is the maximum possible CW size for TCP Westwood that is derived in [82]. Note that the expected transmission duration time is considered to be $T_{min,k}(t)$ that minimizes α to ensure that cooperation is not necessary anymore. By considering the maximum possible rate as the expected rate in (5.4), the cooperation should be stopped if the following inequality holds:

$$l_k(t) \leq [\frac{\omega_{max,k}}{T_{min,k}(t)} - \frac{\omega_1(t)+\omega_2(t)}{T_k(t)}] \cdot T_{min,k}(t),\ k = 1, 2. \tag{5.16}$$

Therefore, using (5.13), we choose ECT as

$$\alpha_k(t) = \frac{1}{\overline{\lambda}_\epsilon} \cdot [\frac{\omega_{max,k}}{T_{min,k}(t)} - \frac{\omega_1(t)+\omega_2(t)}{T_k(t)}] \cdot T_{min,k}(t),\ k = 1, 2 \tag{5.17}$$

and the cooperation should be stopped if the queuing delay decreases to be less than $\alpha_k(t)$. The same as the start-cooperation threshold, we will calculate (5.16) for end-cooperation decision in the cooperative congestion control algorithm.

5.3.2 Packet-Level Cooperation Threshold

From a packet-level perspective, data communication between two end terminals can be described by tracking the behaviour of a number of packets that are sent through the networks [28]. Here, the SCT is calculated based on the packet-level requirements of cooperation. Considering the backlog of a path, cooperation is efficient if the backlog of the transmission path without cooperation, denoted by $B_{p,nc}$, exceeds that of the two paths when cooperation is in place, i.e.,

$$B_{p,nc}(u) \geq B_{p,1}(u) + B_{p,2}(u). \tag{5.18}$$

Here, although one packet is not transmitted through both the paths, we keep the same packet index u for paths 1, 2, and non-cooperative scenario to compare them over the same

RTT. It is proved in [83] that the arrival time of packet u can be described based on the delays of the system in a single path transmission

$$d + \frac{B_p(u)}{c} = A_p(u) - S_p(u). \tag{5.19}$$

The minimum transmission time and the actual transmission time of packet u are $\frac{\psi(u)}{c}$ and $A_p(u) - S_p(u)$, respectively. Therefore,

$$d + \frac{B_p(u)}{c} \geq \frac{\psi(u)}{c}. \tag{5.20}$$

Inequality (5.20) can be extended to the two end-to-end paths, by indexing the formula for paths 1 and 2. Then, by providing inequalities for $B_{p,1}(u)$ and $B_{p,2}(u)$, (5.18) can be rewritten as

$$B_{p,nc}(u) \geq B_{p,1}(u) + B_{p,2}(u) \geq \psi_1(u) - d_1 \cdot c_1 + \psi_2(u) - d_2 \cdot c_2. \tag{5.21}$$

To further proceed with the analysis, we use the following theorem from [83].

Theorem 4.1: Let u and u' denote packet indexes and ψ be the number of PIFs. For all u and u' with $1 < u' < u$, if $\psi(u'), \psi(u'+1), ..., \psi(u)$ are non decreasing, we have

$$B_p(u') + \psi(u) - \psi(u') \geq B_p(u). \tag{5.22}$$

■

When a path is expected to be congested in near future, its PIF number increases with time. Therefore, the condition in Theorem 4.1 for non-decreasing PIFs holds when congestion is predicted. At this moment, cooperation should be started. Using (5.22) for non-cooperative scenario (with subscript nc) and (5.21), two inequalities with the same $B_{p,nc}(u)$ lead to

$$B_{p,nc}(u') + \psi_{nc}(u) - \psi_{nc}(u') \geq \psi_1(u) - d_1 \cdot c_1 + \psi_2(u) - d_2 \cdot c_2 \tag{5.23}$$

where $\psi_{nc}(.)$ is the number of PIFs in the non-cooperative scenario.

While cooperation needs to be beneficial, the overall cooperation CWs should be at least equal to non-cooperative CW, denoted by $\omega_{nc}(u)$ for packet u. Therefore,

$$\omega_{nc}(u) \leq \omega_1(u) + \omega_2(u). \tag{5.24}$$

The same inequality holds for the PIFs,

$$\psi_{nc}(u) \leq \psi_1(u) + \psi_2(u). \tag{5.25}$$

By substituting (5.25) in (5.23), inequality (5.23) reduces to

$$B_{p,nc}(u') \geq \psi_{nc}(u') - d_1 \cdot c_1 - d_2 \cdot c_2. \tag{5.26}$$

From the packet-level perspective, the queuing delay can be estimated as $q_1(u) = T_1(u) - d_1$. Also from [83], we have $q(u) = \frac{B_p(u)}{c}$. Therefore, considering that congestion is predicted for end-to-end path 1 without loss of generality, we have

$$q_1(u') \geq \frac{\psi_{nc}(u')}{c_1} - d_1 - d_2 \cdot \frac{c_2}{c_1}. \tag{5.27}$$

Inequality (5.27) provides the SCT threshold based on the current situation of the path before cooperation ($\psi_{nc}(u')$) and the characteristics of both end-to-end paths (c_1, c_2, d_1, d_2), given by

$$\beta_1(u') = \frac{\psi_{nc}(u')}{c_1} - d_1 - d_2 \cdot \frac{c_2}{c_1}. \tag{5.28}$$

In summary, network characteristics, current traffic condition, and the requested QoS determine the thresholds to start, maintain, and stop cooperation. Based on the calculated thresholds for cooperation decision making, the required functions for cooperation of network 2 with network 1 in congestion control are summarized in Algorithm 1. Note that, in

Algorithm 1: Cooperative congestion control algorithm

```
Result: cooperation between the wireless networks of the uncongested path (i) with
        those of the congested one (j): $i, j \in \{1, 2\},\ i \neq j$
get $d_i, d_j, c_i, c_j, r_{min}$;
for every packet to be sent through path $i$ do
    Attach $i$ and $S_p$ to packet $u$;
    if packet $u$ reaches the destination then
        Measure $T_i, T_{min,i}, q_i$ and $l_i$;
        if ($l_i \geq \omega_i - 2r_{min} \cdot T_{min,i}$ && $q_i \geq \frac{\psi_p}{c_i} - d_i - d_j \cdot \frac{c_j}{c_i}$) then
            $q_i \geq \beta_i$ holds.
        end
        if $l_i \leq \omega_{max,i} - (\frac{T_{min,i}(t)}{T_i(t)}) \cdot (\omega_i(t) + \omega_j(t))$ then
            $q_i \leq \alpha_i$ holds.
        end
    end
    Compare $q_i, \beta_i, \alpha_i$ as in Figure 5.3;
    if cooperation is necessary for path $i$ then
        if path $j$ is not congested then
            Send the ACK through path $j$;
            Perform the cooperation;
        end
    end
end
```

order to take the cooperation start time into account from both packet-level and flow-level perspectives, both packet-level and flow-level inequalities, (5.27)) and (5.11) respectively, should be evaluated.

5.4 Simulation Results

Performance measures of transport layer protocols over wireless links include throughput, goodput, delay, and fairness [74]. Here, as a preliminary study, based on computer simulations, we provide a performance comparison between our proposed cooperative congestion control scheme and the same multi-homed congestion control strategy without any cooperation in a heterogeneous wireless environment with two different wireless networks.

The system model in Section 5.1 is simulated in Matlab using the SimEvents library[1]. To evaluate the performance of the multi-homed cooperative congestion control algorithm with two networks, the following three schemes are considered:

- Our proposed multi-homed cooperative congestion control algorithm with two networks, labeled as “MH - C”;
- Multi-homed non-cooperative congestion control algorithm with two networks, labeled as “MH - NC”;
- Single-network congestion control, labeled as “S”.

All the three schemes are based on the TCP Westwood protocol. The source node transmits packets through the same end-to-end path. The two multi-homed schemes have an additional path in a different network. In the simulation, the packet inter-arrival duration time is exponentially distributed. The packet size is constant, and the packet service time duration with a heavy-tailed distribution is modelled by a Pareto distribution[2]. The end-to-end paths are simulated using their bottleneck queuing model, i.e., an M/G/1 queue for the considered simulation scenario. With the delay-based congestion control, the end-to-end delay in each path can be used to compare the end-to-end congestion control schemes. Three simulation scenarios are considered, one source-destination pair with different packet arrival rates, multiple source-destination pairs, and packet loss due to wireless channel. The simulation results for these scenarios are discussed in the following.

5.4.1 Different Packet Arrival Rates

For the first simulation scenario, we consider a source-destination pair with two available wireless networks. The inter-arrival duration mean is changed from 0.001s to 1s to simulate

[1] Event-driven communication between components, in order to analyze and optimize end-to-end latencies, throughput and packet loss, can be modelled using SimEvents which is included in Simulink software.

[2] Although the packet size does not have a heavy-tailed distribution in this simulation, the service time remains heavy-tailed to reflect the dependency of packet transmission on the ACK reception of previous packets in a congestion control scheme.

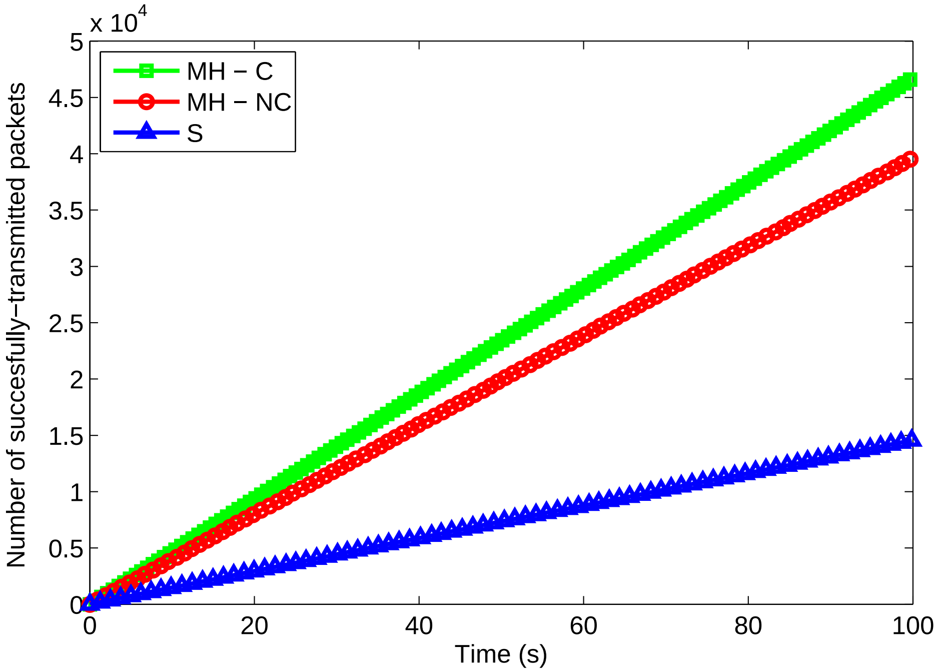


Figure 5.4: The number of successfully-transmitted packets versus time with mean arrival rate of 500 packets/s

all congested (arrival rate $\geq$ service rate) and non-congested scenarios. The simulation parameters are summarized in Table 5.2. The two existing heterogeneous wireless interfaces have different characteristics such as different RTTs.

Table 5.2: Simulation Parameters and their corresponding values

Parameter	Value
Mean packet inter-arrival duration	0.001 - 1 s
Network bottleneck 1 service time mean value	6 ms
Network bottleneck 2 service time mean value	2 ms
Network 1 propagation delay	5 ms [84]
Network 2 propagation delay	6 ms [84]
Packet size	10 Kbits

Figure 5.4 shows the number of successfully-transmitted packets versus time, with the mean packet arrival rate of 500 packets/s. The throughput is improved when multi-homing is provided in a heterogeneous environment over single-network congestion control. However, the multi-homed throughput does not reach the summation of two path throughputs,

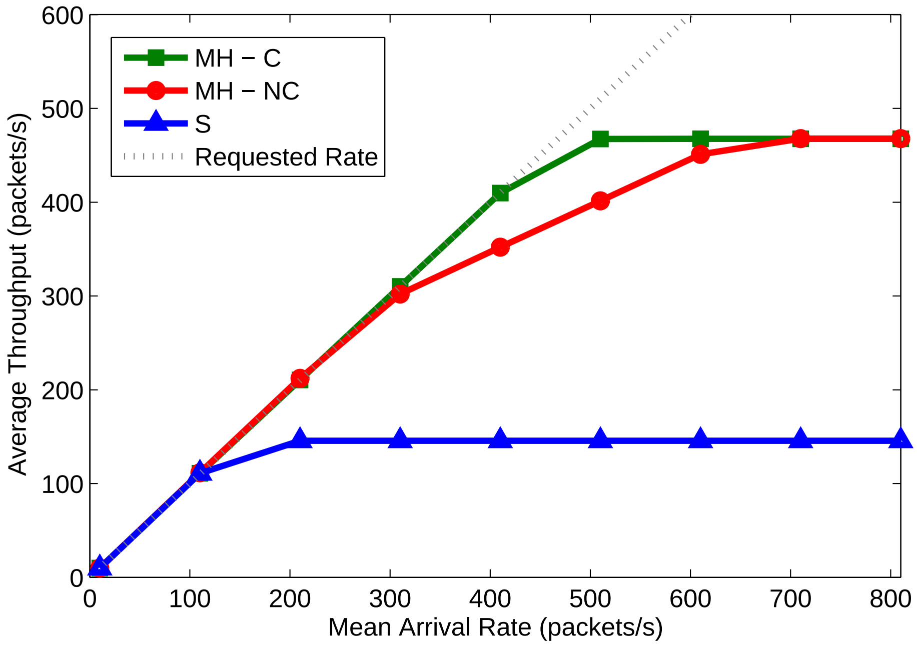


Figure 5.5: Transmission throughput versus arrival rate

similar to the observation in [22]. As can be seen in Figure 5.4, our proposed cooperative congestion control algorithm improves the multi-homed throughput by avoiding congestion and unnecessary queuing delay. Therefore, by means of the two-network cooperation, the number of successfully-transmitted packets increases significantly.

The average throughput of congestion control schemes depends on the arrival rate and service rates in the end-to-end paths. Figure 5.5 shows the transmission throughput in the three considered schemes for different arrival rates. Figure 5.5 also shows a request rate, which is equal to the average arrival rate. Note that the average service rate is 160 packets/s for the first end-to-end path and 500 packets/s for the second end-to-end path. For the average arrival rate less than 160 packets/s, the requested rate is provided by all the three congestion control schemes.

When the arrival rate exceeds the service rate of the first end-to-end path, i.e., the path is congested, the single path congestion control is restricted to its bottleneck capacity and cannot follow the requested rate. However, as it is observed from Figure 5.5, the throughput of the multi-homed congestion control schemes increases.

At the arrival rate of around 300 packets/s, in the multi-homed schemes, the first path becomes congested and the second one is not congested. The non-cooperative scheme cannot manage the available resources efficiently to achieve the requested rate as the two end-to-end paths are independent, meaning that the throughput of multi-homed non-cooperative scheme is restricted in the first end-to-end path, but, it still increases in the second end-to-end path[3]; On the other hand, the cooperative congestion control provides the requested rate by means of cooperation. The cooperative scheme continues to provide the requested rate by means of cooperation for the mean arrival rate less than about 500 packets/s. For larger mean arrival rates, cooperation is still in place, which keeps the throughput of cooperative congestion control more than that of the non-cooperative scheme. However, the cooperation is not sufficient anymore to provide the requested rate. The cooperation between the heterogeneous networks is beneficial until both paths become congested (at around 660 packets/s).

The average end-to-end delay for the three congestion control schemes is illustrated in Figure 5.6. The average waiting time for a packet to reach the destination in all three schemes increases slowly with the average arrival rate, when the arrival rate is less than the service rates (less than 160 packets/s). When the mean arrival rate comes close to 160 packets/s, the first end-to-end path becomes fully congested most of time. Therefore, the end-to-end delay in the single-path scheme sharply increases to infinity. In this scenario, the end-to-end delay in the multi-homed schemes increases slowly. At around 300 packets/s, in which the first end-to-end path becomes congested, the end-to-end delay of the non-cooperative scheme goes to infinity as the end-to-end delay due to the congested path increases the average end-to-end delay. The advantage of cooperation in multi-path congestion control becomes clear in this scenario, in which the two network paths cooperate to avoid congestion in the first path. Therefore, the end-to-end delay does not increase significantly since cooperation

[3] It should be noted that in this scenario, for the multi-homed non-cooperative scheme, half of the source traffic is sent through each of the two end-to-end paths. This assumption is to provide a congestion scenario for a single source-destination pair.

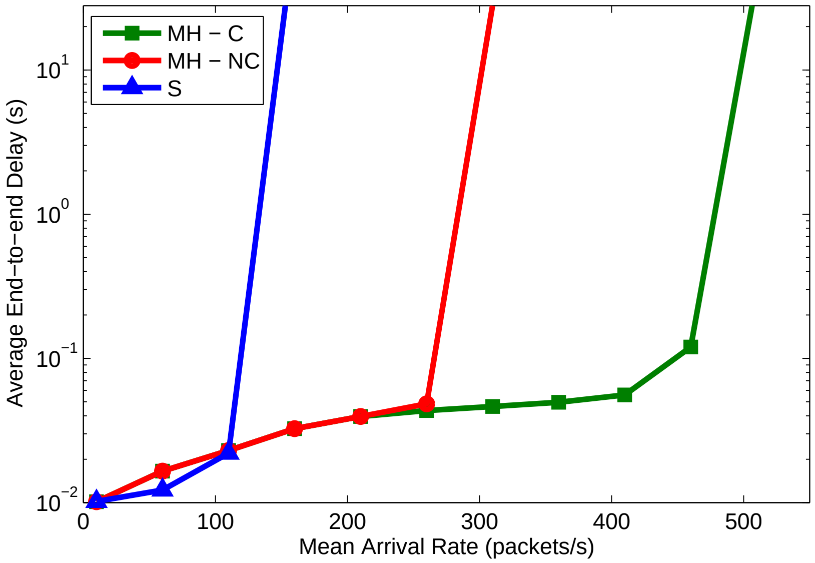


Figure 5.6: End-to-end delay for different arrival rates

between two end-to-end paths is sufficient to avoid congestion in both paths.

As can be seen in Figure 5.6, in the cooperative scheme, the average end-to-end delay increases sharply at around 500 packets/s as cooperation is not sufficient anymore to compensate for the CW size decrement of end-to-end path 1. Therefore, the average end-to-end delay of cooperative scheme experiences a sudden increase.

Table 5.3: Simulation Parameters and their corresponding values

Parameter	Value
Network bottleneck 1 mean service rate value	707 pkts/s
Network bottleneck 2 mean service rate value	141 pkts/s
Pareto shape parameter for service rate 1	3
Pareto shape parameter for service rate 2	3
Pareto scale parameter for service rate 1	0.001
Pareto scale parameter for service rate 2	0.0047
Mean packet arrival rate of the first source	400 pkts/s & 600 pkts/s
Mean packet arrival rate of the second source	370 pkts/s
Network 1 propagation delay	1 ms
Network 2 propagation delay	10 ms
Packet size	10 Kbits

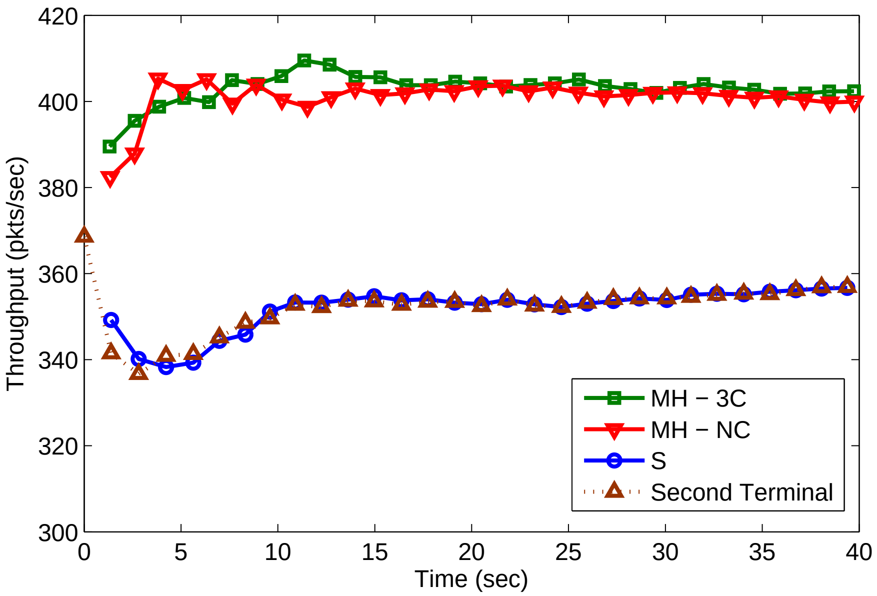


Figure 5.7: Throughput of congestion control schemes over time with the arrival rate of 400 pkts/s

5.4.2 Multiple End-to-end Terminals

In order to evaluate the performance of congestion control schemes for multiple users, two pairs of source-destination terminals are considered. There are also two wireless networks available in this scenario. The first wireless network is available to both pairs and the second wireless network is available only to one of the source-destination pairs. In other words, a source-destination pair with three congestion control schemes (S, MH-C, and MH-NC) co-exists with a second source-destination pair that has access to the first wireless network. Congestion happens in the first network if the aggregate arrival rates exceeds the bottleneck service rate. In such a scenario, we expect that each terminal transmits its packets fairly by means of its congestion control, i.e. the transmission rate is at most half of the first network bottleneck capacity for any of the two source-destination terminals.

The simulation parameters for this scenario are summarized in Table 5.3. The two wireless networks are assumed to be WiFi and cellular networks with the parameters as in [22].

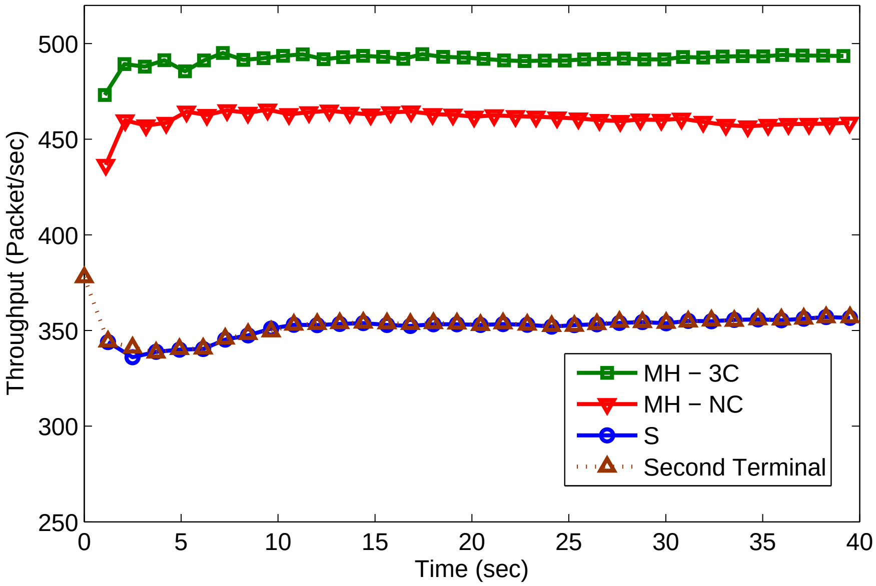


Figure 5.8: Throughput of congestion control schemes over time with the arrival rate of 600 pkts/s

Figure 5.7 illustrates the throughput of each congestion control scheme over a time period. For the single path congestion control, the arrival rate of the first terminal is 400 pkts/s and the arrival rate of the second terminal is 370 pkts/s. The aggregate arrival rate is more than the service rate (707 pkts/s) and, hence, the CW sizes are changed to reach the fair throughput of 707/2 pkts/s for both first and second paths. In multi-homed non-cooperative scheme, 83 percent of packets are transmitted through a path in the first network and 17 percent of them are sent through a path in the second network, based on the network bottleneck capacities. Therefore, all packets are transmitted without congestion. The multi-homed cooperative congestion control is also non-congested and the rate of 400 pkts/s is achieved.

Figure 5.8 shows the throughput of congestion control schemes versus time for the arrival rate of 600 pkts/s. In this scenario, all three congestion control schemes experience congestion in the first path. The single path congestion control throughput remains at the 707/2 pkts/s. The multi-homed non-cooperative scheme experiences congestion in the first path $(500 + 370 > 707)$. The second path transmits 17 percent of packets (102 pkts/s) and the

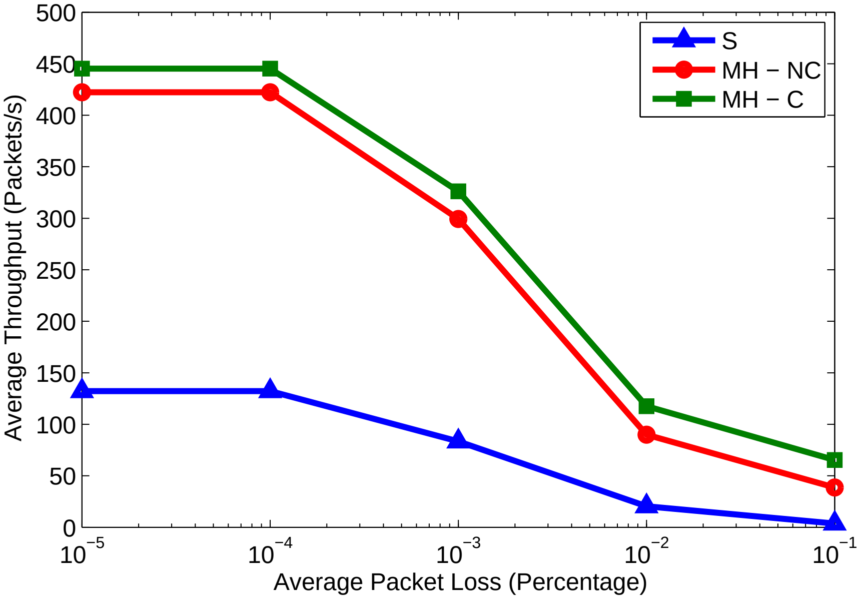


Figure 5.9: Throughput of congestion control in a lossy wireless environment

transmission rate in the first path is limited to 707/2 pkts/s. Thus, as shown in Figure 5.8, the overall throughput in the non-cooperative congestion control is 455 pkts/s. However, the cooperative congestion control makes use of cooperation, which leads to the throughput of around 500 pkts/s which includes 707/2 pkts/s from the first path and 141 pkts/s from the second path. Therefore, in terms of throughput, the cooperative congestion control outperforms the other congestion control schemes.

5.4.3 Packet Loss

In this scenario, we consider packet loss due to wireless network channels and measure the throughput of congestion control schemes for different packet losses with the same parameters in Table 5.2. The average packet loss in the Figure 5.9 is set to be equal to the average packet loss in the first network and 10 times of the average packet loss in the second network. As shown in Figure 5.9, the throughput of all congestion control schemes decreases with increasing the average packet loss. However, cooperative congestion control throughput is always higher than other schemes.

Based on the simulation results, the proposed cooperative congestion control algorithm can enhance the requested rate provisioning, specially when congestion happens in one path. This improvement is achieved by the cooperation of heterogeneous networks in reducing congestion.

5.5 Summary

In this chapter, an end-to-end cooperative congestion control algorithm is proposed when two networks are available to each of the end users. The cooperative scheme is developed without requiring changes to the protocol of the congestion control core. The cooperation measurement and decision-making are performed at the destination and the cooperation notification is sent to the source via an ACK to increase the CW size of the uncongested path. The cooperation start/stop thresholds, SCT and ECT, have been discussed and formulated based on the wireless network characteristics, current traffic condition, and requested QoS. The simulation results show the performance improvement of the cooperative scheme in comparison with the previous multi-homed congestion control protocols.

Chapter 6

Conclusion and Future Work

The objective of this research is to develop an end-to-end cooperative congestion control algorithm in a heterogeneous wireless environment with the Internet backbone. The end terminals are equipped with multiple radio interfaces for connection to the Internet backbone and multiple paths can be established between the end terminals for simultaneous packet transmission. We consider cooperation among heterogeneous wireless networks with overlapped coverage areas as a congestion prevention method without losing service quality.

The cooperation among available networks is used to enhance QoS, especially in a congestion scenario. The cooperative congestion control algorithm should avoid congestion in every path based on the condition of all paths in the association. Therefore, when congestion is predicted for one path, other paths are notified to help the congested path by moving the traffic load away from the congested path and by increasing CW size of non-congested paths to compensate for the CW size reduction of the congested one. However, cooperation among heterogeneous wireless networks in congestion scenarios does not always take place as an uncongested network may not be available for cooperation or cooperation may not be beneficial due to overhead such as cooperation setup time.

In this thesis, the cooperation of heterogeneous wireless networks with multi-homing capability is studied for an end-to-end congestion control. An existing congestion control protocol is chosen as the cooperative congestion control core. The cooperative congestion control is modelled by a finite-state Markov chain, and the estimated congestion level of each path is compared to some calculated thresholds. The thresholds have been calculated to determine when to start/stop cooperation considering wireless network characteristics and current traffic in the network. Simulation results demonstrate that the proposed algorithm reduces congestion occurrences and increases the transmission throughput significantly in a congestion scenario. Moreover, the transmission throughput of cooperative congestion control always outperforms that of non-cooperative congestion control in wireless networks with packet loss.

As a future work, this research can be extended in several directions as follows.

- The CW decrease of a path in a congestion scenario is a serious obstacle to guaranteeing the QoS for the users. Cooperation can enhance QoS in a congestion scenario if enough cooperation from uncongested networks exists. To specify the amount of cooperation, the cooperation rate can be defined as the number of packets sent through the cooperative network normalized to the total number of transmitted packets. The cooperative congestion control rate should be maximized, while congestion occurrences are minimized and QoS degradation is avoided.

- Although heterogeneous wireless networks and the wired Internet backbone provide end-to-end packet transmissions together, they have different network resources, various traffic flows, and dissimilar infra-structures. Therefore, the congestion level of a path in the wireless domain is not generally the same as in the wired one. If a network is congested in the wired Internet backbone, the cooperative congestion control algorithm, which aims to use the heterogeneity in wireless networks to reduce congestion, may not be effective and necessary. Therefore, the cooperation algorithm should be notified whether the cooperation has been effective in congestion reduction. This can be achieved by tracking the queuing delay changes of the end-to-end paths.

- Although the problem of out-of-order packet reception is ignored in most congestion control proposals [52], it should be considered for congestion control in heterogeneous networks because different delays in various end-to-end paths are inevitable. Moreover, the transition between wireless and wired networks may change the order of packets even in a single network path since the switches or routers may have parallel queues with different delays [51].

  The throughput of multi-path congestion control algorithms can be increased significantly by minimizing reordering or its effects [22]. The solution to the reordering problem can take the advantages of cooperation. It can be combined with the cooperative congestion control scheme to provide the best cooperation congestion prevention scenario while the number of out-of-order packets is minimized. To minimize out-of-order packet delivery to improve the transmission throughput, an adaptive ACK strategy should be developed for time-varying delays in the end-to-end paths. The trade-off between the cooperation maximization and reordering minimization can be studied to enhance the end-to-end QoS;

- Fairness is essential for any congestion control scheme since the bottleneck is shared among different entities. It is known from the literature that a fair congestion control protocol in a single-path transmission generally does not act fairly in a multi-path transmission [22]. Thus, a cooperative congestion control algorithm in a heterogeneous environment is not necessarily fair even if the independent congestion control protocols are fair. On the other hand, different definitions are proposed for the fairness concept, as discussed in Subsection 3.1.3. Hence, to achieve the fairness, first, a proper definition of fairness needs to be provided for cooperative congestion control. Then, cooperation among heterogeneous wireless networks should be facilitated to guarantee the target fairness, i.e., a fair rate allocation should be developed to maximize the QoS enhancement via cooperation.

Appendix A

Packet-level congestion control primary equations

Packets are labeled based on their sending order,

$$S_p(u) \leq S_p(u+1),\ \forall u. \tag{A.1}$$

The minimum PIF can be one (just for packet u), which means all the packets except u are received at the destination. The maximum PIF can be equal to the number of packets in flight in the previous RTT plus one, which means non of the packets in flight has reached the destination yet, i.e.,

$$1 \leq \psi(u) \leq \psi(u-1) + 1. \tag{A.2}$$

The relationship between the queuing delay and backlog can be simply described based on the bottleneck of the end-to-end path, given by

$$q(u) = \frac{B_p(u)}{c}. \tag{A.3}$$

There is a limitation on the maximum transmission rate of packets,

$$A_p(u) - A_p(u-1) \geq \frac{1}{c} \; \forall u. \tag{A.4}$$

The arrival time of packet u can be described based on the delays in the system,

$$A_p(u) = S_p(u) + d + \frac{B_p(u)}{c}. \tag{A.5}$$

Bibliography

[1] Cisco Systems Inc., "Cisco visual networking index: Forecast and methodology, 2011-2016," 2012.

[2] J. Postel, "Transmission control protocol," *Technical Report*, 1981.

[3] E. Brosh, S. Baset, V. Misra, D. Rubenstein, and H. Schulzrinne, "The delay-friendliness of TCP for real-time traffic," *IEEE/ACM Trans. Networking*, vol. 18, no. 5, pp. 1478 –1491, 2010.

[4] G. Xylomenos, G. Polyzos, P. Mahonen, and M. Saaranen, "TCP performance issues over wireless links," *IEEE Communications Magazine*, vol. 39, no. 4, pp. 52–58, 2001.

[5] F. Hu and N. Sharma, "Enhancing wireless internet performance," *IEEE Communications Surveys & Tutorials*, vol. 4, no. 1, pp. 2–15, 2002.

[6] B. Bakshi, P. Krishna, N. Vaidya, and D. Pradhan, "Improving performance of TCP over wireless networks," in *Proc. IEEE Int. conf. Distributed Computing Systems*, pp. 365–373, 1997.

[7] S. Koh, M. Chang, and M. Lee, "mSCTP for soft handover in transport layer," *IEEE Communications Letters*, vol. 8, no. 3, pp. 189–191, 2004.

[8] C. Huang, M. Lin, and L. Chang, "The design of mobile concurrent multipath transfer in multihomed wireless mobile networks," *The Computer Journal*, vol. 53, no. 10, pp. 1704–1718, 2010.

[9] F. Kelly, L. Massoulié, and N. Walton, "Resource pooling in congested networks: proportional fairness and product form," *Queueing Systems*, vol. 63, no. 1, pp. 165–194, 2009.

[10] Q. Zhang, W. Zhu, and Y. Zhang, "End-to-end QoS for video delivery over wireless internet," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 123–134, 2005.

[11] L. Cai, X. Shen, J. Mark, and J. Pan, "QoS support in wireless/wired networks using the TCP-friendly AIMD protocol," *IEEE Trans. Wireless Communications*, vol. 5, no. 2, pp. 469–480, 2006.

[12] F. Calì, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *IEEE/ACM Trans. Networking*, vol. 8, no. 6, pp. 785–799, 2000.

[13] J. Iyengar, *End-to-end concurrent multipath transfer using transport layer multihoming.* PhD thesis, University of Delaware, 2006.

[14] R. Stewart and Q. Xie, *Stream Control Transmission Protocol (SCTP).* Addison-Wesley Professional, 2001.

[15] D. Phatak and T. Goff, "A novel mechanism for data streaming across multiple IP links for improving throughput and reliability in mobile environments," in *Proc. Annual Joint Conf. of the IEEE Computer and Communications Societies*, vol. 2, pp. 773–781, 2002.

[16] C. Barakat, E. Altman, and W. Dabbous, "On TCP performance in a heterogeneous network: a survey," *IEEE Communications Magazine*, vol. 38, no. 1, pp. 40–46, 2000.

[17] Z. Sun and W. Wang, "Investigation of cooperation technologies in heterogeneous wireless networks," *Journal of Computer Systems, Networks, and Communications*, vol. 2010, p. 3, 2010.

[18] H.-Y. Hsieh and R. Sivakumar, "A transport layer approach for achieving aggregate bandwidths on multi-homed mobile hosts," *Wireless Networks*, vol. 11, pp. 99–114, Jan. 2005.

[19] A. Abd El Al, T. Saadawi, and M. Lee, "LS-SCTP: a bandwidth aggregation technique for stream control transmission protocol," *Computer Communications*, vol. 27, no. 10, pp. 1012–1024, 2004.

[20] J. Liao, J. Wang, and X. Zhu, "cmpSCTP: An extension of SCTP to support concurrent multi-path transfer," in *Proc. IEEE Int. conf. Communications*, pp. 5762–5766, 2008.

[21] J. Iyengar, P. Amer, and R. Stewart, "Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths," *IEEE/ACM Trans. Networking*, vol. 14, no. 5, pp. 951–964, 2006.

[22] D. Wischik, C. Raiciu, A. Greenhalgh, and M. Handley, "Design, implementation and evaluation of congestion control for multipath TCP," *Usenix NSDI*, 2011.

[23] T. Goff and D. Phatak, "Unified transport layer support for data striping and host mobility," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 4, pp. 737–746, 2004.

[24] L. Ma, F. Yu, and V. Leung, "Performance improvements of mobile SCTP in integrated heterogeneous wireless networks," *IEEE Trans. Wireless Communications*, vol. 6, no. 10, pp. 3567–3577, 2007.

[25] Ł. Budzisz, *Stream Control Transmission Protocol (SCTP), a proposal for seamless handover management at the transport layer in heterogeneous wireless networks.* PhD thesis, Universitat Politecnica de Catalunya, 2009.

[26] S. Barr, C. Paasch, and O. Bonaventure, "Multipath TCP: From theory to practice," in *Networking*, vol. 6640, pp. 444–457, Springer, 2011.

[27] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, “Modeling TCP throughput: A simple model and its empirical validation,” in *Proc. ACM SIGCOMM Computer Communication Review*, vol. 28, pp. 303–314, ACM, 1998.

[28] D. Wei, C. Jin, S. Low, and S. Hegde, “FAST TCP: motivation, architecture, algorithms, performance,” *IEEE/ACM Trans. Networking*, vol. 14, no. 6, pp. 1246–1259, 2006.

[29] K. Jacobsson, *Dynamic modeling of Internet congestion control.* Electrical Engineering, Elektrotekniska system, Kungliga Tekniska högskolan, 2008.

[30] Y. Tian, K. Xu, and N. Ansari, “TCP in wireless environments: problems and solutions,” *IEEE Communications Magazine*, vol. 43, no. 3, pp. S27–S32, 2005.

[31] I. F. Akyildiz, G. Morabito, and S. Palazzo, “TCP-Peach: a new congestion control scheme for satellite IP networks,” *IEEE/ACM Trans. on Networking*, vol. 9, no. 3, pp. 307–321, 2001.

[32] C. Fu and S. Liew, “TCP Veno: TCP enhancement for transmission over wireless access networks,” *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 2, pp. 216–228, 2003.

[33] K. Chandran, S. Raghunathan, S. Venkatesan, and R. Prakash, “A feedback-based scheme for improving tcp performance in ad hoc wireless networks,” *Personal Communications, IEEE*, vol. 8, no. 1, pp. 34–39, 2001.

[34] C. Wang, K. Sohraby, B. Li, M. Daneshmand, and Y. Hu, “A survey of transport protocols for wireless sensor networks,” *IEEE Network*, vol. 20, no. 3, pp. 34–40, 2006.

[35] X. Chen, H. Zhai, J. Wang, and Y. Fang, “A survey on improving TCP performance over wireless networks,” *Resource management in wireless networking*, pp. 657–695, 2005.

[36] K. Read and F. Maurer, “Developing mobile wireless applications,” *IEEE Internet Computing*, vol. 7, no. 1, pp. 81–86, 2003.

[37] S. Tsao, Y. Lai, and Y. Lin, “Taxonomy and evaluation of TCP-friendly congestion-control schemes on fairness, aggressiveness, and responsiveness,” *IEEE Network*, vol. 21, no. 6, pp. 6–15, 2007.

[38] V. Tsaoussidis and H. Badr, “Tcp-probing: towards an error control schema with energy and throughput performance gains,” in *Proceedings Int. Conf. on Network Protocols*, pp. 12–21, 2000.

[39] S. Bhandarkar, N. E. Sadry, A. N. Reddy, and N. H. Vaidya, “TCP-DCR: A novel protocol for tolerating wireless channel errors,” *IEEE Trans. on Mobile Computing*, vol. 4, no. 5, pp. 517–529, 2005.

[40] C. Casetti, M. Gerla, S. Mascolo, M. Sanadidi, and R. Wang, “TCP westwood: End-to-end congestion control for wired/wireless networks,” *Wireless Networks*, vol. 8, pp. 467–479, 2002.

[41] K. Shi, Y. Shu, O. Yang, and J. Luo, "Receiver-assisted congestion control to achieve high throughput in lossy wireless networks," *IEEE Trans. on Nuclear Science*, vol. 57, no. 2, pp. 491–496, 2010.

[42] K. Xu, Y. Tian, and N. Ansari, "Tcp-jersey for wireless ip communications," *Selected Areas in Communications, IEEE Journal on*, vol. 22, no. 4, pp. 747–756, 2004.

[43] S. Biaz and N. H. Vaidya, "De-randomizing congestion losses to improve TCP performance over wired-wireless networks," *IEEE/ACM Trans. on Networking*, vol. 13, no. 3, pp. 596–608, 2005.

[44] Y. Shu, W. Ge, N. Jiang, Y. Kang, and J. Luo, "Mobile-host-centric transport protocol foreast experiment," *IEEE Trans. on Nuclear Science*, vol. 55, no. 1, pp. 209–216, 2008.

[45] G. Rittenhouse, "Next generation wireless networks," *Proc. INFORMS*, 2004.

[46] M. Arshad and M. Mian, "Issues of multihoming implementation using FAST TCP: a simulation based analysis," *Proc. IJCSNS*, vol. 8, no. 9, pp. 104–114, 2008.

[47] D. Wischik, M. Handley, and M. Braun, "The resource pooling principle," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 5, pp. 47–52, 2008.

[48] W. Lou, W. Liu, and Y. Zhang, "Performance optimization using multipath routing in mobile ad hoc and wireless sensor networks," *Combinatorial optimization in communication networks*, pp. 117–146, 2006.

[49] F. Zhong, C. K. Yeo, and B. S. Lee, "Adaptive load balancing algorithm for multiple homing mobile nodes," *Journal of Network and Computer Applications*, vol. 35, no. 1, pp. 316 – 327, 2012.

[50] K. Leung, V. Li, and D. Yang, "An overview of packet reordering in transmission control protocol (TCP): problems, solutions, and challenges," *IEEE Trans. Parallel and Distributed Systems*, vol. 18, no. 4, pp. 522–535, 2007.

[51] N. Piratla and A. Jayasumana, "Metrics for packet reordering a comparative analysis," *Int. Journal of Communication Systems*, vol. 21, no. 1, pp. 99–113, 2008.

[52] A. Afanasyev, N. Tilley, P. Reiher, and L. Kleinrock, "Host-to-host congestion control for TCP," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 3, pp. 304–342, 2010.

[53] F. Paganini, Z. Wang, J. Doyle, and S. Low, "Congestion control for high performance, stability, and fairness in general networks," *IEEE/ACM Trans. Networking*, vol. 13, no. 1, pp. 43–56, 2005.

[54] J. Jaffe, "Bottleneck flow control," *IEEE Trans. Communications*, vol. 29, no. 7, pp. 954–962, 1981.

[55] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. Telecommunications*, vol. 8, no. 1, pp. 33–37, 1997.

[56] J. Mo and J. Walrand, “Fair end-to-end window-based congestion control,” *IEEE/ACM Trans. Networking*, vol. 8, no. 5, pp. 556–567, 2000.

[57] R. Mazumdar, L. Mason, and C. Douligeris, “Fairness in network optimal flow control: Optimality of product forms,” *IEEE Trans. Communications*, vol. 39, no. 5, pp. 775–782, 1991.

[58] 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.

[59] S. Low, L. Andrew, and B. Wydrowski, “Understanding XCP: Equilibrium and fairness,” in *Proc. Annual Joint Conf. of the IEEE Computer and Communications Societies*, vol. 2, pp. 1025–1036, IEEE, 2005.

[60] M. Neely, E. Modiano, and C. Li, “Fairness and optimal stochastic control for heterogeneous networks,” in *Proc. Annual Joint Conf. of the IEEE Computer and Communications Societies*, vol. 3, pp. 1723–1734, 2005.

[61] J. yves Le Boudec, “Rate adaptation, congestion control and fairness: A tutorial,” 2008.

[62] T. Dreibholz, M. Becke, J. Pulinthanath, and E. Rathgeb, “Applying TCP-friendly congestion control to concurrent multipath transfer,” in *Proc. IEEE Int. conf. Advanced Information Networking and Applications*, pp. 312–319, 2010.

[63] A. Matsumoto, M. Kozuka, K. Fujikawa, and Y. Okabe, “TCP multi-home options,” *draft-arifumi-tcp-mh-00, IETF*, 2003.

[64] P. Key, L. Massoulié, and P. Towsley, “Path selection and multipath congestion control,” in *Proc. IEEE Int. conf. Computer Communications*, pp. 143–151, 2007.

[65] M. Honda, Y. Nishida, L. Eggert, P. Sarolahti, and H. Tokuda, “Multipath congestion control for shared bottleneck,” in *PFLDNeT workshop*, 2009.

[66] Y. Yuan, Z. Zhang, J. Li, J. Shi, J. Zhou, G. Fang, and E. Dutkiewicz, “Extension of SCTP for concurrent multi-path transfer with parallel subflows,” in *Proc. IEEE Wireless Communications and Networking Conf.*, pp. 1–6, 2010.

[67] I. Rhee, N. Balaguru, and G. Rouskas, “MTCP: Scalable TCP-like congestion control for reliable multicast,” *Computer Networks*, vol. 38, no. 5, pp. 553–575, 2002.

[68] C. Chang, T. Tsai, and Y. Chen, “Utility and game-theory based network selection scheme in heterogeneous wireless networks,” in *Proc. IEEE Wireless Communications and Networking Conf.*, pp. 1–5, 2009.

[69] E. Halepovic, C. Williamson, and M. Ghaderi, “Wireless data traffic: a decade of change,” *IEEE Network*, vol. 23, no. 2, pp. 20–26, 2009.

[70] Q. Huang, K. Ko, and V. Iversen, "Performance modeling for heterogeneous wireless networks with multi-service overflow traffic," in *Proc. IEEE Global Telecommunications Conf.*, pp. 1–7, 2009.

[71] M. Greiner, M. Jobmann, and L. Lipsky, "The importance of power-tail distributions for modeling queueing systems," *Operations Research*, vol. 47, no. 2, pp. 313–326, 1999.

[72] Z. Sahinoglu and S. Tekinay, "On multimedia networks: self-similar traffic and network performance," *IEEE Communications Magazine*, vol. 37, no. 1, pp. 48–52, 1999.

[73] J. Olsén, *Stochastic modeling and simulation of the TCP protocol.* Matematiska institutionen University, 2003.

[74] A. Gurtov and S. Floyd, "Modeling wireless links for transport protocols," *SIGCOMM Computing Communication*, vol. 34, pp. 85–96, Apr. 2004.

[75] C. Casetti and M. Meo, "A new approach to model the stationary behavior of TCP connections," in *Proc. Annual Joint Conf. of the IEEE Computer and Communications Societies*, vol. 1, pp. 367–375, 2000.

[76] V. Jacobson, "Congestion avoidance and control," in *ACM SIGCOMM Computer Communication Review*, vol. 18, pp. 314–329, 1988.

[77] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. on Networking*, vol. 1, no. 4, pp. 397–413, 1993.

[78] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications.* Wiley-Interscience, 2006.

[79] L. Brakmo and L. Peterson, "TCP Vegas: End to end congestion avoidance on a global Internet," *IEEE Journal Selected Areas Communications*, vol. 13, no. 8, pp. 1465–1480, 1995.

[80] J. D. Little, "Little's law as viewed on its 50th anniversary," *Operations Research*, vol. 59, no. 3, pp. 536–549, 2011.

[81] S.-H. Kim and W. Whitt, "Statistical analysis with littles law," *Operations Research, articles in advance*, pp. 1–17, 2013.

[82] M. Gerla, M. Sanadidi, R. Wang, A. Zanella, C. Casetti, and S. Mascolo, "TCP Westwood: Congestion window control using bandwidth estimation," in *Proc. IEEE Global Telecommunications Conf.*, vol. 3, pp. 1698–1702, 2001.

[83] X. D. Wei, *Microscopic behaviour of internet congestion control.* PhD thesis, California Institute of Technology, 2007.

[84] C. Raiciu, D. Wischik, and M. Handley, "Practical congestion control for multipath transport protocols," 2009.