Bandwidth Allocation with Fairness in Multimedia Networks

A thesis submitted to the
Graduate School of Natural and Applied Sciences

by

Hamed HAMZEH

in partial fulfillment for the
degree of Master of Science

in
Data Science

İSTANBUL
ŞEHİR
UNIVERSITY

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science in Data Science.

APPROVED BY:

Prof. Dr. Shervin Shirmohammadi ....
(Thesis Advisor)

Assist. Prof. Dr. Mehmet Serkan Apaydin ....

Assist. Prof. Dr. Eliya Buyukkaya ....

This is to confirm that this thesis complies with all the standards set by the Graduate School of Natural and Applied Sciences of İstanbul Şehir University:

DATE OF APPROVAL: 30 June 2017

SEAL/SIGNATURE:

Declaration of Authorship

I, Hamed Hamzeh, declare that this thesis titled, 'Bandwidth Allocation with Fairness in Multimedia Networks' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date: 17/07/2017

Bandwidth Allocation with Fairness in Multimedia Networks

Hamed HAMZEH

Abstract

The high demand of bandwidth from multimedia applications, especially video applications which consume the great majority of the Internet bandwidth, has caused a challenge for service providers and network operators. On the one hand, the allocation of bandwidth in a fair manner for multimedia users is necessary, so that the total utility of all users is maximized for higher quality of experience. On the other hand, optimizing the utilization of network resources such as maximizing throughput is also important for network operators to reduce the cost and/or maximize the profits. These two requirements could potentially be conflicting; hence, achieving both at the same time is challenging, and the reason why very few previous efforts have targeted this problem. Examples include Traffic Management Using Multipath Protocol (TRUMP) and Logarithmic-Based Multipath Protocol (LBMP), both of which achieve good results but are not without shortcomings. At the first step, Network Utility Maximization(NUM) problem has been considered as an initial stage to design any traffic engineering method. In this thesis and by considering the mentioned issues, first of all we take into account NUM problem and optimization decomposition methods. We then propose a model based on those methods as Hopcount-Based Fair Allocation(HBFA) to tackle the fair bandwidth allocation issue by comparing it to TRUMP. Although, HFBA tackles the fairness problem, but it cannot reach the desired results in all possible path selections. Hence, in order to address that issue, we propose a model, Price-Based Fair Bandwidth Allocation (PBFA) by implementing a new sending rate adaptation formula and combining it by an intuitive investment method to optimize the feedback prices from the links to achieve an efficient model. The method is evaluated by using different simulations and topologies under various network conditions. Our results show that PBFA not only provides a fairer bandwidth allocation compared to TRUMP, but also improves the link utilization and throughput. The conducted evaluations show that PBFA achieves improvements of as much as 90% in fairness, 207% in throughput, and 91% in utility compared to TRUMP.

. . .

Keywords: Traffic Engineering, network resource management, Multi-path, TRUMP.

Multimedya Ağlarında Adil olarak Bant Genişliği Tahsisi

Hamed Hamzeh

Öz

Bant genişliğini multimedya uygulamaları, büyük çoğunluğu Internet bant genişliği tüketen özellikle video uygulamaları ile yüksek talep bir meydan okuma servis sağlayıcıları ve şebeke operatörleri için neden oldu. Toplam yardımcı programı, tüm kullanıcıların deneyimi daha yüksek kalite için ekranı bir yandan, multimedya kullanıcıları için adil bir şekilde bant genişliği tahsisi, gereklidir. öte yandan, maximizing performans gibi ağ kaynaklarının kullanımını en iyi duruma getirme de maliyeti azaltmak ve/veya kar maksimize etmek ağ operatörleri için önemlidir. Bu iki gereksinim potansiyel çakışan olabilir; Bu nedenle, her ikisini de aynı anda ulaşmak zor, ve neden çok az önceki çabaları bu sorunu hedef neden. örnekler trafik yönetimi kullanarak çok yollu Protokolü (TRUMP) ve Logaritmik tabanlı çok yollu iletişim kuralı (LBMP), ikisi de iyi sonuçlar elde etmek ama eksiklikleri değildir. İlk adımda, ağ yardımcı programı'nı Maximization(NUM) sorun bir ilk aşamada herhangi bir trafik Mühendisliği yöntemi tasarımı için kabul edilmiştir. Her şeyden önce bu tez ve bahsedilen sorunları göz önüne alındığında, hesap NUM sorun ve en iyi duruma getirme ayrıştırma yöntemleri alıyoruz. Biz sonra TRUMP için karşılaştırarak adil bant genişliği ayırma sorunu çözmek için Hopcount-Based adil Allocation(HBFA) olarak bu yöntemlerde temel bir model öneriyoruz. Rağmen HFBA adalet sorunu ele, ama tüm olası yolu seçimleri istenen sonuçları yetişemem. Bu nedenle, sorunu, manken önerdiğimiz adres için yeni bir gönderme uygulayarak fiyat tabanlı adil bant genişliği ayırma (PBFA) oranı uyarlama formülü ve verimli bir modeli elde etmek için bağlantıları geribildirim fiyatlardan en iyi duruma getirmek için bir sezgisel yatırım yöntemi ile birleştirerek. Yöntem farklı simülasyonları ve çeşitli ağ koşulları altında topolojiler kullanarak değerlendirilir. Bizim sonuçlar PBFA TRUMP için karşılaştırıldığında daha adil bir bant genişliği ayırma sağlar, ancak aynı zamanda bağlantı kullanımını ve performansı artırır olduğunu gösterir. Yapılan değerlendirme PBFA gelişmeler kadar 90% adalet, işlem hacmi yüzde 207 ve yardımcı programı TRUMP için karşılaştırıldığında 91%'i elde göster.

Anahtar Sözcükler: Mühendislik, Deneysel PsikolojiTRUMP, mühendislik Rating, PBFA, adil bant genişliği tahsisi, çok yollu.

Acknowledgments

First of all, I would like to express my gratitude to my advisor Prof. Shervin Shirmohammadi for the constant support of my Master study and research, for his patience, motivation, eagerness, and vast knowledge. His guidance helped me in all the time of research and writing of my thesis.

In addition to my advisor, I would like to appreciate Mr. Mahdi Hemmati from University of Ottawa for his encouragement, insightful comments, and hard questions.

Finally, I would like to thank my family, especially my dear wife who supported and tolerated me during my studies and writing this dissertation and an special thanks for my parents and brother for emotionally support me in this way.

Contents

Declaration of Authorship **ii**

Abstract **iii**

Öz **iv**

Acknowledgments **v**

List of Figures **viii**

List of Tables **ix**

1 Introduction **1**
- 1.1 Introduction . . . 1
- 1.2 Motivation . . . 3
- 1.3 Research problem and Objective . . . 4
- 1.4 Research Contributions . . . 4
- 1.5 Publications . . . 5
- 1.6 Thesis Outline . . . 5

2 Background **7**
- 2.1 Optimization in Internet traffic Engineering . . . 8
- 2.2 Multipath Routing . . . 9
- 2.3 Optimization Decomposition . . . 10
  - 2.3.1 Selecting an Objective Function . . . 11
- 2.4 Maximize aggregate user utility (DUMP) . . . 11
- 2.5 Distributed algorithms and Multiple decomposition: . . . 14
  - 2.5.1 Partial-Dual: . . . 14
  - 2.5.2 Primal-Dual: . . . 15
  - 2.5.3 Full-Dual: . . . 15
  - 2.5.4 Direct Path-rate upgrade or Primal Driven: . . . 16
    - 2.5.4.1 Sending Rate Update: . . . 16
    - 2.5.4.2 Feedback Price Update: . . . 16

3 Related Works **17**
- 3.1 TRUMP Algorithm . . . 17
  - 3.1.1 Feedback price update: . . . 18
  - 3.1.2 Sending rate Upgrade: . . . 18
  - 3.1.3 Convergence properties of TRUMP . . . 18

3.1.4 Fair bandwidth allocation: 19
3.1.5 Advantages of TRUMP algorithm 19
3.1.6 Drawbacks of TRUMP algorithm 20
3.2 LBMP Algorithm 20
3.2.1 Main differences between LBMP and TRUMP: 21

4 Hopcount-Based Fair Bandwidth Allocation (HFBA) 22
4.1 System Design: 22
4.1.1 Topology Construction 23
4.1.2 Sending Rate Adaptation 23
4.1.3 Using the Sending Rate Adaptation in MATLAB 24
4.1.4 A simple numerical example to present the working of the model 24
4.2 Performance Evaluations 26
4.2.1 Fairness Index 27
4.3 Results: 27
4.3.1 The simulation of Proposed model 27
4.3.2 NSF Topology 28
4.3.2.1 Evaluations in terms of Sending rate and fairness measure 29
4.3.3 Utility and Throughput 30
4.3.4 Evaluations on Abilene Topology: 31
4.3.4.1 Sending rate allocation and fairness measure: 32
4.3.5 Evaluations on Cost Topology: 33
4.3.5.1 Bandwidth allocation and fairness measure: 35
4.3.6 Evaluations on Core Topology: 35
4.3.6.1 Bandwidth allocation and fairness measure: 36
4.3.7 Evaluations on NTT topology: 37

5 Priced-Based Fair Bandwidth Allocation (PBFA) 39
5.1 System Design 39
5.2 Model Formulation 39
5.3 Performance Evaluation 41
5.4 Results 41
5.4.1 Evaluations for NTT 42
5.4.2 Evaluations for Abilene 43
5.4.3 Evaluations for NSF 44
5.4.4 Evaluations for COST 45
5.4.5 Evaluations for CORE 46
5.4.6 Fairness Measure 46

6 Conclusion and Future Works 48
6.1 Conclusion 48
6.2 Future Works 49
6.2.1 Fairness feature for all ω values 49
6.2.2 Cloud Computing 49
6.2.3 Convergence of TRUMP 49

List of Figures

1.1 Multipath Routing 2

2.1 Multipath Routing 10
2.2 Maximizing aggregate user utility 11

3.1 Sharing a bottleneck link by 3-flows in TRUMP 17
3.2 Share Topology 19
3.3 Gradient projection approach 21
3.4 Sending rate update 21

4.1 Hop-count diversity and bottleneck sharing 22
4.2 A sample topology to implement the working of the system 25
4.3 NSF Topology 28
4.4 Sending rate allocation in TRUMP and HFBA 29
4.5 The value of fairness index for TRUMP and HFBA 29
4.6 Sending rate in different hop-lengths for TRUMP and HFBA 30
4.7 Abilene Topology 31
4.8 The allocation of bandwidth in TRUMP and HFBA 32
4.9 The values for fairness index in TRUMP and HFBA 33
4.10 Cost Topology 34
4.11 Bandwidth allocation in TRUMP and HFBA 35
4.12 Fairness measures for TRUMP and HFBA 35
4.13 Core Topology 36
4.14 Bandwidth allocation in TRUMP and Proposed Model 36
4.15 NTT Topology 37
4.16 Sending rate allocation in TRUMP and HFBA 38

5.1 Bandwidth allocation for 7 competing sources in different link capacities in NTT topology 42
5.2 Bandwidth allocation for 7 competing sources in different link capacities in Abilene topology 43
5.3 Sendin rate allocation in three models 44
5.4 Bandwidth allocation in 3 models in different link capacities 45
5.5 Bandwidth allocation in 3 models in different link capacities 46

List of Tables

2.1 The comparison table for 4 decomposition algorithms 16

4.1 A comparison of working our proposed model and TRUMP 25
4.2 Characteristics of the topologies 27
4.3 Comparing of two models for three factors where w=1 30
4.4 Comparing of two models for three sources where w=1 31
4.5 Bandwidth allocation in TRUMP and HFBA 32
4.6 Utility and throughput 32
4.7 Bandwidth allocation in TRUMP and HFBA 34
4.8 Utility and Throughput 34
4.9 Bandwidth allocation in TRUMP and HFBA 36
4.10 Fairnes index in different flow numbers 37
4.11 Utility and throughput 37
4.12 Bandwidth allocation in TRUMP and HFBA 37
4.13 Utility and throughput in TRUMP and HFBA 38

5.1 Bandwidth allocation in NTT for three models 42
5.2 Utility and throughput in different link capacities 42
5.3 Bandwidth allocation in different capacities 43
5.4 Utility and throughput 43
5.5 Bandwidth allocation in different linc capacities 44
5.6 Utility and Throughput 44
5.7 Bandwidth allocation in 3 models in different link capacities 45
5.8 Utility and throughput in different capacities 45
5.9 Bandwidth allocation in different link capacities 46
5.10 Utility and throughput in different link capacities for 3 models 46
5.11 Fairness measure for NTT 46
5.12 Fairness measure for NSF 47
5.13 Fairness measure for Abilene 47
5.14 Fairness measure for COST 47
5.15 Fairness measure for CORE 47

Chapter 1

Introduction

1.1 Introduction

Bandwidth allocation is one of the most significant issues in today's computer networks; especially for multimedia applications such as video streaming which not only consume huge amounts of bandwidth, but also their users expect a maximum amount of quality of experience. While IP traffic itself has been growing at an annual rate of 21%, it is predicted that up to 90% of global IP traffic will be video by 2018 [1]. This is not surprising considering the popularity of services such as Netflix, YouTube, Amazon Video, etc. In such applications, fairness becomes an important contributor to the user's quality of experience. For example, if two or more users with practically identical subscriptions share the same bottleneck link, it is unfair to allocate considerably more bandwidth to some users compared to others. This is particularly so for video streaming, where the users' utility function is sigmoidal [2], meaning that as the bandwidth allocated to a user decreases linearly, the quality of experience for that user decreases exponentially. So, the fair distribution of bandwidth among the users becomes an indispensable requirement for multimedia applications. At the same time, the network operator has the goal of maximizing the resource utilization of its own network, so as to maximize profits.

This problem can be addressed by proper traffic engineering that can optimize the traffic based on requirements from users, network operators, and network resources [3]. Traffic engineering methods improve the efficiency of bandwidth allocation and control the congestion typically by using some sort of decomposition method [4][5]. To work in today's networks, these methods must be compatible with multipath routing, in which there are different possible paths to send the traffic through. In such a multipath setting shown in figure 1, the network operator adjusts the sending rate of each flow, typically by calculating the path prices derived from the link prices of each path [6][7][8] . So, a specific bandwidth is allocated to each path based on that pathâs price.

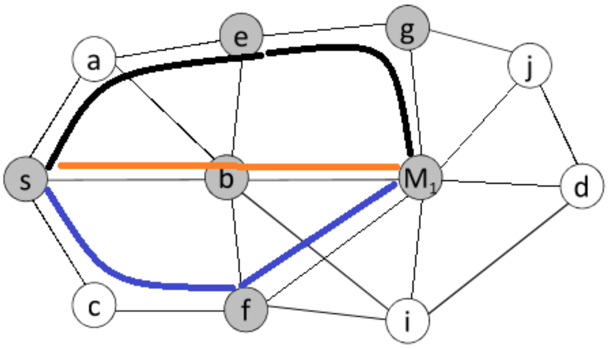

FIGURE 1.1: Multipath Routing

The solutions which are distributed are able to be originated by applying optimization decomposition method which is a regular optimization method to decompose a particular optimization problem into the several sub-problems [4][5]. An individual sub-problem can be resolved through a distinct network component, for instance a router, a terminal host or a link. In order that a distributed solution reach its goals, it is necessary for network components to synchronize with each other by transmitting messages, or calculating recognizable measures such as link weight, packet loss and delay. To guarantee the convergence, distributed solutions originated from optimization decomposition regularly encompasses iterative upgrades through adjustable parameters [3]. The adjustable parameters assist to reasonable amount of adaptation. Optimization decomposition has been extensively applied to originate distributed solutions to a diversity of networking complications [4].

Redesigning an Internet traffic engineering method is a top-down approach, and it is done by choosing an instinctive and applied the objective function that stabilizes the goals of users and operators. Through the means of optimization decomposition methods, in chapter 2 we will introduce four particular distributed solutions wherein sources adjust sending rates on different routes to the end point or destination. A benefit of distributed algorithms is that they adjust the sending rates based on round trip time(RTT) and can reply rapidly to the traffic changes. The calculation of sending rates are different in all four decomposition techniques [4]. The theory of optimization ensures that those algorithms converge to a fixed point. All distributed algorithms are susceptible to the tuning parameters but their functioning are very well. TRUMP algorithm which will be covered later, is created by combining the finest parts of four decomposition methods. All today's distributed solutions are designed to optimize multipath networks.

The optimization methods are used to formulize an Internet traffic management model. Each optimization problem contains an objective function[9][10] and variables. In order that traffic engineering, by taking into account optimization variables like as routing and source rates, there will be more flexible bandwidth allocation.

Accordingly, the question of how to allocate the bandwidth in a fairness manner so that all users benefit the fair bandwidth, is challenging. Traffic engineering methods, provide various ways to improve the efficiency of bandwidth allocation and control the congestion by using the mathematical techniques such as decomposition methods [4][5]. The important thing to be mentioned is that every model should be optimized at the first instead of optimizing an existing model [3]. Among the methods proposed for multipath routing, two of them particularly target fairness: Traffic Management Using Multipath Protocol (TRUMP) and Logarithmic-Based Multipath Protocol (LBMP). While they achieve good results, they are not without shortcomings, as we will show in details in section II. Briefly, their main shortcoming is that both methods do not consider in their fairness metric the hop counts from the source to the user. As a result, users with a higher hop count will be penalized and treated unfairly compared to users with a lower hop count.

To avoid this problem, first of all we propose HBFA to over come the fair bandwidth allocation in diverse hop counts. Then, by taking into account that HBFA does not work in all path selections, we propose Price-Based Fair Bandwidth Allocation (PBFA) method which performs fairer than TRUMP and LBMP for users with varying hop counts. PBFA uses multiple decomposition methods to implement a rate adaptation formula to split the traffic in a fair manner for the users while also increasing the utility and throughput of the system. PBFA is inspired by top-down algorithm design, specifically multiple decomposition [9][10], and calculates the link prices to use them in an investment model where in each iteration, the less bandwidth a flow consumes, the more bandwidth it will get in the next iteration. Using this investment model, the feedback price is optimized.

We designed PBFA in MATLAB and ran multiple simulations under different real-world network topologies to compare it with TRUMP in terms of fairness, throughput, and utility. To measure the fairness, we used Jainâs fairness index [11]. The results show that PBFA achieves improvements of as much as 90% in fairness, 207% in throughput, and 91% in utility compared to TRUMP.

1.2 Motivation

Traffic engineering methods, provide various ways to improve the efficiency of bandwidth allocation and control the congestion by using the mathematical techniques such as decomposition methods. Some of these strategies are investigated until now but there

are many gaps. In a Network topology, there are many nodes and links between source and destination to send the packets. Some of the packets have to travel a long distance to reach the destination. So, they need more bandwidth than the other paths which go through from the paths with the minimum distance. Consequently, paths with minimum hop-length will get the higher bandwidth than the paths with more hop-length. So, by taking into account the mentioned problems, we motivated to propose a new model to allocated the bandwidth in a fairness manner to satisfy the user demands, quality of service and also increase the utility and throughput.
The proposed model is based on decomposition methods which leads to comp up with a new solution for sending rate adaptation in different paths of a network topology. The designed model is relatively flexible in splitting the traffic among the paths with diverse hop counts which is more fairer than previous models such as TRUMP.

1.3 Research problem and Objective

As we discussed in section 1.1, bandwidth allocation is one of the significant factors in computer networks. In most of the algorithms which are proposed until now, the fair bandwidth allocation is taken into the consideration in different ways. In TRUMP algorithm that will be discussed in section 3, there is a gap in terms of fair bandwidth allocation in different hop-counts. In TRUMP algorithm, the paths with more hop-counts, get the minimum bandwidth compared to the paths with minimum hop-counts.

In this dissertation, the aim is to find an effective model to split the bandwidth among the paths with different hop-counts so that all the paths get a fairer allocation of bandwidth. In addition to this, the other purpose is to maximize the link utility and throughput in the network. So, by considering the decomposition models, we can reach our objectives by defining a new sending rate adaptation formula.

1.4 Research Contributions

This dissertation presents a distributed solution to achieve a fair bandwidth allocation model by defining a new sending rate adaptation formula that all paths with different hop-length get the fair allocation of bandwidth. The contributions are listed as follows:

1. In network congestion control, the fair bandwidth allocation among paths with diverse hop-counts is an important issue and the purpose of proposed model is to

optimize the sending rate adaptation formula to achieve the bandwidth allocation with fairness in multipath networks. The designing of the system is based on different simulations by considering already proposed algorithms and our proposed system and analysing the results.

2. The new proposed model is very good in terms of increasing the quality of service which leads to maximize the user utility and overall throughput of the system. Based on the experiments and simulations in MATLAB and also utilizing the optimization models, we could achieve the desired outputs of our new model.

1.5 Publications

The output of this thesis project is a publication published in International Journal of Computer and Communications Engineering(IJCCE) and presented in 3rd International Conference in Computer and Information Technology(ICCIT 2017), Kusadasi, Turkey and The Digital Media Industry and Academic Forum (DMIAF) IEEE, 2017. Athens, Greece.

[1] H. Hamzeh, M. Hemmati and S. Shirmohammadi. Bandwidth Allocation With Fairness in Multipath Networks.*Published in International Journal of Computer and Communication Engineering(IJCCE)*, ISSN: 2010-3743.

[2] H. Hamzeh, M. Hemmati and S. Shirmohammadi. Price-Based Fair Bandwidth Allocation For Networked Multimedia. *International Symposium on Multimedia (ISM)*, Taipei, IEEE 2017.

1.6 Thesis Outline

This thesis implements a design for fair bandwidth allocation in multipath networks and the remainder outline is listed as follows:

1. **Chapter 1-Introduction** This section contains the introduction of thesis subject, motivation, research objectives and contributions as well as research outputs in the form of publications.
2. **Chapter 2-Background** Includes the overview of optimization theory in Internet traffic management, Multipath routing, Maximizing aggregate user utility, decomposition methods, distributed algorithms and different types of them.

3. **Chapter 3-Related Works** Different already proposed distributed methods such as, DATE, LBMP, TRUMP algorithms and their structure and features.
4. **Chapter 4- HFBA Model** This section presents the design and simulation of Hopcount-Based Fair Bandwidth Allocation (HFBA) model and the conducted experiments and results.
5. **Chapter 5- PBFA Model** Including the design and simulations of Priced-Based Fair Bandwidth Allocation (PBFA) model.
6. **Chapter 6-Conclusion** This section is a review of the research project.

Chapter 2

Background

Internet is expanding very quickly and the number of users using it is increasing. So, by increasing the demands, it is important to consider the traffic in transmitting data between the end hosts in the network. This huge amount of data transmission leads to occur the congestion problem. Congestion [13][14][15][18] happens when the demands exceed the actual capacity of the links. Occurring the congestion may degrade the quality of service, user satisfaction and wastage the bandwidth, time and energy. Consequently, the purpose of applying Internet traffic management techniques is to address the problems that are mentioned.

Traffic management determines how much traffic should overpass for every path in the network [15]. The end hosts apply congestion control to adjust the sending rate, and routing protocols choose a single path among the source and the destination. As it is shown in figure 2.1, source A calculates its sending rate for every three paths to destination C according the feedback price.

In today's Internet, terminal hosts applying the TCP [16]to adjust their sending rates with respect to the congestion in the network [17]. In addition, network operators check the network if there are weighty links. If that's the case,they will adjust the transmission rate to avoid the congestion. TCP congestion control [18] considers that the network routs don't shift, and traffic engineering presumes that the introduced traffic persistence. Traffic Management incorporates three main players[9][10]: users, routers and operators. In the Internet, clients run congestion control with adjusting their sending rates In the edge of the Network based on the Network Circumstances. Inside a single self-sufficient framework (AS), Routers run shortest-path algorithm of join weights. Operators tune connection weights with minimizing the congested links.

Internet traffic management is considerably enhanced compared to the first generations of the Networks. Still, because of the biological growth of traffic management, there are numerous weaknesses. First of all, operators adjust link loads supposing that traffic is stationary, and terminal hosts adjust the sending rates supposing that the routing is stable. Secondly, the link-load location issue is computationally puzzling and, meanwhile this offline optimization happens at the period of hours, it does not adjust to variations in existing network traffic. Lastly, today traffic engineering is planned to throughput maximization, and does not think through that applications have diverse performance goals, for example delay minimization.

One of the most considerable issues in internet traffic engineering is the fair bandwidth allocation when there are multiple paths with diverse hop-counts. In some cases, longer hops get the minimum bandwidth compared to the shorter paths and this degrades the fairness feature.

The aim for planning the total traffic engineering structure [4][20] that are listed as follows:

1. **Fairness Feature:** Fair bandwidth allocation is one of the main problems that should be taken into the account when we want to investigate a traffic management system. Overall traffic should be allocated fairly through different paths with different hop-length which is the main concentration of this dissertation.
2. **Efficiency:** This is related to the bandwidth that should be effectively used in order to maximize aggregate efficiency goals.
3. **Robustness:** in terms of topology variations and traffic changes, all the protocols are subjected to be robustness.
4. **Enforcement-able:** By considering available methodologies, all the protocols are subjected to implemented.

The normal aim of a service provider is to maximizing the cumulative efficiency through the numerous traffic modules, wherever each traffic module has a diverse efficiency objective.

2.1 Optimization in Internet traffic Engineering

By considering various existing mathematical tools, the theory of optimization [5][14] is a usual option to investigate and redesign the Internet traffic management. Due to its functionality in investigating and designing of numerous constituents of traffic management, the new traffic management protocols can be located on a robust base by using optimization algorithms. The optimization challenges in the whole network can be solved by applying Internet traffic management and network congestion control. Traffic engineering involves of gathering the quantities of the traffic environment and the perceived weight among each pair of ingress and egress points and accomplishing an intensive reduction of a cost function that reflects the consequential utilization of total links. In the opposite side, optimization problem can be resolved indirectly by using TCP (Transport Control Protocol) in a distributed way, where the numerous alternatives of TCP vary in the form of user utility. In addition, the theory of optimization is handled to examine suggested traffic management protocols, in addition to the designing of distributed congestion control protocols.

Distributed solutions are able to be originated by applying optimization decomposition method which is a regular optimization model to decompose a particular optimization [5][14][21] problem into several sub-problems. Individual sub-problem can be resolved through a distinct network component such as a router, a terminal host or a link. In order that the distributed solution reach its goals, it is necessary for the network components to synchronize with each other by transmitting messages, or calculating recognizable measures for example link weight, packet loss and delay. To guarantee the convergence, distributed solutions originated from optimization decomposition regularly encompasses iterative upgrades through adjustable parameters. Optimization decomposition has been extensively applied to originate distributed solutions to a diversity of networking complications.

2.2 Multipath Routing

Scientists and specialists approve multipath routing [8][22][23] delivers efficiency profits to traffic engineering. Also, many prevalent used routing protocols, choose merely a particular path for the traffic among each end host. Multipath routing has many positive outcomes which are listed as follows:

1. **Improving the efficiency of applications:** According to the requirements of different applications, most of them need high demand of bandwidth, so they can achieve that bandwidth through multiple routes.

2. **Enhancing the reliability:** In a network, there are packet losses and link failures. So, by having multiple routes, traffic is able to shift to another path.
3. **keeping away from congested routes:** When there are different paths, traffic can be switched to substitute routes to avoid congestion problem.

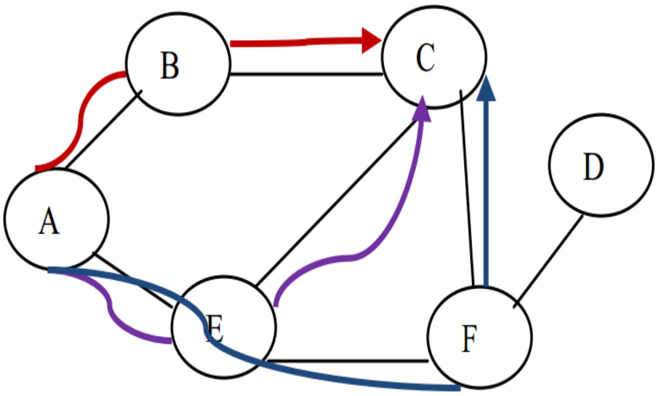

FIGURE 2.1: Multipath Routing

2.3 Optimization Decomposition

Internet traffic engineering is considered by applying decomposition methods [4][10]. It is the method of breaking down an individual optimization problem to the several sub-problems. Internet traffic engineering is a top-down approach, and it is done by choosing an instinctive and applied objective function which will be covered in the next section that stabilizes the objective of the users and the operators. Through the means of optimization decomposition methods, section introduces four particular distributed solutions wherein sources adjust sending rates on different paths to destination.A benefit of distributed algorithms is that they adjust the sending rates based on the round trip time(RTT) and can reply rapidly to the traffic changes. The calculation of sending rates are different in all four decomposition techniques. The theory of optimization ensures that those algorithms converge to a fixed point. All distributed algorithms are susceptible to the tuning parameters but their functioning are very well. TRUMP algorithm which will be covered in section 3, is created by combining the finest parts of four decomposition methods.

2.3.1 Selecting an Objective Function

The optimization methods are used to formulize an Internet traffic management model. Each optimization problem contains an objective function [9][10][24] and variables. In order that traffic engineering, by taking into account optimization variables like as routing and source rates, there will be more flexible bandwidth allocation. There is also a constraint in the objective function where the link load should not surpass the overall capacity of the link.

2.4 Maximize aggregate user utility (DUMP)

Making the purpose of network management as the aggregate utility maximization or *Network Utility Maximization(NUM)* [6][25][26] is subjected to the capacity constraints, where the aggregate utility is the sum of the particular user utilities.
Resource allocation issues can be implemented as a compelled maximization of utility function. User utility $U_r x_r$ is a kind of satisfaction between the sender and the receiver, where U is a concave, non-negative, increasing and twice-differentiable function, which able to symbolize the flexibility of the traffic or control the bandwidth allocation in a fair manner. Also, x_r is the sending rate of source r. In this case, the optimization problem can be written as follows:

$$\begin{gathered}\text{Maximize}\Sigma_r U_r(x_r) \\ \text{Subject to} Rx <= c\end{gathered} \tag{2.1}$$

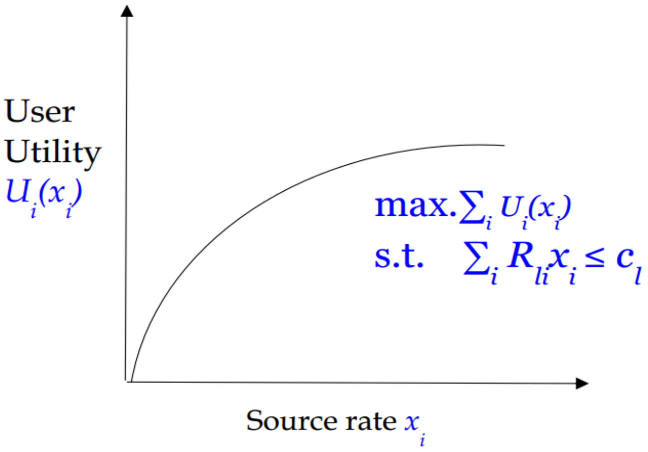

FIGURE 2.2: Maximizing aggregate user utility

According to the above formulation, R is a routing matrix, consists of links l and also sources r. In addition, c is the link capacity where the sending rate of source r should smaller than or equal to c.

In order to obtain the multipath routing, it is better to use the variable z_j^r to show the sending rate of flow i in its jth path. In this case, the routing matrix can be implemented as matrix H [9][26][27]:

$$H = \begin{cases} 1 & ifpath\mathrm{j}ofsource\mathrm{r}useslink\mathrm{l} \\ 0 & otherwise \end{cases} \tag{2.2}$$

The routing matrix H cannot show all the plausible paths in a network[10]. So, the formulation implemented in Figure 2.2 can be updated to:

$$\begin{gathered} \text{Maximize}\Sigma_r U_r(z_j^r) \\ \text{Subject to}\Sigma_r \Sigma_j H_{lj}^r z_j^r <= c_l \end{gathered} \tag{2.3}$$

In this case, the above formulation is a convex optimization [9][10][22][26] problem, which is originated from *dual decomposition*, where a dual variable is presented to reduce the capacity constraint.

Feedback price and Sending rate updates are the most two important factors derived from TCP inverse engineering which are contributed to do the congestion control in the network.

Feedback price update:

$$s_l(t+1) = \Bigg[s_l(t) + \beta_s(t) \Bigg(c_l(t) - \Sigma_r \Sigma_j H_{lj}^r z_j^r(t) \Bigg) \Bigg] \tag{2.4}$$

Sending rate update:

$$z_j^r(t+1) = maximize_{z_j^r} \Bigg(U_r \Bigg) \Bigg(\sum_j z_j^r \Bigg) - z_j^r \Sigma s_l(t) H_{lj}^r \Bigg) \tag{2.5}$$

In formulation 2.4, t refers to the number of iterations and s_l is the feedback price [10][20][27][28] which equals to the differences between the link capacity and the link load. Feedback price has been used to calculate the sending rate adaptation formula and is the summation of the queuing delay and the loss price in a specific link. β_p is the step-size which is greater than zero, to ensure that the aggregate user utility will converge to an optimized point.

Whenever the link load exceeds the capacity of the link, in this case, the value of s_l

will be positive. Sending rate is upgraded by getting the feedback prices of each link. Specifically, link utility is maximized individually by every source.

A network topology consists of links that are presented by l where c_l is the link capacity and also we have N source-destination pairs. Meanwhile, the routing matrix is implemented by R_{li} which captures the proportion of source i. On the other hand, link loads are determined by the network operator. So, the network operator decides to adjust the sending rate for each path to avoid congestion in the network [9][26]. Also, the utility of a link can be implemented as

$$U_l = \Sigma_r R_{lr} x_r / c_l \tag{2.6}$$

So, in order to penalize the links which are want to be overloaded, the f function which is a convex, non-decreasing, and twice-differentiable function, is a kind of punishment when the link load exceeds the actual link capacity. The purpose of selecting this function is summarized in two sentences used in [4][5][10]. In this case, the optimization problem can be written as follows:

$$U_l = \Sigma_l f(\Sigma_r R_{lr} x_r / c_l) \tag{2.7}$$

As a result, all flows will be guided to the links which are less utilized.

By taking into account the formula in 6, the ω variable [4] is introduced to have a balance objective function, where w is a parameter to adapt the balance among the utility and cost functions. So, if we select a small value for ω, in this case the formulation in 5, will be tended to (1) and if we choose a big value for ω, then, it keeps away more link utilization. To have a good traffic management model, it is better to combine the efficiency measures By the robustness of the network, by considering the following formulation:

$$\begin{aligned} maximize \Sigma_r U_r x_r - \omega \Sigma_l f(\Sigma_r R_{li} x_r / c_l) \\ subjectto Rx <= c, x >= 0 \end{aligned} \tag{2.8}$$

This formulation takes into account a solution that impacts a deal among high aggregate utility and a low congestion of the Network, to fulfill the requirement of efficiency and robustness.

ω is a tuning parameter to adjust the harmony between the utility function and the cost function. In order to keep away from the higher link utilization, it is necessary to choose a large value for ω. Nowadays, Network operators carry out traffic management by regulating link loads Based on the immediate traffic load.

Although the efficiency and performance are important factors, on the other hand, the fairness is a crucial factor. By using a Hypothetical viewpoint, the formulation in (2.8) is fair when $\omega = 0$. It does not mean this is applicable for general values of ω.

According to the formula in (2.8), and to capture the convex optimization problem, the formula can be rewritten as follows:

$$\begin{aligned} maximize \Sigma_r U_r(z_j^r) - \omega \Sigma_l f(\Sigma_r R_{lr} x_r / c_l) \\ subject to Rx <= c, x >= 0 \end{aligned} \tag{2.9}$$

2.5 Distributed algorithms and Multiple decomposition:

In this part, we explain different distributed methods [4][15][20][29] created by using optimization Decomposition algorithms. Distributed algorithms are required when the multipath routing is taken into the consideration. All four subsequent methods upgrade the sending rates according to the feedback prices from the links. Also, there are a sort of additional correspondences among the four methods. First of all, the processes executed by the links, containing calculating the link weight. Secondly, entirely four methods experience the same message crossing overhead: From the source to the destination, the summation of link prices should be transferred.

A principled comprehension of the decomposition methods in the network utility maximization, is a base for resource allocation issue in the network. Decomposition methods basically provides the mathematical approaches to construct an analytic base for the creation of distributed algorithms for the networks to do the traffic management. So in this section, we want to review four decomposition methods [4][9][10][26] that are applicable for designing the distributed solutions.

2.5.1 Partial-Dual:

The partial-dual algorithm relates to the efficient capacity y which is a primal variable for this algorithm. The capacity constraint which is shown as $y \leqslant c$ is included. As a result, the following equation, can be used to update the efficient capacity:

$$y_l(t+1) = minimize_{(y_l <= c_l)} \omega f(y_l / c_l) - s_l(t) y_l \tag{2.10}$$

In above formulation, y is updated by the information evolved from the feedback price. It can be also translated that the efficient capacity adjusts the cost of the link utilization which is presented by f.

2.5.2 Primal-Dual:

This algorithm initially breaks down into two sub problems, one of them is accountable for any primal variable. The main issue dissolves for y, presuming a specified x^*. At the same time, the sub problem dissolves for x by assuming a constant y.

$$maximize\Sigma_r U_r(\Sigma_j z_j^r) - \omega\Sigma_l f(y_l/c_l) \tag{2.11}$$

So, the main problem is:

$$maximize\Sigma_r U_r(x^*) - \omega\Sigma_l f(y_l/c_l) \tag{2.12}$$

Also, x^* is a solution to the following function:

$$\text{Maximize}\Sigma_r U_r(x_r) \tag{2.13}$$

Hence, by using an iterative upgrade of efficient capacity, the master issue can be dissolve to:

$$y_l(t+k) = min(c_l, y_l(t) + \beta_y(s_l(t) - \omega f^{(\prime)}(y_l(t))))$$

(12)

So, the only problem between *Partial-dual and primal dual is that in primal-dual, the efficient capacity upgrades in every iteration.*

2.5.3 Full-Dual:

The full-dual algorithm is the same as the partial dual decomposition algorithm though the secondary dual variable p is implemented to weaken the constraint $y \leqslant c$. This dual variable can be translated as constancy price, as it guarantees constancy among the efficient capacity and the capacity constraint in a equivalence scale. The constancy price is upgraded in every iteration by using a sub-gradient procedure:

$$p_l(t+1) = [p_l(t) - \beta_p(c_l - y_l(t))] \tag{2.14}$$

Where, β_p is the step-size for constancy price and this price is a non-negative quantity. The Efficient capacity is upgraded by utilizing the link prices such as *Loss price and feedback price*:

$$y_l(t+1) = minimize_{y_l} \omega f(y_l/c_l) - (s_l(t) + p_l(t))y_l \tag{2.15}$$

2.5.4 Direct Path-rate upgrade or Primal Driven:

Primal-Driven algorithm presents a straight way to relax the constraint where the prior implemented methods were using secondary dual variables.

$$maximize \sum_r U_r \Big(\sum_j z_j^i) \Big) - \omega \sum_l p_l \Big(\sum_r \sum_j H_{lj}^r z_j^r \Big) \tag{2.16}$$

2.5.4.1 Sending Rate Update:

$$z_j^r(t+1) = z_j^r(t) + \beta_z z_j^r(t) \left(\frac{\varphi U_i}{\varphi z_j^r}(x_r(t)) - \sum_l H_{lj}^r s_l(t) \right) \tag{2.17}$$

2.5.4.2 Feedback Price Update:

$$s_l(t+1) = \omega p_l \left(\sum_r \sum_j H_{lj}^r z_j^r(t) \right) \tag{2.18}$$

TABLE 2.1: The comparison table for 4 decomposition algorithms

Method	Characteristics	Number of Variables
Partial-Dual	Efficient Capacity	1
Primal-Dual	Efficient Capacity	3
Full-Dual	Efficient Capacity and let to packet loss	2
Primal-Driven	Upgrading Feedback Price	1

All the algorithms in table 2.1 converges slower for smaller ω due to that when you force the network to bottleneck solutions, it becomes easier to overshoot and go over the capacity, forcing you to select a smaller step-sizes to make the algorithm to converge at all, leading to longer convergence times.

Chapter 3

Related Works

3.1 TRUMP Algorithm

TRUMP can be thought of as the integration of the best parts of the four decomposition algorithms described in [4][9]: primal-dual, full-dual, partial-dual, and direct path rate update, a.k.a. primal driven in order to tackle the poor convergence problem in DUMP. TRUMP converges faster than all 4 decomposition methods by utilizing only one tuning parameter. For a given flow between a source and a user, TRUMP calculates the feedback price of each link and based on that it allocates the sending rate for each user. However, TRUMP does not look at the hop-counts between the source and a given user. As a result, a path with a lower hop-count has a smaller feedback price than a path with a higher hop-count. Accordingly, TRUMP allocates less bandwidth to the longer paths; i.e., it penalizes the longer paths when they compete with shorter paths. For example, can be seen in Fig. 1, assuming that the total bandwidth in the shared link between nodes 10 and 11 is 100 Mbps,TRUMP gives 13 Mbps for path 1 , 23 Mbps for path 2 and 23 Mbps for path 3, because path 1 has higher hop-counts than path 2 and path 3.

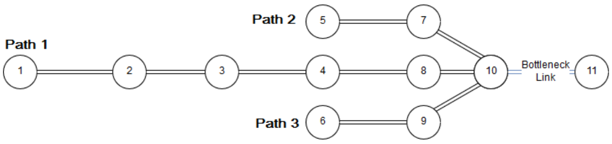

Figure 3.1: Sharing a bottleneck link by 3-flows in TRUMP

This can be unfair in many cases, particularly for multimedia flows that have a minimum bandwidth requirement to achieve a given quality of experience, irrespective of the number of hop counts. For example, if two users are watching the same Netflix program, and assuming that both users are identical in terms of Netflix and ISP subscriptions, then both of them should receive the same video quality even if one of them has more hop counts to the source. TRUMP, on the hand, penalizes the user with more hop counts by assigning less bandwidth to it.

3.1.1 Feedback price update:

$$s_l(t+1) = p_l(t+1) + q_l(t+1) \tag{3.1}$$

$$p_l(t+1) = [p_l(t) - \beta_p(c_l - \sum_r \sum_j H^r_{lj} z^r_j(t))]^+ \tag{3.2}$$

$$q_l(t+1) = \omega f^{'}(\sum_r \sum_j H^r_{lj} z^r_j(t)/c_l) \tag{3.3}$$

In 3.2 and 3.3, p_l is the loss price and q_l is the queuing delay where both of them are updated in every iteration. ω is a tuning parameter, f is the penalty function and c_l is the link capacity. Finally, s_l is the feedback price.

3.1.2 Sending rate Upgrade:

$$z^r_j(t+1) = maximize_{z_{jr}} U_r(\sum_j z^r_j) - \sum_l s_l(t) \sum_j H^r_{lj} z^r_j \tag{3.4}$$

In sending rate adaptation formula, the price for each path, is calculated by using $\sum_l s_l(t) \sum_j H^r_{lj} z^r_j$. So, the important factor in sending rate adaptation is the feedback price which is updated in each iteration. Also, $H^r_{lj} z^r_j$ equals to the effective capacity which is specified by y_l.

3.1.3 Convergence properties of TRUMP

Dissimilar to the previous decomposition methods, TRUMP is an exploratory and does not have a close similarity to a recognized decomposition model. As a result, the convergence feature of the TRUMP algorithm is not proven by the optimization theory [9][10]. On the other hand, TRUMP can converge until the network is lightly loaded. About the step-size (β_p) ,selecting β_p is controversial until there are congested links, where in this case, there will be the pocket loss. Totally, TRUMP is simpler than the other algorithms by having only one tuning parameter.

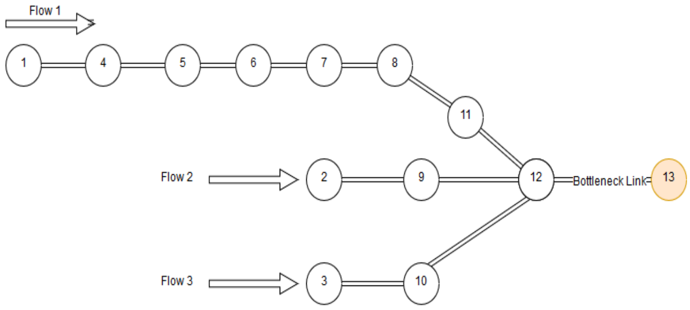

FIGURE 3.2: Share Topology

3.1.4 Fair bandwidth allocation:

Lets consider the fair bandwidth allocation in TRUMP by investigating a given network topology in figure 3.2. According to the figure, there are 3 flows which are competing over a bottleneck link which is specified in the figure. In that topology, the flows 2 and 3 get more bandwidth than flow 1. If we test it in an experiment with 100 mbps link capacity, the flow 1 gets 10 mbps, while flows 2 and 3 get 23 mbps of bandwidth. So, the network operator punish the long paths without considering how much the links are loaded.

Actually, TRUMP algorithm does not look at the hop-counts to adjust the sending rate. It calculates the feedback price of each link and based on it, allocates the sending rate for each source. As a result, a path with minimum hop-counts has less feedback price than a path with more hop-counts. So, the feedback price of long-paths is greater than short paths. Accordingly, TRUMP allocates less bandwidth to long paths and in another word, it penalizes long paths where they compete with the flows that are using short paths.

Lets make an example to have a better understanding of the problem. By considering that each link has the price of 10 and each link has the capacity of 100 mbps,in this case if the hop-count of that path is 8, the total feedback price will be 80. As a result, the allocated bandwidth will be 100-80=20. On the other hand the flows 2 and 3 will get 70 mbps bandwidth. (100-30=70). So, the feedback prices are playing the key role in bandwidth allocation. By using these examples, we can easily figure out that why TRUMP algorithm penalizes long paths.

3.1.5 Advantages of TRUMP algorithm

1. TRUMP is simple and fast algorithm.
2. It's good for big files.

3. It requires a few tunable parameters.

3.1.6 Drawbacks of TRUMP algorithm

1. The convergence of TRUMP algorithm is not proven until now.
2. It's not fair in diverse hop-counts.
3. It does not work well for small files.
4. The fairness of TRUMP is unknown for general ω values.

.. By taking all the aforementioned issues into the consideration, in chapter 4, we will present a new approach to tackle the fairness resource allocation problem, so that all sources get roughly same amount of bandwidth.

3.2 LBMP Algorithm

Logarithm-based multipath protocol(LBMP) [30], is proposed in 2012 to tackle some problems regarding TRUMP algorithm in terms of end-to-end delay and convergence. Although, decomposition methods such as TRUMP reduces the inconsistency, but their convergence and optimality features are not assured. By the way, there are rigid in differentiating among different links.

This algorithm uses logarithmic based approached in order to update the sending rate and also link prices. LBMP interprets the multipath utility maximization into a sub-sequence of unconstrained optimization problems, with unlimited logarithm barriers. Triggering barriers is much easier than selecting the old cost functions and, rather significantly, it enables optimal solution accessible.

The distributed solution for LBMP is based on the *gradient project approach*:

$$p_l(t+1) = \left[p_l(t) + \beta \left(\sum_{s,j:l\in(s,j)} z_j^s(t) - y_l(t) \right) \right]^+ ,$$

$$\mathbf{z}^s(t) = \arg\max_{\mathbf{z}^s} \left(U_s \left(\sum_j z_j^s \right) + \mu \sum_j \ln z_j^s - \sum_j z_j^s \left(\sum_{l:l\in(s,j)} p_l(t) \right) \right)$$

$$y_l(t) = \begin{cases} c_l - \frac{w_l}{p_l(t)}, & \text{if } p_l(t) \geq \frac{w_l}{c_l} \\ 0, & \text{otherwise,} \end{cases}$$

FIGURE 3.3: Gradient projection approach

$$z_j^s(t+T_j^s) = \left[z_j^s(t) + \gamma \left(\frac{1}{p_j^s(t)} - \frac{z_j^s(t)\sum_i z_i^s(t)}{z_j^s(t) + \mu \sum_i z_i^s(t)} \right) \right]^+$$

FIGURE 3.4: Sending rate update

3.2.1 Main differences between LBMP and TRUMP:

1. LBMP algorithm gives control parameters for every link but TRUMP has not this ability.
2. LBMP algorithm has dynamically behavior against the flows.
3. The convergence of LBMP is proved practically but for TRUMP it is not proved yet only under restrict conditions such as no link dynamics.
4. LBMP regulates the throughput and rates in a short time but TRUMP cannot do it.

Although there are some advantages of using LBMP algorithm over TRUMP, but, we in this dissertation, we concentrate on fair bandwidth allocation in diverse hop-counts.

Chapter 4

Hopcount-Based Fair Bandwidth Allocation (HFBA)

4.1 System Design:

In chapter 3, we discussed some algorithms that are implemented to control the congestion in the Networks. We realized that there is a big gap associated with TRUMP algorithm. The problem is fair bandwidth allocation in diverse hop counts, so that TRUMP algorithm is not fair in diverse hop-lengths. In TRUMP, the sending rate is calculated according to the feedback prices from the links in each path. So, without considering the hop-counts and by calculating the feedback prices from the links, it allocates the bandwidth for the competing flows. Also, without considering any delay and RTT and according to figure 4.1, path 1-11 gets the less amount of bandwidth than path 2 and path 1, that is because of path 3 has more hops than path 5-11 and 6-11. In another word, network operator penalizes longer hops and allocates less bandwidth to them. So, we believe that all the sources should benefit the bandwidth in a fair manner.

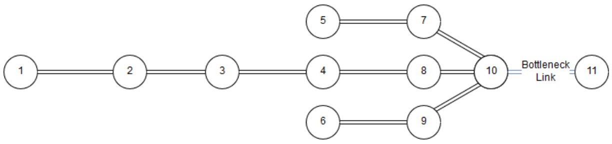

FIGURE 4.1: Hop-count diversity and bottleneck sharing

In order to implement HFBA, we need to use multiple decomposition methods that we discussed in chapter 2. The design of the system consists of three phases: topology construction, modeling the NUM problem and sending rate adaptation formula.

4.1.1 Topology Construction

A Network topology is created by using different links and nodes. The mathematical implementation of each topology is a graph which can be specified by G=(V,E), where V is defined as nodes and E is referred to the edges of the topology. In designing of our system, we specified a matrix to implement the topology. The mentioned matrix has two dimensions for designing the original model and a 3-D matrix has been used to design the new model.

In two dimensional implementation, the matrix is constructed by the links as a first dimension and sources as a second dimension. Also, the links are doubled due to there is sending and receiving operations. So, if we have a topology with 12 links, in this case, we set the number of links to 24. By considering the definition of the graph, the first dimension should present the number of edges. However, in our model designing, we specified the second element as r, which is the production of number of paths(p) and sources(s), to show all number of flows.

$$H=(l,r)$$

In another hand and in order to implement the new proposed model, we need to define a new variable and add it to the H matrix to create a 3-D matrix. The purpose of defining this variable which is specified by *h*, is to present the hop-length of each path.

$$H=(l,s,p)$$

4.1.2 Sending Rate Adaptation

The key part of this thesis is to find a way to optimize the sending rate adaptation formula proposed in 3.2.2 to achieve the fair bandwidth allocation.
By taking into account the formulations in 3.2, we proposed a new model of the sending rate adaptation formula by defining a new variable *h*:

$$h_j^r = \sum_l H_{lj}^r \tag{4.1}$$

By using the formulation in 4.2 and by transferring the formulation 3.1.2 in the network topology, the proposed sending rate adaptation can be written as follows:

$$z_j^r(t+1) = maximize_{z_j r} U_r(\sum_j z_j^r) - (\frac{\sum_l s_l(t) \sum_j H_{lj}^r z_j^r}{h_j^r})$$

(4.2)

In sending rate adaptation formula, source rates are updated in each iteration t. Also, by considering the variable h, the algorithm will experience relatively better fairness, higher link utilization and throughput compared to TRUMP. In formulation(4.3) $\sum_l s_l(t) \sum_j H_{lj}^r z_j^r$ is used to calculate the total path prices. So by dividing it by h variable, we can normalize the path prices, so that the paths prices will be in a same value. As a result, penalization of long paths will be relaxed.

4.1.3 Using the Sending Rate Adaptation in MATLAB

Simulating every mathematical formulation needs to do some changes. In our sending rate adaptation formula and in order to implement it in MATLAB, it is necessary to use the network topology transformation. So, the formula in (4.3) can be written as follows:

$$z_j^r(t+1) = z_j^r(t) - \frac{\left(\gamma \sum\limits_l z_j^r(t) + \frac{\gamma}{\sum\limits_l H_{lj}^r s_l(t)}\right)}{h_j^r} \tag{4.3}$$

In formulation 4.4, the tuning parameter γ is used to control the ratio of the convergence. The value of that is $0 < \gamma < 1$, where we have used $\gamma = 1$ in our simulations.

4.1.4 A simple numerical example to present the working of the model

In order to show the working of the system, HFBA is compared to TRUMP by using a sample example.

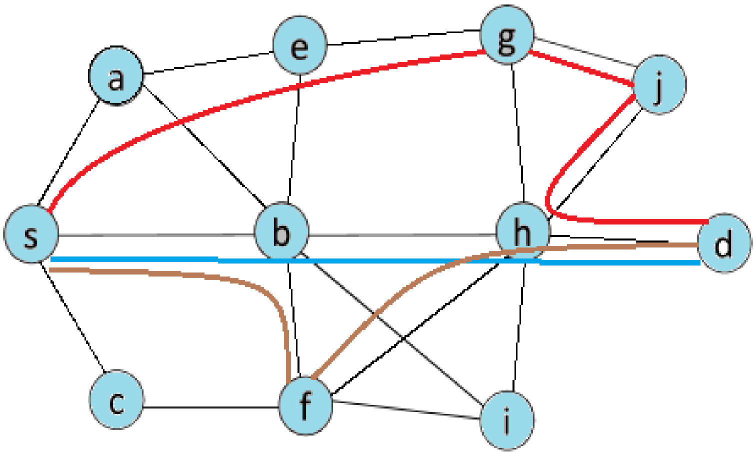

Figure 4.2: A sample topology to implement the working of the system

According to the Figure 4.1, we assume that there are three different flows that we want to send the packet from node s to node d as a destination. In order to show the working of the system in a simple way, we pick an initial 100 mbps sending rate due to that we are investigating it in first iteration. In a real simulation, the initial feedback price for each link is 0.001, but at this example, this value is set up to 1 for each link. The flow 1 is sending through the path 1 (s-a-e-g-j-h-d) where the hop-length of that path is 6. The flow 2, is sending from the path 2 (s-b-h-d) where the hop-length is 3 and finally, the flow 3 is sending from the path 3 (s-b-f-h-d) where the hop-length is 4.

So, by taking into account the formulations in (3.4) and (4.3), we can test the working of two models by using numerical examples.

Table 4.1: A comparison of working our proposed model and TRUMP

Flows	link Capacity	Hop Length	FB	EC	New Model	TRUMP
1	100	6	6	15	85	10
2	100	3	3	8	92	76
3	100	4	4	10	90	60

According to Table 4.1, FB implements the feedback price. Basically, this value for each link is approximately 0.001. However, for more simplicity, it is assumed that the value of FB is 1 for each link. So, by considering the number of hop-counts which reveal the number of the links, the value of FB for instance in the first flow will be 6. Also, EC represents thr sum of the efficient capacity. So, if there are more links, so the value of EC will be high.

By applying the values in Table 4.1 in formulations (3,4) and (4,3), it is clear that our proposed model is relatively fairer than TRUMP and also, we can enjoy high utility and throughput. The results shown in Table 4.1 also reveal that TRUMP allocates minimum bandwidth for the flow 1 with the hop-length of 6 and gives the high amount of bandwidth to the flow 2 with 3 hop-length. In another hand, our proposed model gives approximately the same amount of bandwidth to all flows without any discrimination in terms of hop-counts.

In continue and in chapter 5, we will have actual implementations of HFBA to show it's superiority compared to TRUMP.

4.2 Performance Evaluations

We tested HFBA in MATLAB as defined in phase 4. The path prices are up-to-dated with $\gamma = 0.1$. All the experiments are performed with $\omega = 1$, in which there may be no packet loss; and the iterations are set up to 100. Our simulations use both actual topologies: NSF, CORE, Abilene, NTT and COST. In the next section, we will analyze different setups for the topologies to measure the fairness feature in certain path selections.
We have not used RTT in our simulations, due to that TRUMP is not dependent to RTT values.

The purpose of fair bandwidth allocation is allocating the bandwidth for the competing flows where they share a bottleneck link. In terms of choosing the multiple paths, we use the specific pattern for sending the traffic from a source to a destination. Also, between each pair, we have minimum single-hop path or even more. As it is mentioned in 3.3.1, we have varied path length in the simulations.

In order to examine the fairness measure for TRUMP and HFBA, we need to simulate different topologies to show, how our model behaves fairer than TRUMP. The experiments are based on the impact of various number of sources on fairness index results.

Table 4.2: Characteristics of the topologies

Topology	**Links**	**Nodes**
NSF	42	14
CORE	44	15
Abilene	28	11
COST	11	52
NTT	144	55

4.2.1 Fairness Index

Jain's fairness index is one of the earliest measurements used to calculate the fairness index. This method is implemented by the function of the sending rate x_i and $f(x)$, where $0 \leq f(x) \leq 1$. So, if the value for f is closer to 1, in this case, the system is much fairer.

$$f(x) = \frac{[\sum_{i=1}^{n} x_i]}{\sum_{i=1}^{n} x_i^2} \tag{4.4}$$

In above formulation, x is the sending rate of source i.

4.3 Results:

We set up several simulations by using the mentioned topologies to show the fair behaviour of HFBA in different network conditions.

4.3.1 The simulation of Proposed model

By taking into account the NUM problem which we discussed in section 2.4 and all the formulations described in chapter 2; we simulated HFBA and TRUMP in MATLAB. The steps of simulating and designing the model are listed as follows:

1. Implementing the number of all links in the topology.
2. Presenting all number of sources.
3. Specifying the maximum capacity of each link.
4. Defining the new variable h in order to show the hop length of flows.

5. Initializing the flow rate, effective capacity for the link, loss price, delay price, and feedback price.
6. Calculating the price for every path and updating the flow rate.
7. Simulating the aggregate flow rate for the link and getting the updated loss price, delay price and feedback price.
8. Calculating the objective function by using updated sending rate, effective capacity and also the throughput associated with the updated sending rate adaptation.

4.3.2 NSF Topology

In order to evaluate our proposed model, we used NSF topology which has 21 links and 14 nodes.

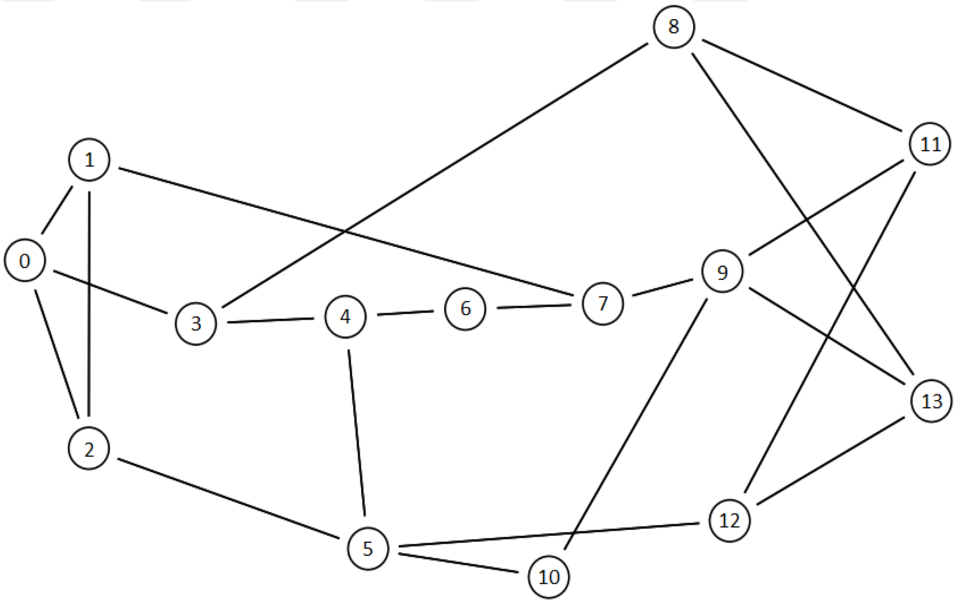

Figure 4.3: NSF Topology

4.3.2.1 Evaluations in terms of Sending rate and fairness measure

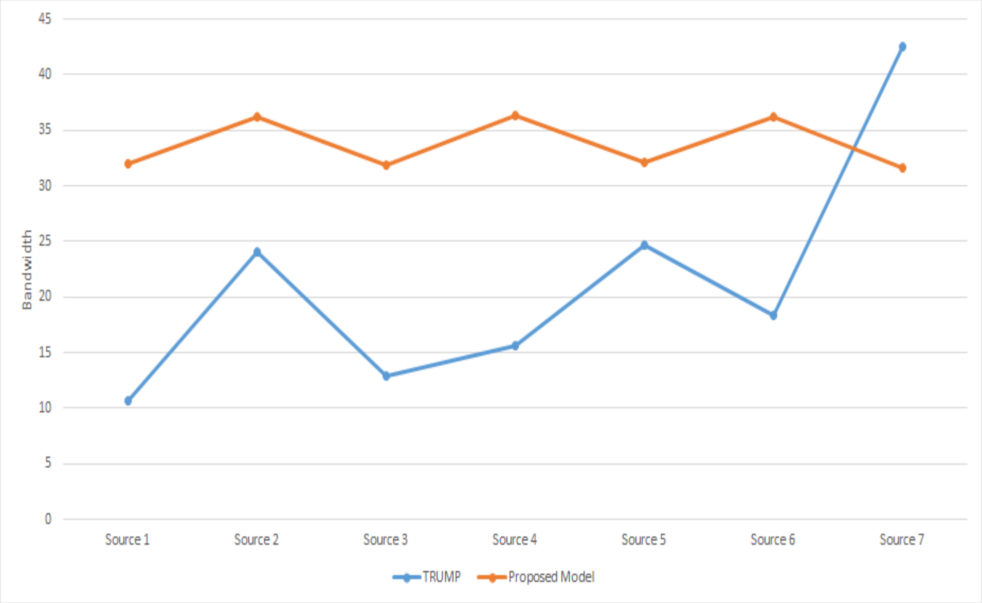

FIGURE 4.4: Sending rate allocation in TRUMP and HFBA

By considering Figure 4.6, for all seven sources, the bandwidth allocation in our proposed model is relatively fairer compared to TRUMP. In another word, the resource allocation in HFBA is much more consistent than TRUMP which has more fluctuations in bandwidth allocation. Also, it's clear that the overall utility and throughput in HFBA is considerably higher than TRUMP.

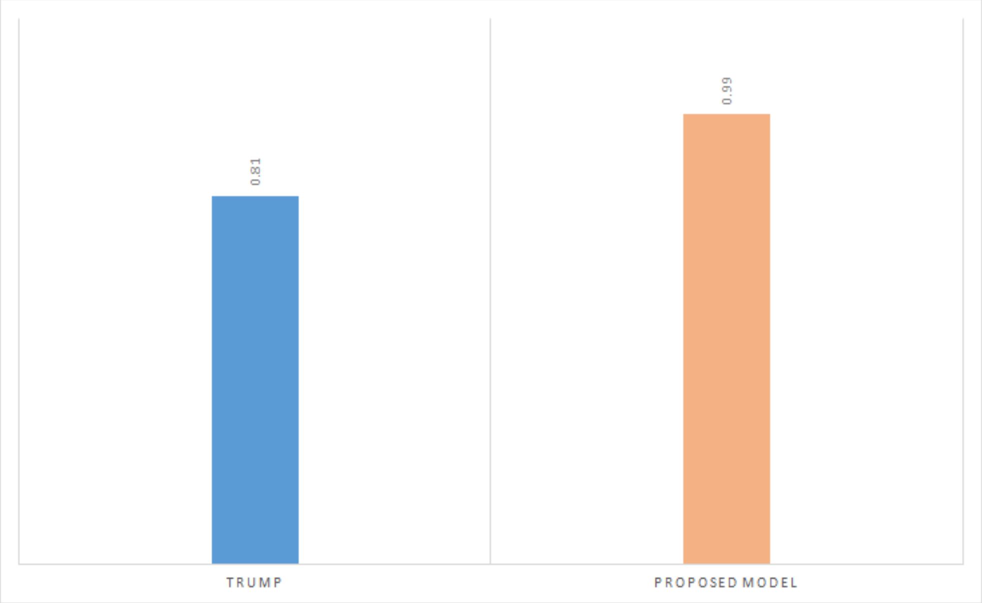

FIGURE 4.5: The value of fairness index for TRUMP and HFBA

The results shown in Figure 4.8 are based on the case study of two models according to the different hop-counts and number of competing flows for a specific topology. For TRUMP algorithm and in case 1, when there is only 2 competing sources, the path with the minimum number of hop-counts, gets the low bandwidth as it can be seen in case 1. In other cases, the paths with more hop-counts, get the more bandwidth than the cases in TRUMP.

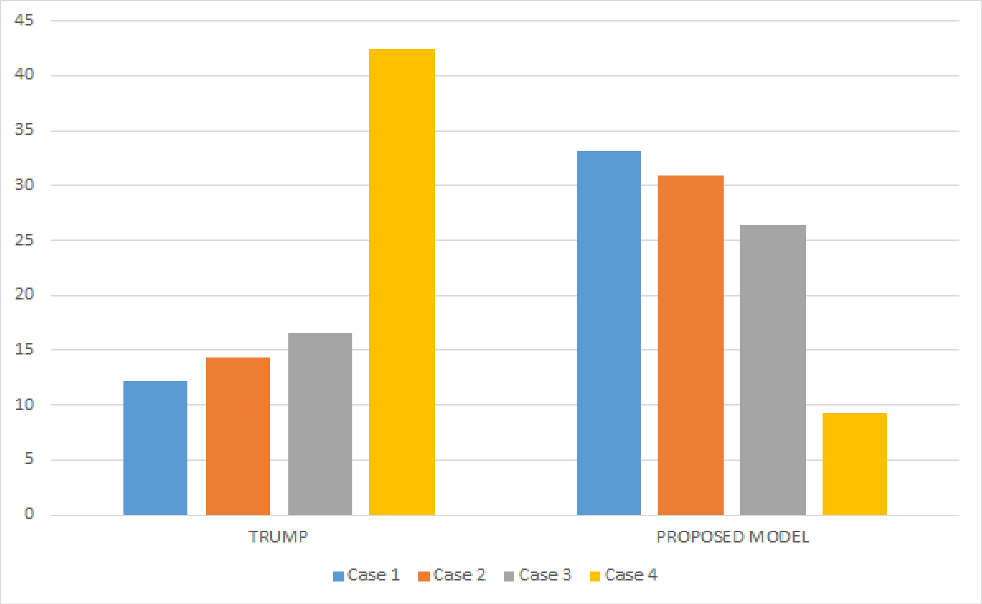

FIGURE 4.6: Sending rate in different hop-lengths for TRUMP and HFBA

4.3.3 Utility and Throughput

According to table 4.7, although our model is similar to TRUMP in terms of convergence feature, but in our model, user can enjoy considerably higher utility and throughput compared to TRUMP algorithm.

TABLE 4.3: Comparing of two models for three factors where w=1

	TRUMP	Proposed Model
Ratio	0.3446	0.9909
Utilty	8.3271	11.6708
Throughput	48.5475	147.0925

Table 4.4: Comparing of two models for three sources where w=1

	TRUMP	Proposed Model
Source 1	19.0726	47.9975
Source 2	14.0864	53.4456
Source 3	15.3885	45.6494

4.3.4 Evaluations on Abilene Topology:

Access-Core topology is one of the topologies which is used in [1] and it has 11 nodes and 28 links.

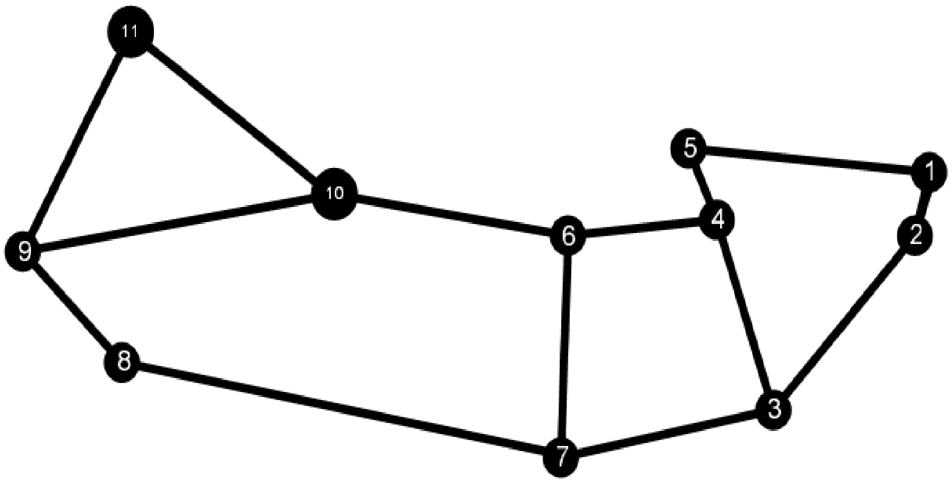

Figure 4.7: Abilene Topology

4.3.4.1 Sending rate allocation and fairness measure:

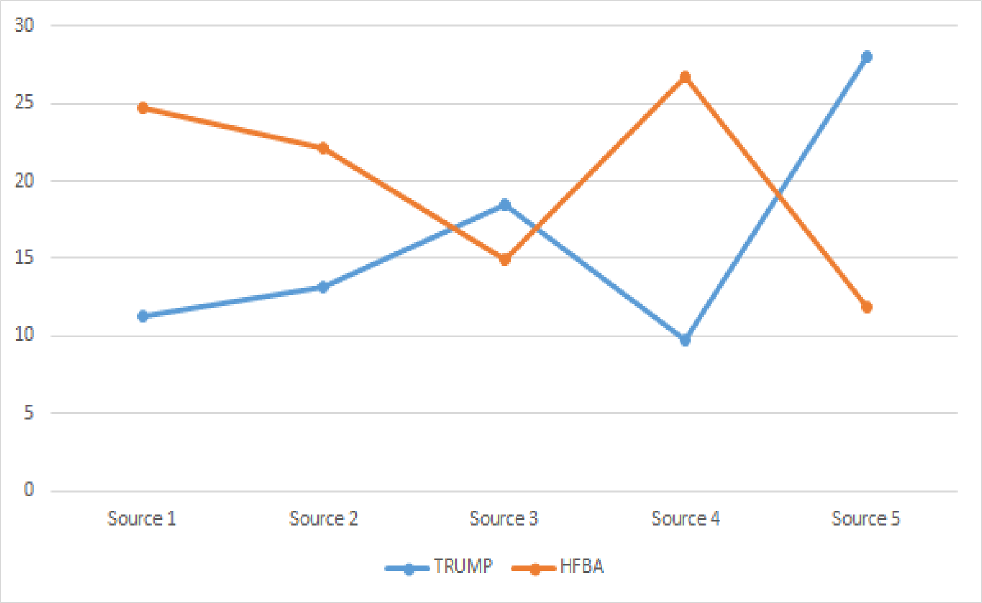

FIGURE 4.8: The allocation of bandwidth in TRUMP and HFBA

TABLE 4.5: Bandwidth allocation in TRUMP and HFBA

	TRUMP	HFBA
Source 1	11.2504	24.7234
Source 2	13.1765	22.1171
Source 3	18.4604	14.9646
Source 4	9.7393	26.659
Source 5	28.0535	11.9004

TABLE 4.6: Utility and throughput

	TRUMP	HFBA
Utility	13.5247	14.7695
Throughput	80.68	100.3646

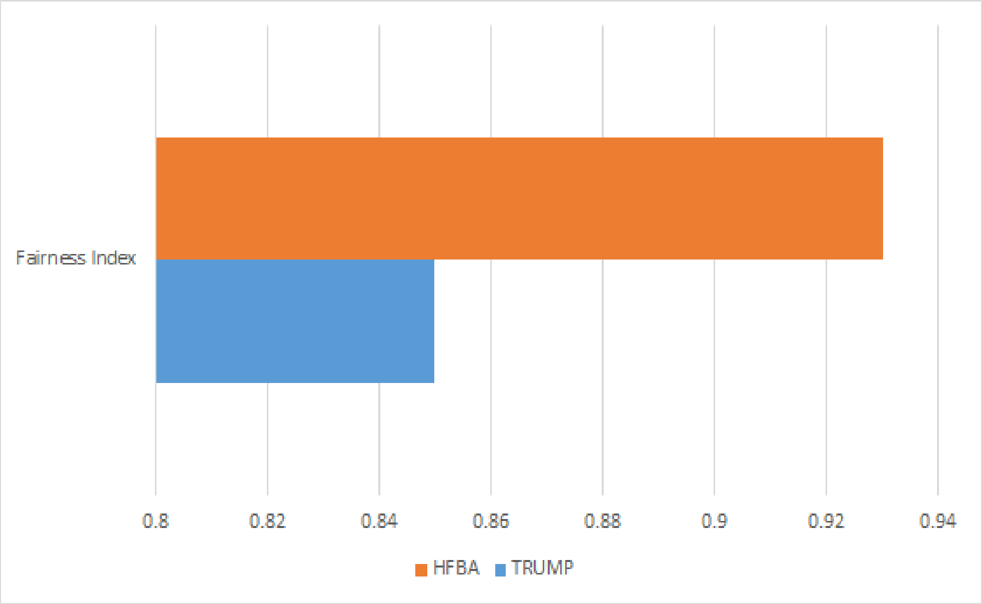

FIGURE 4.9: The values for fairness index in TRUMP and HFBA

Abilene topology is one of the topologies which is used in [10][30]. The fairness measure applied for this topology is more sensible. It is clear that the bandwidth allocation in our proposed model is relatively the same for all sources but, in TRUMP it is different for the flows and there are more fluctuations in bandwidth allocation. As a result, the fairness index for our model is higher than TRUMP.

4.3.5 Evaluations on Cost Topology:

Cost topology is one of the other real topologies for testing purposes and it has 11 nodes and 52 links.

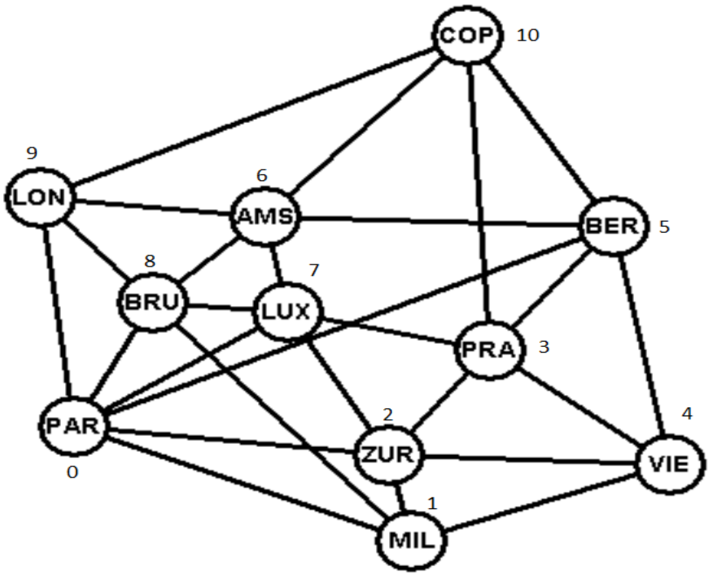

FIGURE 4.10: Cost Topology

TABLE 4.7: Bandwidth allocation in TRUMP and HFBA

	TRUMP	HFBA
Source 1	8.5361	32.1943
Source 2	14.9903	25.1637
Source 3	19.1559	21.8749
Source 4	10.5872	27.2182
Source 5	7.9968	36.7662
Source 6	12.8468	29.9951
Source 7	25.6007	13.6148

TABLE 4.8: Utility and Throughput

	TRUMP	HFBA
Utility	18.0387	22.7032
Throughput	99.7138	186.8272

4.3.5.1 Bandwidth allocation and fairness measure:

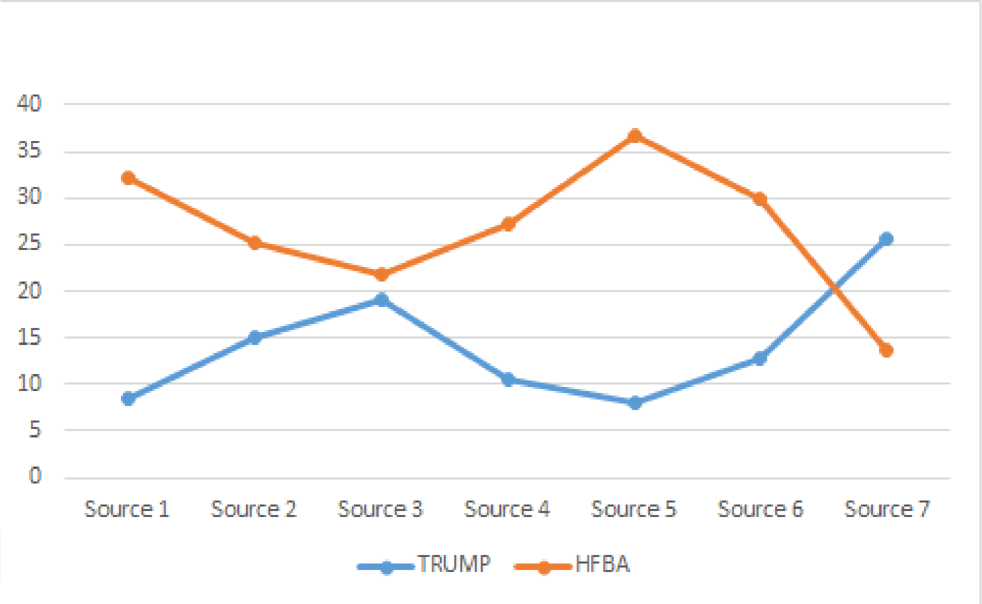

Figure 4.11: Bandwidth allocation in TRUMP and HFBA

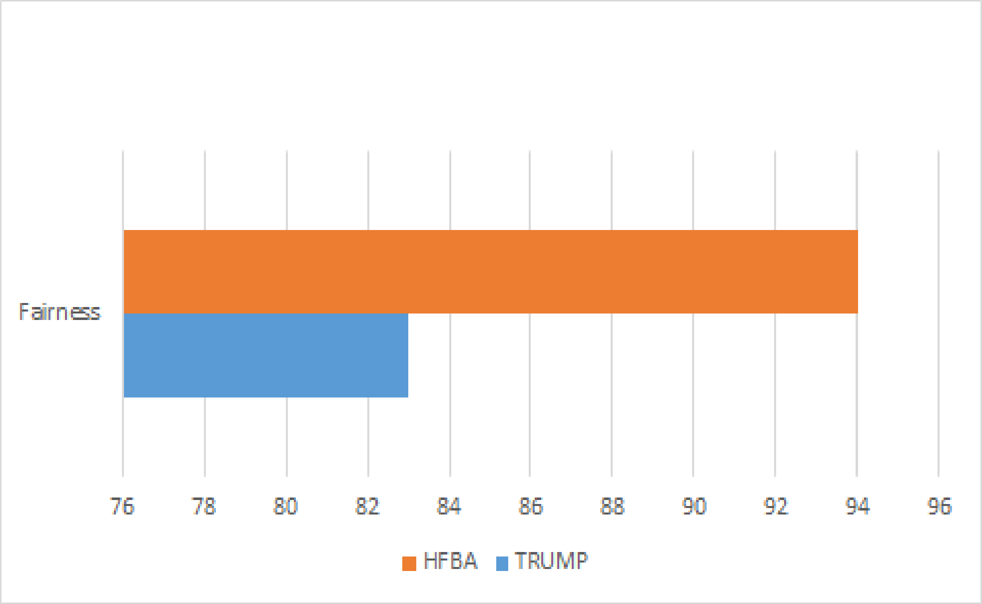

Figure 4.12: Fairness measures for TRUMP and HFBA

4.3.6 Evaluations on Core Topology:

First of all, we assume that we have 15 different paths in order to send the packets from node 1, 2 and 3 as sources to node 15 as a destination. In this case, we have 15 possible ways to send data. In all the experiments, we set the ω to 1.

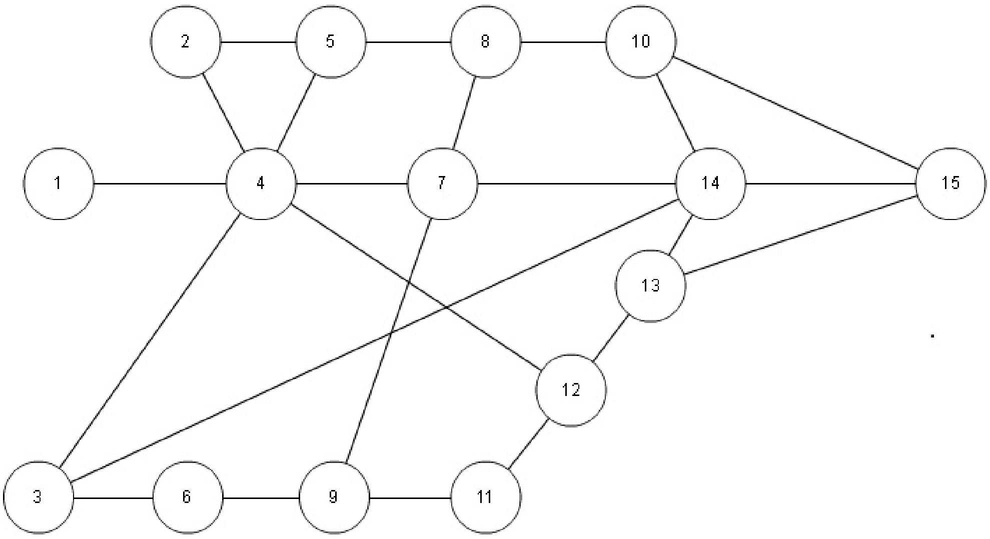

FIGURE 4.13: Core Topology

4.3.6.1 Bandwidth allocation and fairness measure:

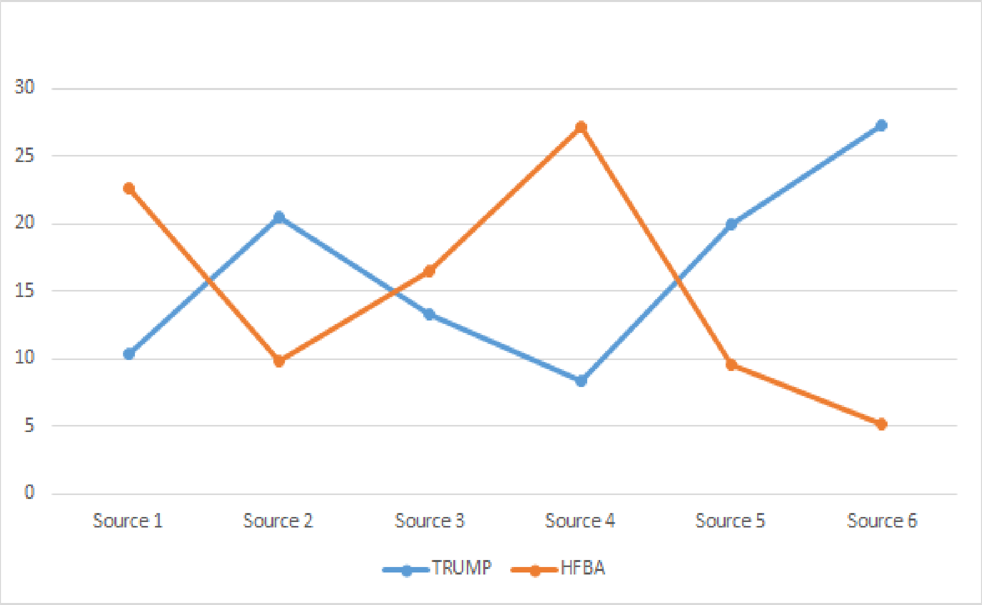

FIGURE 4.14: Bandwidth allocation in TRUMP and Proposed Model

TABLE 4.9: Bandwidth allocation in TRUMP and HFBA

	Source 1	Source 2	Source 3	Source 4	Source 5	Source 6
TRUMP	10.3786	20.4519	13.2487	8.3616	20.0362	27.3115
HFBA	22.6579	9.7607	16.4826	27.1723	9.6076	5.1589

Table 4.10: Fairnes index in different flow numbers

	2 Flows	3 Flows	4 Flows	5 Flows	6 Flows
TRUMP	0.78	0.86	0.84	0.85	0.86
HFBA	0.99	0.93	0.87	0.87	0.79

Table 4.11: Utility and throughput

1	TRUMP	HFBA
Utility	16.3702	15.4067
Throughput	16.3702	15.4067

According to the results and as we mentioned, HFBA performs better in some specific conditions. However, in some topologies like CORE, it does not work well compared to TRUMP. So, in next chapter, we propose a model to tackle this problem associated with HFBA.

4.3.7 Evaluations on NTT topology:

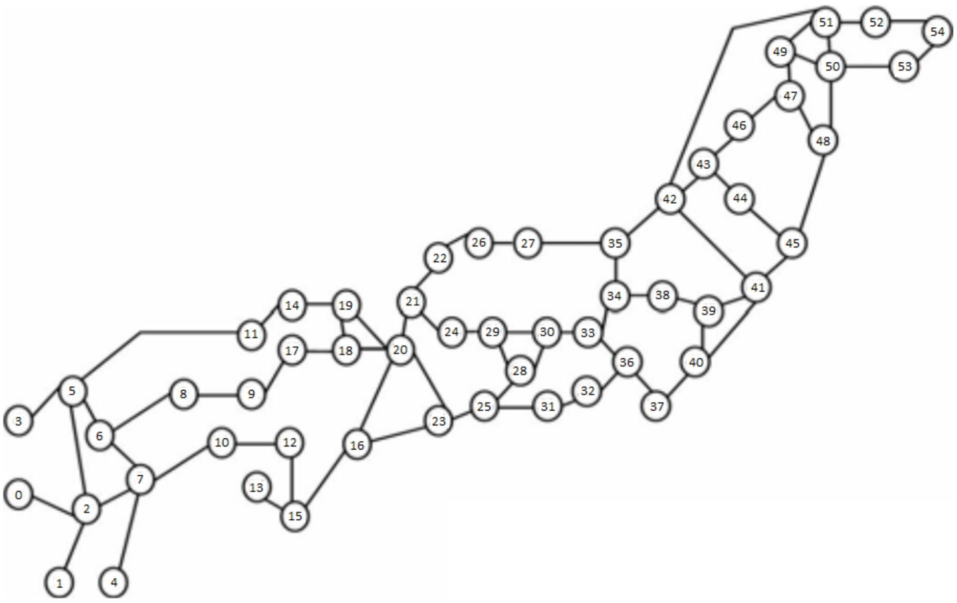

Figure 4.15: NTT Topology

Table 4.12: Bandwidth allocation in TRUMP and HFBA

1	Source 1	Source 2	Source 3	Source 4	Source 5	Source 6	Source 7
TRUMP	5.8526	5.183	3.8	2.8835	2.1401	28.6204	17.3063
HFBA	14.475	15.8438	21.1564	22.9689	27.6312	3.4809	7.4114

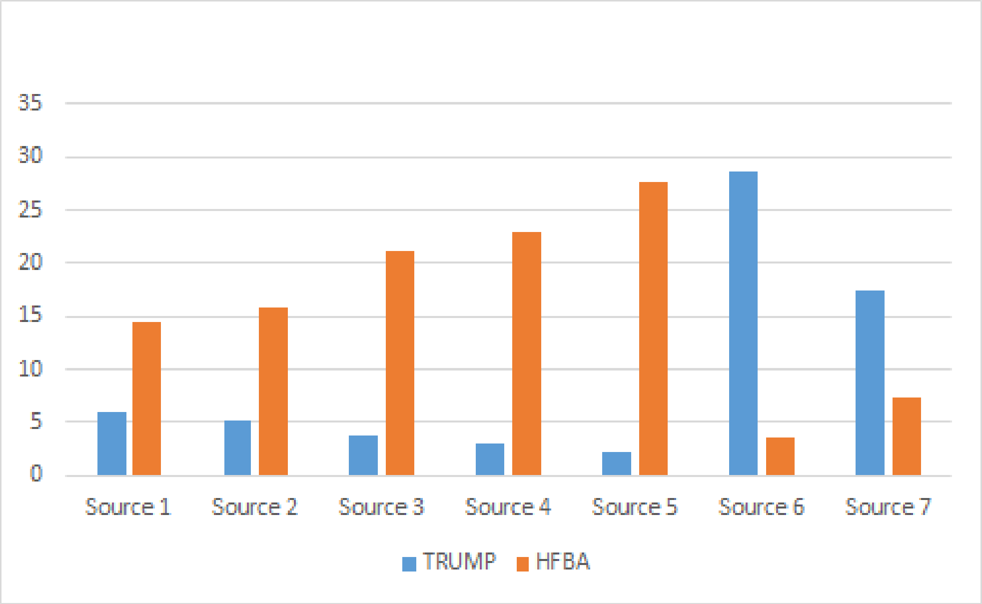

FIGURE 4.16: Sending rate allocation in TRUMP and HFBA

TABLE 4.13: Utility and throughput in TRUMP and HFBA

	TRUMP	HFBA
Utility	12.7723	18.1905
Throughput	65.7859	112.9676

The superiority of HFBA is clear in some conditions of path selections, especially when there are very long paths versus very short paths. Hence, in NTT topology HFBA performs dramatically better than TRUMP in terms of fair bandwidth allocation. In the next chapter, we will propose PBFA model that works better than TRUMP and HFBA and it performs better in all path selection and network conditions, and also provides higher utility and throughput.

Chapter 5

Priced-Based Fair Bandwidth Allocation (PBFA)

In previous section we proposed a model to solve the fair bandwidth allocation problem in TRUMP algorithm. The conducted simulations showed that this model performs better than TRUMP in *some* network conditions.
In this section, PBFA method is proposed to enhance the working of HBFA which is not good in all possible path selections. PBFA works based on the feedback prices of the links. In continue, we will show that PBFA performs significantly better than HBFA and TRUMP.

5.1 System Design

5.2 Model Formulation

In order to implement PBFA, we need to apply optimization problem:

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & \sum_{i=1}^{n} U_i(x_i) \\ \text{subject to} \quad & x \leq c. \end{aligned} \tag{5.1}$$

In formulation(1), U refers to the utility function and x is the sending rate of source i. Considering that the performances of every user can be determined by some utility function. Making the purpose of network management as the aggregate utility maximization or *Network Utility Maximization(NUM)* subjects to the capacity constraints, where the aggregate utility is the sum of the particular user utilities. Resource allocation issues can be implemented as a compelled maximization of utility function. User utility $U_i x_i$ can be considered as a kind of satisfaction between the sender and receiver, where U is a concave, non-negative, increasing and twice-differentiable function, which be able to symbolize the flexibility of the traffic or control bandwidth allocation in a fairness manner.

The formulation(1) is concave and it's suitable for single path routing. By considering multipath routing, we should transfer to a convex optimization problem:

$$\begin{aligned} \underset{z}{\text{maximize}} \quad & \sum_{i=1}^{n} U_i(z_j^i) \\ \text{subject to} \quad & \Sigma_r \Sigma_j M_{lj}^r z_j^r <= c_l. \end{aligned} \tag{5.2}$$

H represents a routing matrix that consists of links and nodes. The hop-length of each path is specified by using the formulation in (4.1).

There are three kinds of link prices: *delay price, loss price and feedback price* [10][25]. The feedback price is the summation of the delay and loss prices. In a network infrastructure, the prices of each link are measured by edge routers to calculate the sending rate. Link prices and sending rate adaptation formulas are normally calculated by using the best parts of the four decomposition methods, as done in TRUMP and LBMP, for example.

Accordingly, in order to have an efficient and fair bandwidth allocation, it is necessary to get more effective prices from each link. In our design, there is an initial link price which is updated in a number of iterations. Hence, in each iteration, every link needs a specific amount of bandwidth to send the traffic through it. In TRUMP, the flows that use the paths with more hop-counts, less bandwidth even though they may be lightly loaded. By taking this into account, we suggest an intuitive investment model to calculate the feedback price from each link by considering the differences between updated and initial link prices. The idea behind our model is that if you consume less, you will get more in the next iterations. We use this idea in our model to optimize the feedback price:

$$p_l(t+1) = p_l(t) + \beta_p(y_l - c_l) \tag{5.3}$$

$$P_{Inv} = p_l(t+1) - p_l(t) \tag{5.4}$$

$$q_l(t+1) = (\omega/c_l)(e^{y_l}/c_l)) \tag{5.5}$$

$$Q_{Inv} = q_l(t+1) - q_l(t) \tag{5.6}$$

$$F_l(t) = P_{Inv} + Q_{Inv} \tag{5.7}$$

In above formulations, q is delay price, p is loss price, P_{Inv} is the investment model for loss price, Q_{Inv} is Investment model for delay price, y_l is effective capacity and F_l is the feedback price. β_p is a step-size and small positive value and ω is a kind of tuning parameter to ensure the balance between cost and utility functions.

Hence, by taking into account the formulations in (3) and (8), the sending rate adaptation formula can be written as follows:

$$z_j^i(t+1) = maximize_{z_{ji}} U_i(\sum_j z_j^i) - (\frac{\sum_l F_l(t) \sum_j M_{lj}^i z_j^i}{h_j^i}) \tag{5.8}$$

5.3 Performance Evaluation

In order to measure the performance of our PBFA method, we created our scenarios on different realistic network topologies. We also applied all the experiments in MATLAB and the fairness measure is calculated by using *jain's fairness index.* The path prices are up-to-date with $\gamma = 0.1$. Most of the experiments are performed with $\omega = 1$, in which there may be no packet loss. The link capacities in the experiment are set up to 100 Mbps, 300 and 600 Mbps.

5.4 Results

In this section, we do some experiments by using various realistic topologies to show the working of PBFA compared to TRUMP and HFBA. The experiments are based on the impact of various number of sources with different hop-counts on fairness index. The competing flows, share the bottleneck link to reach the destination. So, this is important factor to figure out the fair bandwidth allocation in a strict network condition.

5.4.1 Evaluations for NTT

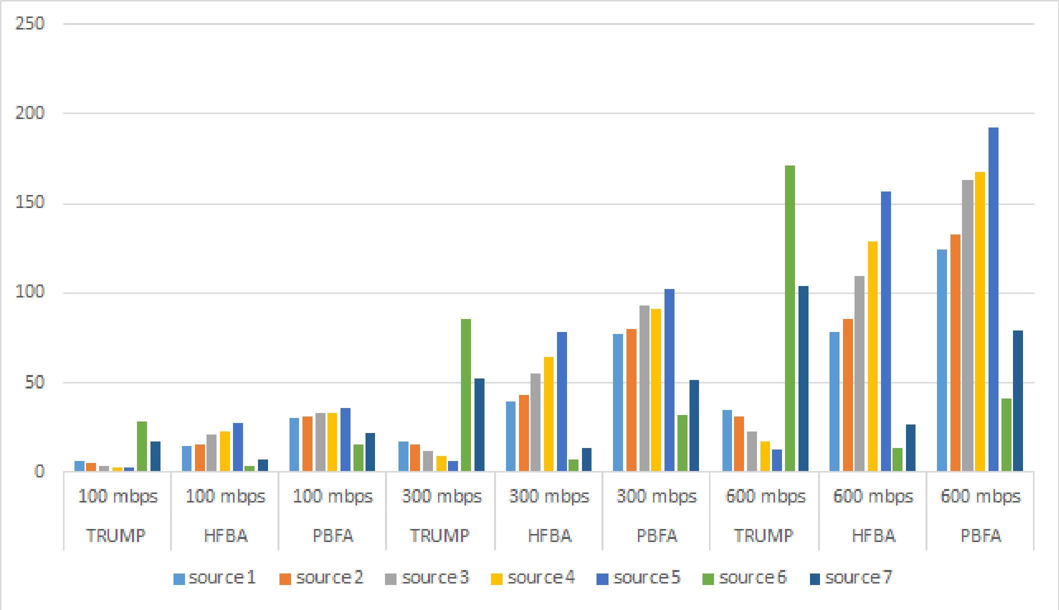

Figure 5.1: Bandwidth allocation for 7 competing sources in different link capacities in NTT topology

Table 5.1: Bandwidth allocation in NTT for three models

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
	100 mbps	100 mbps	100 mbps	300 mbps	300 mbps	300 mbps	600 mbps	600 mbps	600 mbps
source 1	5.8526	14.475	30.321	17.5577	39.0684	76.9561	35.1153	77.8968	124.3559
source 2	5.183	15.8438	30.7767	15.5488	43.0067	80.0126	31.0976	85.7649	132.3344
source 3	3.8	21.1564	33.2055	11.4	54.9045	92.7618	22.7999	109.5462	163.3398
source 4	2.8835	22.9689	32.8	8.6503	64.5807	91.0196	17.3006	128.9114	167.8552
source 5	2.1401	27.6312	35.6054	6.4201	78.5303	101.7143	12.8403	156.8149	192.9063
source 6	28.6204	3.4809	15.2743	85.8614	6.8943	32.2515	171.7229	13.727	41.2865
source 7	17.3063	7.4114	22.0619	51.9192	13.2428	51.6846	103.8385	26.3752	78.8415

Table 5.2: Utility and throughput in different link capacities

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Utility	12.7723	18.1905	23.2243	20.4625	24.4778	29.8074	25.3146	29.3093	33.2776
Throughput	65.7858	112.9676	200.0449	197.3575	300.2276	526.4004	394.7151	599.0364	900.9196

5.4.2 Evaluations for Abilene

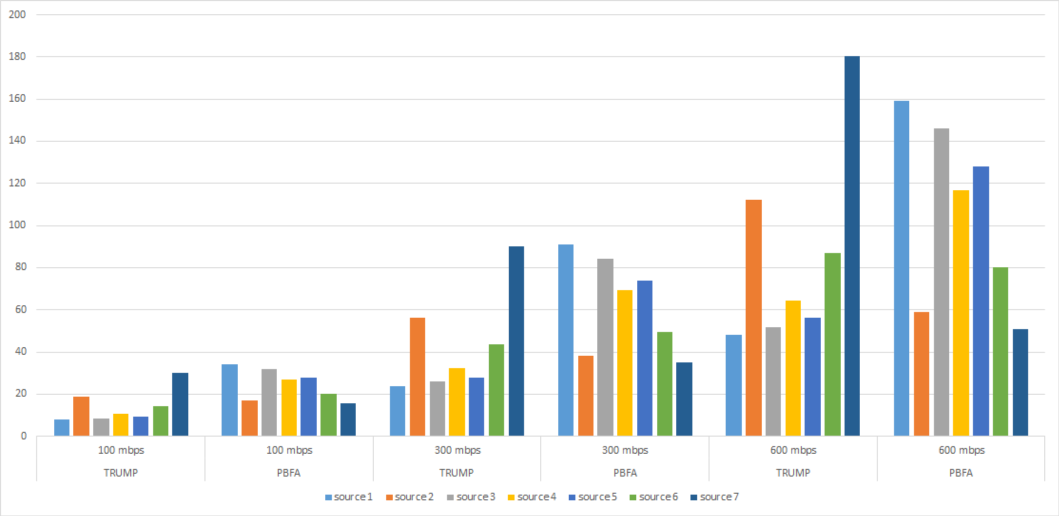

Figure 5.2: Bandwidth allocation for 7 competing sources in different link capacities in Abilene topology

Table 5.3: Bandwidth allocation in different capacities

	TRUMP 100 mbps	HFBA 100 mbps	PBFA 100 mbps	TRUMP 300 mbps	HFBA 300 mbps	PBFA 300 mbps	TRUMP 600 mbps	HFBA 600 mbps	PBFA 600 mbps
source 1	7.9912	21.1584	34.3229	23.9955	67.7851	90.9958	47.9917	135.8974	158.9275
source 2	18.6746	6.8781	16.8673	56.1389	22.8668	38.3075	112.2796	45.886	59.1079
source 3	8.6538	19.2021	31.7838	25.9864	61.7407	84.3054	51.9735	123.7925	146.0073
source 4	10.714	14.8525	27.0509	32.18	48.2212	69.3948	64.3609	96.7096	116.8691
source 5	9.34	16.8264	27.8628	28.0484	54.0872	73.8265	56.0974	108.4492	127.9979
source 6	14.5009	9.7226	20.2164	43.5717	31.9546	49.5285	87.1446	64.1055	80.0205
source 7	29.9272	4.8131	15.585	90.0734	12.7217	34.9106	180.1516	25.4903	50.7874

Table 5.4: Utility and throughput

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Utility	17.8423	17.3023	22.1980	25.5436	25.3430	28.5878	30.3957	30.2135	32.0545
Throughput	99.8017	93.4531	173.6890	299.9944	299.3772	441.2692	599.9994	600.3304	739.7176

5.4.3 Evaluations for NSF

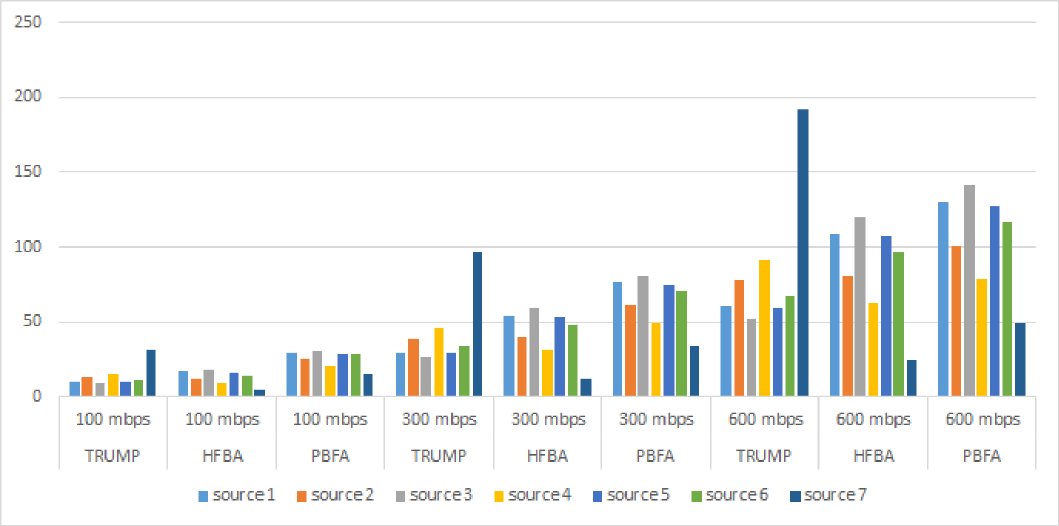

FIGURE 5.3: Sendin rate allocation in three models

TABLE 5.5: Bandwidth allocation in different linc capacities

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 mbps	100 mbps	100 mbps	300 mbps	300 mbps	300 mbps	600 mbps	600 mbps	600 mbps
source 1	10.0183	16.7732	30.1133	30.0817	54.3095	77.1649	60.1643	108.9784	130.7282
source 2	12.9399	12.238	25.1626	38.8634	40.1638	61.6983	77.7283	80.6275	100.0933
source 3	8.7713	18.6263	30.8925	26.3338	59.8321	81.4964	52.6684	120.0324	141.3097
source 4	15.1753	9.5078	20.5338	45.5857	31.3239	49.3939	91.173	62.89	78.962
source 5	9.8235	16.5697	28.7718	29.4957	53.4685	74.6706	58.9922	107.2813	127.7376
source 6	11.1987	14.669	28.1825	33.6289	47.86	70.922	67.259	96.0596	117.4101
source 7	31.9091	4.6984	14.8283	96.0043	12.2097	33.5821	192.0159	24.4798	49.2158

TABLE 5.6: Utility and Throughput

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Utility	17.9193	18.5415	22.4775	25.6184	25.7357	28.8572	30.4706	30.4484	32.3110
Throughput	99.8360	123.0824	178.4848	299.9936	306.9314	448.9282	600.0011	605.4579	745.4566

5.4.4 Evaluations for COST

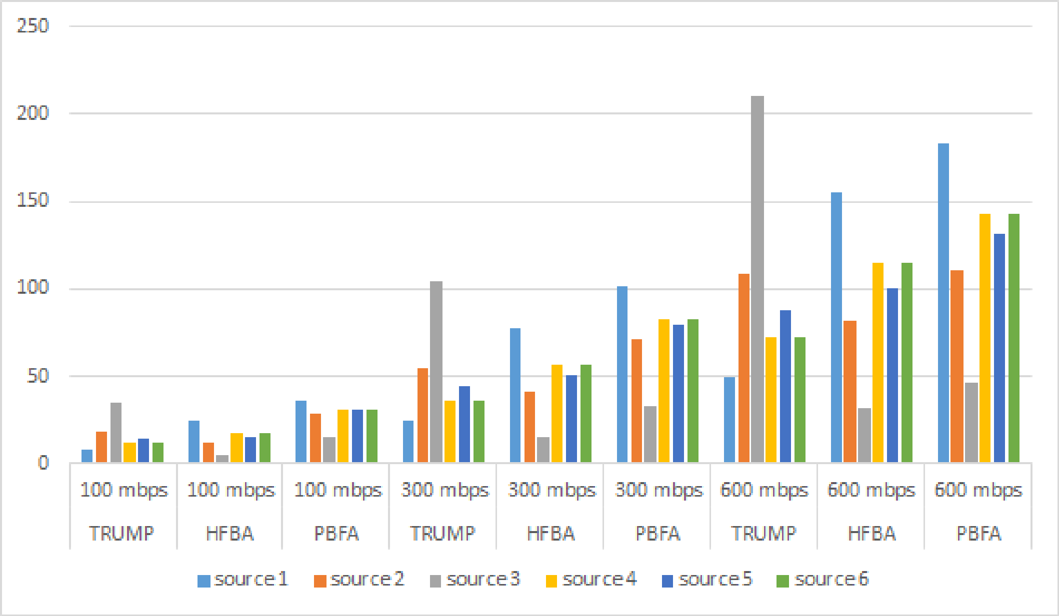

Figure 5.4: Bandwidth allocation in 3 models in different link capacities

Table 5.7: Bandwidth allocation in 3 models in different link capacities

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 mbps	100 mbps	100 mbps	300 mbps	300 mbps	300 mbps	600 mbps	600 mbps	600 mbps
source 1	8.2802	24.4201	35.9856	24.8438	77.5181	101.0863	49.6876	155.493	182.8099
source 2	18.0916	12.3355	29.0868	54.2966	40.8247	71.1419	108.5932	81.9857	111.15
source 3	34.9575	4.654	15.6092	104.95	15.7131	33.1773	209.8979	31.5754	46.4371
source 4	12.0244	17.6833	30.867	36.0822	57.0954	82.8698	72.1643	114.5859	142.3613
source 5	14.6187	15.3295	31.0919	43.8704	50.1572	79.9694	87.7408	100.6962	130.953
source 6	11.9828	17.732	30.9019	35.9571	57.2652	83.1102	71.9142	114.9283	142.8357

Table 5.8: Utility and throughput in different capacities

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Utility	16.2162	15.7234	19.9987	22.8099	22.8219	25.6016	26.9688	27.0033	28.5523
Throughput	99.9552	92.1543	173.5424	300.0000	298.5738	451.3549	599.9981	599.2644	756.5470

5.4.5 Evaluations for CORE

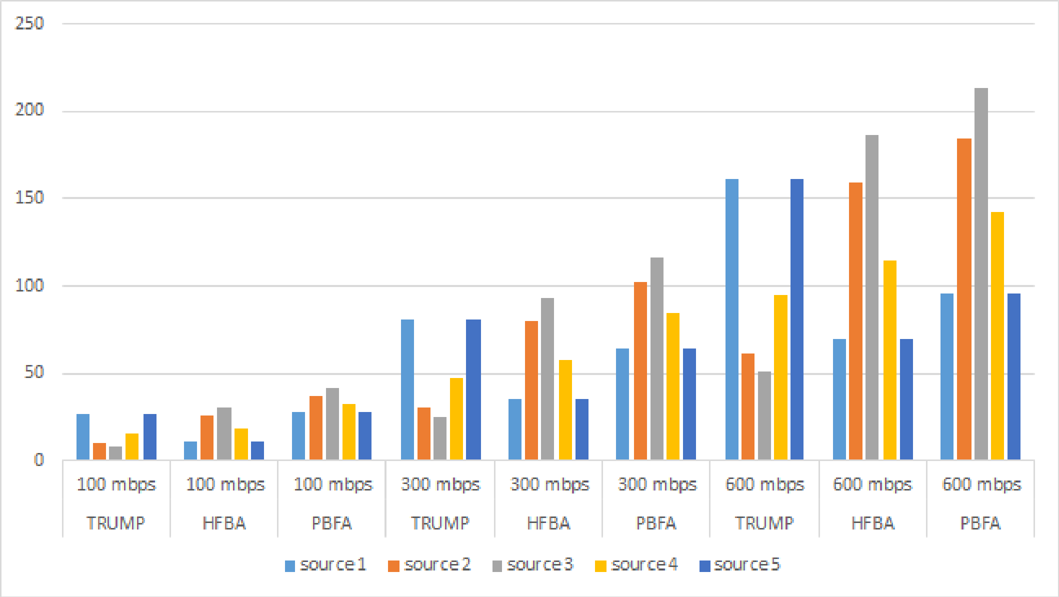

FIGURE 5.5: Bandwidth allocation in 3 models in different link capacities

TABLE 5.9: Bandwidth allocation in different link capacities

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 mbps	100 mbps	100 mbps	300 mbps	300 mbps	300 mbps	600 mbps	600 mbps	600 mbps
source 1	26.8519	10.9042	27.6634	80.5557	34.9716	63.8959	161.1115	70.0934	95.8663
source 2	10.1555	25.6437	36.7652	30.4664	79.5478	102.3931	60.9329	159.2751	184.8711
source 3	8.4771	30.1453	41.547	25.4315	93.1836	116.6829	50.863	186.5564	213.127
source 4	15.8787	18.1628	32.2296	47.6362	57.2168	84.5015	95.2724	114.6181	142.834
source 5	26.8519	10.9042	27.6634	80.5557	34.9716	63.8959	161.1115	70.0934	95.8663

TABLE 5.10: Utility and throughput in different link capacities for 3 models

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Utility	13.8010	14.3280	17.4445	19.2941	20.0668	22.1396	22.7598	23.5406	24.6691
Throughput	88.2150	95.7602	165.8686	264.6456	299.8913	431.3693	529.2912	600.6364	732.5647

5.4.6 Fairness Measure

TABLE 5.11: Fairness measure for NTT

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Fairness Index	0.51	0.80	0.94	0.51	0.75	0.92	0.51	0.75	0.88

Table 5.12: Fairness measure for NSF

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Fairness Index	0.78	0.89	0.96	0.78	0.88	0.94	0.78	0.88	0.92

Table 5.13: Fairness measure for Abilene

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Fairness Index	0.79	0.84	0.93	0.79	0.83	0.90	0.79	0.83	0.89

Table 5.14: Fairness measure for COST

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Fairness Index	0.78	0.86	0.95	0.78	0.87	0.93	0.78	0.86	0.91

Table 5.15: Fairness measure for CORE

	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA	TRUMP	HFBA	PBFA
Bandwidth	100 Mbps			300 Mbps			600 Mbps		
Fairness Index	0.83	0.86	0.96	0.83	0.86	0.94	0.82	0.85	0.91

According to the results and as an average, for NTT, there is approximately 76% improvement in fairness, for NSF, there is about 20% enhancement, for Abilene, there is 15%, for COST topology, there is approximately 19% improvement and for CORE topology, we have 18% enhancement in fairness compared to TRUMP. As a result, PBFA behaves fairer than TRUMP especially for large networks such as NTT.

In terms of utility and according to table 1, PBFA achieves approximately 91% improvement for NTT, 30% for NSF and 30% for Abilene, 12% for COST and 15% for CORE topologies compared to TRUMP. It is interesting to see that our PBFA is especially better in large networks such as NTT. This is important since network operators have to manage large networks. So, it is one of the big advantages of our model to differentiate the bandwidth in large and short distance paths.

In terms of the throughput shown as figure 4, there is a dramatic improvement for NTT of approximately 207%, as well as 79% and 64% improvements for NSF and Abilene topologies, respectively.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

The aim of this dissertation was to propose a model in order to address the fairness problem associated with TRUMP algorithm. The thesis began with discussing about different issues in traffic management. Optimization and using distributed algorithms are very important to implement traffic management methods to deal with multipath routing. So, these methods and algorithms took into the consideration by concerning the optimization decomposition algorithm. In continue, distributed algorithms and decomposition methods introduced as Partial-dual, Primal-dual, primal-driven and path rate update. Also, there is discussed that sending rate update and feedback price update are derived by using those four decomposition methods. In section 3, and in related works, we explored the TRUMP algorithm which is flexible and a simple method to control the congestion in Networks by using only one tuning parameter. However, it has a problem in terms of fair bandwidth allocation in diverse hop-counts. So, we considered this issue to propose a new model to tackle this issue in TRUMP algorithm. Also, we have looked briefly at LBMP algorithm to tackle the convergence problem in TRUMP algorithm. In chapter 4, we proposed HFBA to overcome the fair bandwidth allocation problem in TRUMP. The results from HFBA presented that this model works better than TRUMP in some network conditions and also certain path selections. In Chapter 5 and by taking into account the problems in our HFBA model which does not work in all network conditions, we implemented PBFA which optimizes the feedback prices from the links and accordingly we proposed a new sending rate adaptation formula so that the new model could work in all possible path selections that the previous model could not achieve that. According to the results in chapter 6, we showed that PBFA works significantly better than TRUMP and HFBA in terms of fair bandwidth allocation. In addition, by applying the new model, we can enjoy high utility and throughput.

6.2 Future Works

In this section we would like to implement the works which will be done later in order to improve the working of our proposed model and also other the improvements of TRUMP algorithm.

6.2.1 Fairness feature for all ω values

As it's mentioned in chapter 3 regarding TRUMP, there is an important challenge in TRUMP algorithm. The problem is that the fair bandwidth allocation in TRUMP is not specified and known for general ω values. Because of that we need to do all the experiments so that the values of ω is set up to 1. Doing the experiments in other values for ω give us different results which is not consistent. So, it requires to apply optimization methods to solve that problem which is my first priority to develop my work in this scope.

6.2.2 Cloud Computing

Cloud computing and it's related technologies are developing very fast. Accordingly and by considering the arising of the 5G networks and Mobile Edge Computing (MEC) technology, the requirements for increasing the QoE will be critical. Hence, we will take into account the optimization of resource allocation in MEC by mixing our proposed models and other optimization techniques to propose novel approaches to manage the resources in MEC and cloud environments.

6.2.3 Convergence of TRUMP

It is obvious that one of the main problem associated with TRUMP is its convergence feature, so that it's convergence is not proven yet. So, We expect to propose a new model to tackle this problem by using mathematical models such as optimization decomposition algorithms.

Bibliography

[1] A. Buzzi, S. Choi, W. Hanly, S. V. Lozano, A. Soong, A. C. K. Zhang, J. C. 2014. What Will 5G Be?. *IEEE Journal on Selected Areas in Communications*, 32 (6), 1065-1982.

[2] M. Hemmati, B. McCormick, and S. Shirmohammadi. QoE-Aware Bandwidth Allocation for Video Traffic Using Sigmoidal Programming. *IEEE Multimedia*, 2017.

[3] J.He and J. Rexford. Design for Optimizability: Traffic Management of a Future Internet. *Springer, Algorithms for Next Generation Networks, Part of the series Computer Communications and Networks*, pp 3-18, 20 January 2010.

[4] D. P. Palomar and M. Chiang. A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1439-1451, Aug 2006.

[5] M. Chiang, S. Low, R. Calderbank and J. Doyle. Layering as optimization decomposition. *Proceedings of the IEEE*, 95(1):255312, January 2007.

[6] X.Lin and N.B.Shroff. Utility Maximization for Communication Networks with Multi-path Routing. *IEEE Trans. Automatic Control*, vol. 51, May 2006.

[7] F. Paganini. Congestion Control with Adaptive Multipath Routing Based on Optimization. *40th Annual Conference on Information Sciences and Systems*, Princeton University,Princeton, NJ, USA, 2006.

[8] J. He and J. Rexford. Towards Internet-wide Multipath Routing. *IEEE Network Magazine*, 22(2):16â21, March 2008.

[9] J. He, M. Suchara, M. Bresler, J. Rexford, and M. Chiang. Rethinking Internet traffic management: From multiple decompositions to a practical protocol. *Proceedings of the 2007 ACM CoNEXT conference*, pp. 1-12,New York, New York â December 10 - 13, 2007.

[10] J. He, M. Suchara, M. Bresler, and J. Rexford. From Multiple Decompositions to TRUMP: Traffic Management Using Multipath Protocol. *Published in: Proceeding*

CoNEXT '07 Proceedings of the 2007 ACM CoNEXT conference Article No. 17. New York, New York â December 10 - 13, 2007.

[11] H. SHI, R. Prasad, E. Onur , and J. Zhang. Fairness in Wireless Networks - Issues, Measures and Challenges. *Published in: IEEE Communications Surveys Tutorials*, Volume: 16, Issue: 1, First Quarter 2014.

[12] J. He and J. Rexford. Towards Internet-wide Multipath Routing. *IEEE Network Magazine*, 22(2):16â21, March 2008.

[13] D.P. Wagner. Congestion Policing Queues - A new approach to managing bandwidth sharing at bottlenecks. *Institute of Communication Networks and Computer Engineering*, University of Stuttgart, Germany.

[14] J. He, M. Bresler, M. Chiang, and J. Rexford. Towards Robust Multi-layer Traffic Engineering: Optimization of Congestion Control and Routing. *IEEE J. On Selected Areas in Communications*, 25(5):868â880, June 2007.

[15] J. He, M. Chiang, and J. Rexford. Can Congestion Control and Traffic Engineering Be at Odds?. *In Proc. IEEE GLOBECOM*, November 2006.

[16] H. Han, S. Shakkottai, C. Hollot, R. Srikant, and D. Towsley. Multi-Path TCP: A Joint Congestion Control and Routing Scheme to Exploit Path Diversity on the Internet. *IEEE/ACM Trans. Networking*, 14(6):1260â1271, December 2006.

[17] J. He, M. Chiang, and J. Rexford. TCP/IP Interaction Based on Congestion Price: Stability and Optimality. *In Proc. International Conference on Communications*, June 2006.

[18] R. Gao, D. Blair, C. Dovrolis, M. Morrow, and E. Zegura. Interactions of Intelligent Route Control with TCP Congestion Control. *In Proc. of IFIP Networking*, May 2007.

[19] D. P. Palomar and M. Chiang. A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1439-1451, aug 2006.

[20] J. He, M. Chiang, and J. Rexford. Distributed Adaptive Traffic Engineering. *IEEE J. Poster session at INFOCOM*, 2005.

[21] R. Srikant. The Mathematics of Internet Congestion Control. *Birkhauser*, 2004.

[22] D. Katabi, M. Handley, and C. Rohrs. Congestion Control for High Bandwidth Delay Product Networks. *In Proc. ACM SIGCOMM*, August 2002.

[23] W. Xu and J. Rexford. MIRO. Multi-path Interdomain Routing. *In Proc. ACM SIGCOMM*, August 2006.

[24] S. Balon, F. Skive, and G. Leduc. How well do traffic engineering objective functions meet TE requirements?. *In Proc. IFIP Networking*, May 2006.

[25] F. P. Kelly, A. K. Maulloo, and D. K. H. 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..

[26] S. Shakkottai and N. Srikant. Network Optimization and Control. *the essense of knowledge.* DOI: 10.1561/1300000007.

[27] J.He and J.Rexford. Towards Internet-wide Multipath Routing. June 2007. *Princeton University Tech. Report*, TR-787-07www.cs.princeton.edu/research/techreps/TR-787-07.

[28] W. He, X. Liu and K. Nahrstedt. A feedback control scheme for resource allocation in wireless multihop Ad Hoc Networks. *Computer Science Department University of Illinois at Urbana-Champaign.*

[29] R. Teixeira, K. Marzullo, S. Savage and G. M. Voelker. Characterizing and measuring path diversity of Internet topologies. In Proc. ACM SIGMETRICS, June 2003.

[30] K. Xu, H. Liu, J. Liu and J. Zhang. LBMP: A Logarithm-Barrier-based Multipath Protocol for Internet Traffic Management.

[31] D. Vyas, R. Patel and A. Ganatra. Survey of Distributed Multipath Routing Protocols for Traffic Management. *Journal of Computer Applications*, (0975â8887) ,Volume 63âNo.17, February 2013