

# Bandwidth Allocation with Fairness in Multipath Networks

Hamed Hamzeh<sup>1\*</sup>, Mahdi Hemmati<sup>2</sup>, Shervin Shirmohammadi<sup>1,2</sup>

<sup>1</sup> Multimedia Systems Laboratory, Istanbul Sehir University, Istanbul, Turkey.

<sup>2</sup> Distributed and Collaborative Virtual Environment Research Laboratory (DISCOVER Lab), University of Ottawa, Ottawa, Canada.

\* Corresponding author. Email: hamedhamzeh@std.sehir.edu.tr

Manuscript submitted March 14, 2017; accepted June 30, 2017.

doi: 10.177606/ijcce.6.3.151-160

---

**Abstract:** Network resource management and traffic engineering are important subjects in today's Internet. In terms of traffic engineering, bandwidth allocation and splitting it in a fair manner among different users have become challenging. In addition, optimizing the utilization of network resources, increasing the user utility and throughput are also considerable. So, the user satisfaction with regard to the resource allocation and Quality of Service (QoS) are the most important factors that should be taken into the consideration. At the first step, Network Utility Maximization (NUM) problem has been considered as an initial stage to design any traffic engineering method. In this paper and by considering the mentioned issues, first of all we take into account the NUM problem and optimization decomposition methods by focusing on Traffic Management Using Multipath Protocol (TRUMP), and its weaknesses to tackle the fair resource allocation problem associated with it. We then propose a model to tackle the fair bandwidth allocation issue by implementing an optimized sending rate adaptation model using an intuitive investment method to optimize the link prices (delay and loss) to achieve an efficient fair bandwidth allocation model. The model is evaluated by using different simulations and different topologies under various network conditions. Our results show that the proposed model behaves fairer than TRUMP in certain path selections. As an average from the results and at a minimum point our model achieves 26% improvement in fairness in contrast to TRUMP. In addition, for large networks we can enjoy approximately 90% improvement in fairness measure.

**Key words:** Traffic engineering, network resource management, multi-path, TRUMP.

---

## 1. Introduction

Bandwidth allocation is one of the most significant issues in today's computer networks. The question of how allocate the bandwidth in a fair manner has become challenging. So, the fair distribution of bandwidth among the Internet users is an indispensable part of the improvement of QoS. In order to have a better traffic management, it's necessary to consider the important parts of a traffic management model [1]. These elements include users, network operator and routers. Actually, the network operator does the traffic engineering by using different algorithms. These algorithms are based on the optimization decomposition and distributed algorithms such as TRUMP [2] and Logarithmic-Based Multipath Protocol (LBMP) [3]. A benefit of using distributed algorithms is that they adjust the sending rates based on the round trip time (RTT) and can reply rapidly to traffic changes. The theory of optimization [4] 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 known as *Primal-Dual*, *Partial Dual*, *Primal Driven* and *Full Dual* [5].

In a multipath routing which is implemented in Fig. 1, the network operator adjusts the sending rate by using the mentioned algorithms by calculating path rates derived from the links and the routers [6], [7]. By collecting the link prices from each path, the best route will be chosen to send the traffic. So, a specific bandwidth will be allocated to each path. Accordingly, the question of how to allocate the bandwidth in a fair manner so that all the users benefit the desired amount of bandwidth has become 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. The important thing to be mentioned is that every model should be optimized at the first instead of optimizing an existing model [8].

Some of these strategies are investigated until now but there are many gaps. In a Network topology, there are many nodes and links between the source and the destination. Some of the packets have to be sent in a long distance to reach the destination. So, they need more bandwidth than other paths which go through from the paths with minimum distance and also their feedback price will be higher than the shorter paths. The feedback price is the summation of delay and loss prices from the links. Consequently, paths with less hop-counts will get the higher bandwidth than the paths with more hop-counts. This problem is associated with TRUMP. So, by taking into account the mentioned problems, in this paper we proposed a fair bandwidth allocation model by utilizing the multiple decomposition methods to present a rate adaptation formula to split the traffic in a fair manner to satisfy the user demands and the quality of service. Our model is inspired by a top-down algorithm designing. Our method optimizes the feedback price from the links.

All the implementations are done in MATLAB by comparing the results with TRUMP to see the superiority of our model. In order to compare the fairness measures in the proposed model and TRUMP, we used Jain's fairness index [9]. Finally, all the simulations and performance evaluations are performed by using actual network topologies. The results from the experiments show that as an average, the model achieves 26% improvement in fairness and especially 90% for big networks compared to TRUMP.

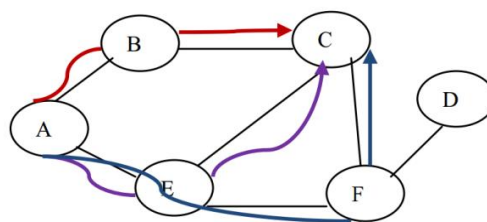


Fig. 1. Multipath routing.

## 2. Background and Related Works

Maximizing aggregate utility [2] is a basic approach to network traffic management as for  $U_i(x_i)$  where  $U$  is concave, increasing and twice-differentiable. So, the optimization problem can be written as follows:

$$\begin{aligned} & \text{maximize } \sum_i U_i(x_i) \\ & \text{subject to } Rx \leq c, x \geq 0 \end{aligned} \quad (1)$$

Optimization problem aims to maximize the user utility  $U$  regarding the sending rate  $(x)$  of source  $i$ , that

should not exceed the actual link capacity  $c$ , and  $R$  is the routing matrix. We assume, there is a network consists of different paths to send the traffic. In this case, we need a routing matrix to show the paths in the network as  $H_{lj}^i$ . According to this, the value of the routing matrix will be 1, if path  $j$  of the source  $i$  utilizes link  $l$ , otherwise it will be 0.  $H$  does not depict all the paths in the network; so, we should rewrite the maximization problem as:

$$\begin{aligned} \maximize & \sum_i U_i \left( \sum_j Z_j^i \right) \\ \text{subject to} & \sum_i \sum_j H_{lj}^i Z_j^i \leq c_l, \forall l \end{aligned} \quad (2)$$

If we assume the  $c$  parameter as the link capacity, then the transmission rate should not exceed the capacity  $c$  (It can be interpreted as the capacity constraint). To get the multipath routing, the  $z$  variable is introduced to show the sending rate of the source  $i$  in its  $j$ th path. Feedback price update and also path rate update which are implemented in [2] can be derived as:

- **Feedback price update at link  $l$ :**

$$s_l(t+1) = [s_l(t) - \beta_s(t)(c_l(t) - \sum_i \sum_j H_{lj}^i z_j^i(t))]^+ \quad (3)$$

- **Sending rate update at source  $i$  and path  $j$ :**

$$z_j^i(t+1) = \maximize_{z_j^i} (U_i(\sum_j z_j^i) - z_j^i \sum_l s_l(t) H_{lj}^i) \quad (4)$$

In which  $t$  implements the iteration number,  $s_l$  is the feedback price and  $\beta$  is the step size greater than 0.

## 2.1. The Novel Purpose of Today's Traffic Management

An improved traffic management objective can be done by combining the performance metrics and the network robustness as [2]:

$$\begin{aligned} \maximize & \sum_i U_i(x_i) - \omega \sum_l f(\sum_i R_{li} x_i / c_l) \\ \text{subject to} & R x \leq c, x \geq 0 \end{aligned} \quad (5)$$

This objective provides a solution that strikes a trade-off between high aggregate utility and a low overall network congestion, to satisfy the requirement for performance and robustness. Here  $\omega$  is a tuning parameter which adjusts the balance between the utility function and the cost function. The formulation in (5) can be converted to the convex optimization problem [2], [10]:

$$\begin{aligned} \maximize & \sum_i U_i(\sum_j z_j^i) - \omega \sum_l f(y_l / c_l) \\ \text{subject to} & y_l \leq c, y_l = \sum_i \sum_j H_{lj}^i z_j^i \end{aligned} \quad (6)$$

## 2.2. TRUMP Algorithm

TRUMP algorithm is a fast, simple and also easy to manage algorithm. It is designed by using the best parts of four decomposition algorithms: *Primal-Dual*, *Partial Dual*, *Primal Driven* and *Full Dual Feedback price update* and also the sending rate adaptation formula can be written as follows:

- **Feedback price update**

$$s_l(t+1) = p_l(t+1) + q_l(t+1) \quad (7)$$

- **Loss price update**

$$p_l(t+1) = [p_l(t) - \beta_p(c_l - \sum_i \sum_j H_{lj}^i z_j^i(t))]^+ \quad (8)$$

- **Delay price update**

$$q_l(t+1) = \omega f'(\sum_i \sum_j H_{lj}^i z_j^i(t) / c_l) \quad (9)$$

- **Sending rate update**

$$z_j^i(t+1) = \max_{z_j^i} \text{imize } U_i(\sum_j z_j^i) - \sum_l s_l(t) \sum_j H_{lj}^i z_j^i \quad (10)$$

TRUMP is an algorithm with fewer tuning parameters and it's based on  $p_l$  as the loss price and  $q_l$  as the queuing delay. In this algorithm, loss and delay prices [11] are calculated and then updated in every iteration. The weakness of this algorithm is its convergence, which is not proven yet.

The fairness feature of TRUMP is unknown for general  $\omega$  values and because of that most of the experiments for this algorithm are set to  $\omega=1$ . According to the Fig. 2, it's assumed that there are three sources competing over a bottleneck link specified in Fig. 2. In this case, paths 2 and 3 with three hop counts get more bandwidth than path 1 consists of six hop counts. This is the main problem of fair bandwidth allocation associated to the TRUMP algorithm where Network operator penalizes longer hops.

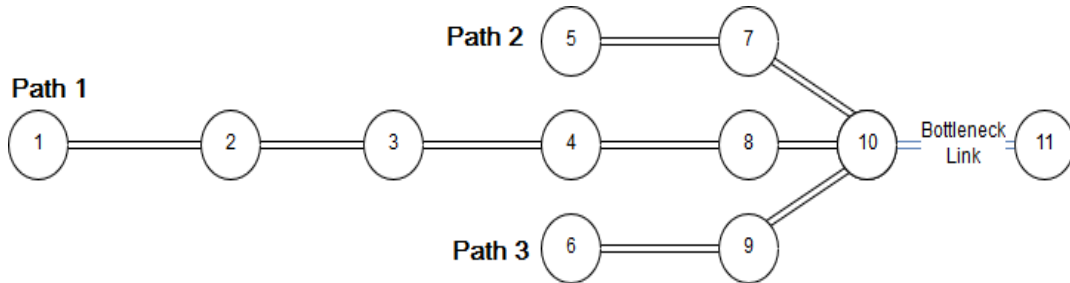


Fig. 2. Sharing a bottleneck link by 3-flows in TRUMP.

Actually, TRUMP algorithm does not consider 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 longer paths is greater than the shorter paths. Accordingly, TRUMP allocates less bandwidth to longer paths and in another word, it penalizes longer paths where they compete with the flows that are using shorter paths.

### 2.3. LBMP Algorithm

LBMP is designed to tackle the convergence problem associated with TRUMP algorithm by using a logarithmic function to deal with the congestion problem in the network, as same as the later algorithms that we discussed. This algorithm is designed by using the multiple decomposition methods. Compared to TRUMP, LBMP converges faster than TRUMP, and it provides better throughput and utility.

LBMP is originated by using a Barrier function method, which translates a constrained optimization problem into a sequence of simpler unconstrained optimization problems; it then constructs infinite barriers at the constraint bounds and ensures every optimization iteration strictly meet the respective constraints. LBMP uses different tuning parameters to tune the link weights to avoid the congestion problem in a network, where  $\mu$  is a small positive parameter.

The fairness feature of LBMP is similar to TRUMP, and it penalizes longer hops regarding the bandwidth allocation. Also, compared to TRUMP, LBMP is flexible in differentiating the control at different links, and its optimality and convergence are theoretically guaranteed.

### 3. Proposed Model

#### 3.1. Introduction

By taking into account the mentioned problems in TRUMP algorithm, we realized that there is a big gap associated with it. The problem is the fair bandwidth allocation in diverse hop counts; so that TRUMP algorithm is not fair in diverse hop-counts, and without considering any delay and RTT the paths with shorter hop lengths, get more bandwidth than the paths with longer hops. In another word, network operative punishes longer hops and assigns less bandwidth to them. So, this is not fair due to the end to end delay. So, it's necessary that all the sources get approximately the equal bandwidth or according to their hop-lengths.

#### 3.2. Model Definition

By considering the formulations in 2.2, we revised the sending rate adaptation formula to achieve fairness resource allocation by defining a new variable  $h$  in order to show the hop-length of each path:

$$h_j^i = \sum_l H_{lj}^i \quad (12)$$

In this case, the new formulation of sending rate can be shown as follow:

$$z_j^i(t+1) = \max_{z_j^i} U_i(\sum_j z_j^i) - \frac{\sum_l s_l(t) \sum_j H_{lj}^i z_j^i}{h} \quad (13)$$

In sending rate adaptation formula mentioned in (13), the path prices are divided equally among different paths in the network.

#### 3.3. Experimental Setup

We simulated TRUMP and the proposed model in MATLAB. The path prices are up to date with  $\gamma = 0.1$  in order to ensure the convergence. Most of the experiments are done by  $\omega = 1$ , in which there may be no packet loss. Our simulations use both unreal and practical topologies which are NSF, CORE, ACCESS-CORE, NTT and COST. For those topologies, we used the link capacities of 100 (Mbps).

##### 3.3.1 Topological implementation

The purpose of fair bandwidth allocation in this paper is allocating the traffic for competing flows that they use a bottleneck link [12]. We want to look at the conditions for different topologies shown in Fig. 3 and mentioned in Section 3.3. As it's mentioned, the specific path selection patterns are chosen.

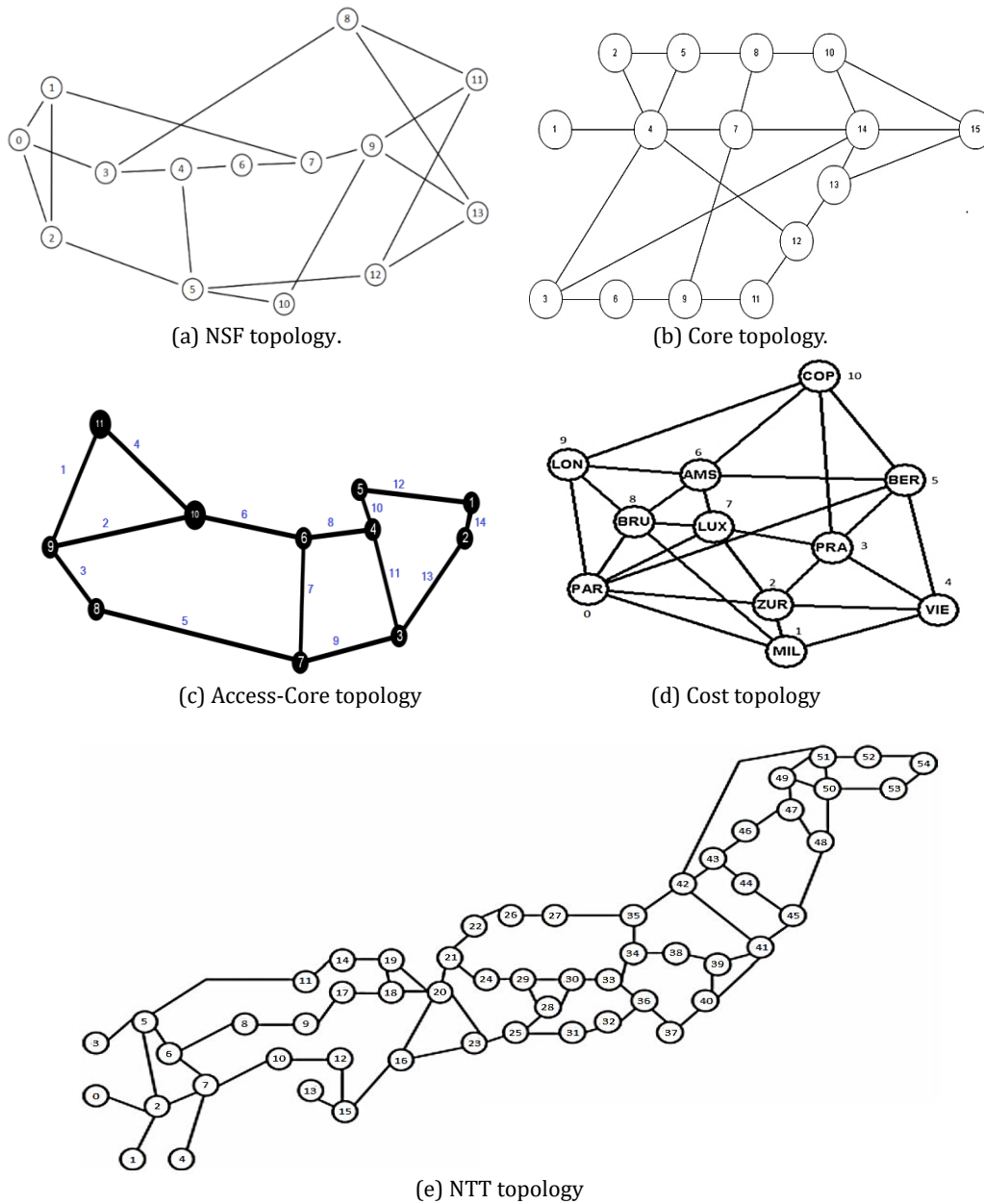


Fig. 3. Topologies which are used in simulations.

In all the topologies, the link capacities are set up to 100 mbps and also link delays are not considered directly into the simulation.

Table 1. Characteristics of Topologies

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

### 3.3.2 Fairness index

Jain's fairness index is one of the earliest measurements which is used to calculate how a network topology behaves fairly among different flows. This method is implemented by  $f(x)$ , where  $0 \leq f(x) \leq 1$ . So, if the value of  $f$  is closer to 1, in this case, the system is much fairer.

$$f(x) = \frac{\left[ \sum_{i=1}^n x_i \right]^2}{\sum_{i=1}^n x_i^2} \quad (14)$$

In equation (14),  $x_i$  is the sending rate of source  $i$  and  $n$ , is the number of flows. So, the equation gives us an overall fairness index value for the system.

### 3.4. Simulation Results

In this part, we implement a demonstrative results according to the topologies presented in Fig. 3. During the simulations, the delay price is set up to 0 at the beginning. First of all, we start by testing the fairness index for TRUMP and our proposed model to see the differences between them.

In terms of choosing the multiple paths, we used the specific pattern to see how the system can achieve fairness bandwidth allocation. As it's mentioned in 3.3.1, we have the various path lengths in the simulations.

The superiority of our proposed model is more obvious, when there are very longer paths versus very shorter paths. Hence, in NTT topology, our proposed model works dramatically better than TRUMP in terms of fair resource allocation. In another word, our model is very good for big networks.

#### 3.4.1 Case study of fairness allocation

The experiments are based on the impact of various number of sources on fairness index results. Competing flows, share a bottleneck link to reach the destination. So, this is important factor to show the fair bandwidth allocation in a strict network condition. By considering the type of each topology, there will be different results.

Table 2. Fairness Measure for NSF Topology

Cases	TRUMP	Proposed Model
Case 1	0.73	0.97
Case 2	0.90	0.83
Case 3	0.73	0.90
Case 4	0.87	0.99
Case 5	0.74	0.98

Table 3. Fairness Measure for NTT Topology

Cases	TRUMP	Proposed Model
Case 1	0.56	0.96
Case 2	0.50	0.84
Case 3	0.44	0.97
Case 4	0.45	0.95
Case 5	0.53	0.91

Also, as we mentioned, the hop lengths are different and they have at least 1 to maximum 48 hop counts, such as NTT topology. The results from Tables 2, 3, 4, 5 and 6, reveal that the proposed model is better than TRUMP algorithm. Also, based on Fig. 4, we can clearly see that the bandwidth allocation in our model is relatively better than TRUMP in different topologies.

Table 4. Fairness Measure for Cost Topology

Case:	TRUMP	Proposed Model
Case 1	0.81	0.99
Case 2	0.85	0.96
Case 3	0.86	0.96
Case 4	0.84	0.97
Case 5	0.80	0.97

Table 5. Fairness Measure for Access-core Topology

Cas	TRUMP	Proposed Model
Case 1	0.77	0.98
Case 2	0.84	0.95
Case 3	0.73	0.84
Case 4	0.71	0.84

Table 6. Fairness Measure for Core Topology

Cases	TRUMP	Proposed Model
Case 1	0.76	0.99
Case 2	0.82	0.95
Case 3	0.79	0.93
Case 4	0.71	0.9
Case 5	0.69	0.95

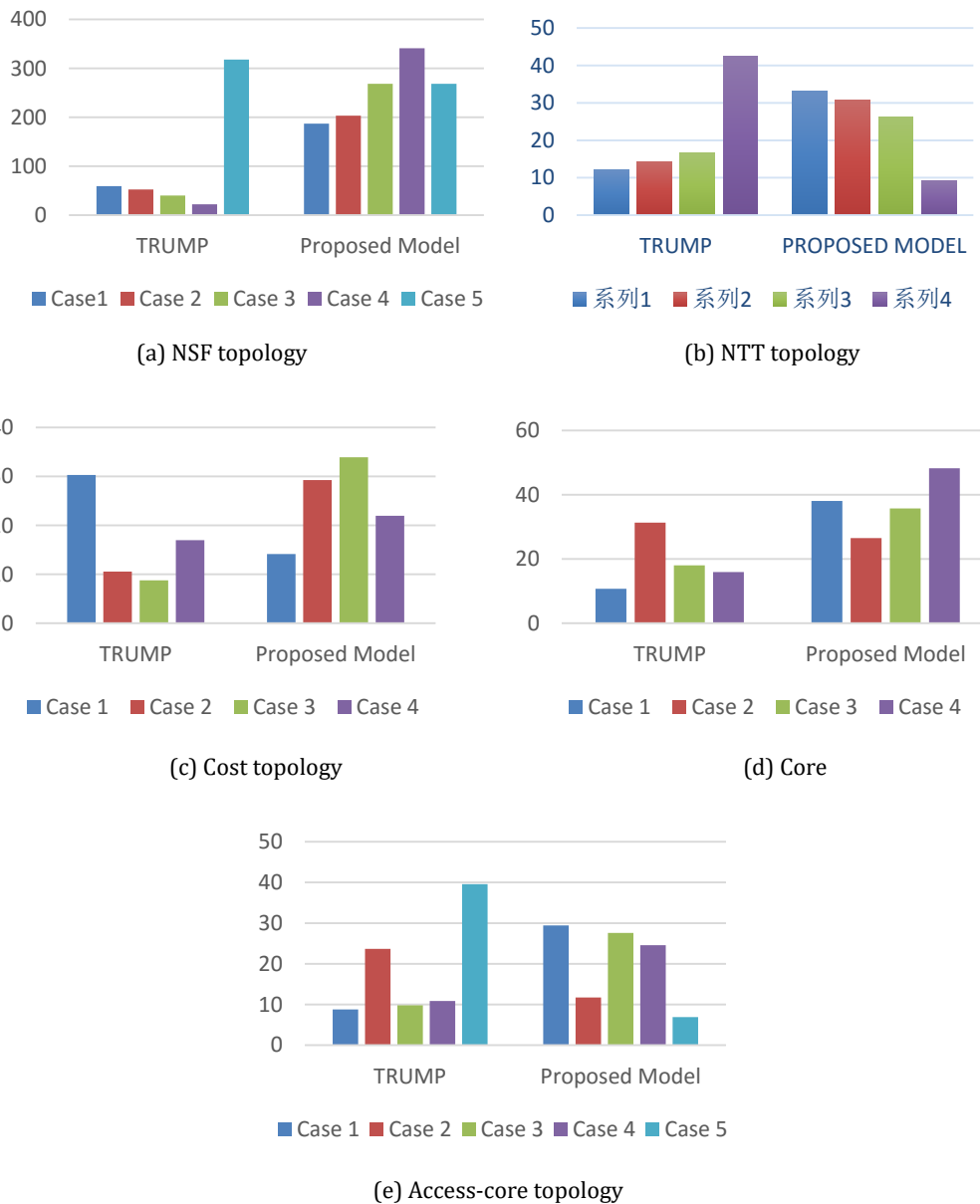


Fig. 4. Bandwidth allocation in different topologies.

## 4. Conclusion

In this paper we considered one of the most important issues in computer networks in terms of fair bandwidth allocation. Then, by taking into account TRUMP algorithm as one of the distributed algorithm, we highlighted one of the drawbacks of it in case of fair bandwidth allocation and we then proposed a model to address that problem. The results showed that our new model provides significantly better fair allocation compared to TRUMP.

## References

- [1] He, J., Suchara, M., Bresler, M., Rexford, J., & Chiang, M. (2007). Internet traffic management: From multiple decompositions to a practical protocol. *Proceedings of the 2007 ACM CoNEXT Conference* (pp. 1-12).
- [2] He, J., Suchara, M., Bresler, M., & Rexford, J. (2008). From multiple decompositions to TRUMP: Traffic management using multipath protocol.

- [3] Xu, K., Liu, H., Liu, J. & Zhang, J. (2011). LBMP: A logarithm-barrier based multipath protocol for internet traffic management. *IEEE Transactions on Parallel and Distributed Systems*, 22(3).
- [4] Rexford, J. (2006). Route optimization in IP networks. *Handbook of Optimization in Telecommunications*. Springer Science + Business Media.
- [5] Palomar, D. P., & Chiang, M. (2006). A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communications*, 24(8), 1439-1451.
- [6] Lin, X., & Shroff, N. B. (2017). Utility maximization for communication networks with multipath routing. *IEEE Trans. Automatic Control*, 51.
- [7] He, J., & Rexford, J. (2007). *Towards Internet-wide Multipath Routing* (Report TR-787-07). Princeton University.
- [8] He, J., & Rexford, J. (2010). Design for optimizability: Traffic management of a future internet. *Springer, Algorithms for Next Generation Networks, Part of the Series Computer Communications and Networks*, 3-18.
- [9] Shi, H., Prasad, R., Onur, E., & Zhang, J. (2014). Fairness in wireless networks-issues, measures and challenges. *IEEE Communications Surveys Tutorials*, 16(1).
- [10] Palomar, D., & Chiang, M. (2006). A tutorial on decomposition methods and distributed network re-source allocation. *IEEE J. on Selected Areas in Communications*, 24, 1439-1451.
- [11] Kelly, F. P., Maulloo, A. K., & Tan, D. K. H. (1998). Control for communication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3), 237-252.
- [12] Wagner, D. P. *Congestion Policing Queues-A New Approach to Managing Bandwidth Sharing at Bottlenecks*. Stuttgart: Institute of Communication Networks and Computer Engineering, University of Stuttgart, Germany.



**Hamed Hamzeh** is a graduate student in data science program at Istanbul Sehir University, Istanbul, Turkey, where he is a teaching and research assistant at the school of computer engineering. His research interests are computer networks, cloud computing, data bases and big data analysis.



**Mahdi Hemmati** is a Ph.D. candidate and research assistant in DISCOVER Lab in University of Ottawa, Ottawa, Canada. He has different awards and funds. His research interests are distributed learning-based optimization in video streaming applications fair and efficient resource allocation in SDN for multimedia traffic.



**Shervin Shirmohammadi** received the Ph.D. degree in electrical engineering from the University of Ottawa, Canada, where he is currently a professor with the School of Electrical Engineering and Computer Science. He is the director of the Distributed and Collaborative Virtual Environment Research Laboratory, and an affiliate member of the Multimedia Communications Research Laboratory, conducting research in multimedia systems and networks, specifically video systems, gaming systems, and multimedia-assisted biomedical systems. The results of his research, funded by more than \$13 million from both industry and public sectors, have led to more than 300 publications, over 80

researchers trained at the postdoctoral, Ph.D., master's, and research engineer levels, over 20 patents and technology transfers to the private sector, and a number of awards and prizes. He is the associate editor-in-chief of IEEE Transactions on Instrumentation and Measurement, senior associate editor of ACM Transactions on Multimedia Computing, Communications, and Applications, and associate editor of IEEE Transactions on Circuits and Systems for Video Technology. He was the associate editor-in-chief of IEEE Instrumentation and Measurement Magazine from 2014 to 2015, and associate editor of Springer's Journal of Multimedia Tools and Applications from 2004 to 2012. Dr. Shirmohammadi is an IEEE fellow for contributions to multimedia systems and network measurements, a University of Ottawa Gold Medalist, a licensed professional engineer in Ontario, and a lifetime professional member of the ACM.