January 2014, Volume 1, Number 1 (pp. 61–71)
http://www.jcomsec.org

Journal of Computing and Security

An Optimal Traffic Distribution Method Supporting End-to-End Delay Bound

Touraj Shabanian [a,*]
Massoud Reza Hashemi [a]
Ahmad Askarian [a]
Behnaz Omoomi [b]

[a] *Department of Electrical and Computer Engineering , Isfahan University of Technology, Isfahan, Iran.*
[b] *Department of Mathematical Science, Isfahan University of Technology, Isfahan, Iran.*

ARTICLE INFO.

Article history:
Received: 21 November 2012
Revised: 31 July 2013
Accepted: 14 September 2013
Published Online: 20 December 2013

Keywords:
Traffic Distribution, Routing, Convex Optimization, Subgradient Method.

ABSTRACT

Routing methods for optimal distribution of traffic in data networks that can also provide quality of service (QoS) for users is one of the challenges in recent years' research on next generation networks. The major QoS requirement in most cases is an upper bound on end-to-end path delay. In multipath virtual circuit switched networks each session distributes its traffic among a set of available paths. If all possible paths are considered available, then the source's decision on its traffic distribution can be considered as routing. A model of the routing function as a mathematical problem which distributes the input traffic over possible paths for each session is proposed here. A distributed and iterative algorithm which will keep the average end-to-end delay for individual paths below a required bound is introduced. This algorithm minimizes the total average delay of all packets in the network. The convergence of the algorithm is illustrated.

1 Introduction

Computer networks have evolved into a new generation where a wide range of new services are provided to various network users [1]. For many of these new services, such as VOIP, IPTV, Network Games, etc, it is not sufficient just to transfer the information to the destination, but for the users' satisfaction it is necessary to guarantee their required QoS as well. In this manner, the new services with arbitrary QoS requirements can be deployed in the network. Providing the QoS must be achieved by utilizing the least possible resources of the network such that the network can be optimized in terms of resource utilization [2]. Network optimization algorithms determine traffic distribution for a given traffic demand so that the optimum resource utilization can be achieved. But the research results so far show that providing QoS in cases where routing is performed without paying attention to the QoS requirements is difficult. Therefore, considering the required QoS in the optimization algorithms and determining the routes accordingly is one of the challenges of the next generation networks [3].

Recently several new services have become popular in the internet qualities of which depend on the end-to-end delay experienced by the packets in the network

[*] Corresponding author.

Email addresses: shabanian@ec.iut.ac.ir (T. Shabanian), hashemim@ cc.iut.ac.ir (M. R. Hashemi), a.askarian@ec.iut.ac.ir (A. Askarian), bomoomi@cc.iut.ac.ir (B. Omoomi)

ISSN: 2322-4460 © 2014 JComSec. All rights reserved.

[4–8]. For an acceptable QoS it is required that the end-to-end delay is kept under a threshold level. Providing QoS is not an easy task in datagram networks. In new generation networks, virtual circuit switched networks such as MPLS is used to provide a better framework to implement QoS.

Most of the QoS provisioning algorithms in the literature exploit certain mechanisms to guarantee the delay for a given path. Nen Jin, et.al show that for providing QoS in a DiffServ network, the price per unit of traffic rate for each traffic class can be adjusted. They assume a given path for a user. The satisfaction of the user is modeled through a convex function of the traffic passing through that given path and the QoS level of the assigned traffic class [9]. In [5] QoS is proposed to be provided by adjusting the capacity allocated to each DiffServ class. The QoS measure is the exact proportion of the average delay of two different traffic classes. Each user's traffic is routed through a predetermined path and depending on the amount of traffic of each class, the traffic over this path experiences a delay which is considered as its cost. In [6] a dynamic method is used to adjust the users' traffic rate in a manner that a minimum rate and a maximum delay threshold are guaranteed. A predetermined path used for routing the traffic and its rate is determined by solving a convex optimization problem which satisfies the user's delay requirements.

Most of the articles that study the traffic distribution in virtual circuit switched networks assume a set of known paths for each source-destination pair. To simplify the problem, usually, a small set of paths is selected from all possible paths beforehand [10, 11]. In the articles that find routes based on QoS requirements, the QoS is mostly measured based on m parameters. Each QoS parameter for a path is sum of the QoS parameters of its links. The links are modeled by an m-dimensional weight vector $W = (w_1, ..., w_m)$ the components of which represent the QoS parameters of links. Paths with QoS parameters lower than the threshold levels will satisfy the required QoS and can be selected. In this manner the QoS-based routing problem is modeled as a multi-constraint (optimal) problem. Since these problems are NP-hard, in most cases heuristic methods are adopted in solving them [3].

Here the objective is to introduce a scalable method in terms of the number of sessions, in order to distribute the network's traffic over available paths in a virtual circuit switched network that would minimize the average delay for all packets as the total cost of the network, while guaranteeing a bounded end-to-end path delay as the users' QoS requirement. The proposed method in this article is based on the analysis of the traffic distribution problem with delay constraints. As a result, this problem is modeled as a constrained convex optimization problem and the routing algorithm is provided in accordance to the analytical solution of this problem.

In Section 2 an analytical model for distributing traffic is introduced where the traffic distribution is modeled as a constrained convex optimization problem. In Section 3 the Lagrangian dual method is adopted for solving this problem. An algorithm that can be realized in a data network based on the dual method is proposed here. In Subsection 3.1 the implementation method of the proposed algorithm in real networks is explained. In section 4 the simulation results are provided expressing that this proposed method converges and can achieve its objective in an effective manner. This article will be concluded in Section 5. The analysis of the proposed model is provided in the Appendix A.

2 Traffic Distribution Model

The objective in common for all the routing algorithms is to determine the appropriate paths for carrying the users' traffic from source to destination. All or part of each user's traffic is assigned to each selected path; therefore, a direct output of a routing algorithm is the amount of traffic allocated to each path. In fact, routing can be modeled as a mathematical problem which determines the distribution of all sessions' traffic over the network graph.

In this article source-destination pairs are assumed to be known and are presented by the set W. Each source-destination pair $w \in W$ is considered as a session and its average input traffic is presented by r_w. A data network is modeled as a stationary and directed graph $G(A, V)$. The graph nodes, represented by set V model the network routers or gateways and graph links represented by set A, model the physical links between the routers. Some of the nodes of the graph are source or destination of the sessions in the network (Figure 1). A session path is a set of links that connects the source of the session to its destination. The set of the paths of each session is called P_w. Thus the routing problem is similar to finding the distribution of each session's traffic over its paths.

The parameters and notations which are used in the rest of this article are introduced in the following Nomenclature:

- W: The set of all existing sessions, where N_W shows the total number of these sessions
- P: The set of available paths of all sessions $w \in W$ in $G(A, V)$,where N_P shows the total number of these paths
- P_w: The set of available paths of session w

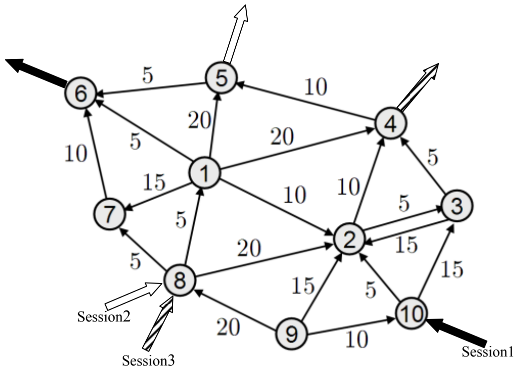


Figure 1. A network graph with three sessions

- r_w: Average traffic rate of session w
- x_p: Traffic assigned to path $p \in P$
- X: A vector of N_P components whose *pth* component is the assigned traffic to path p, x_p
- λ_p: The lagrangian multiplier according to the delay constraint of path p
- Λ: A vector of N_P components whose *pth* component is the λ_p
- th_p: The threshold level of average delay of packets in path p
- Th : A vector of N_P components whose *pth* component is th_p
- f_{ij}: The flow crossing from link (i, j) of $G(A, V)$
- $h_p(X)$: The cost function associated with path p
- $H(X)$: A vector of N_p components whose *pth* component is $h_p(X)$
- $D_{ij}(f_{ij})$: The cost function associated with link (i, j)

Based on the above definitions the following relations hold:

$$x_p \geq 0 \qquad \forall p \in P \qquad (1)$$

$$\sum_{p \in P_w} x_p = r_w \qquad \forall w \in W, \forall p \in P_w \qquad (2)$$

$$f_{ij} = \sum_{p|(i,j) \in p} x_p \qquad \forall (i, j) \in A \qquad (3)$$

$$h_p(X) = \sum_{(i,j) \in p} D_{ij}(f_{ij}) \qquad \forall p \in P \qquad (4)$$

If the average delay of the packets over a link is considered as the link's cost function, $D_{ij}(f_{ij})$, and the messages are delayed only by the links of the network, then (5) expresses the expected delay for all packets over the network [12]. Equation (5) indicates the average time that packets remain in the network and use network resources; thus, it can be considered as the overall system cost.

$$D = \sum_{(i,j) \in A} D_{ij}(f_{ij}) \qquad (5)$$

Even in a virtual circuit network minimizing (5) can be a good objective for traffic distribution since it can improve network resource utilization [13, 14]. In the virtual circuit switched networks, each session's traffic is distributed among the available paths. By assuming a stable network and assuming that the traffic of the sessions is stationary, this problem is modeled and analyzed as the problem of distributing the average input traffic of each session r_w, over the set of session's paths P_w, which will result in the sessions' path flows x_p, for all sessions. Thus, f_{ij}, the total flow of link (i, j), can be expressed by the different path flows. As a result f_{ij} equals the sum of all path flows traversing link (i, j), (3). Here each session represents a customer. The expectation of each customer from the network is defined based on the customer's traffic's delay tolerance. In this case the customer will be satisfied if the average delay is bounded to a certain threshold. Therefore, considering the delay of each link as its cost is deemed to be appropriate. In this model the sum of the cost function of the links which compose a path, is considered as the path cost, $h_p(X)$, which is equal to the sum of the costs of the path's links (4). Considering (5) as the overall cost function of the network and (4) as the customer cost, the limitation of which is required by the customers, the routing in the network can be modeled as Problem 1.

Problem 1.

$$minimize \;\; D(X) = \sum_{(i,j) \in A} D_{ij}\big(\sum_{p|ij \in p)} x_p\big) \qquad (6)$$

$$\sum_{p \in P_w} x_p = r_w \qquad \forall w \in W, \forall p \in P_w \qquad (7)$$

$$x_p \geq 0 \qquad \forall p \in P \qquad (8)$$

$$h_p(X) \leq th_p \qquad \forall p \in P \qquad (9)$$

In this problem, the path flows x_p, are the variables. The objective function $D(X)$ is considered as the overall system cost. The purpose of this problem is to find the distribution of the traffic among the available paths in order to minimize the overall system cost while the constraints (7) to (9) are satisfied. Constraints (7) and (8) guarantee the acceptable allocation of the traffic over the session's paths, and constraint (9) guarantees the delay limitation or users' expectation. If constraint (9) is ignored, Problem 1 is converted to Problem 2. Problem 2 is known as optimal routing problem introduced in [15] and improved in [14, 16–18].

Problem 2.

$$minimize D(X) = \sum_{(i,j)\in A} D_{ij}(\sum_{p|ij\in p} x_p)$$

$$\sum_{p\in P_w} x_p = r_w \qquad \forall w \in W, \forall p \in P_w$$
$$x_p \geq 0 \qquad \forall p \in P$$

3 Solving The Problem

Usually the cost function $D_{ij}(f_{ij})$ is expressed as a convex, non-decreasing, continuous and differentiable function; therefore, the path cost will have the above characteristics. Since the cost functions $h_p(X)$ are convex, Problem 1 is a constrained convex optimization problem [19, 20], which can be solved using any of the existing methods, such as Projected Gradient, Interior Point, etc. But here the objective is to find a solution that can also be implemented in real networks. In this regard the Lagrange dual problem is formulated and solved. In other words, since Problem 1 is a convex optimization problem the duality theorem is adopted in solving it. The fact that strong duality holds is presented in Proposition 1. Since there is a practical solution to solve Problem 2 [15], the dual problem is described using the Lagrange multipliers related to (9). Thus the Lagrangian is (10) where only constraint (9) is relaxed by introducing Lagrange multiplier λ_p for each path $p \in P$. The resultant partial dual function is Problem 3 [19].

$$L(X, \Lambda) = D(X) + \sum_{p\in P} \lambda_p.(h_p(X) - th_p) \quad \forall \Lambda \geq 0 \tag{10}$$

Problem 3.

$$q(\Lambda) = \quad minimize L(X, \Lambda)$$

$$\sum_{p\in P_w} x_p = r_w \qquad \forall w \in W, \forall p \in P_w$$
$$x_p \geq 0 \qquad \forall p \in P$$

Considering Problem 3 as the dual function of Problem 1, the dual problem will be Problem 4.

Problem 4.

$$maximize \quad q(\Lambda)$$
$$\lambda_p \geq 0 \qquad \forall p \in P$$

As mentioned in Proposition 2, the $-q(\Lambda)$ is a convex function which is not necessarily differentiable in general, but it is sub-differentiable at all points. Therefore, Problem 4 can be solved iteratively by adopting the subgradient method [21]. In this method an initial value is given to variable Λ, (Λ^0), and in each iteration according to (11) a new value is calculated which will be closer to the optimum value.

$$\Lambda^{k+1} = [\Lambda^k + \alpha^k.g^k]^+ \tag{11}$$

To calculate the new value of Λ in the kth iteration, first a subgradient of function $-q(\Lambda)$ called $-g^k$ is calculated at Λ^k, and then Λ^{k+1} is calculated by using (11) where, α^k is a positive step size and "+" denotes projection on the set R^+. As result-4 indicates, in order to find a vector g^k the traffic must be distributed based on Problem 3 solution according to $\Lambda = \Lambda^k$, denoted by $X^*(\Lambda^k)$. In this case the deviation of the cost of a path from its threshold th_p, is equal to the associated component of g^k, (12).

$$g^k = h_p(X^*(\Lambda^k)) - th_p \tag{12}$$

Eventually, the iterative algorithm finds Λ^* which is the best solution for Problem 4. Obviously in this iteration the input traffic is distributed similar to that of the path flows which are the solution of Problem 3 for the amount of $\Lambda = \Lambda^*$. Since the conditions for strong duality exists according to Proposition 1, this distribution will be the optimum solution of Problem 1 as well. In the following section the proposed algorithm is explained. The convergence proof of this problem is presented in the Appendix A.

Algorithm Steps:

Step1: A feasible value is given to Λ. Since in Problem 4 every $\Lambda \geq 0$ is acceptable, the $\Lambda^0 = 0$ is used as the initial value. In this step, the initial value of q^{best} is 0.

Step2: In iteration k, Problem 3 must be solved based on the value of Λ^k, leading to the optimum value $q(\Lambda^k)$ and the optimum point $X^*(\Lambda^k)$. The components of this vector are represented by $x_p^*(\Lambda^k)$. In other words a mechanism must be used to determine path flows, for the optimal routing problem when (13) is considered as the cost function of each link. Therefore the Lagrange multipliers can be interpreted as the bottleneck indicators of the paths.

$$D_{ij}^k = (1 + \sum_{p|ij\in p} \lambda_p^k).D_{ij}(\sum_{p|ij\in p} x_p^*(\Lambda^k)) \tag{13}$$

Step3: In iteration k with respect to the value of $X^*(\Lambda^k)$ which is calculated in step2, the deviation of each path's cost from the threshold level of the same path is calculated. Considering the Proposition 3, the negative of this value can be considered as the *pth* component of the subgradient vector of $-q(\Lambda)$ at Λ^k or

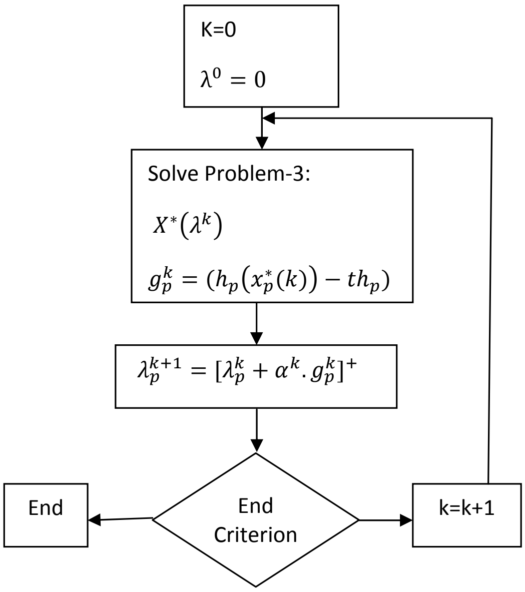

K=0
$\lambda^0 = 0$
Solve Problem-3:
$X^*(\lambda^k)$
$g_p^k = (h_p(x_p^*(k)) - th_p)$
$\lambda_p^{k+1} = [\lambda_p^k + \alpha^k . g_p^k]^+$
End
End Criterion
k=k+1

Figure 2. The flowchart of flow distribution algorithm

$-g_p^k$. After calculating the deviation for all paths, the value of Λ for next iteration or Λ^{k+1} can be calculated using (11).

Step4: The value of $q^{best} = max\{q^{best}, q(\Lambda^k)\}$is calculated and k is increased by one. Then if the condition of ending the algorithm is met, the algorithm terminates, otherwise, it goes back to step2 for next iteration.

Condition of ending the algorithm: In a simple case, the condition which leads to the algorithm termination can be the maximum number of iterations (Figure 2).

3.1 Matching the algorithm with real networks

As mentioned before, the main objective of this article is to distribute the input traffic of a session over its known paths. A session can be equivalent of a source and destination pair in virtual circuit switched networks such as ATM and MPLS, or in general in any network that uses explicit routing or source routing. Even a certain DiffServ class traversing the same LSP in these networks can be considered as a session. In practice this proposed algorithm is implemented for each session iteratively and in parallel for all sessions.

Here each iteration of the algorithm is assumed to be performed in one time slot. At the end of a time slot, destination nodes calculate the deviation of the average delay for each path from the required delay bound. The bottleneck multiplier of the path is calculated based on its cost deviation and is sent to the source node. The average delay of packets in each iteration can be determined by the destination either using analytical modeling or just by measurement. In a case where the path delay is estimated by using measurement methods, based on the assumptions about the link cost in this article, this proposed algorithm will definitely converge according to the Proposition 4. During each time slot the source nodes distribute the input traffic according to the optimal point of Problem 3. In each iteration, the Problem 3 is an optimal routing problem where the cost function of each link is defined by (13). This problem can be solved by one of the existing methods [13, 14, 16–18].

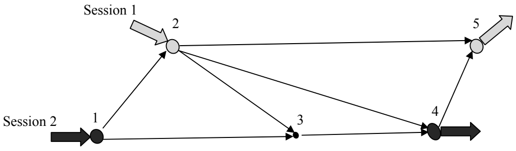


Figure 3. Network simulation graph

Each time slot can be in the order of the end-to-end trip time in the network. The algorithm is scalable because it is implemented independently for each session. If the set of the paths for each session can be assumed to include all possible paths for the session based on the topology of the network, the algorithm will practically select the routes; therefore, a separate method for determining the possible routes will not be necessary.

4 Simulation

The algorithm for two sessions is simulated over the network graph in Figure 3. The algorithm is executed independently for each session in an iterative and synchronized manner. All possible paths for session 1 are P1(14a), P2(14b) and P3(14c) and for session 2 are P4(14d), P5(14e) and P6(14f).

$$P1 = \{(1,2),(2,4)\} \tag{14a}$$
$$P2 = \{(1,2),(2,3),(3,4)\} \tag{14b}$$
$$P3 = \{(1,3),(3,4)\} \tag{14c}$$
$$P4 = \{(2,3),(3,4),(4,5)\} \tag{14d}$$
$$P5 = \{(2,4),(4,5)\} \tag{14e}$$
$$P6 = \{(2,5)\} \tag{14f}$$

In this simulation the average delay of the links is modeled as (15) which is a convex, continuous, and differentiable function of its average traffic. In this equation C_{ij} is the capacity of the link (i,j) and K_{ij} is a positive coefficient of the link. The domain of this function covers the traffic flows between 0 and C_{ij} only and as the flow gets closer to C_{ij} the delay increases exponentially. The function is undefined for values

Table 1. Parameters of the Network links

Link(i,j)	K(i,j)	C(i,j)
(1,2)	2	44.7
(1,3)	4	16
(2,3)	3	44.7
(2,4)	1	44.7
(2,5)	8	44.7
(3,4)	4	44.7
(4,5)	2	16

equal to or above C_{ij}. The coefficient and capacity of the links of Figure 3 are proposed in Table 1.

$$D_{ij}(f_{ij}) = \frac{(K_{ij} * f_{ij}^2)}{(C_{ij} - f_{ij})} \qquad (15)$$

The constant input traffics are used in the simulation as the expected values of the sessions' traffics in general. The average input traffic for each session is assumed to be 20 Mbps. In this simulation the attempt is made to clarify two important points: to show that the iterative algorithm converges to the optimal point of Problem 1 and that this algorithm achieves its objective in limiting the end-to-end delay of the paths in addition to minimizing the total network delay. Since the main objective of this proposed model is similar to the optimal routing problem, the Problem 2, the results of the proposed algorithm are compared with the Problem 2, for the above scenario.

In the first step, the path flows for each session are calculated based on solving the optimal routing problem, the Problem 2, by applying CVX package in MATLAB. In this case the end-to-end delay for each path as well as the expected delay of packets are calculated (see Table 2).

In the second step, the path flows for each session are calculated based on the optimal routing problem with end-to-end delay constraint, Problem 1. The end-to-end delay bound for each path is assumed to be 76 units in this simulation. The path flows are calculate by solving Problem 1 applying CVX package in MATLAB (see Table 3).

The total cost of the network in step 2 is slightly higher than the optimum total cost in step 1. Yet in step 1 the individual path cost, for paths 1 and 6, is beyond the end-to-end delay bound. This means that this proposed algorithm is able to limit the delay with a minimum increase in the total cost. Also it can be seen that based on the Complementary Slackness condition, x_p of paths 1 and 6 is decreased from the optimum values of step 1, down to a point that their average delays are decreased to the threshold level. As such, the optimum dual variable, DV, of these two paths is expected to be higher than zero while DV of the other paths expected to be zero. It can be interpreted that the marginal cost of the paths 1 and 6 should be lower compared to that of path 3 for the calculated traffic.

In the final step, the proposed algorithm is simulated through MATLAB. Here the step size is 0.008. The simulation finishes after 1000 iterations. The final results of the algorithm are presented in Table 4. The stepwise results of the algorithm for Lagrange multipliers and two of the link flows as a sample are presented in Figure 4 and Figure 5.

The results in Table 4 are the same as the results in Table 3. This means that the iterative algorithm converges to the same results of the centralized solution.

Figure 4 shows that the path flows converge to the same results as the results of the case where the Problem 1 is solved in a central manner.

Figure 5 shows that the Lagrangian multipliers of the distributed solution converge to the optimal dual variable values obtained from the centralized solution of the Problem 1.

5 Conclusion

In this article a new method is introduced for traffic distribution in virtual circuit switched networks which can be implemented in real networks. In this method the input traffic of each session is distributed among the possible paths, in a manner that the total system cost is minimized at the same time as the average cost for each path is kept bounded below a required threshold level. This method is scalable as its operation is per session. It is analytically proven in this article that this algorithm converges under the assumptions that are feasible in real networks. The simulation results approve the effectiveness of the algorithm. The results obtained from the simulation are in line with the results obtained from analytical resolution of the convex optimization problem.

Appendix A Mathematical Analysis

In this section the analysis of the proposed algorithm is provided. First some parameters used in this section are defined

- $\overline{x}_p$: Flow of the path p that is held in Assumption 1
- $H(X)$: Cost vector of all sessions with N_P components where the pth component represents the cost of the pth path
- $A(X)$: Deviation vector with N_P components where the pth component represents the deviation

Table 2. Simulation results of step1

a) Link Information

Link	Flow	Cost
(1,2)	17.92	47.96
(1,3)	2.08	4.98
(2,3)	0	0
(2,4)	25.47	33.73
(3,4)	2.08	6.50
(2,5)	12.45	76.91
(4,5)	7.55	26.97
$\sum D_{ij}(f_{ij})$		197.05

b)Path Information

Path	Path Flow	Dual Varaible(DV)	E2E Delay
1	17.92	11.55	81.68
2	0	13.55	54.46
3	2.08	11.55	11.48
4	0	16.74	33.48
5	7.55	14.74	60.7
6	12.45	14.744	76.91

Table 3. Simulation results of step 2

a) Link Information

Link	Flow	Cost
(1,2)	17.39	44.28
(1,3)	2.61	8.16
(2,3)	0	0
(2,4)	25	31.72
(3,4)	2.61	10.38
(2,5)	12.39	76
(4,5)	7.61	27.62
$\sum D_{ij}(f_{ij})$		198.16

b)Path Information

Path	Path Flow	Marginal Cost	Lag Multiplier	E2E Delay
1	17.39	10.87	0.385	76
2	0	14.91	0	54.66
3	2.61	15.04	0	18.53
4	0	18.74	0	38
5	7.61	14.7	0	59.35
6	12.39	14.62	0.115	76

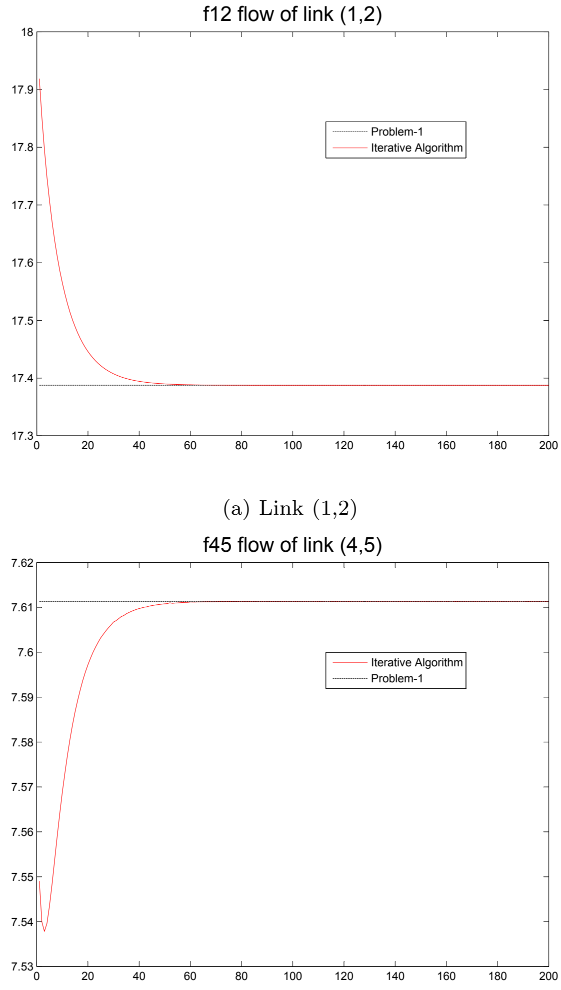
Table 4. Final results of Step3 for 1000 iterations and step size 0.008

a) Link Information

Link	Flow	Cost
(1,2)	17.39	44.28
(1,3)	2.61	8.16
(2,3)	0	0
(2,4)	25	31.72
(3,4)	2.61	10.38
(2,5)	12.39	76
(4,5)	7.61	27.62
$\sum D_{ij}(f_{ij})$		198.16

b)Path Information

Path	Path Flow	Marginal Cost	Lag Multiplier	E2E Delay
1	17.39	10.87	0.385	76
2	0	14.91	0	54.66
3	2.61	15.04	0	18.53
4	0	18.74	0	38
5	7.61	14.7	0	59.35
6	12.39	14.62	0.115	76



(a) Link (1,2)

(b) Link (4,5)

Figure 4. The flow of links per 200 iterations

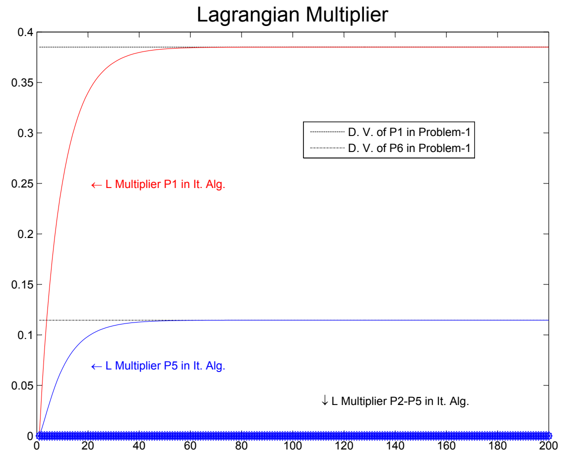


Figure 5. The Lagrangian multipliers corresponding to paths per 200 iterations

of the *pth* path from its threshold

- Th: Threshold vector with N_P components and the *pth* component represents the maximum delay bound of the path p
- Λ^*: Optimum solution of Problem 4 which is a vector with N_P components
- λ_p^*: The *pth* component of the optimum vector Λ^*, which is the optimum Lagrange multiplier of the *pth* path

Assumption 1. *The value of r_w's are such that Problem 1 has at least one strictly feasible point, in other words (16) is held.*

$$\exists \overline{X} | \sum_{p\in P_w} \overline{x}_p = r_w \ \ \& \ \ \overline{x}_p \geq 0 \ \ \& \ \ h_p(\overline{X}) < th_p$$
$$\forall w \in W, \forall p \in P_w \tag{16}$$

Result-1: Since the feasible set of the Problem 1 is not empty, this problem has at least one optimal point [19, 20].

Proposition 1. *The optimum solution of Problem 4 is equal to the optimum solution of Problem 2.*

Proof. Since Problem 2 is a convex optimization problem, if the Slater conditions apply then the strong duality will also apply [19]. According to Assumption 1 the Slater condition is held; therefore strong duality is held □

Result-2: Assuming that the input traffic of sessions w meet (16), a strong duality exists and the optimum solution of Problem 4 is equal to the optimum solution of Problem 2.

Result-3: Because of strong duality, (17) should hold for the optimum points of Problem 2 and Problem 4 as follow:

$$\lambda_p^*.(h_p(x_p^*)) = 0 \equiv \begin{cases} (h_p(x_p^*) - th_p < 0 \Rightarrow \lambda_p^* = 0 \\ (h_p(x_p^*) - th_p = 0 \Rightarrow \lambda_p^* \geq 0 \end{cases} \tag{17}$$

According to (17), at the optimum point of Problem 4, the Lagrange Multiplier of the paths with lower costs than that of the threshold level is 0, and for the paths with Lagrange Multipliers greater than 0, the final traffic amount assigned to them will be such that the cost of these paths will be exactly equal to the threshold level.

Proposition 2. *A) The function $-q(\Lambda)$ defined in Problem 4 is a convex function of Λ.*

B) This function has subgradient at all of the points in its domain.

Proof. If

$$C \triangleq \{\{(x_1 ... x_n)| \sum_{p\in P_w} x_p = r_w, x_p \geq 0 \ \ \forall p \in P_w\}\}$$

Then

$$-q(\Lambda) = maximize_{X\in C}\{-L(X,\Lambda)\}$$

A) $-q$ is a convex function: Defining vector $A(X)$ and function $b(X)$ by (18) and (19), $-L(X,\Lambda)$ can be considered as a linear function of Λ for a given value of vector X, as in (20)

$$A(X) \triangleq Th - H(X) \quad (18)$$

$$b(X) \triangleq \sum_{p\in P} h_p(x_p) \quad (19)$$

$$-L(X,\Lambda) = (A(X)^T.\Lambda + b(X)) \quad (20)$$

Taking into account the definition given in (20) for function $L(X,\Lambda)$, $-q(\Lambda)$ can be considered as the point-wise maximum of the family of linear functions at all points Λ according to (21); therefore $-q(\Lambda)$ is a convex function [19].

$$-q(\Lambda)|_{\Lambda^1} = max_{X\in C}\{(A(X)^T.\Lambda + b(X))|_{\Lambda^1}\} \quad (21)$$

B)Function $-q(\Lambda)$ has subgradient at all points Λ:

The $-q(\Lambda)$ is differentiable at all points Λ where only one X, $X^*(\Lambda)$, maximizes (21), i.e. at these values of Λ, only one of the functions $A(X)^T.\Lambda + b(X))$ is greater than the others; therefore at these points, the subgradient of the function is unique and is equal to its gradient which is calculated through (22).

$$\frac{\partial -q(\Lambda)}{\partial \Lambda} = \nabla(-q(\Lambda)) = A(X^*(\Lambda)) = Th - H(X^*(\Lambda))$$

&

$$X^*(\Lambda) = arg(max_{X\in C}\{(A(X)^T.\Lambda + b(X))\}) \quad (22)$$

The $-q(\Lambda)$ is not differentiable at the points Λ where (21) is at its maximum at some points. At these Λ some of the functions $(A(X)^T.\Lambda + b(X))$ have the greatest value at the same time. In this case, although $-q(\Lambda)$ is not differentiable, it has subgradient which is calculated through (23).

$$\frac{\partial -q(\Lambda)}{\partial \Lambda}|_{\Lambda^1} = Convexhull_{X_i}\{(-A(X_i^*(\Lambda))^T\}$$

& (23)

$$X^*(\Lambda) = arg(max_{X\in C}\{(A(X)^T.\Lambda + b(X))\})$$

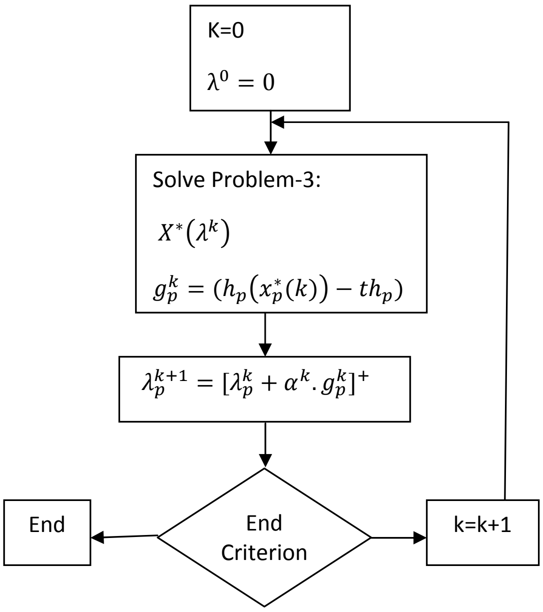

K=0
$\lambda^0 = 0$
Solve Problem-3:
$X^*(\lambda^k)$
$g_p^k = (h_p(x_p^*(k)) - th_p)$
$\lambda_p^{k+1} = [\lambda_p^k + \alpha^k . g_p^k]^+$
End
End Criterion
k=k+1

Figure 6. $-q(\lambda)$ for one dimensional λ

□

According to Proposition 2, the function $q(\Lambda)$ is the point-wise infimum of a family of affine functions (21); hence, it is concave and sub-differentiable at any point (Figure 6). In Proposition 3 an equation is provided to calculate one of the subgradient vectors of function $-q(\Lambda)$ that can be used in the algorithm in Figure 2.

Proposition 3. *At each point $\widehat{\Lambda}$ (24) gives the subgradient of $-q(\Lambda)$ at that point*

Proof. According to (22,23) for a given $\widehat{\Lambda}$, each optimal solution of (21), $X_i^*, -A(X_i^*)$, is one of the subgradient vectors of $-q(\Lambda)$ at point $\widehat{\Lambda}$. According to (21), the optimal point of this equation at point $\widehat{\Lambda}$ can be obtained by solving Problem 3 based on $\widehat{\Lambda}$.

$$-g(\widehat{\Lambda}) = (Th - H(X^*)) \in \frac{\partial q(\Lambda)}{\partial \Lambda}|_{\widehat{\Lambda}}$$

& (24)

$$X^*(\Lambda) = arg(max_{X\in C}\{(A(X)^T.\Lambda + b(X))\})$$

In other words, $X^*(\Lambda)$ is an optimal point of Problem 3 based on $\widehat{\Lambda}$.

□

Result-4: Considering (24) the number of components of vector $g(\widehat{\Lambda})$ is equal to the total number of paths of session w. The *pth* component of this vector is equal to the deviation of the cost of path p from its threshold level. In this equation the path cost should be calculated when the traffic is the optimum solution of Problem 3 for vector $\widehat{\Lambda}$. To calculate the subgradient vector at point $\widehat{\Lambda}$, solving Problem 3 at vector $\widehat{\Lambda}$ and finding its optimum solutions suffices. Following this, the cost of each path is calculated for this traffic and its deviation from the threshold level is considered as the component of the subgradient vector.

Proposition 4. *The algorithm introduced in Section 3 converges:*

Proof. As shown in Figure 2, this algorithm describes the steps of the subgradient method in solving Problem 4. According to the proof given in [21], if the value of the subgradient of function $-q(\Lambda)$ in all points has an upper bound such as G and if the distance from the initial point of the algorithm and the optimum point is less than R, the subgradient method converges [21]. To prove the convergence of the algorithm, first, an upper bound for the distance of the initial point of this algorithm and the optimum point is introduced, and then the upper bound for the value of the subgradient vector of function $-q(\Lambda)$ at all acceptable points is calculated.

A) Upper bound for the distance between the initial point Λ^0 and optimal point Λ^*:

The initial point of the proposed algorithm in this article is $\Lambda^0 = 0$. Assume a component λ_p^* is infinite. Considering Assumption-1 the amount of $L(\overline{X}, \Lambda^*)$ and also $g(\Lambda^*)$ is $-\infty$. The optimal value of Problem 3 will be$-\infty$, while the optimal values of Problem 3 and Problem 1 were expected to be equal. considering Assumption-1 the optimal value of Problem 1 is finite (a contradiction); therefore all components of Λ^* are finite, hence $|\Lambda^* - \Lambda^0|$ is bounded.

B) The norm of the subgradient vector in all iterations is upper bounded:

In iteration k, the component p of the subgradient vector is equal to the difference of $h_p(X^*(\Lambda^k))$ with th_p. Considering Assumption-1, $(X^*(\Lambda^k)$ is a finite vector and since the optimal value of Problem 1 is finite then $h_p(X^*(\Lambda^k))$ must be finite, hence, the norm of the vector is finite. Based on the maximum distance between the initial and the optimal points of the algorithm and the upper bound calculated for the subgradient at every step of the algorithm, the subgradient method for solving this problem will converge.

□

References

[1] Ning Wang, Kin-Hon Ho, George Pavlou, and Michael P. Howarth. An overview of routing optimization for internet traffic engineering. *IEEE Communications Surveys and Tutorials*, 10(1-4):36–56, 2008. URL http://dblp.uni-trier.de/db/journals/comsur/comsur10.html#WangHPH08.

[2] W.C.Hardy. *Qos Measurement and Evaluation of Telecommunications Quality of Service*. John Wiley & Sons, 2001.

[3] Xavier Masip-Bruin, Marcelo Yannuzzi, Jordi Domingo-Pascual, Alexandre Fonte, Marília Curado, Edmundo Monteiro, Fernando A. Kuipers, Piet Van Mieghem, Stefano Avallone, Giorgio Ventre, Pedro A. Aranda-Gutiérrez, Matthias Hollick, Ralf Steinmetz, Luigi Iannone, and Kavé Salamatian. Research challenges in qos routing. *Computer Communications*, 29(5):563–581, 2006.

[4] Seung Yeob Nam, Sunggon Kim, and Dan Keun Sung. Measurement-based admission control at edge routers. *IEEE/ACM Trans. Netw.*, 16(2): 410–423, 2008. URL http://dblp.uni-trier.de/db/journals/ton/ton16.html#NamKS08.

[5] Ishai Menache and Nahum Shimkin. Capacity management and equilibrium for proportional qos. *IEEE/ACM Trans. Netw.*, 16(5):1025–1037, 2008. URL http://dblp.uni-trier.de/db/journals/ton/ton16.html#MenacheS08.

[6] Mohamed E. M. Saad, Alberto Leon-Garcia, and Wei Yu. Optimal network rate allocation under end-to-end quality-of-service requirements. *IEEE Transactions on Network and Service Management*, 4(3):40–49, 2007.

[7] Xavi Masip-Bruin, Marcelo Yannuzzi, Rene Serral-Gracia, Jordi Domingo-Pascual, Jose Enriquez-Gabeiras, Maria Angeles Callejo, Michel Diaz, Florin Racaru, Giovanni Stea, Enzo Mingozzi, Andrzej Beben, Wojciech Burakowski, Edmundo Monteiro, and Lus Cordeiro. The euqos system: a solution for qos routing in heterogeneous networks [quality of service based routing algorithms for heterogeneous networks]. *Communications Magazine, IEEE*, 45(2):96–103, Feb. 2007. ISSN 0163-6804. doi: 10.1109/MCOM.2007.313402.

[8] Wojciech Burakowski, Andrzej Beben, Halina Tarasiuk, Jaroslaw Sliwinski, Robert Janowski, Jordi Mongay Batalla, and Piotr Krawiec. Provision of end-to-end qos in heterogeneous multi-domain networks. *Annales des Tlcommunications*, 63(11-12):559–577, 2008. URL http://dblp.uni-trier.de/db/journals/adt/adt63.html#BurakowskiBTSJBK08.

[9] Nan Jin and Scott Jordan. On the feasibility of dynamic congestion-based pricing in differentiated services networks. *IEEE/ACM Trans. Netw.*, 16(5):1001–1014, 2008.

[10] Srikanth Kandula, Dina Katabi, Bruce S. Davie, and Anna Charny. Walking the tightrope: re-

sponsive yet stable traffic engineering. In Roch Guérin, Ramesh Govindan, and Greg Minshall, editors, *SIGCOMM*, pages 253–264. ACM, 2005. ISBN 1-59593-009-4.

[11] Neda Moghim, Seyed Mostafa Safavi, and Massoud Reza Hashemi. Performance evaluation of a new end-point admission control algorithm in ngn with improved network utilization. *International Journal of Innovative Computing, Information and Control*, 6(7), 2010.

[12] Leonard Kleinrock. *Communication Nets; Stochastic Message Flow and Delay*. McGraw-Hill Book Company, New York, 1964. (Out of Print.) Reprinted by Dover Publications, 1972 and in 2007. Published in Russian, 1971, Published in Japanese, 1975.

[13] R. Gallager. A Minimum Delay Routing Algorithm Using Distributed Computation. *IEEE Transactions on Communications*, 25(1):73–85, 1977. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1093711.

[14] D.P. Bertsekas, E. Gafni, R.G. Gallager, Massachusetts Institute of Technology. Laboratory for Information, and Decision Systems. *Second Derivative Algorithms for Minimum Delay Distributed Routing in Networks*. LIDS-P. Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 1981. URL http://books.google.com/books?id=V5zpHAAACAAJ.

[15] Dimitri P. Bertsekas and Robert G. Gallager. *Data Networks, Second Edition*. Prentice Hall, 1992. ISBN 978-0-13-201674-2.

[16] Christos G. Cassandras, M. Vasmi Abidi, and Don Towsley. Distributed routing with on-line marginal delay estimation. *IEEE Transactions on Communications*, 38(3):348–359, 1990.

[17] A. Segall. The modeling of adaptive routing in data-communication networks. *Communications, IEEE Transactions on*, 25(1):85 – 95, jan 1977. ISSN 0090-6778.

[18] A. Segall, M. Sidi, Massachusetts Institute of Technology. Laboratory for Information, and Decision Systems. *A Failsafe Distributed Protocol for Minimum Delay Routing*. LIDS-P. Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 1981. URL http://books.google.com/books?id=2YOxtgAACAAJ.

[19] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[20] D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, 1999.

[21] S. Boyd, L. Xiao, and A. Mutapcic. Subgradient methods, 2003.

Touraj Shabanian received the B.Sc. and M.Sc. in electrical engineering from the Isfahan University of Technology, Isfahan, Iran, in 2001, 2005, and Ph.D. degrees in electrical engineering from the Isfahan University of Technology, Isfahan, Iran, in 2013. His research interests include QoS and traffic management in large-scale networking systems, network optimization and graph theory with emphasis on routing algorithms.

Massoud Reza Hashemi received his BSc and MSc from Isfahan University of Technology in 1986 and 1988 respectively, and his PhD from the University of Toronto in 1998, all in Electrical and Computer Engineering. From 1988 to 1993 he was with Isfahan University of Technology as a faculty member. From 1998 to 1999 he was a postdoctoral fellow at the University of Toronto. From 1999 to 2002 he was a founding member and lead systems architect of AcceLight Networks, where he developed some of the key system elements of a multi-terabit multiservice core switch. Since 2003 he has been with Isfahan University of Technology. His current research interests include Software Defined Networks, Data Centric Networks, DSM in Smart Grid, and Alert Correlation in Network Intrusion Detection.

Ahmad Askarian received the Bachelor degree in electrical engineering from the Shahed University, Tehran, Iran, in 2008 and the M.E. degree from the Isfahan University of Technology (IUT), Isfahan, Iran, in 2010. He is currently pursuing the doctoral degree at the University of Texas at Dallas (UTD). Since 2012, he is a research assistant in the Scalable Network Engineering Techniques Laboratory, UTD, Dallas, US. His research interests include network optimization and graph theory with emphasis on routing algorithm.

Behnaz Omoomi received the Ph.D. degree from University Putra Malaysia in 2001, the M.Sc. degree from Isfahan University of Technology in 1997, and the B.Sc. degree from Isfahan University of Technology in 1994, all in Mathematical Science. She is currently an Associate Professor in the Department of Mathematical Science at the Isfahan University of Technology. Her research interests include Graph Theory, Combinatorics, Graph Coloring and Game Theory.