

OLP-E: A New Algorithm for Setup and Optimizing the LSP Flows

Ali Parhizkari¹, Mohsen Jahanshahi², Hamid Maraqi³

¹Department of Computer Engineering, Buin Zahra Branch, Islamic Azad University, Buin Zahra, Iran ²Department of Computer Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran ³Department of Computer Engineering, Islamic Azad University of Qazvin, Qazvin, Iran

ABSTRACT

In MPLS (Multi-Protocol Label Switching) networks, packets of different flows may be categorized to several classes named Forward Equivalence Class (FEC) regarding their corresponding traffic characteristics. For each FEC, the appropriate QoS parameters as well as Service Level Agreement (SLA) can be defined individually. Various algorithms for FEC specification and label-switched path (LSP) planning have been proposed yet. In this article, a new on-line algorithm called End-to-End Optimized LSP Planning (OLP-E) for this problem is presented which is able to cope with unpredictable traffic situations and guarantees the pre-specified QoS requirements in differentiated service (DiffServ) high speed networks. Experimental results demonstrate that the OLP-E outperforms the Widest Shortest Path (WSP) in terms of resource utilization, achieved throughput, transmission cost, average packet delivery ratio, end-to-end delay, packet loss, and traffic distribution.

KEYWORDS: MPLS, LSP optimization, LSP setup

INTRODUCTION

Traffic engineering concentrates on how to assign network recourses to network flows [1]. There are three mappings in MPLS networks; first, packets-to-FECs, then FECs-to-LSPs, and finally LSPs to physical network configurations [2,3]. Focusing on two latter mappings, in this article, the main objective is to establish the logical LSPs within the network resource which involves determining both ends of LSPs, the corresponding paths, and capacity of each one. Clearly, OLP-E tries to specify the optimal network topology via LSP planning on a given network and traffic characteristics as well as QoS constraints namely, end-to-end delay, packet loss, and transmission cost. Aforementioned optimization problem has been proved to be NP-Hard due to exist of Binary decision variables [6,18]. In a case that the decision variables are real which specify load of the links, the LSP planning might be solved in polynomial time [4,17]. Considering the several QoS parameters conjointly as well as physical constraints in LSP planning is a must which is not paid attention in the reported researches yet (e.g. [7,8,9]).

The rest of this paper is organized as follows. Section 2, is dedicated to related works. In section 3, the proposed algorithm is presented. In Section 4, the proposed methods is analyzed. Section 5, concludes the article.

2. Related works

In this section, we survey the reported works;

The objective of the lowest cost algorithm (LC) can be stated as follows. Assuming two nodes V_s and V_d in network graph coupled with two constraints D (flow required bandwidth) and C (cost) it is needed to find a path between them such that $b(p) \ge D$ and $C(p) \le C$, where b(p) is the minimum residual bandwidth of all links forming the path and C(P) is the sum of all links' cost. The algorithm is online and designed for a single path.

Minimum Interference Routing Algorithm (MIRA)[10]is the most well-known online algorithm in this area. The scheme is one of the most prominent algorithms to analyze similar works. Nonetheless, MIRA suffers from many shortcomings that have been resolved in latter works. The rationale behind the proposed design in [11] is that establishment of a LSP between two nodes can reduce available bandwidth for other input/output nodes that caused by the interference. Therefore, the main idea in this algorithm is that if the paths with the more interference are ignored, then the bottleneck occurrence probability is reduced. To do this, in this method initially, the set of all critical links for every pairs of input/output nodes is founded. Afterwards, the weight of each link (C_{sd}) can be computed as (1):

(1)
$$w(l) = \sum_{(s,d):l \in \mathcal{C}_{sd}} a_{sd}$$

^{*}Corresponding Author: Mohsen Jahanshahi, Department of Computer Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran

where, a_{sd} is the weight of each LSP.

In [6], the idea is to reduce the reserved bandwidth which in turn leads to accept the more flows. Like [12], downside with this method is that traffic characteristics supposed to be known a priori. This information can be obtained in different time slots (e.g., $\{f^k(\tau), \tau = 1, 2, ..., \theta\}$) which represents the amount of variations of k_{th} input/output pairs in discrete times 1 to θ . In this way, the requests of different time slots can be better supported. To do this, for each remaining link *m*, the weight w_m can be computed as follows:

(2)
$$\chi_m = \max \left\{ f^k(\tau) + \mu_m(\tau) \right\} \quad \tau = 1, 2, ..., \ell$$

(3) $W_m = \frac{C_m}{C_m - \chi_m} + \varepsilon.$

Where $u_m(\tau)$ is consumed bandwidth on link *m* and $f^k(\tau)$ is the amount of request k_{th} pair in time slot τ . Defining of this weight function is contribution of this article. In the next step of this algorithm shortest path is computed using Dijkstra. Indeed, this algorithm can be in category of RISP algorithms in which weigh of each link is defined as inverse of residual bandwidth. Also, this algorithm has been devised for single path scenarios and well works with time-variable traffics. Its cost function (w_m) can be computed as follows that its objective is to minimize the following weighted function (4):

(4) Minimaize
$$c = \alpha \cdot C_{Max} + (1-\alpha) \cdot C_{mean}$$

In which C_{max} and c_{mean} are maximum and mean load of links.

In [13,19], a new method was proposed to dynamically tune network load among LSPs. In this method, it is assumed ingress and egress nodes are priory known. LSPs has been established a prior and the objective is to dynamically distribute traffic among them. In this method, probe packets are sent across the links and then their round trip time (RTT) are computed to estimate the load of links. In summarize, traffic engineering (TE) is composed of two phases; 'traffic monitoring' and 'traffic balancing among the paths'. If traffic load among the links varies too much, TE unit goes on with the traffic balancing phase. The aim of measurement and analyze unit is to obtain the statistics of each LSP (e.g., end-to-end delay or packet loss). This algorithm can be placed into hybrid (online / offline) category.

In what follows, the cons and pros of the reported methods are argued. One of the most shortcomings of the proposed methods is that just an aspect of the problem has been considered and the other issues ignored. The main problem of LC, SWP and WSP is that they don't consider the future requests. This problem using MIRA and the concept of interference has been relatively solved. However, MIRA suffers from following disadvantages; First, since MIRA takes interference into account for each input/output pair individually, its efficiency is reduced when several nodes use the common links. Second, its computational overhead ($O(n^3)$) for computation of flow maximization and $O(m^2)$ to computation of critical links in which *n* and *m* are the number of nodes and paths, respectively[7,8]. That is why a network with fewer nodes is used during performance evaluation of it [14].In MIRA, impact of interference is considered for the potential requests not for actual traffics. That is, the most of request are rejected with high probability. The shortcoming of Profile Based Routing (PBR) [25] is that traffic characteristic should be known priori [11]. This assumption in bursty traffic scenarios leads to arising many difficulties which will result in unreliable gathered information.

In [16], OLP has been proposed in which besides minimization of transmission cost as well as processing load on every node, throughput is increased. In this article, the scheme proposed in [16] is improved and moreoverlink/flow capacities are optimized.

In sum, the most of the reported algorithms establish the LSPs regardless of traffic specifications which in turn leads to unrealistic results. In other hand, among the surveyed works just [9] considered the end-to-end delay assuming M/M/1.

In this article, a new method to cope with the aforementioned problems will be proposed. Advantages of the proposed algorithm are discussed in details:

• Optimizing call admission control: since the non-atomic input flows can be disseminated to destination via different *k* paths, the acceptance rate of the input packets can be improved.

• Priority consideration: the proposed algorithm sorts and routes the input flows regarding their priority. The priority can be defined as a function of the following criteria:

• Atomic flows are more eligible to route across the network.

• The flows which are sensitive to delay, packet loss, and jitter have the more chance to forward throughout the network.

• The large flows may be delivered instantly.

Albeit other metrics can be defined.

3. OLP-E_E Algorithm: the proposed algorithm

In this research, it is interested to optimize the following three parameters; first, LSP establishment planning, second, LSP capacity, and finally traffic distribution. The last parameter determines how the traffic can be delivered between two Label Edge Routers (LER) using parallel paths. Hereafter, we call the aforementioned parameters as topology optimization, capacity optimization, and flow optimization which are dependent to each other. Configuration of LSP establishment as well as its capacity dependents on traffic distribution and vice versa [1].

3.1 Preliminaries

In this section, the required definitions and assumptions, input/output of the proposed algorithm, followed by the constraints of the problem are explained in details;

Un-directed graph G(V, L) represents a connected V-node network. Capacity of link $l \in L$ is defined as C_i . Routing the traffic across the logical paths named LSP is accomplished using the links' IDs which in turn it reduces the processing load in routers. λ_i^{od} corresponds with the computed bandwidth for flow *i* between source node *o* and destination node *d* in bit per second. λ_j^{od} also represents bandwidth of LSP *j* established between source o and destination d. Parameter $T_i = \begin{bmatrix} r_p^i & r_m^i \\ r_p^r & r_m^r \end{bmatrix} \in R^3$ determines the traffic

characteristic of flow *i*, in which r_n^i and r_m^i denote peak and mean rate of flow *i*, respectively. Also b^i denotes the average burst time of flow *i*. Parameter $R_i = (s_i, e_i, T_i, k_i)$ specifies source S_i and destination e_i , traffic characteristic_T, and distribution factor k of flow i, in order. Similarly, q is the number of requests. m and j are indexes of nodes and LSPs, respectively. P_i denotes the LSP of path j. Z_i is the number of hops in the LSP of path j. Finally, the input of the proposed algorithm are the following items:

Graph G(N,L) as the network topology.

 $T_i = \left[r_p^i r_m^i b^i \right]_{\text{is the traffic characteristic.}}$

 $R_i = (s_i, e_i, T_i, k_i)$ is flow specification for each request.

 D_{i}, ε_{i} is the QoS requirements, where D and ε are the maximum acceptable delay and packet loss, respectively.

 $d_{\rm w}$ is the cost of delivering a unit of data from source to destination through IP network without LSP switching. Therefore, D_j denotes the above cost while LSP is utilized and computed as (5).Where $0 < \alpha < 1$ and β is the processing cost in LER of LSP j.

(5) $D_i = \beta + \alpha d_v$

Also, the proposed algorithm outputs the following three items:

- Source and destination of each LSP
- P_{j}
- λ_i^{od}

And, finally the constraints of the problem can be stated as follows:

End-to-end LSP planning(6):

(6)
$$\forall i, \exists j \text{ so that } S_i = O_j \text{ and } e_i = d_j$$

Atomic / Non-atomic flow i; if $K_i = 1$, flow i is atomic and must be routed just across a single path. Otherwise, if $K_i^{>1}$ flow i can be forwarded via K_i paths.

 C_l denotes the physical bandwidth of link l.

$$(7) \sum_{i} \sum_{k=1}^{k_{i}} \lambda_{ik} \cdot \delta_{ikl} \leq C_{l}, \quad \forall i, k_{i})$$

Such that:

Parhizkari et al., 2012

$$\delta_{ikl} = \begin{cases} 1 & \text{if flow i travers through link l} \\ 0 & \text{otherwise} \end{cases}$$

3.2 Computation of equivalence capacity for every flow

To guarantee least end-to-end delay and packet loss it is desired to compute the equivalence capacity for each traffic. For this, several approaches have been proposed yet [5]. In what follows, a method named exponential traffic generator (On/Off)' for capacity computing is analyzed. Distribution function of a two state Fluid-Flow model with a buffer can be computed as follows:

(8)
$$z_{0} = \frac{(T_{b} + T_{idle})C - T_{b}r_{p}^{i}}{(C - r_{p}^{i})C}$$
(9)
$$F(x) = 1 - e^{z_{0}x}$$

Where *x* is the buffer size. Packet loss probability equals to the buffer override probability:

$$(10) P_{plp} = e^{-z_0 x}$$

Equivalence capacity to guarantee packet loss can be computed as follows:

(11)
$$C_{pbp}^{i} = \frac{r_{p}^{i}}{2} + \frac{(T_{b} + T_{idle})x}{2\ln(\varepsilon_{i})} + \sqrt{\left(\frac{r_{p}^{i}}{2} + \frac{(T_{b} + T_{idle})x}{2\ln(\varepsilon_{i})}\right)^{2} - \frac{T_{idle}r_{p}^{i}x}{\ln(\varepsilon_{i})}}$$

Using the above equations and Little equation, mean delay D and the size of buffer x can be inserted in the following Eq. (12).

(12)
$$D r_{p}^{i} \frac{T_{idle}}{T_{b} + T_{idle}} z_{0} + (z_{0}x - 1)e^{z_{0}x} + 1 = 0$$

With the same rationale it can be shown that if $\mathcal{E}_i < 10^{-2}$ then

$$(13) \quad (z_0 x - 1)e^{z_0 x} < 10^{-2}$$

Now, it can be well approximated:

(14)
$$D = -\frac{T_b + T_{idle}}{T_{idle} r_p^i z_0}$$

In the other hands, the equivalence capacity C_{Delay} of flow *i* for a given mean delay is:

(15)
$$C_{Delay}^{i} = \frac{r_{p}^{i}}{2} \left[(1 - T_{idle} \cdot D_{i}) + \sqrt{(1 - T_{idle} \cdot D_{i})^{2} + \frac{4 T_{idle}^{2} \cdot D_{i}}{T_{idle}^{2} + T_{b}}} \right]$$

In sum, λ_i can be computed as (16):

(16)
$$\lambda_i = Max(C'_{plp}, C'_{Delay})$$

3.3 Initial distribution of flows and its routing algorithm

As it is shown in Fig 1, the proposed routing algorithm uses the minimal number of routes to distribute non-unique flows.

Procedure Route-flow $(s_i, e_i, \lambda_i, k_i)$ { Compute shortest path with Dijkstra. Alg Between s_i, e_i and compute C_{\min} ; // C_{\min} is minimum residue capacity link of path. If $C_{\min} \ge \lambda_i$ Routing flow *I* in the founded path and exit; Else If $k_i = 1$ Reject Flow *I* and Exit; If $k_i > 1$ and $C_{\min} < \lambda_i$ { Routing flow *I* amount C_{\min} in the founded path; $\lambda^{-e}C_{\min}$ If $\lambda_i > 0$ Call Route-flow $(s_i, e_i, \lambda_i, k_i)$; } **Fig1. Pseudo code for flow initial routing**

3.4 Optimization constraints

In this sub-section the required constraints are discussed. The routing protocol tries to minimize the interference among different LSPs [14]. More LSPs more cost to manage the MPLS network. Thus, it is necessary to minimize the number of LSPs [11]. Albeit in this optimization problem other cost functions can be used. For instance, minimization of delivery cost [12], maximization of link utilization [13], and finally minimization of path delay [12].

The main objective of this research is to maximize network throughput but minimize overall transmission cost. For this, we define D_{od} as transmission cost of traffic r_m^i between source o and destination d.

Therefore, D_t is the overall transmission cost and can be computed as Eq. (17):

(17)
$$D_{t} = \sum_{od} D_{od} \left(r_{m}^{i} \right)$$

3.4.1 LSPs capacities optimization

LSP throughput optimization problem can be accomplished using periodically resource reallocations with regard to traffic demands [11,12]. In our model, every flow *i* can be distinguished with its

end-users, path ID, and equivalent capacity(λ_i). Network-Utilization optimization involves to several constraints (e.g., physical constrains) as well as some planning objectives as follows:

1. Physical constraints: the sum of all flows on a link must be less than or equal to the link capacity.

2.If R(od) is the set of all parallel path between source o and destination d, then $\sum_{i \in R(od)} \lambda_i = \lambda_{od}$.

3. $\lambda_i \geq 0, \lambda_i \geq 0$

Many works in the literature accomplished above problem via modeling in LP or NLP. But, these analytical frameworks regarding traffic modeling, utilization constrains, and computational complexity are undesired. Therefore, in what follows, a new algorithm is used to cope with the aforementioned constrains and can be applied in large scale networks distributively. Since the algorithm has high convergence speed and acceptable functionality, it is used in this work.

Given: \boldsymbol{r}_{p}^{i} , Burst _Time , idle _Time , \boldsymbol{r}_{m}^{j} , q. **INPUT**: a Graph G(N,L), α,β , All Request $R_i = \left| s_i e_i T_i k_i \right| = 1, 2, \dots, q$, Traffic particulars: $T_i = \left| r_p^i r_m^i b^i \right|$, QoS parameter D_i , ε_i OUTPUT: A Set Of Optimal LSP, V: G (N, L+V). **1**. For Flow *I* compute C_{plp}^{i} , C_{Delay}^{i} according to below equations: $1.1.^{T} idle = \frac{1}{idle - Time} C_{Delay}^{i} = \frac{r_{p}^{i}}{2} \left[(1 - T_{idle} \cdot D_{i}) + \sqrt{(1 - T_{idle} \cdot D_{i})^{2} + \frac{4 \cdot T_{idle}^{2} \cdot D_{i}}{T_{idle} + T_{b}}} \right]$ $C_{plp}^{i} = \frac{r_{p}^{i}}{2} + \frac{(T_{b} + T_{idle})x}{2\ln(\varepsilon_{i})} + \sqrt{\left(\frac{r_{p}^{i}}{2} + \frac{(T_{b} + T_{idle})x}{2\ln(\varepsilon_{i})}\right)^{2} - \frac{T_{idle}r_{p}^{i}x}{\ln(\varepsilon_{i})}}$ 1.2. Or: For Packet Loss Probability Guarantee: $\lambda_i = C_{plp}^i$ 1.3. Or: For Delay Guarantee: $\lambda_i = C_{Delay}^i$ 1.4. Or: For both Delay and Packet Loss Probability Guarantee: $\lambda_i = Max(C_{plp}^i, C_{Delay}^i)$ **2**. For Flows *i* //Routing flow *i* from *s* to *e* Call procedure Route-flow $(s_i, e_i, \lambda_i, k_i)$ **3**. Establish *LSP's* between all S_i , e_i and add it to set *V* and N++. //set LSP between S_i, e_i 4. Compute for each LSP: 4.1. $T_{j}(r_{m}^{i},\lambda_{j}) = \left| 1 - \frac{r_{m}^{i}(\lambda_{j})!}{\frac{\lambda_{j}(r_{m}^{i})^{k}}{\sum_{j}(r_{m}^{i})^{k}}} \right| * Z_{j}$ Where Z_j is number of Hob in path P_j : $R_j(\lambda_i) = r_m^i T_j(r_m^i, \lambda_j) w(r_m^j)$ B = 1// some of increase bandwidth 4.2. For all links $L \in \mathbf{P}_j$ $R_l = \min residual link capacity;$ 4.3. Do If $R_1 \ge B$ $W_j(\lambda_j, B) = \mathbf{R}_j(\lambda_j + B) - R_j(\lambda_j)$ If $(W_i(\lambda_i, B) > \varepsilon)$ { $R_{l} = B;$ $\lambda_i + = B$ } While $(R_l \ge B \text{ and } W_i(\lambda_i, B) > \varepsilon)$ 5. IF N>0 for each set of parallel LSP $\boldsymbol{R}_{od}^{j} = \sum_{\substack{alllink \\ P_{j}}} (1 - \boldsymbol{B}_{j}(\boldsymbol{r}_{m}^{j}, \boldsymbol{\lambda}_{j}))$ 5.1. Compute with Erlong Formula: $R_m^{od} = \frac{\sum_{set \ parallel \ LSP} R_{od}^j}{number \ of \ parallel \ LSP}$ 5.2. For each LSP j $R_{od}^{j} > R_{m}^{od}$ Increase r_m^j 5.3. For each LSP j $R_{od}^{j} < R_{m}^{od}$ Decrease r_m^j 6. for all LSP: Checked and Remove LSP which increase transmission cost. 7. Allocate minimal rest bandwidth in path LSP_{j} . Fig.2- pseudo code for OLP-E algorithm

However, other optimization algorithms can be used efficiently. optimal LSPs' capacity involves to the following factors:

- The LSPs compete with each other on the common physical resources.
- Given traffic to each LSP depends on the blocking probability of end-to-end flows which in turn depends on the links capacities forming the path.
- A link with minimal residual capacity may be bottleneck for LSP optimal capacity

The algorithm is composed of two phases; in the first phase a low boundary for LSP optimal capacity is computed across the nodes locally. In the second phase, the initial capacity increases to satisfy bandwidth constraints. Local optimization for each node m can be modeled as(18):

(18)
$$C_{j,opt,m} = f(\vec{\lambda}_J, \vec{B}_J)$$

Where \vec{B}_{j} , $\vec{\lambda}_{j}$ are end to end blocking probabilities and flow rate respectively. Unlike the reported algorithms, this algorithm improves the LSP capacity via computation of blocking probability iteratively. Advantage of this approach is to allocate the LSP capacities based on node level instead of network level. The stop condition can be either epsilon improvement in utilization is met or path residual capacity gets to zero.

In our model $W_j(\lambda_j, B)$ specifies how improvement in LSP utilization is met per B bps. It is assumed that capacity of other links is fixed the overall utilization is maximal if the computed utilization in the Erlang Eq(19,20) gets to maximum.

(19)
$$T_{j}(r_{m}^{j},\lambda_{j}) = \left(1 - \frac{r_{m}^{j}(\lambda_{j})!}{\sum_{k=0}^{\lambda_{j}}(r_{m}^{j})^{k}/k!}\right)^{*} Z_{j}$$

(20)
$$R_{j}(\lambda_{j}) = r_{m}^{j} \cdot T_{j}(r_{m}^{j},\lambda_{j}) \cdot w(r_{m}^{j})$$

Where $R_j(\lambda_j)$ is the LSP utilization function. Also, $W(r_m^j)$ is the weight of LSP. If $R_l \ge B$ and $R_l = C_{\min} \in P_j$ then $W_l(\lambda_j) = P_l(\lambda_j + B) = P_l(\lambda_j + B)$

(21)
$$W_j(\lambda_j, B) = R_j(\lambda_j + B) - R_j(\lambda_j)$$

If $W_j(\lambda_j, B) > \varepsilon$ then $R_i = R_i - B$ and $\lambda_j = \lambda_j + B$. This algorithm proceed until $R_i >= B_{\text{and}} W_j(\lambda_j, B) > \varepsilon$

3.4.2 Flow allocation optimization problem

Flow optimization is defined as a function of traffic distribution across parallel paths based on end to end blocking probability. In [15] an algorithm has been proposed in which utility functions can be specified and moreover their derivations should be compute which in turn makes the problem complicated. Instead, we apply a novel weighted pass probability function that is simple but effective. Weighted pass probability for the flow of each LSP is defined as follows (22):

(22)
$$R_{od}^{j} = \sum_{\substack{alllink\\p_{j}}} (1 - B_{j}(r_{m}^{j}, \lambda_{j}))$$

supposing that R_m^{od} is mean of weighted pass probability for all $i \in od$ and is computed as follows(23):

(23)
$$R_m^{od} = \frac{\sum_{set \ parallel \ LSP} R_{od}^j}{number \ of \ parallel \ LSP}$$

for each the flow i if $R_{od}^{j} > R_{m}^{od}$ rate r_{m}^{j} increased and otherwise it decreased.

4. Performance Evaluation

In order to evaluate the efficiency of the proposed algorithm, we used Exponential (on/off) as traffic generator. We also modeled the requests with two different random traffic scenarios named Uniform Traffic and Hub Traffic. Uniform traffic is used when all nodes can send traffic to each other while Hub Traffic is used when a node is going to send traffic to all other nodes. Then, achieved delay and packet loss have been compared with the results of WSP. Furthermore, we used two different topologies Abilene that is representative of a real network as well as a random generated graph. Equation (23)is used to compute the cost of LSP transmission.

$$(23) \quad D_{j} = \beta + \alpha . d_{v}$$

where d_v is hop-by-hop transmission cost. Also, $\alpha \boxtimes \in [0,1]$ and β is computational cost of both end nodes of a LSP.

4.1Experimental Results

In this section, the objective is to analyze performance of the proposed algorithms in terms of transmission cost, end to end delay, packet loss, packet delivery ratio, throughput, and load balancing. In all simulations, the term WSP denotes that routing is solely based on IP. Also, WSP_LSP is associated with routing through LSP establishment. In addition, the Base Alg. represents IP routing without LSP, and whenever this algorithm is used to LSP planning it is named LP Alg. Finally, we optimize the LP Alg. regarding LSP flows and capacities and name it as OLP-E.

Experiment 1

In this experiment, the aim is to compute transmission cost as a function of parameter β in single path scenario. As it can be seen in Fig.3, applying OLP-E method on the given network leads to achieve minimum transmission cost among all methods for all values of β . As it is expected, results also demonstrate that WSP and Base Alg. are independent of β .



Experiment 2

Fig 3: Transmission cost for single path scenario

In this experiment, the objective is to compute transmission cost while a number of parallel paths are considered. The result show that distribution of flows on different paths leads to more traffic acceptance rate (Fig. 4). In other hands, once more traffic is distributed on different paths, more links and nodes are involved in routing problem. That is why more traffic acceptance rate leads to more transmission cost. From the results, the proposed algorithms OLP-E and LP perform better than other comparable schemes.



Fig 4:Transmission cost for multi path scenario

Experiment 3

In this experiment, we are eager to analyze transmission cost as a function of given traffic. From the shown results in Fig. 5, while traffic is less than 45Mbps, OLP-E algorithm outperforms other schemes in terms of transmission cost. But when we increase traffic beyond this threshold, the OLP-E algorithm can accept more traffic. That is why transmission cost in OLP-E algorithm is slightly more than that of LP algorithm.



Fig 5:Transmission cost as a function of flow

Experiment 4

This experiment is manly conducted to evaluate average packet delivery ratio as a function of given traffic. As it can be observed in Fig. 6 the OLP-E mechanism works better than other schemes. The reason is that in the Base algorithm traffic efficiently is distributed over different paths. In this experiment, it is also desired to observe the network throughput. Figure7 depicts the obtained results. As it is expected, applying OLP-E algorithm leads to achieve more throughputs. As a hint, while traffic is increased it will result in more throughputs but less average packet delivery ratio. That is why average packet delivery ratio and throughput trends in Figs. 6 and 7 are descending and ascending, respectively.



Fig 6:Average packet delivery ratio for different algorithms

Parhizkari et al., 2012



Fig 7:Throughput for different algorithms

Experiment 5

In this experiment, we vary delay requirement from 0.0001 to 0.1 *ms* as well as packet loss probability from 0.0001 to 0.1. Results come in Fig 8.



Figure 8: Delay and packet loss guarantee in WSP and OLP-E

From Figure 8, OLP-E outperforms WSP in terms of different delay and packet loss requirements.

Experiment 6

In this experiment, the objective is to analysis processing load for different algorithms. The results demonstrate that LP and OLP-E algorithms in addition to reduce the processing load on each node, they can utilize the network resources efficiently.



5. Conclusion

LSP capacity and flow optimization problem has been proved to be NP-complete. Therefore, in this article, two novel algorithms namely, LP and OLP-E were proposed to optimize LSPs. It is noted that, in both proposed schemes, the minimum number of flow divisions is used to route the dividable flows. Furthermore, the proposed scheme scan efficiently utilize the network resources. OLP-E optimizes the LSPs' capacity as well as their flows.Clearly, it guarantees packet loss and end-to-end delay for different pre-specified levels. Furthermore, in OLP-E, just required bandwidth is dedicated to each flow, and consequently network bandwidth is utilized efficiently. That is why in OLP-E, the more flows are accepted.

REFERENCES

- G. Kaur, D. Kumar, "MPLS Technology on IP Backbone Network", International Journal of Computer Applications, Vol. 5, No.1, 2010
- [2] A. Riedl, "Optimized Routing Adaptation in IP Networks Utilizing OSPF and MPLS", HSRS 2004.
- [3] M.K.Girish, B. Zhou and J. Qiang, "Formulation of the traffic engineering problems in MPLS based IP networks", *Fifth IEEE Symposium on Computers and Communications ISCC*, 2000.
- [4] Y.WangandZh, Wang,"Explicit routing algorithms for Internet traffic engineering", *The 8th International Conference on Computer Communications and Networks*, 1999.
- [5] Z. Zhifei and Q. Zhengding, "A Novel Approach for Real-time Equivalent Bandwidth Estimation", International Conference on Communication Tecnology Proceedings, Vols I & II, PP. 1470-1474, 2000.
- [6] F. Ricciato and U. Monaco, "Routing demands with time-varying bandwidth profiles on a MPLS network", Computer Networks, Vol. 47, Issue 1, PP. 47-61, 2005
- [7] A. Halimi, K. Hendling, B.S. Halimi and K. Bengi,"Impact of different link cost metrics on the performance of routing algorithms for traffic-engineered MPLS networks", *The 8th International Conference on Communication Systems, ICCS*, 2002.
- [8] K.M. Elsayed, "HCASP: A hop-constrained adaptive shortest-path algorithm for routing bandwidth-guaranteed tunnels in MPLS networks", *Ninth International Symposium on Computers and Communications ISCC*, 2004.

- [9] S. Beker, D. Kofmanet and N. Puech, "Off Line Reduced Capacity Layout Design for MPLS Networks", *IEEE workshop on IP Operations and Management*, 2003.
- [10] T.V. Lakshman, K. Karand, M.Kodialam, "Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications", *IEEE Journal on elected Areas in Communications*, Vol. 18, No. 12, PP. 2566–2579, 2000
- [11] B. Wang,X. Su, "A new bandwidth guaranteed routing algorithm for MPLS traffic engineering", IEEE International Conference onCommunications, 2002.
- [12] S. Suri, M.Waldvogel, and P.Warkhede, "Profile-based routing: A new framework for MPLS traffic engineering", Washington University Computer Science Technical Report WUCS-00-21, July 2001.
- [13] A. Elwalid, C. Jin, S. Low and I. Widjaja, "MATE: MPLS Adaptive Traffic Engineering", in Proc. IEEE Infocom, 2001.
- [14] P. Aukia, M. Kodialam, P.Koppol, T.V. Lakshman, H.Sarin and B.Suter, "RATES: a server for MPLS traffic engineering", *IEEE Network*, Vol. 14, No. 2, 2000.
- [15] A.Farago, et al, "A New Degree of Freedom in ATM Network Dimensioning: optimizing the Logical Configuration", IEEE JSAC, Sept. 1995.
- [16] A. Parhizkari and S. Khorsandi, "OLP algorithm for resource management in DiffServe MPLS", 15thIranian Conference on Electrical and Electronics Engineering, in Farsi, 2006.
- [17] T. Usui and Y. Kitatsuji, "A study on traffic management cooperating with IMS in MPLS networks", *Telecommunication Systems*, 2011
- [18] J. Jelinskis and G.Rutka, "Fuzzy-CAC for LSP Setup in GMPLS Networks", *Electronics and Electrical Engineering*, 2010
- [19] L.Georgiadis, K.Floros, P.Georgatsos, S.Sartzetakis, Lexicographically, "optimal Balanced Networks", INFOCOM 2001.