A New Algorithm to Route Multimedia Traffic in Optical Networks with Improved Call Blocking Characteristics

[1]Mrs. R. Deepalakshmi, Assistant Professor,
[2]Dr. S. Rajaram, Associate Professor
[1]Department of Computer Science and Engineering,
Velammal College of Engineering and Technology,Madurai Tamil Nadu, India
e-mail:deepajeyaseelan@yahoo.com
[2]Department of Electronics and Communication Engineering,
Thiagarajar College of Engineering,Madurai Tamil Nadu, India
e-mail:rajaram_siva@tce.edu

Abstract

Multimedia applications present new challenges to the current networking technology. One of them is Quality of Service (QoS) requirement. Due to the increasing dynamics of traffic introduced by multimedia applications, dynamic and flexible QoS (Quality of Service) control is needed to ensure both QoS satisfaction and resource efficiency. Because integrated service networks are designed to support a wide range of traffic classes, which have different service requirements, developing metrics to evaluate them is complex. Optical networks provides Quality of Service (QoS), guarantees are also resilient to failures. Supporting QoS connections requires the existence of routing mechanisms that computes the QoS paths, where these paths satisfy the QoS constraints. Resilience to failures, on the other hand, is achieved by providing, each primary QoS path, a set of alternative QoS paths, upon a failure of either a link or a node. We aim at to minimize the total bandwidth reserved on the backup edges. The above objectives, coupled with the need to minimize the global use of network resources, imply that the cost of both the primary path and the restoration topology should be a major consideration of the routing process. It turns out that the widely used approach of disjoint primary, restoration paths is not an optimal strategy. Hence, the proposed Multimedia Traffic Routing Algorithm for Optical Networks with improved call blocking Characteristics construct a topology, and this topology protects a portion of the primary QoS path. This approach guarantees to find a topology with optimal cost which satisfies the QoS constraints.

Keywords: *optical network, routing, Schemes, Restoration Schemes, QoS Models, QoS Constraints, Shortest path, Primary path and Restoration Topology.*

1. Introduction

OPTICAL networks are promising for serving as the backbone networks of the next-generation Internet due to their potential ability to meet the ever-increasing demands of high bandwidth. Internet applications can differ with respect to many characteristics, such as burst, high data transaction Bandwidth, multi-granularity of traffic flows, and high resilience. For example, data-intensive distributed applications may require moving different amounts of data from Kilobytes to Terabytes or even Petabytes among multiple sites. These applications may require different quality of service (QoS). Optical networking technologies offering enormous transmission bandwidth and advanced capabilities are expected to play an important role in creating an efficient infrastructure for supporting internet applications. In current optical network, the capacity of each light path is usually restricted to the granularities such as STM-N, N=1, 4, 16, 64. For high data transport bandwidth requires, single light path couldn't satisfied the demand. So we need to setup multi-paths and split the traffic to these paths to implement data transport within the limited scope of time. It has been proved that a reasonable and efficient multi-path routing could increase the availability and decrease the rejection ratio of the whole network [8]. It is possible that the overall utilization efficiency of the network resource can be further improved through an appropriate traffic-distributing algorithm in multiple paths and an appropriate protected path provisioning [6].

This raises the research questions of how to decide the numbers of working paths and protection paths, how to design an adaptive algorithm to distribute the traffic onto the available multiple paths, and how to allocate the bandwidth resource to protection path to satisfy the QoS under a dynamic traffic requests.

A number of routing algorithms have been proposed in traditional networks [3] - [5]. However, all these algorithms addressed the problem of TCP packets for IP networks, they do not consider important issues involved in optical technologies, and therefore do not meet the requirements and exigencies of optical network. Reference [2] investigated the problem of reliable multi-path routing for high-capacity backbone optical networks. It proposed and investigated the characteristics of effective multi-path bandwidth as a metric to provision a connection to multiple paths while satisfying its availability requirements. It developed two efficient heuristics for solving the multi-path routing problem. The results showed that the heuristics perform significantly better (decreasing blocking probability) than conventional single-path routing approaches for high bandwidth connections.

This paper focus on optical network supporting application in which large and long-holding requests count more. And as a practical application, the protection and restoration will be taken into consideration. The objective of this paper is to develop a Routing and Restoration Algorithm for Optical Networks with enhanced performance for splitting traffic and to configure the protection path to satisfy the users' availability requirements. More precisely, under a dynamic traffic model, we investigate the problem of distributing the traffic among the available k-paths and configuring the bandwidth of protection path in order to maximize the efficiency of network resources consumption or minimize the congestion on the network resources, while the users' availability requirements are satisfied. The remainder of the paper is structured as follows: Section II states the definitions and the objectives. Section III describes our algorithms of provision the bandwidth and configuring protection paths. Section IV proposes the results and comprises the performances between single-path routing and multi-path routing for optical multimedia application. Finally, the conclusion of this paper is given.

2. Definitions and The Objectives

In this paper, we develop dynamic algorithms for routing in optical networks. In order to take advantage of the multiplexing gain, we connect optical endpoints using a tree structure (instead of independent point-to-point paths between optical endpoints). An optical network tree has several benefits, as listed below.

1) **Sharing of Bandwidth Reservation**. A single bandwidth reservation on a link of the tree can be shared by the entire traffic between the two sets of optical endpoints connected by the link. Thus, the bandwidth reserved on the link only needs to accommodate the aggregate traffic between the two sets of optical endpoints.

2) **Scalability**. For a large number of optical endpoints, a tree structure scales better than point-to-point paths between all optical endpoint pairs.

3) **Simplicity of Routing**. The structural simplicity of trees ensures that if MPLS [3] is used for setting up paths between optical endpoints, then fewer labels are required and label stacks on packets are not as deep. With MPLS, between each pair of optical endpoints, label switched paths (LSPs) along links of the tree can be set up using the explicit routing capabilities of either RSVP-TE or CR-LDP[3].

4) **Ease of Restoration**. Trees also simplify restoration of paths in case of link failures, since all paths traversing a failed link can be restored as a single group, instead of each path being restored separately.

We develop algorithms for computing *optimal optical* trees, that is, trees for which the amount of total bandwidth reserved on edges of the tree is minimum. Initially, we assume that network links have infinite capacity, and show that even for this simple scenario, the general problem of computing the optimal optical tree is NP-hard. However, for the special case when the incoming and outgoing bandwidths for each optical endpoint are equal, we are able to devise a *breadth-first search* (STEINER TREE) algorithm for computing the optimal tree whose time complexity depends on the number of links and nodes in the network, respectively.

We present a dynamic integer programming formulation for the general optical tree computation problem (that is, when incoming and outgoing bandwidths for each optical endpoint are arbitrary) and develop an algorithm, we also extend our proposed algorithms for computing optical trees to the case when network links have capacity constraints. We show that in the presence of link capacity constraints, computing the optimal optical tree is NP-hard even when incoming and outgoing bandwidths of each endpoint are equal. Further, we also show that computing an approximate solution that is within a constant factor of the optimum is as difficult as computing the optimal optical tree itself However, our experimental results with synthetic network graphs indicate that the optical trees constructed by our proposed algorithms require dramatically less bandwidth to be reserved (in many instances, by more than a factor of 2) compared to *Steiner Tree*. Further, among the two algorithms, the Multimedia Traffic Routing algorithm performs the best, reserving less bandwidth than either the Steiner Tree algorithms over a wide range of parameter values.

3. Multimedia Traffic Routing Algorithm (Mtra)

INPUT:A optical Network G=(N,L),optical network units $O_{XC}=(O_{XC1}, O_{XC2},\ldots, O_{XCp})\Sigma$ N, Residual Bandwidth constraints B on L and a VPN setup request $Vr=(r_1, r_2,\ldots, r_p)$

OUTPUT: A minimum cost optical tree OT_{MC} for V_{ri} on which all leaf nodes are optical network units O_{XCp} with $r_i > 0$.

The optical network backbone is modeled by a graph G= (N, L), where N and L are the set of optical endpoints and the set of links, respectively. Let n and m denote the cardinality of N and L, respectively. Let B be the residual bandwidth of links on L and the amount of residual bandwidth on link l (l€L) is denoted by B(l). A subset $O_{XC}=(O_{XC1}, O_{XC2},\ldots, O_{XCp})$of N (Ap□N) is the set of optical network units. Each endpoint e_i of a optical gains access to optical service by connecting to a specific optical network unit OXCp in OXC. In other words, for each endpoint of a optical network, there is a corresponding optical network unit in OXC.

The elliptic region in figure 1 is an example of optical network backbone G. The round regions (A to G) inside G are optical network units in N. The solid lines between two optical network units are links in L. The number beside each link is the amount of residual bandwidth on it (B (l) =5 for all l€L in this figure). The round regions (1, 2 and 3) outside G are optical network units (e_1, e_2 and e_3, respectively, in our notation) of an optical networks, which gain access to optical network service via optical network units in O_{XCp}. The dotted lines labeled as path i, j is the transmission path for optical network traffic between e_i and e_j.

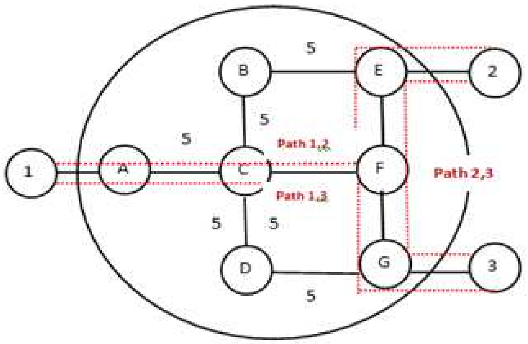

Figure.1. Optical Backbone Network

i) Optical Setup Request Modeling:

The demands for optical network service of customers are described by optical network setup requests. In this paper consider that the bandwidth requirement of each endpoint ej is symmetric. Let b(ej) denotes the bandwidth requirement of endpoint e_j, and Max_r denote the maximum bandwidth guarantee provided by optical network unit. The i^{th} setup request, denoted by v_{ri} .Each v_{ri} is represented by a p-tuple vector $(r_1, r_2, \ldots, r_p)$, where p is the cardinality of optical network units(OXCs). The number of nonzero elements in v_{ri} represents the number of network unit contained in the corresponding optical networks. The value of j^{th} element $r_1$of v_{ri} represents the bandwidth requirement of endpoint e_j.

ii) Establishment Problem

The *algorithm* defined in this paper which mainly considers on-line establishment of bandwidth guaranteed point-to-point OXCs. The optical network setup requests of customers are sent to ORS (Optical Request Server).

In this paper consider the situation where (a) optical setup requests arrive one by one independently and (b) information about future optical setup requests is unknown. This information includes the number of future optical setup requests, the number of OXCs contained in each optical setup request, and the bandwidth requirement of each endpoint. The service provider must process each setup request in an on-line manner, the off-line model in not suitable. Upon receiving a setup request vri, the service provider triggers the routing algorithm to establish a corresponding connection. The routing algorithm performs this task by first choosing a path between each endpoint pair and then allocating bandwidth on each link on the paths. If there is not enough residual bandwidth on the link when the bandwidth is being allocated, vri will be rejected. In this paper the rejection ratio is taken as the performance metric to compare different optical network routing algorithms. The *rejection ratio* is defined as:

$$\text{Rejection ratio} = \frac{\text{Number of requests rejected}}{\text{Total number of request received}}$$

The optimization goal of provisioning algorithms is to minimize the rejection ratio, which in turn will maximize the number of requests successfully established on the network backbone. Although the main performance metric here focus on rejection ratio.

Other important performance metrics (eg: link utilization and bandwidth allocation efficiency) are also investigated in the experimental simulations. The *ORS* also keeps the complete link state topology database and is responsible for finding an explicit path for each endpoint pair of a OXCs. Then the explicit paths can be setup using a signaling protocol such as RSVP. For computing the explicit paths, the *ORS* needs to know the current network topology and link residual bandwidth. We assume that there exists a link state routing protocol for information acquisition.

The Factors Influencing Rejection Ratio:

In this case, the links of the network backbone have a finite amount of bandwidth and the service provider needs to establish multiple connections on the network backbone on-line. The two most important factors influencing the rejection ratio achieved by routing algorithms are: (1) Bandwidth allocation efficiency (2) Load balance mechanism that considers the amount of residual bandwidth on links. Routing algorithms must take the residual bandwidth of links into account, and avoid using links that are thinly spread. This will balance the load on links of G and reduce rejection ratio.

Multimedia Traffic Routing Algorithm (MTRA)

To alleviate the drawbacks of (a) inefficiency on bandwidth allocation, and (b) disregarding the amount of residual bandwidth for links selection , we propose a new provisioning algorithm called the *Multimedia Traffic Routing Algorithm* **(**MTRA).The *Steiner Tree* and *MTRA* both are tree-based (i.e., they establish a connection based on tree topology (*optical tree*)). While *Steiner Tree* has excellent bandwidth allocation efficiency, it does not consider maximizing the accommodation of on-line requests. On the contrary, *MTRA* considers both bandwidth allocation efficiency and accommodation of on-line requests by achieving balance of link residual bandwidths.

The major difference between *Steiner Tree* and *MTRA* is that the cost function they defined for *optical tree* selection. Let *T* be a *optical tree* consisting of *k* links. The cost functions of *Steiner Tree* and *MTRA* are defined as following:

$$\text{Cost}_{\text{MTRA}}\,(\text{T}) = \sum_{l \leq x \leq k} \frac{RS(lx)}{B(lx)}, \quad \text{and}$$

$$\text{Cost}_{\text{STEINERTREE}}\,(\text{T}) = \sum_{l \leq x \leq k} RS\,(lx)$$

where $RS(l_x)$ and $B(l_x)$ represent the amount of bandwidth allocation needed and the amount of residual bandwidth on the xth link, l_x, respectively. The cost function of *MTRA* is derived by the cost function defined in the routing algorithms proposed in [6,7] for route selection. When processing a request, *MTRA* tries to find an *optical tree* that minimizes the cost function defined above. It is clear the additional cost for using a link l_x in building a *optical tree* is proportional to the value of $RS(l_x)$ and is reciprocal to the value of $B(l_x)$. Therefore, *MTRA* tries to find a *optical tree* with links of abundant residual bandwidths and low overall bandwidth allocation. As a result, *MTRA* can satisfy both bandwidth allocation efficiency and balance of residual bandwidths. The pseudo code of *MTRA is* described in Table 1**.**

Table 1. Pseudo code for *MTRA*

Dynamic Cost Optimized Provisioning Algorithm(DCOPA)
Input: A Network graph *G*=(*N*,*L*), Optical Cross Connects (OXCs) AR=(ar1,ar2,…,arp)□N, residual bandwidth constraints B on L, and a optical connection setup request vri =(r1,r2,…,rp).
Output: A minimum cost optical tree OT_{MC} for vr_i, on which all leaf nodes are Optical Cross Connects ar_j with r_j>0.

```
$OT_{MC}$ :=Ø;
For each v Є N
    {
        Tv:= BFS_Tree(G,v);
        PTv:=Prune_Tree($T_v$, $v_{ri}$);
        Compute_RS($PT_v$, $v_{ri}$);
        if(Cost($PT_v$)<Cost($OT_{MC}$) ) $OT_{MC}$:= $PT_v$;
    }
if (Cost($OT_{MC}$) = ∞)
{ Reject($v_{ri}$); Return Ø;}
else   {
For each link lx Є $OT_{MC}$ {B($l_x$):= B(lx)-RS($l_x$);}
        Accept($v_{ri}$); Return($OT_{MC}$);
 }
```

Given a network graph G consisting of n nodes, to process a optical setup request vr_i, MTRA iterates totally n times, once for each v Є N. In each iteration, MTRA first finds a candidate VPN tree PT_v rooted at v for vr_i, and then computes the amount of bandwidth needed to be allocated to each link l_x of PT_v. Finally the cost value associated with PT_v can be computed. After finding all PT_v (v Є N), if there is not any PTv (v Є N) on which all links have enough residual bandwidth for allocation, MTRA will reject vr_i. In the case of accepting vr_i, MTRA will return the optical tree with the minimum cost value among all PT_v (v Є N) for vr_i, which is denoted by OT_{MC}. In addition, MTRA then allocates bandwidth to each link l_x of OT_{MC} by performing $B(l_x)$:= $B(l_x)$-$RS(l_x)$.

To find a candidate optical tree PT_v rooted at v, MTRA first find a BFS tree (breadth first search tree [18]) T_v rooted at v (by calling Function BFS_Tree). T_v contains all nodes in G and in addition, T_v may contain nodes that are not VPN access routers used in vr_i as leaf nodes. Therefore, MTRA prunes T_v and obtains a candidate optical tree PT_v, on which all leave nodes are optical cross connects used in vr_i (by calling Function Prune_Tree).

MTRA computes the amount of bandwidth needed for each link lx of a optical tree T according to the bandwidth requirement information in vri (by calling Function Compute_RS in Table 2). To compute the value of $RS(l_x)$ (l_x Є T), we first remove lx from T which partitions the optical tree into two subtrees Tx a and Tx b. Let BR_ Tx ^{a} and BR_ Tx ^{b} denote the accumulated bandwidth requirement of the OXCs (endpoints) on Tx ^{a} and Tx ^{b}, respectively. Then RS(lx) is determined by the minimum value of BR_ Tx a and BR_ Tx b. For more details about computing the $RS(l_x)$ value for each l_x on a optical tree.

Given a optical tree T, in a normal case, the function Cost of MTRA returns the cost value computed by the cost function defined previously. However, where T is null (Ø), or there are links on T that do not have enough bandwidth for allocation, the function Cost will return ∞.The time complexity of each iteration in MTRA is O(m), which is determined by the function BFS_Tree. To process a request, a total of n iterations are required. So, It is clear that the time complexity of MTRA for processing a request is O(mn).

Table 2. Pseudo code for *Compute_RS*.

Function *Compute_RS*(*T*, *vr_i*)

Let *lx* be the *x*th link on *T*.

Let $RS(l_x)$ be the amount of bandwidth allocation needed on l_x with respect to the bandwidth requirement specified in vr_i.

Let T_x^a and T_x^b be the two subtrees obtained by remove l_x from *T*.

```
for (each $l_x$ in $T$)
{
    Initialize two variable $BR\_ T_x^a$, $BR\_ T_x^b$ to value 0;
    For (each element $r_j \neq 0$ $(1 \leq j \leq p)$ of $vr_i$)
      {
         if( $a_{rj}$ Є Tx ) then add rj to BR_ Tx a
        else add $r_j$ to $BR\_ T_x^b$
      }
 $RS(l_x) := min(BR\_ T_x^a, BR\_ T_x^b)$;
}
```

Fig.2. Pseudo code to compute _RS

We now consider Scenario disregarding the amount of residual bandwidths on links in BSF routing algorithm results in a higher rejection ratio. Initially, B(l)=5 for all l in L. The sketch of G, after accepting vr_1, is shown in Figure 2. The number beside each link in *G* is its residual bandwidth after accepting vr_1. The dotted lines form the minimum cost *optical tree* OT_{MC} that *DCOPA* will output for vr_1.

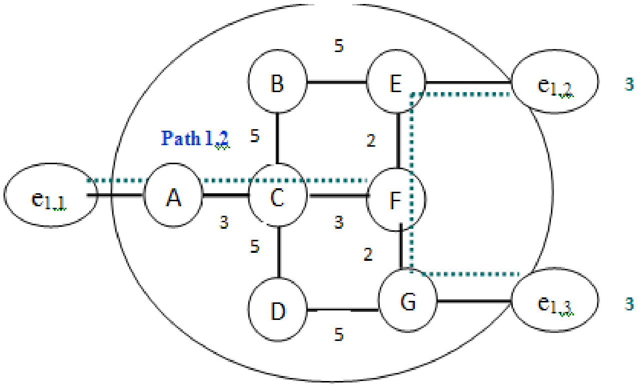

Figure 2: A Sketch of G after processing vr_1

After accepting vr_1, *MTRA* then processes vr_2. Each candidate *optical tree* PT_v ($v \in N$) for vr_2 considered by *MTRA* is shown in Figure 3. We can find four different types of candidate *optical tree* for vr_2. Note that PT_A, PT_B, PT_C and PT_D are identical. The number beside each link of PT_v ($v \in N$) is the amount of bandwidth that needs to be allocated to it. The cost value associated with each PT_v ($v \in N$) is:

$$\mathrm{Cost}(PT_A) = \mathrm{Cost}(PT_B) = \mathrm{Cost}(PT_C) = \mathrm{Cost}(PT_D) =$$

$$\frac{RS(l_{A,C})}{B(l_{A,C})} + \frac{RS(l_{B,C})}{B(l_{B,C})} + \frac{RS(l_{C,D})}{B(l_{C,D})} + \frac{RS(l_{B,E})}{B(l_{B,E})} + \frac{RS(l_{D,G})}{B(l_{D,G})} =$$

$$\frac{3}{3} + \frac{3}{5} + \frac{3}{5} + \frac{3}{5} + \frac{3}{5} = 3.4$$

and $\mathrm{Cost}(PT_E) = \mathrm{Cost}(PT_F) = \mathrm{Cost}(PT_G) = \infty$

It is clear that *MTRA* will return PT_A for vr_2 (vr_2 is accepted by *MTRA*). Hence the *rejection ratio* achieved by *MTRA* in *Scenario* is 0%.

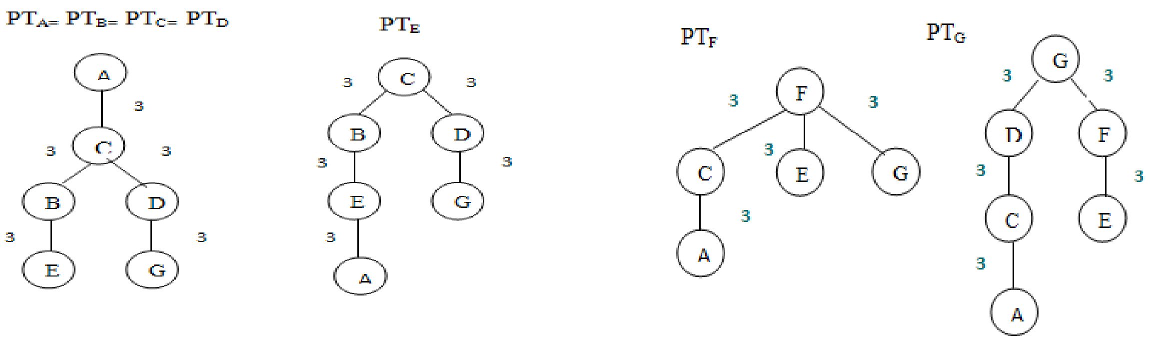

Figure 3: Candidate optical trees considered for vr$_2$

4. Comparison and Analysis

i) Simulation Environment

To evaluate the performance of *MTRA* we set up a *network model* for optical network provisioning algorithms simulator (*OPNET*). We implemented all components, except topology generator, in c programming language. The topology generator of *OPNET randomly* generates the network backbone *G* administrated by the NSP. Because Brite [11] has been widely used in a lot of research literature to generate random network topologies, we also adopt it as the topology generator in *OPNET*. We generate randomly a connected network graph *G* by assigning proper values in the configuration file used by Brite. We have implemented two provisioning algorithms in the simulation: (1) *MTRA*(2) *Steiner Tree routing*.

The optical network setup requests generator randomly generates a set of optical setup requests according to the given parameters *K*, *p*, and *Maxr*. The request set contains *K* requests. The number of endpoints contained in each OXCs is generated randomly between 2 and *p*, and the bandwidth requirement $b(e_i)$ for each endpoint e_i is generated randomly between 1 and *Maxr*.

ii) Performance Results

In this subsection, we describe the simulations. The simulation compares the rejection ratio achieved by various provisioning algorithms implemented and the last simulation investigates the bandwidth allocation efficiency of MTRA.

iii) Experimental Results

(Designed and developed MTRA Vs Steiner Tree Routing)

We compared the routing cost (that is, the total bandwidth reserved on links of the optical tree) and the running times of the algorithms for the symmetric as well as the asymmetric bandwidth models. In the study, we examined the effect of varying the following two parameters on routing cost: 1) network size 2) number of OXCs. Most of the plots in the following subsections were generated by running each experiment five times (with different random networks) and using the average of the cost/execution times for the five repetitions as the final result.

Simulation 1: (Network Size)

Figure.4 and 5 depicts the routing cost of the Steiner tree and MTRA algorithms as the number of network nodes is increased from 10 to 100. OXCs are assigned equal incoming and outgoing bandwidths and unequal incoming and outgoing bandwidth the number of OXCs set to 10% of the network size. The MTRA algorithm is provably optimal for the symmetric case. Further, unlike the Steiner tree algorithm which is oblivious to the bandwidths of endpoints, the Steiner tree algorithm does take into account the bandwidth requirements for OXCs. As a result, it outperforms by almost a factor of 2 for a wide range of node values.

Table 3 Parameter configuration of Simulation 1 (Symmetric)

Number of OXCs	MTRA-cost	Steiner - cost
10	2050	3000
30	2500	4400
50	4500	7000
75	5500	9000
100	6500	10000

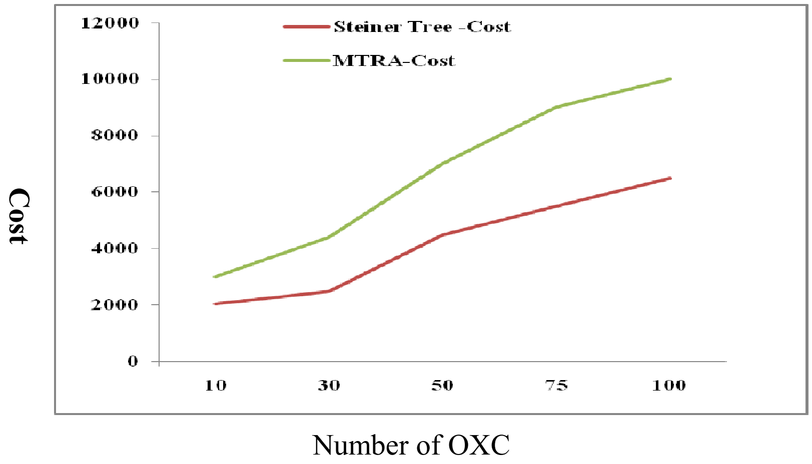

Figure. 4. Effect of Number of OXC in performance of Algorithm (Symmetric)

Table 4 Parameter configuration of Simulation 1 (Asymmetric)

Number of OXCs	MTRA-cost	Steiner -cost
10	2050	2500
30	2100	3000
50	4000	5000
75	4500	6000
100	5500	7500

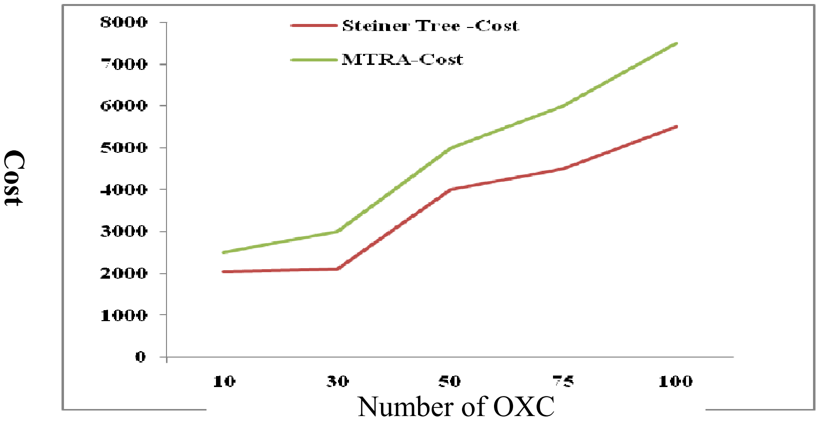

Figure. 5. Effect of Number of OXC in performance of Algorithm (Asymmetric)

Simulation 2: (Rejection Ratio)

We conduct 15 runs with various number of topology, in each of which, 100 requests are randomly generated. The rejection ratios achieved by the two routing algorithms are (see Figure 6). The x-axis represents the number of OXCs. and the y-axis represents the rejection ratio and average link utilization achieved by each routing algorithm in each run.

Table 5 Parameter configuration of Simulation 1

Number of OXCs	MTRA-Rejection Ratio	Steiner -Rejection Ratio
10	2	6
30	1	5
50	1	5
75	1	4
100	0	4

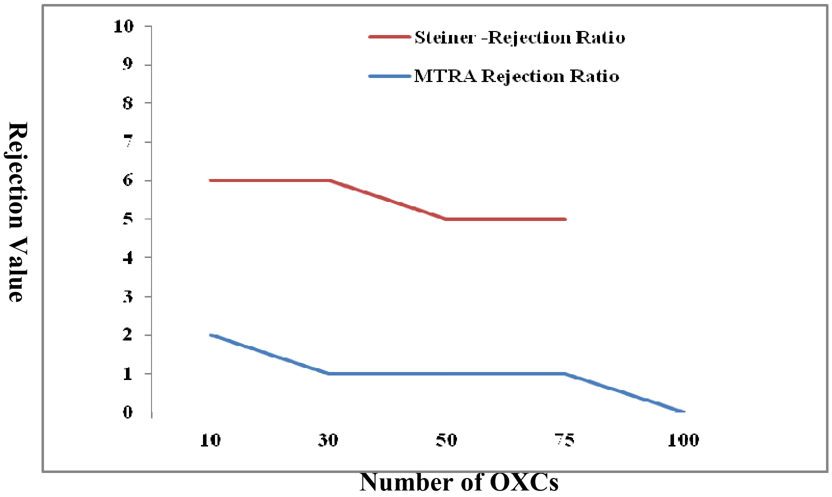

Figure. 6. Effect of Rejection Ratio

We can see that the rejection ratio achieved by MTRA is much less than that achieved by Steiner tree routing.

5. Conclusion

In this research work, the designed novel algorithms for routing optical network MTRA connected OXC using a tree structure and attempted to optimize the total bandwidth reserved on edges of the optical tree. The algorithm showed that even for the simple scenario in which network links are assumed to have infinite capacity, the general problem of computing the optimal tree is NP-hard. However, for the special case when the incoming and outgoing bandwidths for each OXC are equal, MTRA proposed a breadth-first search algorithm for computing the optimal tree. According the simulation results MTRA can indeed reduce the rejection ratio effectively.

We can think of extending the research work for using Alternative Cost Functions as the Performance Metric for Asymmetric case of bandwidth allocation in optical networks.

References

[1] Sudipta Sengupta and Ramu Ramamurthy, “From Network Design to Dynamic Provisioning and Restoration in

Optical Cross-Connect Mesh Networks: An Architectural and Algorithmic Overview”, IEEE Network, July/August 2001.

[2] L. A. Cox, Jr., J. R. Sanchez, and L. Lu, “Cost Savings from Optimized Packing and Grooming of Optical Circuits: Mesh vs. Ring Comparisons,” Opt.Networks, May–June 2001, pp. 72–90.

[3] R. Ramamurthy et al., “Capacity Performance of Dynamic Provisioning in Optical Networks,” J. Lightwave Tech., vol. 19, no. 1, Jan. 2004, pp. 40–48.

[4] D. Awduche and Y. Rekhter, “Multiprotocol Lambda Switching: Combining MPLS Traffic Engineering Control with Optical Crossconnects,” IEEE Commun.Mag., vol. 39, no. 3, Mar. 2003.

[5] Kefei Wang et at., “Multicast Routing in 40Gb/s Heterogeneous Optical Networks”, 0-7803-8938-7/05/ IEEE, 2005.

[6] Jun Gu et at., “Routing algorithm for multicast under multi-tree model in optical networks”, Elsevier Science Publishers Ltd, Volume 314, Issue 1 ,February 2004.

[7] Vladica Tintor and Jovan Radunović , “Distributed Dijkstra sparse placement routing algorithm for translucent optical networks”, Photonic Network Communications Volume 18, Number 1, 55-64, 2008.

[8] Meeyoung Cha et al., “Integer Linear Programs for Routing and Protection Problems in Optical Networks”, Photonic Network Communications , 2009.

[9] S. Ganguly., “Waveband Routing and Merging in Hybrid Optical Networks”, IEEE Communications Society, 2006.

[10] Ralf H¨ulsermann et al., “Dynamic Routing Algorithms in Transparent optical networks”, IEEE Commun. Mag., 37:67–73, 2009.

[11] A. Medina, A. Lakhina, I. Matta and J. Byers, BRITE: Universal Topology Generation from a User’s Perspective, www.cs.bu.edu/brite/publications/usermanual.pdf, April 2001.