Virtual Topology Expansion Method For Dynamic Traffic Grooming In WDM Mesh Networks

M.kalaiyarasi and S.Alagumani

Abstract- This paper deals with virtual topology expansion method (VTE) for dynamic traffic grooming in wavelength division multiplexing networks. Dynamic grooming deals with requests for wavelength allocation based on a dynamic pattern of arrivals in contrast to the situation of static grooming in which the pattern of arrivals must be previously known. Solutions for dynamic grooming typically involve the construction of an auxiliary graph for deciding on the routing and wavelength assignment. An auxiliary graph can represent the network partially leading to scalable solutions. However, a previous proposal employing a reduced auxiliary graph produces blocking that is not fairly distributed among calls. A novel algorithm is thus proposed in this paper for achieving fairness in relation to the blocking of calls. This algorithm uses alternative routing rather than shortest-path routing as well as auxiliary graphs based on the virtual topology. Results give a low cost for path establishment by virtual topology expansion (VTE) algorithm than by previously proposed zone based algorithms (ZBA) algorithms.

Keywords: Grooming, static grooming, dynamic traffic grooming

1. INTRODUCTION

Now we are in the twenty first century, the era of 'Information technology' .There is no doubt that information technology has had an exponential growth through the modern telecommunication systems. Particularly, optical fiber communication plays a vital role in the development of high quality and high-speed telecommunication systems. Today, optical fibers are not only used in telecommunication links but also used in the Internet and local area networks (LAN) to achieve high signaling rates.

2. TRAFFIC GROOMING

Traffic grooming is the process of grouping many small tele-communications flows into larger units, which can be processed as single entities. For example, in a network using both time-division multiplexing (TDM) and wavelength-division multiplexing (WDM), two flows which are destined for a common node can be placed on the same wavelength, allowing them to be dropped by a single optical add-drop multiplexer. Often the objective of grooming is minimizing the cost of the network. The cost of line terminating equipment (LTE) (also called add/drop multiplexers or ADMs) is the most dominant component in an optical WDM network's cost. Thus grooming typically involves minimizing the usage of ADMs.

A large body of research on traffic grooming for synchronous optical network (SONET) ring networks has accumulated, since the technology is widely used. However, ring topology does not scale well to accommodate the growth of traffic. Moreover, the topology of future networks will probably consist of irregular meshes, and there has thus been a great deal of interest in solving dynamic grooming problems for mesh topologies.

2.1 TYPES OF TRAFFIC GROOMMING

There are two types of traffic grooming,
(i) Static grooming
(ii) Dynamic grooming

In Static traffic grooming, the pattern of arrivals and set of connections must be previously known.

In dynamic traffic grooming connection requests are presented one at time and deals with requests for wavelength allocation based on dynamic pattern of arrivals.[2]

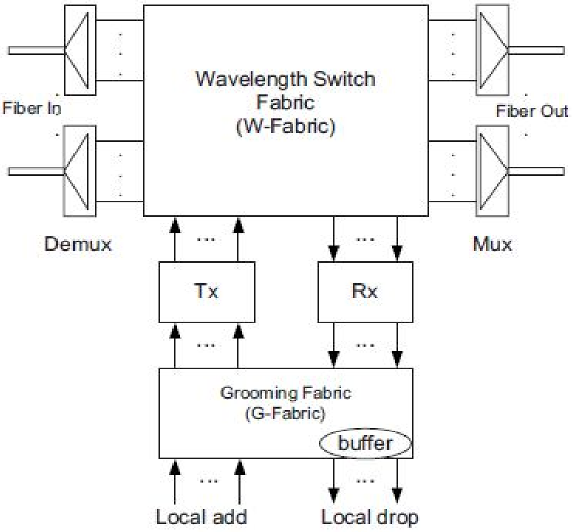


Fig.2.1 Grooming node architecture

3.1 WAVELENGTH DIVISION MULTIPLEXING

One of the most promising concepts for high capacity communication systems is wavelength division multiplexing (WDM). Each communication channel is allocated to a different frequency and multiplexed onto a single fiber. At the destination wavelengths are spatially separated to different receiver locations. In this configuration the high carrier bandwidth is utilized to a greater extent to transmit multiple optical signals through a single optical fiber. [9]

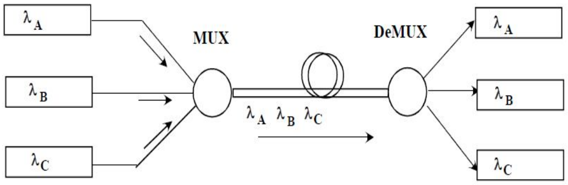


Fig.3.1 Wavelength division multiplexing

The use of WDM can multiply the effective bandwidth of a fiber optic communications system by a large factor. But its cost must be weighed against the alternative of using multiple fibers bundled into a cable.

3.2 ROUTING AND WAVELENGTH ASSIGNMENT (RWA)

The general objective of the RWA problem is to maximize the number of established connections. Each connection request must be given a route and wavelength. The wavelength must be consistent for the entire path, unless the usage of wavelength converters is assumed. Two connections requests can share the same optical link, provided a different wavelength is used.

Given the complexity of RWA, there are two general methodologies for solving the problem:

1. The first method is solving the routing portion first, and then assigning a wavelength second. Three types of route selection are Fixed Path Routing, Alternate path routing, and virtual topology expansion.

2. The second approach is to consider both route selection and wavelength assignment jointly.

Since light paths are the basic building block of this network architecture, their effective establishment is crucial. It is thus important to provide routes to the light path requests and to assign wavelengths on each of the links along this route among the possible choices so as to optimize a certain performance metric. This is known as the *routing and wavelength assignment* (RWA) problem [3].The wavelengths assigned must be such that no two light paths that share a physical link use the same wavelength on that link. Moreover, in networks without wavelength converters, the same wavelength must be used on all links of the lightpath (wavelength continuity constraint). The RWA problem is critically important in increasing the efficiency of wavelength-routed optical networks. With a good solution of this problem, more customers can be accommodated by the given system, and fewer customers need to be rejected during periods of congestion.

4. ROUTING IN OPTICAL NETWORKS

Routing is the process of selecting paths in a network along which to send network traffic. The routing process usually directs forwarding on the basis of routing tables which maintain a record of the routes to various network destinations. Thus, constructing routing tables, which are held in the router's memory, is very important for efficient routing. Most routing algorithms use only one network path at a time, but multipath routing techniques enable the use of multiple alternative paths.[7]

4.1. SHORTEST PATH ALGORITHM

In graph theory, the shortest path problem is the problem of finding a path between two vertices (or nodes) in a graph such that the sum of the weights of its constituent edges is minimized. An example is finding the quickest way to get from one location to another on a road map; in this case, the vertices represent locations and the edges represent segments of road and are weighted by the time needed to travel that segment.

There are several variations according to whether the given graph is undirected, directed, or mixed. For undirected graphs, the shortest path problem can be formally defined as follows. Given a weighted graph (that is, a set *V* of vertices, a set *E* of edges, and a real-valued weight function f: $E \rightarrow \mathbf{R}$), and elements *v*and *v'* of *V*, find a path *P* (a sequence of edges) from *v* to a *v'* of *V* so that

$$\sum_{p \in P} f(p)$$

Is minimal among all paths connecting *v* to *v'*.

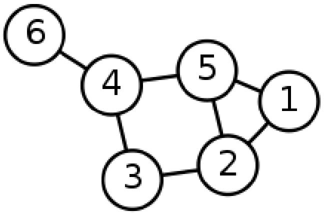


Fig 4.1A graph with 6 vertices and 7 edges

4.1.1 Dijkstra's algorithm

Dijkstra's algorithm is a graph search algorithm that solves the single-source shortest path problem for a graph with nonnegative edge path costs, producing a shortest path tree. This algorithm is often used in routing and as a subroutine in other graph algorithms.

For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost (i.e. the shortest path) between that vertex and every other vertex. It can also be used for finding costs of shortest paths from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined. For example, if the vertices of the graph represent cities and edge path costs represent driving distances between pairs of cities connected by a direct road, Dijkstra's algorithm can be used to find the shortest route between one city and all other cities. As a result, the shortest path first is widely used in network routing protocols, most notably OSPF (Open Shortest Path First).

Dijkstra's original algorithm does not use a min-priority queue and runs in $O(|V|^2)$. The implementation based on a min-priority queue implemented by a Fibonacci heap and running in $O(|E| + |V| \log |V|)$.This is asymptotically the fastest known single-source shortest-path algorithm for arbitrary directed graphs with unbounded nonnegative weights. (For an overview of earlier shortest path algorithms and later improvements and adaptations, see: Single-source shortest-paths algorithms for directed graphs with nonnegative weights).

Dijikstra algorithm steps:

Let the node at which we are starting be called the initial node. Let the distance of node Y is the distance from the initial node to Y. Dijkstra's algorithm will assign some initial distance values and will try to improve them step by step.

1. Assign to every node a tentative distance value: set it to zero for our initial node and to infinity for all other nodes.
2. Mark all nodes unvisited. Set the initial node as current. Create a set of the unvisited nodes called the *unvisited set* consisting of all the nodes except the initial node.
3. For the current node, consider all of its unvisited neighbors and calculate their *tentative* distances. For example, if the current node A is marked with a tentative distance of 6, and the edge connecting it with a neighbor B has length 2, then the distance to B (through A) will be 6+2=8. If this distance is less than the previously recorded tentative distance of B, then overwrite that distance. Even though a neighbor has been examined, it is not marked as *visited* at this time, and it remains in the *unvisited set*.
4. When we are done considering all of the neighbors of the current node, mark the current node as visited and remove it from the *unvisited set*. A visited node will never be checked again; its distance recorded now is final and minimal.
5. If the destination node has been marked visited (when planning a route between two specific nodes) or if the smallest tentative distance among the nodes in the *unvisited set* is infinity (when planning a complete traversal), then stop. The algorithm has finished.
6. Set the unvisited node marked with the smallest tentative distance as the next "current node" and go back to step 3.

We consider a weighted connected simple graph G with vertices a = v0, v1, . . . , vn = z and weights w(vi, vj) > 0 where w(vi, vj) = 1 if {vi, vj} is not an edge. Dijkstra's algorithm finds the cost of the "cheapest" path between vertices a and z.

Example: Use Dijkstra's algorithm to find the cost of the cheapest path between a and z inthe following weighted graph described at each iteration the function L and set S.[7]

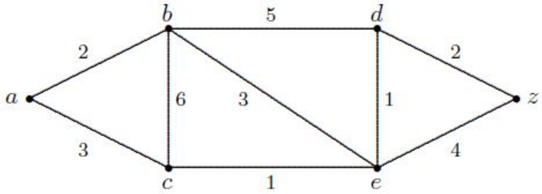
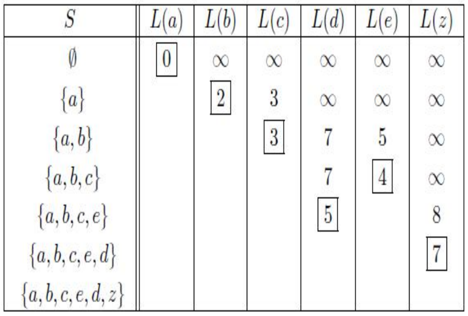


Fig 4.2 Mesh network with weights

The iterations of Dijkstra's algorithm are described in the following table.



S	$L(a)$	$L(b)$	$L(c)$	$L(d)$	$L(e)$	$L(z)$
$\emptyset$	$\boxed{0}$	∞	∞	∞	∞	∞
$\{a\}$		$\boxed{2}$	3	∞	∞	∞
$\{a,b\}$			$\boxed{3}$	7	5	∞
$\{a,b,c\}$				7	$\boxed{4}$	∞
$\{a,b,c,e\}$				$\boxed{5}$		8
$\{a,b,c,e,d\}$						$\boxed{7}$
$\{a,b,c,e,d,z\}$						

Table: 4.1Iterations of Dijkstra's algorithm

At the last iteration, z € S andL (z) = 7.We conclude that the cheapest pathfrom a tozhas a cost of 7.

4.2. ALTERNATE PATH ROUTING(APR)

This determination of the shortest path using traditional shortest-path algorithm on a graph with edges would involve high costs if used by other source–destination pairs. The goal is to have a set of unique paths, even if these do not represent the shortest possible routes from the source node to the destination. The idea is that thesedisjointed paths will orient the establishment of lightpaths without the creation of network bottlenecks, a procedure that has been entitled alternative routing.

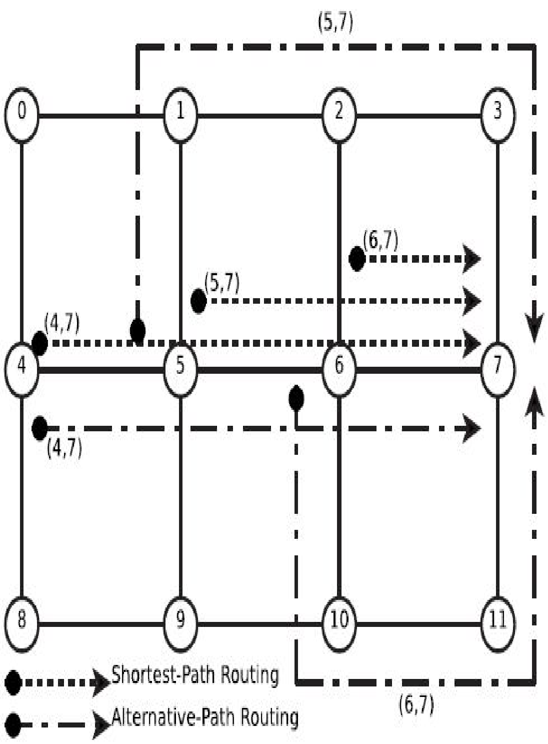


Fig 4.3shortest path v_s alternate path routing

Figure 4.3 provides an example in which routes are defined for the pairs 4→7, 5→7, and 6→7. When using traditional shortest-path algorithms, all three paths include the edge (6, 7), thus creating a bottleneck, whereas the use of alternative routing would avoid such a bottleneck.

4.3. ZONE BASED ALGORITHMS

One possibility for reducing the growth in computational complexity, as well as reducing the probability of blocking, is the establishment of a lightpath in a reduced network region, called a zone shown in fig4.4. This zone will be proportional in size to the number of nodes in the shortest physical route from the source to the destination node of each connection request. The central idea here is to increase scalability and reduce computational complexity by avoiding lightpaths that may exhaust network resources. Moreover, a zone can be dynamically enlarged as needed to ensure flexibility in lightpath selection. Although this approach reduces the probability of blocking, as will be shown here, this reduction is generally not fairly distributed among all source–destination pairs. [1]

4.3.1. ZONE BASED WITH NEIGHBOUR EXPANSION METHOD

One of the developments for the solution of the traffic grooming problem is the use of auxiliary graphs to represent network resources and topology. [6]

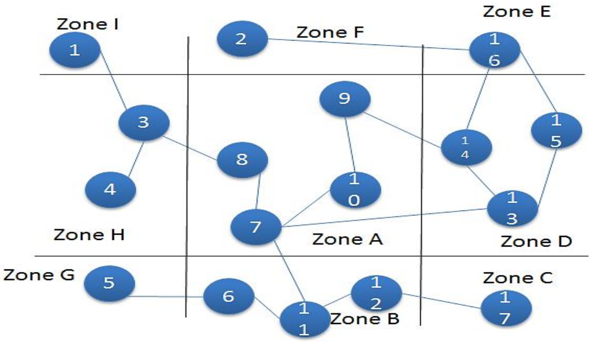


Fig 4.4 Node level topology of zones

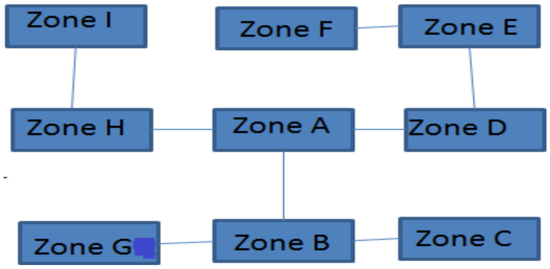


Fig 4.5 Zone level topology of network

This algorithm tries to establish a route from source to destination using a traditional shortest-path algorithm in the auxiliary graph. If no path can be found, the call is considered to be blocked. The major drawback of this algorithm is a lack of scalability given that the entire network must be represented. Auxiliary graphs can, however, represent single regions (zones) rather than the whole network, thus leading to solutions with low computational complexity.The zone-based with neighbor expansion (ZWNE) algorithm proposed in employs such an auxiliary graph, with a size proportional to the number of hops on the shortest path, it produces less blocking than do algorithms based on a full topology. The auxiliary graph in this solution includes the vertices of the shortest path between source and destination of a connection request, with edges between these nodes representing existing or allocable light paths given the solution of a routing and wavelength assignment problem upon the arrival of a request. If such a light path cannot be found in this restricted topology, an expanded auxiliary topology is introduced, adding neighboring vertices to those already present in the auxiliary graph as well as new edges representing the relevant light paths of the physical topology. Expansion continues until an adequate route can be determined. [5]

5. ALTERNATE ROUTING WITH VIRTUAL TOPOLOGY EXPANSION METHOD(ARVTE)

The novel algorithm proposed here, the Alternative Routing with Virtual Topology Expansion (ARVTE) algorithm, is based on the same principals as the ZWNE algorithm, but the reduced auxiliary graph differs fundamentally in construction.[4]

5.1. VIRTUAL TOPOLOGY EXPANSION METHOD

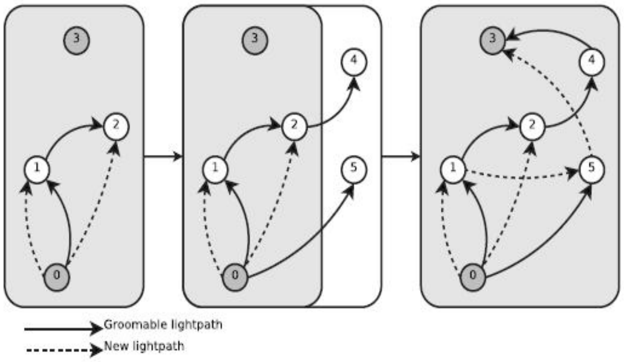


Fig 5.1example of the virtual topology expansion method

Fig 5.1illustrates the expansion procedure of the VTE algorithm. The leftmost part of the figure shows an auxiliary graph constructed for the pair 0→3. In this figure, the solid arrows indicate existing light paths, whereas the dashed ones are allocable. Because there is no existing path from source to destination, neighboring vertices 4 and 5 are added to the graph in the central part of the figure, which leads to the addition of the edges at the rightmost part of the figure, thus enabling the establishment of a multi hop path between source and destination nodes.

5.2. ALTERNATE ROUTING WITH VIRTUAL TOPOLOGY EXPANSION (ARVTE)

The construction of the auxiliary graph, G(V,E), using the vertices of the designated path, P(s,d), for the source–destination path (s,d) is carried out, and the edges corresponding to either existing or allocable light paths are added .Then attempt to find a light path for the establishment of the requested connection between the pair(s,d) with demand r, Req(s,d,r). For the selection of this light path, a shortest-path algorithm is executed in the auxiliary graph, P0(s,d). If a path is found, the ligh t path is established. Otherwise, the auxiliary graph is expanded by introducing vertices, either from the virtual topology or from the physical topology. After that, an expanded auxiliary graph is constructed and a new search initiated. P(s, d) is defined by an algorithm, called alternative path routing, adopted in the offline phase. [4]

6.ALGORITHM

```
1.Get number of nodes and enter the connections
2.If selected S and D is neighbor to each other,
Delay = 1
Else
Assign delay == 1000 i.e., infinity
3.Run any one of the existing ALTERNATE PATH ROUTING ALGORITHM to find all possible paths between S and D
4. Choose any path to transmit data from the possible paths
5. Enter the wavelength requirement, W
If W exists,
Display all paths with their delays for specified delays,
Choose minimum delay from displayed paths to transmit,
Display transmission path with average W available to each link,
6. Check any W common in all links in the path
        If present
          Transmit data, and
          Calculate COST for transmission path
COST = 1xHc(L) + 0.1xGp (L) + 0.01xWc(L)
          Wc(L) = Wavelength conversions
          Gp (L) = Grooming port pairs
           Hc(L) = hop count i.e., delay
        Else,
                Use wavelength converters,
                Repeat **,
        Else
Choose VIRTUAL TOPOLOGY EXPANSION METHOD
        Display all paths from VTE
          Repeat 5,
```

7. CONCLUSION

Recent advances in research on traffic grooming have shifted the focus from homogeneous static traffic and restricted topologies to more general scenarios that include heterogeneous dynamic traffic over mesh topologies. One type of solution for the dynamic traffic grooming problem adopts an auxiliary graph to represent the network topology and the availability of resources .Such a solution, when accounting for an entire network, implies high computational complexity of paths and leads to a solution that is non-scalable for large networks.

The present paper has introduced a novel algorithm, the Alternative Routing with Virtual Topology Expansion (ARVTE) algorithm, which borrows the idea of a restricted auxiliary graph from the ZWNE algorithm, but introduces a new graph expansion procedure, as well as a routing algorithm to be used with the new procedure for improving performance. The emphasis is at the avoidance of the bottlenecks resulting from the traditional shortest-path algorithm. The ARVTE algorithm produces lower cost for path establishment values than does that of ZWNE; moreover, these values are more balanced in distribution.

Given this evidence, the ARVTE algorithm seems to be a good candidate in path establishment at low cost for dynamic traffic grooming in mesh networks, especially for asymmetric networks such as India-net topology.

REFERENCES

[1]. Quang-Dzung Ho and Man- Seop Lee, “A Zone-Based Approach for Scalable Dynamic Traffic Grooming in Large WDM Mesh Networks”, in IEEE proceedings.
[2]. J. Hu and B. Leida, “Traffic grooming, routing, and wavelength assignment in optical WDM mesh networks,” in *Proc. Of IEEE INFOCOM*, 2004, pp. 1340–1351.
[3]. Routing and Wavelength Assignment in Optical Networks, byAsuman E. Ozdaglar and Dimitri P. Bertsekas.
[4]. “Virtual-Topology Adaptation for WDM Mesh Networks Under Dynamic Traffic’’, Aysegul Gencata, Student Member, IEEE, and Biswanath Mukherjee, Member, IEEE.
[5]. A.C.Drummond and N.L.S.da Fonseca, “On the effectiveness of expansion mechanisms in zone-based dynamic traffic grooming algorithms,” in IEEE ICC 2010 proceedings, 2010, pp.1-5.
[6]. Q.D. Ho and M.S. Lee, “A zone-based approach for scalable dynamic traffic grooming in large wdm mesh networks,” IEEE Journal of Light wave Technology, 2007, vol.25, no. 1,pp. 261–270.
[7]. http://www.wikipedia.org/search.
[8].http://www.optical-network.com/topology.php.
[9]. http://www.electronics.dit.ie