LOAD BALANCING IN WDM NETWORKS THROUGH DYNAMIC ROUTE CHANGES

S.Ramanathan\textsuperscript{1}, G.Karthik\textsuperscript{1}, Ms.G.Sumathi\textsuperscript{2}

\textsuperscript{1} Dept. of computer science
Sri Venkateswara College of engineering, Sriperumbudur, 602 105.
\textsuperscript{2} Asst.professor, Dept. of computer science,
Sri Venkateswara College of engineering, Sriperumbudur, 602 105.

Abstract:

This study deals with the formulation of an effective and efficient algorithm for load balancing in WDM networks. Here we develop a Load Balancing algorithm for IP-like Routing. The considered networks are based on a routing protocol where the next hop at a given node depends only on the destination of the communication.

The aim of this work is to consider the applicability of this approach in the context of optical networks, by taking into account the routing and wavelength assignment constraints simultaneously which is a NP-Complete problem.

Our algorithm performs at each iteration, a basic change in the network by rerouting the traffic to the destination from the given source by finding an alternate path thereby reducing the load on the common link. As a result a single entry change in the routing table is performed, which is very important for the stability of the network.

KEYWORDS: WDM, virtual load, Routing and Wavelength assignment, Load Balancing, Dynamic Routing.
1. INTRODUCTION:

In the present days, we experience an increasing bandwidth demand caused by the exponential growth of the Internet and the introduction of communication-intensive applications. In this framework, techniques such as Optical Wave Division Multiplexing (WDM) and routing schemes such as Generalized Multi Protocol Label Switching (G-MPLS) have been proposed.

In wavelength division multiplexing (WDM) networks, communication between optical cross-connect (OXC) switches takes place along all-optical WDM channels, which are commonly referred to as light paths.

The central algorithmic problem in WDM routing [1] between source-destination pairs is:

- **route assignment**: Compute a route for a light path between each source-destination pair, and
- **wavelength assignment**: for each link traversed by this light path, determine the wavelength to be allocated for the light path on the given link.

Together, these two assignment problems are often referred to as the Routing and Wavelength Assignment (RWA) problem [7]. The Routing and Wavelength problems in WDM networks are generally addressed disjointly because they both combine to form a NP-complete [2] problem.

In our work, we address the routing and wavelength problems simultaneously, find an optimal solution to this problem.

2. ROUTING AND WAVELENGTH ASSIGNMENT:

The RWA problem is complicated by the fact that an OXC switch may be (optionally) equipped with wavelength conversion hardware, which permits light paths transient through the switch to enter and leave the switch on different wavelengths.

If no switches are equipped with wavelength converters, then a light path in this network must always occupy the same wavelength on every fiber link it traverses. This restriction is commonly known as the **wavelength continuity**
constraint [3] for non-wavelength converting switches. An OXC switch that is equipped with wavelength conversion hardware is exempt from this constraint.

2.1. ROUTING ASSIGNMENT:

There are three broad classes of strategies being used presently to address the routing assignment problem. These strategies are referred to as fixed routing, fixed alternate routing, and adaptive routing [3].

**Fixed routing:**

Fixed Routing is a simple technique, which involves maintaining a fixed routing table at each candidate source node. The routing table consists of one entry for each candidate destination node, where the entry specifies the path from the source to the destination. Fixed routing is simple to implement, but is subjected to unacceptably high blocking probabilities when wavelength availability on links becomes scarce and when link failures occur.

**Fixed Alternate Routing:**

Fixed alternate routing attempts to address the shortcomings of fixed routing by augmenting each entry in the routing table to be a prioritized set of paths from source to destination, rather than just a single one. By doing so, fixed alternate routing is less sensitive to link failures and wavelength scarcity and thus offers lower connection blocking probabilities.

**Adaptive Routing:**

Adaptive routing schemes attempt to select a path between a source-destination pair based on dynamically collected information concerning the network's state. This technique is much more resilient to link failures and less sensitive to wavelength scarcity, thus offers lower connection blocking probability than fixed adaptive routing approach.

2.2. WAVELENGTH ASSIGNMENT:

The second half of the RWA problem, wavelength assignment, is addressed algorithmically using strategies based either on heuristics, or on graph coloring algorithms. Presently considered heuristics include Random Wavelength Assignment, First-Fit, Least-Used (SPREAD), Most-Used (PACK),
Min-product, Least Loaded, MAX-SUM, Relative Capacity Loss, Distributed Relative Capacity Loss, Wavelength Reservation, Protecting Threshold etc.

2.3. STATIC OR DYNAMIC NETWORKS:

In a wavelength-routed network, the traffic can be either static or dynamic [3]:

In a static traffic pattern, a set of light paths are set up all at once and remain in the network for a long period of time. The RWA problem for static traffic is known as the “static light path establishment“ (SLE) problem.

In a dynamic traffic pattern, a light path is set up for each connection request as it arrives, and the light path is released after some finite amount of time. The problem of light path establishment in a network with dynamic traffic demands is called the *dynamic light path establishment* (DLE) problem.

One of the challenges involved in designing wavelength-routed networks with dynamic traffic demands is to develop efficient algorithms and protocols for establishing light paths. The algorithms must be able to select routes and assign wavelengths to connections in a manner that efficiently utilizes network resources and maximizes the number of light paths established. Signaling protocols for setting up light paths must effectively manage the distribution of control messages and network state information in order to establish a connection in a timely manner. Typically, a network control and management protocol is employed to perform the RWA and signaling tasks mentioned above.

Another issue in dynamic light path establishment is how to initiate requests for light path establishment and removal. There are a number of possible approaches for generating a connection request.

2.4. ADAPTIVE ROUTING AND LOAD BALANCING:

Load balancing in WDM networks consists of two sub problems:

- The light path connectivity
- The traffic routing problem.

The routing problem has its origin at the beginning of the networking research. In particular, adaptive routing, that incorporates network state
information into the routing decision is considered in the context of all-optical networks, while previous work on state-dependent routing with trunk reservation is used in traditional telecommunications networks. It is also known that flow deviation methods although computationally demanding, can be used to find the optimal routing that minimizes the maximum link load for a given network topology.

Let us now define the context and the notation. By **physical topology** we mean the actual network composed of passive or configurable optical nodes and their fiber connections. The **logical topology** is given by the light paths between the electronic routers, determined by the configuration of the OADMs (Optical Add Drop Multiplexers) and transmitters and receivers on each node. The **traffic pattern** is available as an N x N matrix (N being the number of nodes in the network) T- (t\(_{ij}\)) where t\(_{ij}\) denotes the number of light paths required from node i to j. We assume that t\(_{ij}\) values are non negative integers and t\(_{ij}\) =0 if i=j. A **routing table** is an array associated to each node in the network, containing next hop required for routing. For a given traffic pattern routing tables associated to the nodes, the sum of the number of light paths passing through each link is called the **virtual load** of the link. Finally, the maximum virtual load on each link of a path is called the **congestion** of the path. The maximum virtual load on each link of the network is called the **congestion** of the network.

Because global changes of the logical topology and/or routing scheme can be disruptive to the network, algorithms that are based on a sequence of small steps (i.e., on local search from a given configuration) are considered.

The Load Balancing problem can be defined as follows.

**Load Balancing** -- Given a physical network with the link costs and the traffic requirements between every pair of source-destination (number of light paths required), find a routing of the light paths for the network with the least congestion.
3. DEFINITIONS AND EXPLANATIONS OF THE PARAMETERS:

Imminent path set:
This set contains all paths that are candidate to replace those passing through the most congested links of the network. It consists of all paths having enough space to route the virtual load removed from the most congested link, without causing their virtual load to be higher than the current congestion of the network.

Shortest path routing (network):
It calculates the shortest path tree for each destination node and returns the corresponding routing table as a matrix.

rtable[n]:
It is the routing table of node n, whose i-th entry rtable[n][i] is the next hop node index for light paths passing through node n and with destination d.

calculate load (network, traffic, rtable):
It returns the network congestion given the network topology, the traffic pattern and the current routing scheme. The function also returns the set of links having maximum loads.

Best_path (pfrom , pto) :
It gives the set of paths that are eligible candidates to replace the existing congested path.

NAW ( ):
It identifies the optimal pair (source-destination) communicating using the congested link which when rerouted reduces the virtual load of the network.

Log_and ( src, dest):
This gives the number of wavelengths allocated for traffic in the path between src and dest.
MAXITER:

This defines the maximum number of times the balancing part loop has to run before adapting to the dynamic load changes in the network.

Fig 1 THE SEARCH SPACE
4. Algorithm:

The search for the algorithm is presented in Figure 1. The algorithm employed is described as follows:

1. rtable $\leftarrow$ shortest path routing (network)
2. Initial network with the given traffic { init_route() }
3. $<$congestion, congested link set> $\leftarrow$ calculate load(network, traffic, rtable)

Repeat

Imminent path $= \phi$

for each link $<$cfrom, cto$> \in$ congested link set

excess $\leftarrow$ No. of $\lambda$’s allocated more than threshold

for each $<$src,dest $> \in$ communicating pair (cfrom , cto)

$<$pfrom , pto$> \leftarrow$ NAW( )

remove partial load (pfrom,pto)

Imminent path $\leftarrow$ best_path (pfrom,pto)

restore partial load (src, dest)

if (imminent path $\neq \phi$)

change rtable (imminent path)

$<$congestion, congested link set> $\leftarrow$ calculate load (network, traffic, rtable)

until MAXITER

4.1. EXPANSION OF THE SUB-MODULES USED IN THE ALGORITHM:

4.1.1. NAW () – NEAR MAXIMUM AVAILABLE WAVELENGTH:

for all $<$src,dest$> \in$ communicating pair (cfrom ,cto)

$<$pfrom,ppto$>$ ( communicating pair satisfying following condition

src $\leftarrow$ link src & dest $\leftarrow$ link dest &

if ( log_and ( src , dest) $\leq$ excess)

$<$pfrom,pto$>$ ( min { excess - log_and ( src , dest) })

if there exists no such pair $<$src , dest $>$

$<$pfrom , pto $>$ ( min { log_and ( src,dest) - excess})

return this path and the minimum load.
4.1.2 Best_Path (pfrom, pto):

Best candidate load (virtual load on imminent path <pfrom, pto>)

For each neighbour n’bour (neighbourhood (pfrom))

find a path from pfrom to pto thro’ n’bour such that
- The path does not form a cycle
- The path does not travel thro’ the same congested link

vl (virtual load on imminent path <pfrom, pto> thro’ n’bour)

If (vl < best candidate load)

best candidate load (vl)

Imminent path ( <pfrom, pto , n’bour>)

Else

If (vl = best candidate load)

Imminent path = imminent path U {<pfrom, pto , n’bour>}

Best path = random pick( imminent path)

4.1.3 change rtable (n’bour):

rtable[src][dest] = n’bour

NOTE: \(\rightarrow\) : Denotes the looping structure with the start and the end lines of the loop.

5. DESCRIPTION OF THE ALGORITHM FUNCTION:

Global changes of the logical topology and/or routing scheme can be disruptive to the network, hence the algorithm performs load balancing in a sequence of small steps [4].

The basic idea of the new load balancing scheme is as follows: we find the shortest path route in the network taking into account the wavelength constraint between the source and destination. We iteratively find the most congested edges and re-route the traffic passing through these edges, until no more re-routing is possible within network constraints or a fixed maximum number of iterations is reached.
In the algorithm, the initialization section (lines 1-2) starts by generating the routing tables through the application of the shortest path routing algorithm [6] to the specific network. Using the function calculate load (line-3) we calculate the load on each link of the network, the initial value of the congestion and the set of congested links congestedlinkset. The imminentpathset is empty at the beginning of each iteration.

The local search algorithm (lines 4-16) consists of two distinct parts, which are

- Selection of an alternate path.
- Correction of routing table.

First, a set of alternative paths where to re-route part of the traffic passing through the most congested links is found (lines 5-12); then we correct the routing table of the network (line 13,14) and start over again by considering the new congested links.

The first part includes the core of the algorithm. We consider each congested link in the congestedlinkset; identified with its end points (cfrom, cto). Then we iterate through all light paths using that congested link. We then identify the communicating pair (src, dest) that uses the congested link which when rerouted, avoiding the congested link, reduces the congestion in the network. The NAW segment identifies this communicating pair.

For every identified (src, dest) pair through the link (cfrom, cto) we try to re-route the traffic. The function NAW (network) finds if an alternate route exists between the (src, dest) pair, which has more available wavelength than the present route, using the log_and.

The best_path ( ) finds the optimal path to reroute the traffic from (src, dest). For all the neighbourhood nodes n’bour of src, this finds the virtual load for the route through n’bour reaching dest. This returns the path with least virtual load as the best alternate path.

The second part is that the routing tables of the nodes are updated for the new path to take effect. Finally a new value of congestion and the relative set of most loaded links congestedlinkset is calculated again in order to start a new search of alternate paths through the network.
This local search algorithm continues searching for better values of congestion until no more re-routes can be performed or until a given number of iterations have been performed.

6. IMPROVEMENTS OVER THE EXISTING ALGORITHM:

- Efficient re-routing:
  Our algorithm finds the best route between a source and destination with wavelength being the primary constraint, unlike other algorithms where alternate paths[5] are chosen with number of hops as the primary constraint and the important wavelength constraint getting less attention and a non-optimal path being chosen.

- Minimal changes to routing table:
  We designed the algorithm based on a sequence of small steps because global changes of the logical topology and/or routing scheme can be disruptive to the network[2].

- Useful for dynamically evolving traffic:
  This iterative scheme suits in a very appropriate way to a dynamic environment where traffic requirements evolve with time. This algorithm is sufficient to keep the system in a suitable state as the traffic matrix changes. Of course, only lines 2-14 must be executed. Moreover, a very low number of iterations of the outer loop must be performed at each step (i.e) MAXITER must be small to avoid excessive traffic disruption.

- Minimal global state information is required:
  This algorithm performs source routing using only the global state information of each wavelength. This is in contrast to several other algorithms, which assumes that each optical node exchanges full link state information with other nodes.
7. CONCLUSION:

The considered load balancing algorithm at each iteration modifies only a single entry in the routing table of a node. Hence the stability of the network is unaffected since too many changes in the routing table entries at the same time will disrupt the traffic flow and make the network unstable. This algorithm takes care of this problem.

Although our algorithm might not find the best route which can be found through exhaustive search, this is far more practical when time complexity and the number of routing table changes which decides the stability of the network is considered.

Hence our algorithm addresses the routing and wavelength problem simultaneously and an optimal solution can be obtained using the algorithm.

Many extensions can be envisioned from our work. One such is the problem of network having links of different capacities.

8. REFERENCES:


