Heuristic Algorithms Considering Various Objectives for Virtual Topology Design in WDM Optical Networks

Nina Skorin-Kapov and Mladen Kos
Department of Telecommunications, FER
University of Zagreb,
Zagreb, Croatia
e-mail: {nina.skorin-kapov|mladen.kos}@fer.hr

Abstract

This paper addresses the problem of virtual topology design in wavelength routed WDM optical networks with no wavelength converters. To solve this problem, a virtual topology composed of a set of lightpaths must be determined, routed over the physical topology and assigned individual wavelengths. Furthermore, traffic between each pair of nodes must be routed over the established virtual topology. There can be several objectives in virtual topology design such as the minimization of congestion, average packet hop distance, the number of tranceivers and wavelengths used, and the average physical length of the established lightpaths. This problem has been shown to be NP-complete so heuristic algorithms have been developed to solve it. We propose variations of a simple and fast greedy algorithm for virtual topology design. The solutions obtained by the proposed algorithms are compared with respect to several optimization criteria. Some of the results are compared with existing algorithms for the same problem. Presented is an extensive analysis of the obtained results which indicates the advantages and disadvantages of each of the algorithms with respect to various objective criteria. This analysis is meant to help determine which algorithm performs best depending on the specific circumstances in a network such as resource availablity, network topology, and traffic trends.

Keywords-Routing and wavelength assignment, virtual topology design, WDM optical networks

1 Introduction

1.1 Wavelength Routed WDM Networks

Wavelength division multiplexing (WDM) can exploit the large bandwidth of optical fibers by dividing it among different wavelengths creating several WDM channels. As a result, end nodes need to operate only at the speed of a single WDM channel (e.g. peak electronic rate). Wavelength routed WDM networks can also take advantage of spatial wavelength reusability which is important if there is a limited number of wavelengths available. These networks are equipped with configurable WDM nodes which enable us to set up and tear down all-optical connections, called *lightpaths*, between pairs of nodes. These all-optical connections can traverse multiple physical links in the network. Information sent via a lightpath does not require any opto-electronic conversion at intermediate nodes.

Establishing a set of lightpaths creates a virtual topology on top of the physical topology. The physical topology represents the physical interconnection of WDM nodes by actual fiber links in the WDM optical network. The links in the virtual topology represent all-optical connections or lightpaths established between pairs of nodes. The virtual topology problem in wavelength routed WDM networks is as follows. Given is a physical topology, the number of available wavelengths on each link, the number of available transmitters and receivers at each node, and a traffic matrix representing the long-term average traffic flows between nodes. To create a virtual topology, a set of lightpaths which forms the virtual topology over the physical one must be determined. In order for these lightpaths to be established, each lightpath must be routed over the physical topology and assigned a particular wavelength. We will refer to a combination of these subproblems as the Virtual topology and Routing and Wavelength Assignment problem ($VRWA$). Lastly, packet switched traffic must be routed over the established virtual topology. This will be referred to as Traffic Routing (TR).

In order to set up a lightpath, nodes on its corresponding physical path must be configured to do so. If the network lacks wavelength converters, the lightpath must use the same wavelength along its entire physical path. This is called the *wavelength continuity constraint.* Two lightpaths that share a common physical link cannot be assigned the same wavelength. The source and destination nodes of the lightpath must have available transmitters and receivers respectively in order for the lightpath to be successfully established. Due to a limited number of transmitters and receivers at each node and a limited number of available wavelengths on each link, it is usually not possible to set up a lightpath between every pair of nodes.

Determining a good virtual topology with respect to various optimization criteria given a limited amount of resources is a complex problem that has been studied extensively [2] [3] [5] [8] [9]. Most algorithms suggested for virtual topology design concentrate on a single optimization criterion and the quality of their obtained solutions are considered with respect to only that criterion. Here, we suggest heuristic algorithms for the $VRWA$ problem and analyze their performance by considering several aspects of the obtained solutions.

1.2 Optimization Criteria in Virtual Topology Design

A brief description of various optimization criteria in virtual topology design follows.

1.2.1 Congestion

The most common optimization criterion in virtual topology design is the minimization of congestion. Congestion is defined as the maximum traffic load on any virtual link.

1.2.2 Packet Hop Distance

If delay is an important issue, it is desirable to minimize the average number of lightpaths traversed by a unit of traffic (packet) on its path from source to destination in the virtual network. This is called the average packet hop distance and is a function of the virtual topology and the long term traffic matrix.

1.2.3 Wavelengths Used

In order to leave more room for future expansion of the virtual topology, minimizing the total number of distinct wavelengths used is desirable. In [7], the authors consider the maximum number of lightpaths routed on any physical link to be a measure of the expandability of the virtual topology. This is equal to the maximum number of wavelengths used on any link and is essentially the lower bound on the total number of distinct wavelengths used. This measure of network expandability is only sufficient if the network is equipped with wavelength converters at each node. If the network lacks wavelength converters, a request to add a new lightpath may be rejected even though there exists a path on which all links have available wavelengths due to the unavailability of the *same* wavelength on the entire path. If such is the case, reconfiguration or wavelength rerouting[1] must be performed or the request is blocked. As a result, it seems that minimizing the total number of distinct wavelengths used, instead of minimizing the maximum number of lightpaths on a physical link, is a more appropriate objective criterion. Using less distinct wavelengths, i.e. leaving more entirely free wavelengths, decreases the chances that a new request will be blocked due to the wavelength continuity constraint.

1.2.4 Tranceivers Used

Transmitters and receivers, commonly referred to as tranceivers, are fairly expensive. As a result, it is desirable to set up a virtual topology with fewer tranceivers (i.e. fewer lightpaths) as long as the congestion and average packet hop distance are acceptable.

1.2.5 Physical Hop Length

In opaque networks where electronic regeneration is performed at each node, minimizing the *physical* hop length of individual lightpaths is important. Such networks require a transmitter and receiver at the head and tail nodes respectively of each physical link included in the

[1] Wavelength rerouting is a mechanism that switches a certain number of existing lightpaths to a different wavelengths in order to create a wavelength continuous path for a new request.

lightpath. As a result, longer physical paths dramatically increase the cost of the network. In addition, the minimization of the physical length of a lightpath, not only in terms of hops but also in terms of actual distance, is desirable in all WDM networks due to signal degradation.

1.2.6 Virtual Hop Distance

An optimization criterion that has not been considered much in research dealing with virtual topology design is a measure which we refer to as the average *virtual* hop distance. The average virtual hop distance is the average hop distance in the virtual topology between all source - destination pairs. This is a function of the virtual topology alone and is entirely independent of the traffic matrix. We feel that this criterion, in combination with the average *packet* hop distance, is relevant due to the following. If the average *packet* hop distance is low but the average *virtual* hop distance is high, this means that most of the lightpaths are concentrated around a small number of nodes with high traffic. Since traffic can be prone to change and reconfiguration of the virtual topology can be costly, it seems that such a virtual topology could perform poorly in the long run as traffic changes. On the other hand, if the virtual topology has not only a low average *packet* hop distance but a low *virtual* hop distance as well, we know that *all* the source - destination pairs are fairly well connected. Therefore, in addition to performing well for current traffic trends, the virtual topology would perform well for changing traffic and thus postpone reconfiguration for a longer period of time.

Furthermore, ensuring a finite average virtual hop distance would eliminate unconnected virtual topologies. Suppose that there is zero traffic between a pair of nodes in the current traffic matrix. If such is the case, the hop distance between these nodes would not enter into the calculation of the average *packet* hop distance since there are no packets delivered between these two nodes. Therefore, without considering the average virtual hop distance, the distance between these nodes could be arbitrarily long or the two could even be unconnected. In the case of the latter, not even a single packet could be sent between these nodes without reconfiguration of the virtual topology.

1.2.7 Execution Time

The execution time of virtual topology design algorithms, although not a direct optimization criterion for the virtual topology problem, is certainly an important aspect of the algorithms to consider. Most algorithms proposed for virtual topology design have large execution times and thus become intractable for larger networks.

1.3 Related Work

In networks equipped with wavelength converters, the virtual topology design problem is less complex since the wavelength continuity constraint does not apply. An exact mixed integer linear formulation (MILP) for complete virtual topology design in WDM networks with full wavelength conversion is given in [3]. The objective is to minimize the average packet hop distance. Heuristic algorithms for the same problem are suggested in [8]. In [9], the authors formulate a MILP for virtual topology design with the objective to minimize congestion. There is no constraint on the number of wavelengths used.

In the absence of wavelength converters, the virtual topology problem becomes more difficult. A MILP which minimizes congestion in networks with no wavelength converters is given in [5]. This formulation is not computationally tractable, hence a heuristic approach is suggested. The suggested MILP is relaxed and iteratively run 25 times using a cutting plane. The variables representing the virtual topology and physical paths are rounded while a wavelength assignment heuristic is applied to assign wavelengths to individual lightpaths. Traffic is routed over the virtual topology using a linear programming formulation (LP) consisting of only the traffic constraints of the relaxed MILP. We will refer to this heuristic as $MILP+WA$. One of the drawbacks of the $MILP+WA$ heuristic is the following. Supposing there are W available wavelengths on each fiber, the relaxed MILP obtains a solution which satisfies this constraint. However, since the wavelength assignment algorithm which is subsequently applied gives suboptimal solutions, it does not guarantee a successful wavelength assignment with at most W wavelengths. As a result, the $MILP+WA$ algorithm does not necessarily give feasible solutions for all cases.

A well-known heuristic algorithm for the Virtual topology and

*R*outing and *W*avelength *A*ssignment problem ($VRWA$) problem called $HLDA$ is given in [9]. *T*raffic *R*outing (TR) is solved subsequently using an LP formulation which minimizes congestion. $HLDA$ attempts to establish lightpaths between pairs of nodes in decreasing order of their corresponding traffic. These lightpaths are routed on the shortest available path and assigned the lowest available wavelength found on that path. After establishing a lightpath between a pair of nodes, the value of their corresponding traffic is decreased by the value of the next highest traffic demand and all the demands are again sorted in decreasing order. This allows multiple lightpaths to be established between pairs of nodes with high traffic. After the procedure terminates, tranceivers may be left over at some nodes in the network. If such is the case, $HLDA$ establishes lightpaths at random between these nodes until all the available resources are exhausted. This algorithm although simple, performs very well with respect to congestion.

1.4 Contributions of the Paper

Determining a good virtual topology with respect to the optimization criteria described in Section 1.2 is a complex problem. In this paper we suggest 4 simple greedy algorithms motivated by $HLDA$ [9] which tackle the $VRWA$ problem. The variations between the algorithms are each meant to better satisfy different optimization criteria. Traffic routing over the virtual topology is done using the LP formulation suggested in [5]. Most algorithms suggested for virtual topology design are evaluated by considering a single optimization criterion to be the measure of quality of their obtained solutions. In this paper, we analyze the performance of the proposed algorithms with respect to several aspects of the obtained solutions. We also compare some of our results to that of existing algorithms for virtual topology design. Furthermore, we analyze the benefits and drawbacks of establishing multiple lightpaths between pairs of nodes. We discuss the trade-offs associated with each algorithm and the network scenarios in which they may perform best.

1.5 Outline of the Paper

The rest of the paper is organized as follows. In Section 2, we informally define the $VRWA$ problem and suggest heuristic algorithms to help solve it in Section 3. Numerical results and a detailed analysis of the obtained results are given in Sections 4 and 5. We finish with some suggestions for further research and concluding remarks in Section 6.

2 Problem Definition

The physical optical network is modelled as a graph $G_p = (V, E_p)$, where V is the set of nodes and E_p is the set of physical edges. Edges are assumed to be bidirectional (each representing a pair of optical fibers - one fiber per direction) and have assigned weights representing their length or cost. Given is a long term traffic matrix $\Lambda = (\lambda^{sd}), s, d \in V$, where each element represents the average traffic flow from source node s to destination node d. The number of available wavelengths W on each link and the number of available transmitters Tr and receivers Re at each node are given. We include an additional parameter (value h) which represents an upper bound on the physical length[2] of a lightpath.

The $VRWA$ problem searches for a set of lightpaths which creates a virtual topology on top of the physical topology. The virtual topology can be modelled as a directed graph $G_v = (V, E_v)$. Each directed edge in E_v represents one lightpath $(i, j), i, j \in V$, defined by the source node i and destination node j of the lightpath. No more than Tr lightpaths can share the same source node and no more than Re lightpaths can share the same destination node. In other words, Tr and Re are the maximum out-degree and in-degree respectively of any node in G_v. Furthermore the $VRWA$ problem searches for a set of physical paths $P = \{P_1(i_1, j_1), \ldots, P_{|E_v|}(i_{|E_v|}, j_{|E_v|})\}$ in G_p, each corresponding to one lightpath or virtual link from G_v, and assigns wavelengths to these paths. Paths $P_k(i_k, j_k)$ and $P_l(i_l, j_l)$ where $k \neq l$, $k, l = 1, \ldots, |E_v|$, cannot be assigned the same wavelength if they share a common edge in G_p. At most W distinct wavelengths can be assigned to the paths in P. The length of any path $P_k(i_k, j_k)$, $k = 1, \ldots, |E_v|$, is upper bounded by value h.

There are several objectives to consider when solving the $VRWA$ problem. The most common is to design such a virtual topology G_v and corresponding routing and wavelength assignment which enables

[2] Length can be considered in terms of hops or actual distance.

traffic Λ to be routed over G_v with the minimal congestion. It is also desirable that the virtual topology have small packet and virtual hop distances and yet consist of a small number of lightpaths to reduce total tranceiver cost. With respect to routing and wavelength assignment, the number of distinct wavelengths used and the lengths of physical routes of individual lightpaths should both be minimized.

3 The proposed algorithms: TSO_SP, TSO_FS, $TSBS\_SP$, $TSBS\_FS$

We propose four fast greedy algorithms for the $VRWA$ problem in networks with no wavelength converters. Traffic Routing over the virtual topology is solved subsequently using the same LP which minimizes congestion used by the $MILP+WA$ heuristic in [5].

3.1 The TSO_SP algorithm

The TSO_SP algorithm is a simple virtual topology design algorithm where Traffic is Sorted Overall and routed on the Shortest Path available. Algorithm TSO_SP is similar to the $HLDA$ algorithm suggested in [9] except that is does not establish multiple lightpaths between nodes and does not subsequently assign lightpaths at random until all the tranceivers or wavelengths are exhausted. The reason for this is that our objectives include minimizing the number of tranceivers and wavelengths used.

The algorithm is as follows. Since we have W available wavelengths, suppose W copies of graph G_p referred to as $G_p^1, \dots, G_p^W$ each representing one wavelength. This "layered graph" approach was first introduced in [1] for the Routing and Wavelength Assignment problem. For each traffic demand in Λ in decreasing order, the shortest path available in any graph $G_p^1, \dots, G_p^W$ is found[3]. Suppose this path is found on graph G_p^w, where $w \in \{1, \dots, W\}$. If the length of this path is less than h and there is an available transmitter and receiver at the source and destination nodes respectively, the lightpath is established and assigned wavelength w. The edges found along the path are deleted from graph G_p^w and the number of available tranceivers is updated. The procedure terminates when all the tranceivers or wavelengths are exhausted or until we have tried to establish a lightpath between every source - destination pair in the network.

[3]If the shortest path exists in more than one graph, the graph representing the lowest wavelength is chosen.

The pseudocode of TSO_SP is as follows:

Input:
$G_p = (V, E_p)$;*//physical network*
Λ;*//$|V| * |V|$ long term traffic matrix*
W; *//available wavelengths*
Tr, Re;*//available transmitters and receivers respectively at each node*
h; *//max physical length of lightpath*
Initialization:
$Transmitters := \{t_1, \dots, t_{|V|}\}$, $t_i = Tr$, $i = 1, \dots, |V|$; *//available transmitters per node*
$Receivers := \{r_1, \dots, r_{|V|}\}$, $r_i = Re$, $i = 1, \dots, |V|$; *//available receivers per node*
$k = 1$; *//index of the potential lightpath under consideration*
Begin:
Sort traffic demands between source - destination pairs in Λ in decreasing order creating a list $\tau = \{(s_1, d_1), \dots, (s_{|\tau|}, d_{|\tau|})\}$ of potential lightpaths, where $|\tau| = |V| * (|V| - 1)$;
Create W copies (layers) of $G_p : G_p^1, \dots, G_p^W$;
while $k \le |V| * (|V| - 1)$ and there are available tranceivers **do**
  For demand (s_k, d_k) from τ, find its shortest path P_k available in $G_p^1, \dots, G_p^W$.
  (If more than one shortest path exists, route on lowest wavelength layer);
  if the length of $P_k < h$ **then**
    Establish lightpath (i, j), where $i = s_k$, $j = d_k$;
    If routed on graph G_p^w, delete from G_p^w all edges in P_k and assign wavelength w to lightpath (i, j);
    $t_i = t_i - 1$; $r_j = r_j - 1$;
  end if
  Remove (s_k, d_k) from τ;
  $k = k + 1$;
end while
End

3.2 The TSO_FS algorithm

The TSO_FS algorithm is a virtual topology design algorithm where Traffic is Sorted Overall and routed on the First Satisfactory route available. The traffic demands are sorted in decreasing order as in TSO_SP but only *one* copy or layer of graph G_p referred to as G_p^1

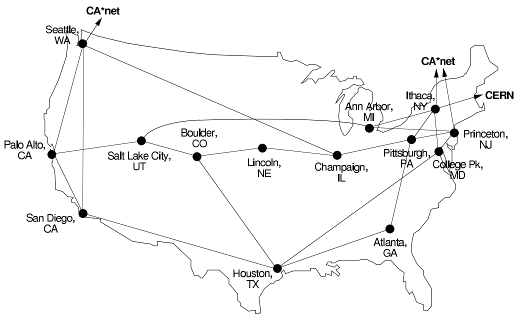


Figure 1: The NSF network

is created. After routing the highest traffic demand on its shortest path in G_p^1, to route the second highest traffic demand we again try and route it in G_p^1. If there is a *satisfactory* path in G_p^1 (i.e. if its length is less than h), the lightpath is routed in G_p^1 even though there may be a shorter path in the original graph G which we could have used if we routed the lightpath on a higher layer as in TSO_SP. If there is no satisfactory path in G_p^1, a second copy of G_p, called G_p^2, is created on which we route the lightpath and assign to it wavelength 2. For each subsequent traffic demand, we search for the shortest path in each existing graph in sequential order until the first satisfactory route is found. If no satisfactory route is available on any graph and there are less than W graphs, a new one is created. If there already exist W graphs and the traffic demand cannot be routed on any graph $G_p^1, \ldots, G_p^W$, the corresponding lightpath is not established.

The motivation for sequentially 'filling up' wavelengths as described above is to minimize the total number of distinct wavelengths used. This leaves more room for future expansion of the virtual topology. Routing lightpaths in such a manner may result in longer physical paths as a trade off to using less wavelengths. This problem is solved by bounding the physical length of the lightpaths (value h) with an acceptable value. Separate hop bounds for each source - destination could also be specified. Furthermore, TSO_FS is faster than TSO_SP since it routes lightpaths on the first found satisfactory route instead of searching for the overall shortest path. This difference in execution time may be significant for larger networks, particularly when a large number of wavelengths and tranceivers are available.

The pseudocode is as follows:

Input:
$G_p = (V, E_p)$;*//physical network*
Λ;*//$|V| * |V|$ long term traffic matrix*
W; *//available wavelengths*
Tr, Re;*//available transmitters and receivers respectfully at each node*
h; *//max physical length of lightpath*
Initialization:
$Transmitters := \{t_1, \ldots, t_{|V|}\}$, $t_i = Tr$, $i = 1, \ldots, |V|$; *//available transmitters per node*
$Receivers := \{r_1, \ldots, r_{|V|}\}$, $r_i = Re$, $i = 1, \ldots, |V|$; *//available receivers per node*
$k = 1$; *//index of the potential lightpath under consideration*
Begin:
Sort traffic demands between source - destination pairs in Λ in decreasing order creating a list $\tau = \{(s_1, d_1), \ldots, (s_{|\tau|}, d_{|\tau|})\}$ of potential lightpaths, where $|\tau| = |V| * (|V| - 1)$;
Create 1 copy (layer) of $G_p : G_p^1$;
$GRAPHS := \{G_p^1\}$;
while $k \leq |V| * (|V| - 1)$ and there are available tranceivers **do**
  For first demand (s_k, d_k) in τ, find the shortest path in each graph in $GRAPHS$ in sequential order until the first satisfactory path P_k is found
  if the length of $P_k < h$ **then**
    Establish lightpath (i, j), where $i = s_k$ and $j = d_k$;
    If routed on graph G_p^w, delete from G_p^w all edges in P_k and assign wavelength w to lightpath (i, j);
    $t_i = t_i - 1$; $r_j = r_j - 1$;
  else if the length of $P_k \geq h$ and $|GRAPHS| < W$ **then**
    Create $G_p^{|GRAPHS|+1}$ and route lightpath (i, j), where $i = s_k$ and $j = d_k$, on shortest path P_k in $G_p^{|GRAPHS|+1}$;
    Delete edges in P_k from $G_p^{|GRAPHS|+1}$
    Assign wavelength $(|GRAPHS| + 1)$ to lightpath (i, j);
    Add $G_p^{|GRAPHS|+1}$ to $GRAPHS$;
    $t_i = t_i - 1$; $r_j = r_j - 1$;
  end if
  Remove (s_k, d_k) from τ;
  $k = k + 1$;
end while
End

3.3 The $TSBS\_SP$ algorithm

The $TSBS\_SP$ algorithm is a virtual topology design algorithm where Traffic is Sorted By Source and routed on the Shortest Path available. When resources are scarce it seems the above mentioned algorithms could obtain unconnected or poorly connected virtual topologies where lightpaths are concentrated around a small number nodes with high traffic. Such solutions may be infeasible. Intuitively it seems that a virtual topology more evenly spread out among the nodes may perform better with respect to the average virtual hop distance, particularly when resources are vary scarce. This line of thought is the basis for the $TSBS\_SP$ algorithm.

The $TSBS\_SP$ algorithm essentially works the same way as the TSO_SP algorithm, but sorts the traffic demands (potential lightpaths) differently. Here the traffic originating from each source is sorted separately. In other words, we have $|V|$ separate lists, one for every node, each containing $|V| - 1$ traffic demands to all of the remaining nodes in decreasing order. A new list is created by taking the highest traffic demand from each node, starting with the highest one overall, and continuing in decreasing order. Then the second highest traffic demand from each node is selected and so on. This procedure is repeated until all the traffic demands are in the list. The remaining steps of the algorithm are identical to those of the TSO_SP algorithm described in section 3.1. The pseudocodes of the algorithms are the same except for the method of sorting the traffic demands.

3.4 The $TSBS\_FS$ algorithm

The $TSBS\_FS$ algorithm is a virtual topology design algorithm where Traffic is Sorted By Source and routed on the First Satisfactory path available. The $TSBS\_FS$ algorithm sorts the traffic demands as done by the $TSBS\_SP$ algorithm but routes lightpaths on the first satisfactory route as done by the TSO_FS algorithm. The pseudocode of the $TSBS\_FS$ algorithm is identical to that of TSO_FS except for the method of sorting the traffic demands.

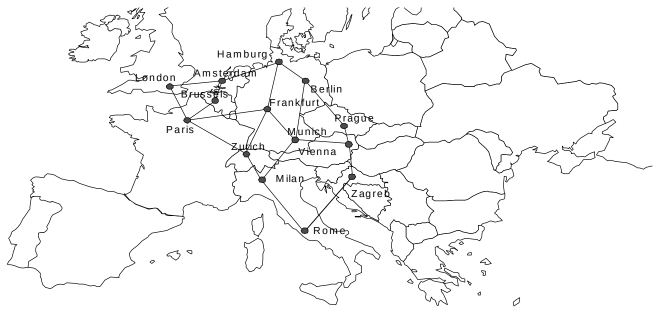
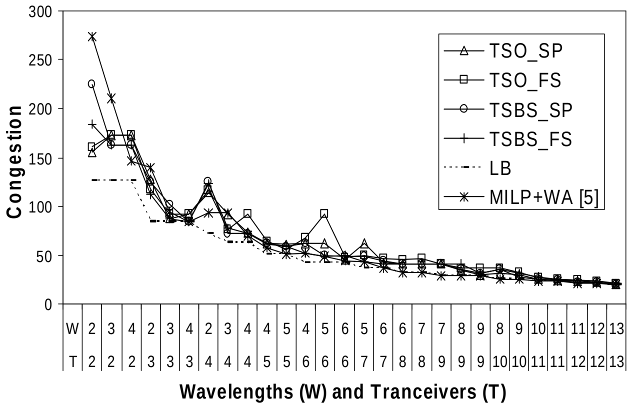


Figure 2: The European optical core network

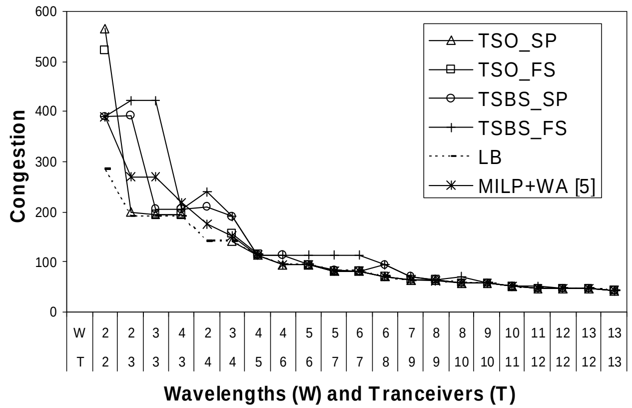
4 Numerical Results

The TSO_SP, TSO_FS, $TSBS\_SP$, and $TSBS\_FS$ algorithms for the $VRWA$ problem were implemented in C++ and run on a PC powered by a P4 2.8GHz processor. To solve the LP for traffic routing, the CPLEXv6 solver was used. The algorithms were tested on data from the NSF US backbone network and a reference pan-European core topology designed as part of the COST Action 266 project [4], shown in Figures 1 and 2 respectively. Both of these networks consist of 14 nodes. The algorithms were tested for two traffic matrixes: p1 and p2 which correspond to Tables III and IV in [9]. These traffic matrixes were used to test the $MILP + WA$ heuristic in [5]. In traffic matrix p1, most of the traffic is concentrated around 42 pairs of nodes while traffic in p2 is more evenly distributed.

Furthermore, the algorithms were tested on 5 randomly generated 30 node networks where the probability P_e of there being an edge between node was set to 0.2. Two types of traffic matrixes were generated for each test case. The first was a uniform traffic matrix where traffic was uniformly distributed over $[0, 100]$ for each source-destination pair. The second type of traffic matrix, which we will refer to as nonuniform, was generated by the method used in [3] where a fraction F of the traffic is uniformly distributed over $[0, C/a]$ while the remaining traffic is uniformly distributed over $[0, C*\Upsilon/a]$. The values were set to $C = 1250$, $a = 20$, $\Upsilon = 10$ and $F = 0.7$ as in [3].



(a)



(b)

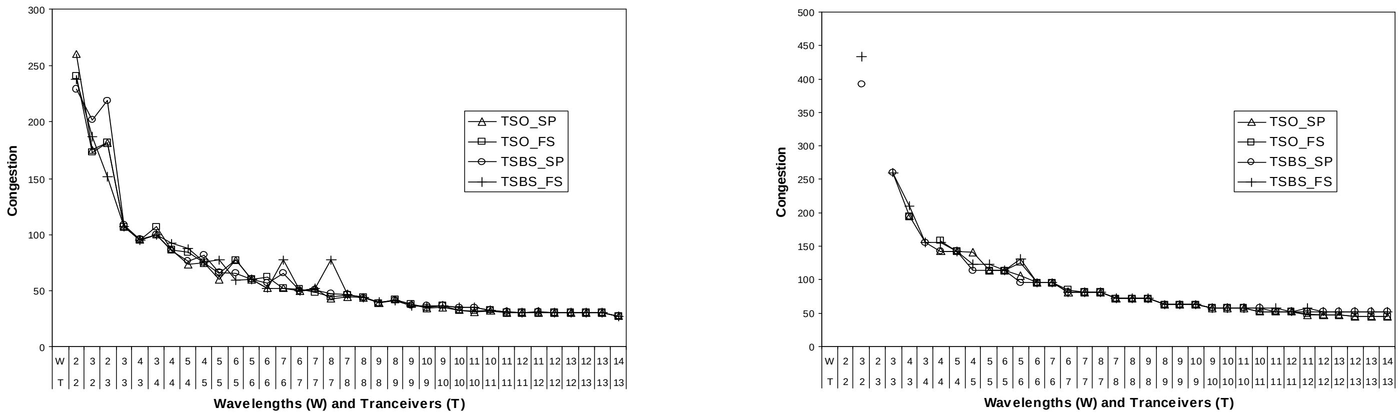
Figure 3: Comparison of congestion of the solutions obtained by the TSO_SP, TSO_FS, $TSBS\_SP$, $TSBS\_FS$, and $MILP+WA$ [5] heuristics and the lower bound (LB) for traffic matrix (a) p1 and (b) p2 in the NSF network.

We ran the TSO_SP, TSO_FS, $TSBS\_SP$, and $TSBS\_FS$ algorithms with the number of tranceivers T ranging from 2 to 13. T represents the number of transmitters and the number of receivers at each node, i.e. we assume that $Tr = Re = T$. T is therefore the maximum in-degree as well as the maximum out-degree of each node in the virtual topology. The number of wavelengths W ranged from $W = T - 1$ to $W = T + 1$ for each value of T. Value h which restricts the physical length of a lightpath was set to $\max(diam(G_p), \sqrt{|E_p|})$ as suggested in [6] for a routing and wavelength assignment algorithm. Length was considered in terms of hops.

In addition, to compare the congestion of the solutions obtained by the proposed algorithms with those obtained by the $MILP + WA$ algorithm for the NSF network, we ran the proposed algorithms with the values T and W corresponding to those in [5]. In Fig. 3, we plot the corresponding values for congestion for traffic matrixes (a) p1 and (b) p2. The lower bound on congestion, denoted as LB, and the congestion obtained by the $MILP + WA$ heuristic are taken from Tables V and VI in [5].

The TSO algorithms performed better than the $TSBS$ algorithms for traffic matrix p2 but had a few peaks when run for traffic matrix p1. All the greedy algorithms give similar results as the number of tranceivers and wavelengths increases and for many cases are close to or equal to those obtained by the $MILP + WA$ heuristic. For a larger number of wavelengths, they often give the optimal solution (i.e. the lower bound), particularly for traffic matrix p2. For some cases when resources are very scarce, the greedy algorithms perform better than $MILP + WA$.

It is important to note that for many cases the $MILP + WA$ algorithm uses more than the available number of wavelengths since wavelength assignment is performed subsequently. This occurs for the following test cases. For p1: (T=3, W=2), (T=4, W=2), (T=6, W=4), (T=6, W=5), (T=7, W=5), (T=9, W=7), (T=9, W=8), (T=10, W=8), and (T=12, W=11), while for p2: (T=2, W=1), (T=4, W=2), (T=6, W=4), (T=7, W=5), (T=9, W=7), (T=10, W=8), (T=10, W=9),



(a) (b)

Figure 4: Comparison of congestion of the solutions obtained by the TSO_SP, TSO_FS, $TSBS\_SP$, and $TSBS\_FS$ heuristics for traffic matrix (a) p1 and (b) p2 in the European core network.

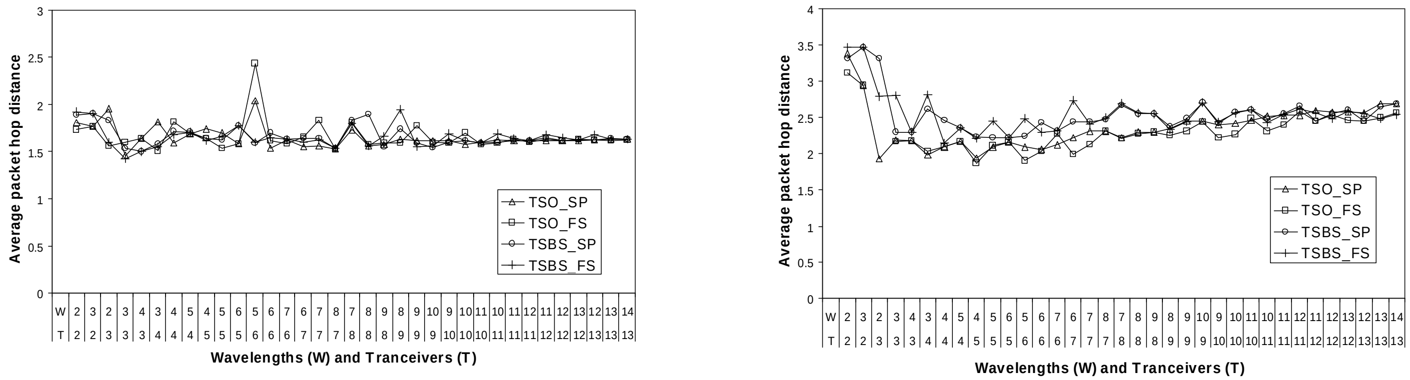
(T=12, W=11), and (T=12, W=12). These solutions are thus infeasible.

The execution times between the $MILP + WA$ algorithm and the proposed greedy algorithms differ substantially. As mentioned in [5], the average execution time to solve the relaxed MILP in the $MILP+WA$ heuristic took about 5 minutes on an IBM 43P/RS6000. This is done iteratively 25 times and then a rounding heuristic is used which runs about 1 minute. The execution time of the wavelength assignment heuristic is not mentioned. This means that the average execution time of $MILP + WA$ was at least 5*25+1 = 126 minutes = 2 hours and 6 minutes. The average execution times of the TSO_SP, TSO_FS, $TSBS\_SP$, and $TSBS\_FS$ algorithms for the same test cases when run on a PC powered by a P4 2.8GHz processor were all under half a second[4].

In Fig. 4 we plot the congestion of the solutions obtained for the European core network for traffic matrixes (a) p1 and (b) p2 by each of the proposed algorithms. All 4 algorithms behave similarly although the SP algorithms perform slightly better than the FS algorithms which tend to have peaks for some test cases.

All four algorithms performed almost the same with respect to congestion when run for the 30 node networks with nonuniform and uniform traffic. Namely, the 30 node networks, which have an average degree of 6, are denser and better connected than the NSF and European core networks. In such well connected networks there exist several edge disjoint paths and therefore it is possible to set up a several lightpaths even when wavelengths are scarce. In other words, for most cases the algorithms terminated when all the available tranceivers were exhausted and not as a result of the lack of wavelengths. This means that the vir-

[4] Note that the same LP was used for Traffic Routing in the $MILP$+WA heuristic and in combination with the TSO_SP, TSO_FS, $TSBS\_SP$, and $TSBS\_FS$ algorithms so these execution times were omitted. They ranged from 1 to 20 seconds depending on values T and W.



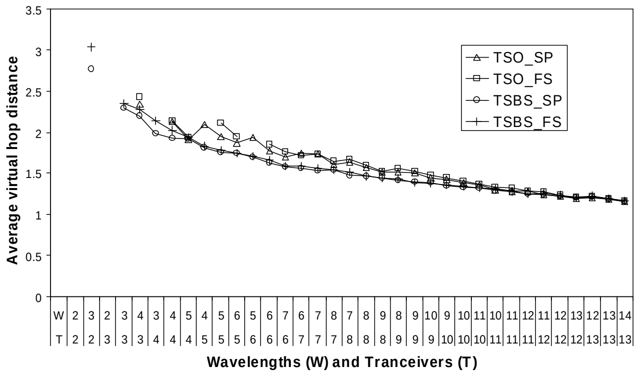
(a) (b)

Figure 5: Comparison of average packet hop distance of the solutions obtained by the TSO_SP, TSO_FS, $TSBS\_SP$, and $TSBS\_FS$ algorithms for traffic matrix (a) p1 and (b) p2 in the NSF network.

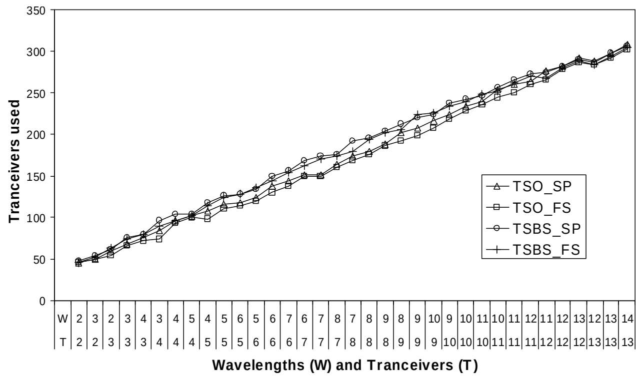
tual topologies obtained by each of the algorithms consist of the same number of lightpaths. Since several alternative paths are available and it is thus unlikely that a lightpath be rejected due to unavailability of a physical route, the method of routing (i.e. SP and FS) does not make much of a difference with respect to the obtained virtual topology. Furthermore, as the number of tranceivers at each node increases, the method of sorting lightpaths does not make a significant difference either with respect to the lightpaths established. It follows that the congestion, average packet hop distance, virtual hop distance and the number of tranceivers used do not differ significantly since these measures are functions of the obtained virtual topology. The differences in the behavior of the proposed algorithms for the 30 node networks are evident with respect to other optimization criteria such as the number of wavelength used, physical hop length of the lightpaths and execution time. Here, the SP and FS aspects of the algorithms play a significant role.

When the algorithms were run for the NSF and European core networks, they usually terminated when all the available wavelengths were exhausted. As a result, the established virtual topologies differed somewhat leading to differences in congestion, average packet hop distance, average virtual hop distance and the number of tranceivers used. The average packet hop distances of the solutions obtained by each of the proposed algorithms in the NSF network are shown in Fig. 5. The results for the European core network are analogous. We can see from the graphs that the TSO algorithms perform better than the $TSBS$ algorithms for most cases, particularly for traffic matrix p2. This makes sense since the main objective of the TSO algorithms is to establish lightpaths between nodes with the overall highest traffic without considering overall connectivity. As a result they maximize single hop traffic.

Sorting the traffic overall may not be desirable if traffic is prone to change since the obtained virtual topology may be very poorly connected or even unconnected. This is particularly true for cases where the current traffic matrix has zero or very little traffic flowing between some pairs of nodes as is the case in traffic matrix p2. In Fig 6.(a), the average virtual hop distances of the solutions obtained by the TSO_SP, TSO_FS, $TSBS\_SP$, $TSBS\_FS$ algorithms for the European core network for traffic matrix p2 are shown. For test cases where there is no



(a)



(b)

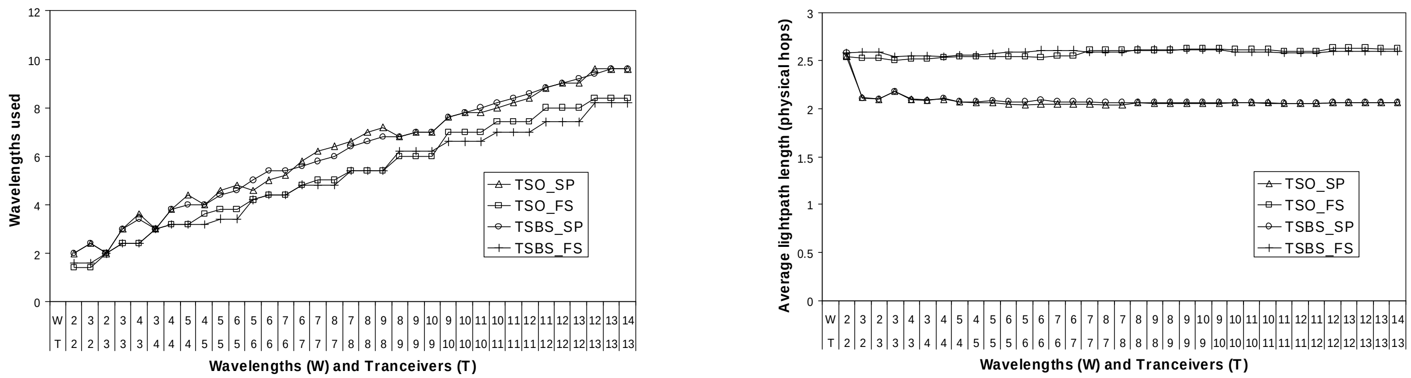
Figure 6: Comparison of (a) the average virtual hop distance and (b) the number of tranceivers used by the TSO_SP, TSO_FS, $TSBS\_SP$, and $TSBS\_FS$ algorithms in the European core network for traffic matrix p2.

point plotted, the corresponding algorithm did not obtain a feasible solution (i.e. a connected virtual topology) and thus the average virtual hop distance is infinite. We can see that the $TSBS$ algorithms yield virtual topologies that are better connected overall and as a result may perform better as the traffic matrix changes. The situation is analogous but to a lesser degree for traffic matrix p1.

To determine the behavior of the algorithms with respect to virtual connectivity for cases when resources are very scarce, we ran each algorithm for the number of tranceivers ranging from 2-5 and the number of wavelengths ranging from 2-4. The cases where the algorithms failed to find feasible solutions for the NSF and European core networks for traffic matrix p2 are shown in Table 1. (The column entitled $HLDA^*$ will be explained later on.) We can see that the TSO_SP and TSO_FS algorithms yielded infeasible solutions for the NSF network in 4 and 2 cases respectively, while the $TSBS\_SP$ and $TSBS\_FS$ algorithms obtained feasible solutions in all cases. For the European core network, both TSO algorithms yielded unconnected virtual topologies in 9 cases, while the $TSBS$ algorithms did so in only 3 cases. All four algorithms obtained feasible solutions for all cases for traffic matrix p1.

It is logical that the virtual topologies that are better connected overall establish more lightpaths and therefore use more tranceivers. The number of tranceivers used to create virtual topologies in the European core network for traffic matrix p2 are shown in Fig. 6.(b). We can see that the TSO algorithms use fewer tranceivers than the $TSBS$ algorithms but as a trade-off to virtual hop distance. Since the algorithms behave analogously with respect to virtual hop distance and the number of tranceivers used for the NSF network, these results are omitted for lack of space.

Recall that all four algorithms usually terminated due to lack of available wavelengths when run for the NSF and European core networks. As a result, they use the same number of distinct wavelengths. On the other hand, when run for the 30 node networks the algorithms terminated when all the tranceivers were used up. The average number of wavelengths used in the 30 node networks for nonuniform traffic are shown in Fig. 7.(a). We can clearly see the FS algorithms use fewer wavelengths than the SP algorithms. Recall that congestion, average packet and virtual hop distance and the number of tranceivers used were almost the same for the 30 node networks. Therefore, to establish



(a) (b)

Figure 7: Comparison of (a) the average number of distinct wavelengths used and (b) the average physical length of the established lightpaths in the 30-node networks for nonuniform traffic.

virtual topologies which perform equally well, the FS algorithms use significantly fewer wavelengths than the SP algorithms. This leaves more room for future expansion of the virtual topology.

Table 1: Cases where the algorithms failed to find a feasible solution in the (a) NSF and (b) European core networks for traffic matrix p2. (Cases marked 'x' are those where the obtained solutions were infeasible.)

(A)

T	W	TSO_SP	TSO_FS	TSBS_SP	TSBS_FS	HLDA*
2	2	x				x
2	3					
2	4					
3	2	x				x
3	3					x
3	4					
4	2	x	x			x
4	3					
4	4					
5	2	x	x			x
5	3					
5	4					

(B)

T	W	TSO_SP	TSO_FS	TSBS_SP	TSBS_FS	HLDA*
2	2	x	x			x
2	3	x	x			x
2	4	x	x			x
3	2	x	x	x	x	x
3	3					
3	4	x	x			x
4	2	x	x	x	x	x
4	3	x	x			x
4	4					
5	2	x	x	x	x	x
5	3	x	x			x
5	4					x

Since the FS algorithms route lightpaths on satisfactory but not necessarily shortest paths, it is to be expected that their corresponding physical hop lengths will be longer. The average lengths of the established lightpaths in the 30 node networks for nonuniform traffic are shown in Fig. 7.(b). Since we limit the hop distance to an acceptable value using parameter h, this is not necessarily a problem. The results obtained with respect to wavelengths and physical lengths of lightpaths for the 30 node networks with uniform traffic are analogous to those with nonuniform traffic.

To determine the benefits of establishing multiple lightpaths between source-destination pairs with respect to the various objective criteria, we implemented the well known $HLDA$ algorithm [9] and compared it to TSO_SP. $HLDA$ sorts traffic overall, routes it on the shortest path available but also allows multiple lightpaths to be established between nodes with heavy traffic. We eliminated step 4 of the $HLDA$ algorithm which randomly establishes lightpaths until all the tranceivers or wavelengths are exhausted. This step is eliminated since two of our objectives for the $VRWA$ problem are to minimize the number of tranceivers and wavelengths used. We refer to this algorithm as $HLDA^*$.

In Fig. 8 we plot the (a) congestion, (b) average packet hop dis-

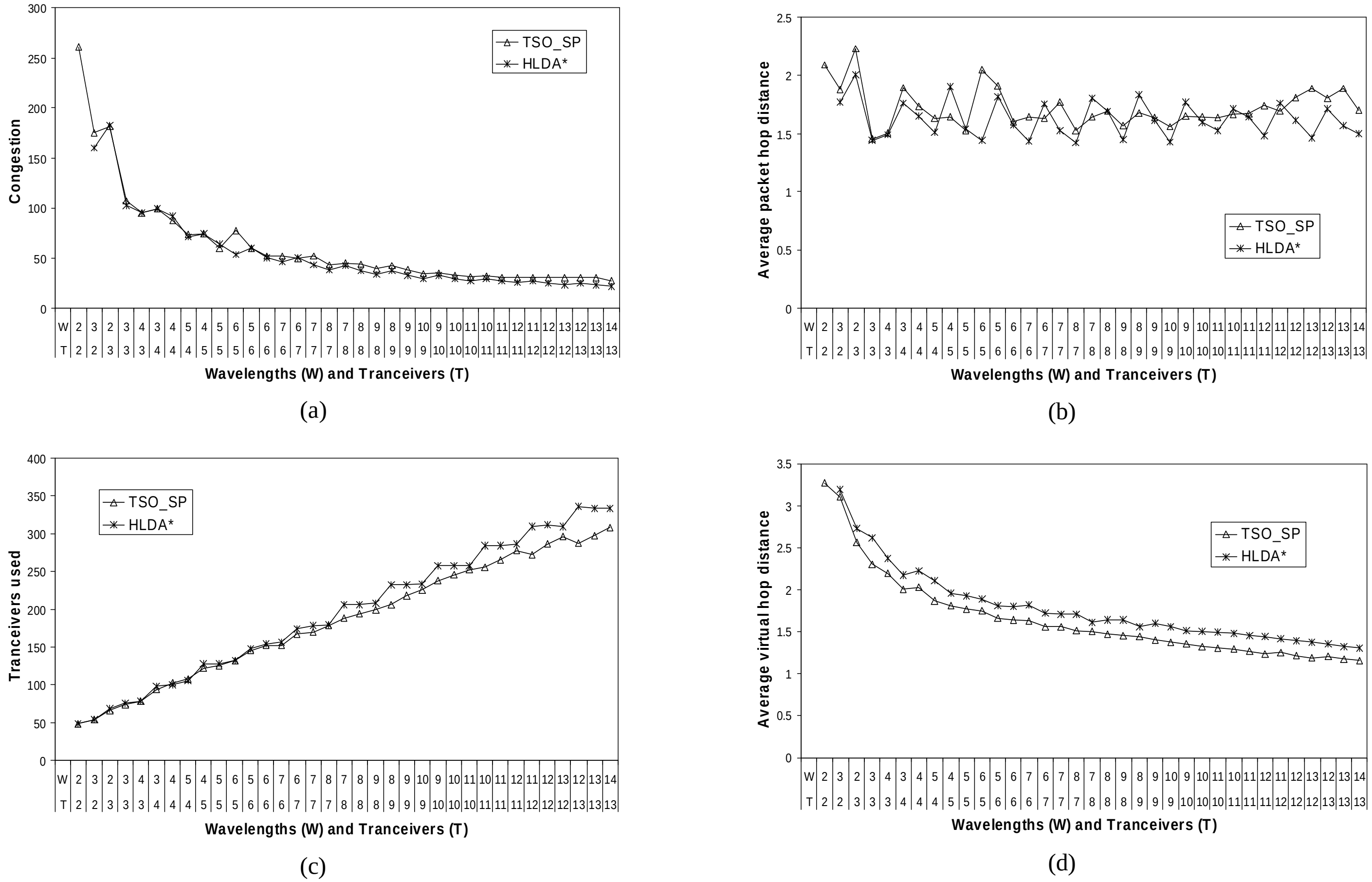


Figure 8: Comparison of the (a) congestion and (b) average packet hop distance, (c) tranceivers used and (d) average virtual hop distance of the solutions obtained by the TSO_SP and $HLDA^*$ algorithms for traffic matrix p1 in the European core network.

tance, (c) number of tranceivers used and the (d) average virtual hop distance of the solutions obtained by $HLDA^*$ and TSO_SP for the European core network for traffic matrix p1. Since the results obtained for traffic matrix p2 as well as the results obtained for the NSF network for both traffic matrixes are fairly similar, they are omitted for lack of space. With respect to congestion, $HLDA^*$ performed slightly better. This seems logical since $HLDA^*$ can establish multiple lightpaths where traffic is high. Neither algorithm performed consistently better with respect to average packet hop distance. The average packet hop distance obtained by the TSO_SP algorithm seems less dependent on the available number of wavelengths than that obtained by $HLDA^*$ which tends to vary more. On average, $HLDA^*$ obtained virtual topologies with lower average packet hop distance. In the 30 node networks, $HLDA^*$ and TSO_SP yielded virtual topologies with the same values for congestion and average packet hop distance.

The number of wavelengths used by both algorithms was the same and is therefore not plotted. The TSO_SP algorithm used less tranceivers as can be seen in Fig. 8.(c) which makes sense since it cannot set up multiple lightpaths between nodes. The average virtual hop distances of the obtained solutions are shown in Fig. 8.(d). Even though the TSP_SP algorithm established fewer lightpaths (i.e. used fewer tranceivers), it obtained virtual topologies with lower average virtual hop distances than $HLDA^*$. As a result, such virtual topologies are cheaper and yet better connected overall. For cases when wavelengths are scarce, $HLDA^*$ performed worst with respect to feasibility when compared to the proposed algorithms for traffic matrix p2 (see Table 1).

In Fig. 9, the average execution times of the algorithms when run for the 30 node networks are shown. Although all the proposed algorithms are very fast, it is evident that the execution times of the SP algorithms grow faster with the number of available resources than those of the FS algorithms. This makes sense since the SP algorithms search for the shortest path available while the FS algorithms establish a lightpath as soon as a feasible path is found. Since today there can be over 100 wavelengths available, this growth in execution time could be significant, particularly for larger networks. This could especially be important if lightpath requests arrive dynamically over time requiring new virtual topologies to be created in real time. The differences in execution times of the proposed algorithms with respect to $HLDA^*$ become substantial as the network grows. We ran the $HLDA^*$, TSO_SP and TSO_FS algorithms for a randomly generated network with 250 nodes and the probability of there being an edges between nodes 0.02. The algorithms were run for cases with up to 8 tranceivers. The corresponding execution times are as follows. The TSO_FS algorithm ran between 5-35 seconds, the TSO_SP ran between 25-135 seconds, while $HLDA^*$ ran approximately an hour.

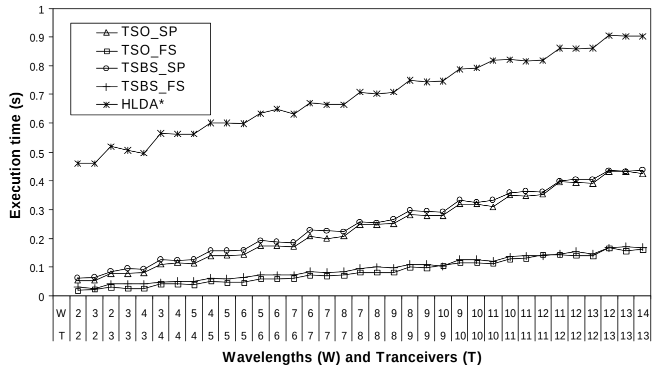


Figure 9: Comparison of average execution times of the TSO_SP, TSO_FS, $TSBS\_SP$, $TSBS\_FS$ and $HLDA^*$ algorithms when run for the 30 node networks.

5 Discussion

Several conclusions can be drawn from the obtained results. When comparing the effects of sorting the traffic demands differently, we can see that the TSO algorithms perform better than or equal to the $TSBS$ algorithms in most cases with respect to congestion and average packet hop distance. Even so, if traffic is prone to change, $TSBS$ may be the wiser choice since these algorithms yield better connected virtual topologies (i.e. have lower average virtual hop distances). Lower virtual hop distances may postpone the need for reconfiguration for a longer period of time but as a trade-off use more tranceivers increasing network cost. This is particularly true in sparse networks. If wavelengths are very scarce, $TSBS$ may also help prevent from establishing unconnected virtual topologies. Since all the algorithms are fast, for moderate size networks it may be best to run a TSO algorithm and if no feasible solution is found, run a $TSBS$ algorithm. If congestion is critical and traffic is predicted to remain constant for a long period of time, establishing multiple lightpaths as done by $HLDA^*$ may be desirable. Otherwise, the gain of using multiple lightpaths does not seem significant and yet increases the network cost by establishing more lightpaths.

The method of routing and assigning wavelengths does not significantly effect the objective criteria which are functions of the virtual topology. These include congestion and average packet and virtual hop distance. Still, the SP algorithms perform slightly better than the FS algorithms for these criteria. The main advantage of the FS algorithms over the SP algorithms is that they perform routing and wavelength assignment using less distinct wavelengths, particularly in dense networks. If virtual topology expansion is anticipated, FS may be the better choice for routing. This is certainly not the case in opaque networks since the resulting physical lengths of the established lightpaths are longer. For very large networks with many available wavelengths, using an FS algorithm may be desirable due to shorter execution times.

6 Conclusion

In order to efficiently utilize resources in wavelength routed optical networks it is necessary to successfully solve the virtual topology design problem. This problem is very complex and several aspects of the obtained solutions should be considered. In this work, efficient heuristic algorithms are proposed for the Virtual topology and Routing and Wavelength Assignment problem in networks with no wavelength conversion. The algorithms differ with respect to the order in which the lightpaths are established and the method of routing and assigning wavelengths. These variations are intended to improve the performance of the algorithms with respect to various objective criteria such as congestion, average virtual, physical and packet hop distances, and the number of tranceivers and distinct wavelengths used. A detailed analysis and testing on real and randomly generated networks with uniform and nonuniform traffic indicate the advantages and disadvantages of the suggested variations. Further avenues of research will include developing similar algorithms for virtual topology design in networks with full or limited wavelength conversion. Designing virtual topologies which support multicast traffic by establishing light trees will also be considered.

References

[1] C. Chen and S. Banerjee, "A New Model for Optimal Routing and Wavelength Assignment in Wavelength Division Multiplexed Optical Networks," in *Proc. IEEE Infocom '96*, 1996, pp.164-171.

[2] D. Banerjee and B. Mukherjee, "A Practical Approach for Routing and Wavelength Assignment in Large Wavelength-Routed Optical Newtorks," *IEEE J. Select. Areas Commun.*, vol. 14, no. 5, pp. 903-908, June 1996.

[3] D. Banerjee and B. Mukherjee, "Wavelength-Routed Optical Networks: Linear Formulation, Resource Budgeting Tradeoffs, and a Reconfiguration Study," *IEEE/ACM Trans. Networking*, vol. 8, no. 5, pp. 598-607, Oct. 2000.

[4] R. Inkret, A. Kuchar, B. Mikac "Advanced Infrastructure for Photonic Networks: Extended Final Report of COST Acteion 266," Zagreb: Faculty of Electrical Engineering and Computing, University of Zagreb, 2003. pp. 19-21.

[5] R. M. Krishnaswamy and K. N. Sivarajan, "Design of logical topologies: a linear formulation for wavelength-routed optical networks with no wavelength changers," *IEEE/ACM Trans. Networking*,vol. 9, no. 2, pp.186-198, April 2001.

[6] P. Manohar, D. Manjunath and R. K. Shevgaonkar, "Routing and Wavelengths Assignment in Optical Networks From Edge Disjoint Paths Algorithms," *IEEE Communication Letters*, vol. 6, no. 5, pp. 211-213, May 2002.

[7] P. Manohar, D. Manjunath and R. K. Shevgaonkar, "Effect of Objective Function on Performance in Wavelength Routed Optical Networks," *Proc of Eighth National Communications Conference (NCC-2002)* , Mumbai, Jan 2002.

[8] B. Mukherjee, D. Banerjee, S. Ramamurthy and A. Mukherjee, "Some Principles for Designing a Wide-area WDM Optical Network," *IEEE/ACM Trans. Networking*,vol. 4, no. 5, pp.684-696, Oct 1996.

[9] R. Ramaswami and K. N. Sivarajan, "Design of Logical Topologies for Wavelength-Routed Optical Networks," *IEEE J. Select. Areas Commun.*, vol. 14, no. 5, pp. 840 - 851, June 1996.