Improved Lightpath (Wavelength) Routing in Large WDM Networks

Weifa Liang[1,2] George Havas[1] Xiaojun Shen[3]

Centre for Discrete Mathematics and Computing
[1] Department of Computer Science and Electrical Engineering
[2] Department of Mathematics
The University of Queensland, Queensland 4072, Australia

[3] Computer Science Telecommunications Program
University of Missouri - Kansas City
Kansas City, MO 64110, USA

Abstract

We address the problem of efficient circuit switching in wide area networks. The solution provided is based on finding optimal routes for lightpaths and semilightpaths. A lightpath is a fully optical transmission path, while a semilightpath is a transmission path constructed by chaining several lightpaths together, using wavelength conversion at their junctions. The problem thus is to find an optimal lightpath/semilightpath in the network in terms of the cost of wavelength conversion and the cost of using the wavelengths on links. We first present fast, efficient algorithms both for the general problem and for a natural restricted version. The new algorithms outperform earlier work, providing time improvements amounting to an almost linear time factor in most cases. Also, all our algorithms can be implemented on the network in a distributed way.

Keywords: Optical networks, optimal semilightpaths, algorithm design and analysis, graph theory.

1 Introduction

The emerging optical network offers the possibility of interconnecting hundreds of thousands of users, covering local to wide area, and providing capacities exceeding substantially those conventional networks. The network promises data transmission rates several orders of magnitudes higher than current electronic networks. The key to high speed in the network is to maintain the signal in optical form rather than traditional electronic form. The high bandwidth of fiber-optic links is utilized through *Wavelength-Division Multiplexing* (WDM) technology which supports the propagation of multiple laser beams through a single fiber-optic link provided that each laser beam uses a distinct optical wavelength. Major applications of such networks are video conferencing, scientific visualization, real-time medical imaging, supercomputing, and distributed computing [2, 12, 13]. A comprehensive overview of the physical theory and applications of this technology can be found in the books by Green [8] and McAulay [11].

One major topic in WDM networks is routing. Message transfer in such networks is through first establishing the *lightpath* and then proceeding with the transfer. Lightpaths thus provide a powerful approach to utilize the vast available bandwidth in WDM networks [1, 5, 10]. A lightpath is an all-optical transmission path between two nodes in the network, implemented by assigning a unique wavelength throughout the path. Data transmitted through a lightpath do not need wavelength conversion or electronic processing at intermediate nodes. Therefore, lightpaths enable an efficient utilization of the optical bandwidth in WDM networks, and reduce electronic processing delay at the intermediate nodes, thereby improving reliability and the quality of the services provided to data communications.

While transmitting all traffic between every pair of nodes over lightpaths is desirable, it is not generally feasible to establish lightpaths between every pair of nodes and accommodate all the traffic by lightpaths, due to physical constraints such as limited number of wavelengths, limited number and tunability of optical transceivers at each node as well as lightwave dispersions that limit the physical length of a lightpath. Additionally, given the network conditions, in general a single optical wavelength may not be available between a given source and the destination because some of the resources are already occupied by existing lightpaths. To cope with the above limits, Chlamtac et al [4] introduced the *semilightpath* concept in which a transmission path is obtained by establishing and chaining several lightpaths together. Thus, wavelength conversions on a semilightpath are required at some intermediate nodes, but generally not at every node. A lightpath is a special case of the semilightpath when the number of intermediate nodes on the path with wavelength conversion is zero.

The objective of this paper is to present algorithms for finding an efficient routing semilightpath between a given source and destination such that the cost sum of links and nodes of the path is minimum in terms of the following cost measurements: 1) the cost for traversing a link on some wavelength; and 2) the cost for wavelength conversion when the path has to switch to a different wavelength at some intermediate nodes.

Clearly if we only consider the first cost factor, the problem becomes the traditional single-source shortest path problem which has been well studied. The additional cost for wavelength conversion at nodes, however, makes the direct application of the single-source shortest path algorithms to the network graph inappropriate. In this paper we show that, through a series of appropriate transformations, the problem can be reduced to a single-source shortest path problem.

The above problem was first formalized by Chlamtac et al [4]. They presented an $O(k^2n + kn^2)$ time algorithm (which we denote Algorithm CFZ) for it, where n is the number of nodes and k is the number of wavelengths in the network. Their algorithm is optimal when $k \geq n$ and the network is a dense network, i.e. $m = O(n^2)$, where m is the number of links in the network. However, their algorithm is a centralized algorithm. It may not be suitable for distributed computing environments. Besides, their algorithm does not take into account some important network parameters such as the number of links m and the maximum degree d of the nodes in the network. Since the considered network is a large wide area network, it is usually a sparse, planar or approximately planar graph, thus, $m = O(n)$. In practice d is usually a constant. Even if d is not a constant, it should be a very slowly increasing function of n, e.g. $O(\log n)$, say. Taking these network parameters into account, we may ask whether there is a faster algorithm for the problem. Motivated by these concerns, we answer this question by presenting faster algorithms which take the above network parameters into consideration.

We first present an efficient algorithm for the problem, which requires $O(k^2n + km + kn\log(kn))$ time, where k is the number of wavelengths, and n and m are the number of nodes and links in the network. We mention another simple algorithm which takes $O(k^2n\log k + km\log(kn))$ time. Our algorithms can be implemented on the network in a distributed way. The communication and time complexities of the distributed version of our algorithms are $O(km)$ and $O(kn)$, respectively. Then we introduce a restricted version of the problem in which the number of wavelengths assigned to every link is bounded by k_0. Assume that k_0 is strictly less than k; in most cases k_0 is a very slowly increasing sublinear function of n even if k is quite large (e.g. $k > n$). For this case, we present an algorithm which takes $O(d^2nk_0^2 + mk_0\log n)$ time. It is nice that this time complexity is independent of k.

Compared with Algorithm CFZ, our algorithms have two major advantages. First, we take into account the physical topology of the network which makes our first algorithm outperform Algorithm CFZ in most cases. In particular, when k is small (e.g. $k = O(\log n)$) and $m = O(n)$, our first algorithm runs in time $O(n\log^2 n)$, while Algorithm CFZ runs in time $O(n^2\log n)$. (Our second algorithm is also better than Algorithm CFZ in most cases, except where either k is quite large (e.g. $k \geq cn/\log n$) or the network is very dense (e.g. $m = O(n^2)$), but this rarely happens in practice). Second, because our algorithms have high locality, they can be efficiently implemented on the network in a distributed fashion.

2 The Model and the Problem

The optical network is modeled by a directed graph $G = (V, E)$, where V and E are a set of nodes and a set of directed links of the network. Let $n = |V|$ and $m = |E|$. Let $d_{in}(G, v)$ and $d_{out}(G, v)$ be the in-degree and out-degree of v in G. Define $d_{in} = \max\{d_{in}(G, v) \mid v \in V\}$, $d_{out} = \max\{d_{out}(G, v) \mid v \in V\}$, and $d = \max\{d_{in}, d_{out}\}$ the *maximum degree* of G. It is well known that $\sum_{v\in V} d_{in}(G, v) = \sum_{v\in V} d_{out}(G, v) = m$. Clearly, $m \leq dn$. Note that the undirected version of the network can be modeled by replacing an undirected link with two oppositely directed links.

Suppose that a set $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_k\}$ of wavelengths is available for the network. Following Chlamtac et al [4], the cost of using the resources in the network is represented as follows. For each link e and wavelength λ_i a nonnegative weight $w(e, \lambda_i)$ is associated, representing the "cost" of using wavelength λ_i in link e. If λ_i is not available on the link then, the weight is infinite. The "cost" of wavelength conversion is modeled via cost factors of the form $c_v(\lambda_p, \lambda_q)$, which is the cost of wavelength conversion at node v from wavelength λ_p to wavelength λ_q. For some given v, p and q, if the conversion at v is not available, then $c_v(\lambda_p, \lambda_q)$ is infinite. If the two wavelengths are equal (i.e, $p = q$), then $c_v(\lambda_p, \lambda_p) = 0$. The above defined wavelength conversion costs accommodate the general case where conversion cost depends on the nodes and the wavelengths involved.

A *semilightpath* $\mathcal{P}$ is defined as a sequence $e_1, e_2, \ldots, e_l$ of directed links such that the tail of e_{i+1} coincides with the head of e_i, $i = 1, \ldots, l-1$. Further, a wavelength $\lambda_{j_i} \in \Lambda$ is associated with each e_i, which is the wavelength used by the path on link e_i. The *cost* $C(\mathcal{P})$ of the semilightpath $\mathcal{P}$ is thus defined as follows. Denote by $head(e)$ and $tail(e)$ the head and the tail of a directed link e, which are the two endpoints of e. Then the cost is

$$C(\mathcal{P}) = \sum_{i=1}^{l} w(e_i, \lambda_{j_i}) + \sum_{i=1}^{l-1} c_{head(e_i)}(\lambda_{j_i}, \lambda_{j_{i+1}}) \quad (1)$$

where the first sum in Equation 1 is the cost of traversing the links and the second sum is the cost of wavelength conversion at intermediate nodes.

The *optimal semilightpath problem* then is defined as follows. Given a directed network $G = (V, E)$, let $\Lambda(e)$ be the set of available wavelengths associated with each directed link $e \in E$. Suppose the wavelength conversion function at each node $v \in V$ is also given. Let s and t be two nodes in G which are the *source* and the *destination* respectively, the problem is to find a semilightpath from s to t such that the path cost defined in Equation 1 is minimized. For this problem, not only do we need to find such an optimal semilightpath, but also do we need to assign each link e on the path a specific valid wavelength $\lambda(e) \in \Lambda(e)$ and to set the wavelength conversion switch at every intermediate node on the path, if necessary.

3 The Optimal Semilightpath Problem

In this section we present an efficient algorithm for the problem. The basic idea behind our algorithm is to transform the problem to a shortest path problem on an auxiliary directed graph $G_{s,t}$ which will be defined later. Then a solution of this shortest path problem for $G_{s,t}$ corresponds to a solution for the optimal semilightpath problem on G exactly. Note that the algorithm by Chlamtac et al [4] also first constructs an auxiliary graph WG which they call the *wavelength graph* of G, then finds a solution in WG which corresponds to a solution for the original problem. However, the construction of $G_{s,t}$ is totally different from their WG. Later we will see that the performance of our algorithm is much better than theirs.

3.1 An auxiliary construction

Given the network G, there is a set $\Lambda(e)$ ($\subseteq \Lambda$) of available wavelengths for each directed link $e \in E$. We first construct a directed *multigraph* $G_M = (V_M, E_M)$ as follows. $V_M = V$, and for each directed link $e = \langle u, v\rangle$ in E construct $|\Lambda(e)|$ parallel directed links in E_M from u to v, each of which is associated with a distinct wavelength $\lambda \in \Lambda(e)$ and the weight $w(e, \lambda)$. Thus, G_M has $|V_M| = |V| = n$ nodes and $m_1 = |E_M| = \sum_{e\in E} |\Lambda(e)|$ directed links. Obviously $m_1 \le km$.

Let $E_{in}(G, v)$ and $E_{out}(G, v)$ be the set of incoming links and the set of outgoing links of v respectively. By the definition, $head(e) = v$ for each $e \in E_{in}(G, v)$ and $tail(e') = v$ for each $e' \in E_{out}(G, v)$. Now for each node $v \in V_M$, it is obvious that
$|E_{in}(G_M, v)| + |E_{out}(G_M, v)|$
$= \sum_{e\in E_{in}(G,v)\cup E_{out}(G,v)} |\Lambda(e)|$
$\le k(d_{in}(G, v) + d_{out}(G, v)) \le 2kd$.

Let $\Lambda_{in}(G_M, v)$ be the set of wavelengths for all links in $E_{in}(G_M, v)$, i.e, a wavelength λ_{j_i} belongs to $\Lambda_{in}(G_M, v)$ if there is an $e \in E_{in}(G_M, v)$ which is assigned the wavelength λ_{j_i}. Similarly, $\Lambda_{out}(G_M, v)$ is defined as the set of wavelengths for all links in $E_{out}(G_M, v)$.

For each node $v \in V_M$, we construct a directed bipartite weighted graph $G_v = (X_v, Y_v, E_v)$ as follows. For each $\lambda \in \Lambda_{in}(G_M, v)$, there is a corresponding node x in X_v, and for each wavelength $\lambda' \in \Lambda_{out}(G_M, v)$, there is a corresponding node y in Y_v. There is a directed link $e = \langle x, y\rangle \in E_v$ from $x \in X_v$ to $y \in Y_v$ if and only if one of the following conditions hold: (i) $\lambda = \lambda'$. The weight of e is assigned $w(e) = C_v(\lambda, \lambda) = 0$; (ii) $\lambda \ne \lambda'$ and wavelength conversion from λ to λ' at v is allowed. The weight of e is assigned $w(e) = C_v(\lambda, \lambda')$.

Observation 1 *Let $G_v(X_v, Y_v, E_v)$ be the bipartite weighted graph defined as above, then $|X_v| + |Y_v| = |\Lambda_{in}(G_M, v)| + |\Lambda_{out}(G_M, v)| \le 2k$ and $|E_v| \le k^2$.*

Next we construct another directed auxiliary graph $G' = (V', E')$ from G_M as follows. $V' = \bigcup_{v\in V_M}(X_v \cup Y_v)$. Let $\langle u, v\rangle \in E_M$ be any directed link with wavelength λ. Suppose that $u' \in Y_u$ and $v' \in X_v$ are the corresponding nodes of u and v in G_u and G_v. Then, $\langle u', v'\rangle \in E'$ and the weight of this link is $w(\langle u, v\rangle, \lambda)$. Let E_{org} be the set of the links of G' obtained from E_M by the above transformation, then $|E_{org}| = |E_M| = \sum_{e\in E} |\Lambda(e)| \le km$. Define $E' = \bigcup_{v\in V_M} E_v \cup E_{org}$.

Observation 2 *Let $G'(V', E')$ be the directed weighted graph defined as above, then $|V'| = \sum_{v\in V_M}(|X_v| +$*

$|Y_v|) = \sum_{v \in V_M}(|\Lambda_{in}(G_M, v)| + |\Lambda_{out}(G_M, v)|) \leq 2kn$ *and* $|E'| = \sum_{v \in V_M} |E_v| + |E_{org}|$
$\leq k^2 n + m_1 \leq k^2 n + km.$

Since we deal with large wide area networks, which are sparse in practice (i.e, they have a large n, a small m, e.g., $m = O(n)$, and a bounded degree d, etc.), the efficient representation of G uses adjacency lists.

Observation 3 *Let $G'(V', E')$ be the directed weighted graph defined as above. Then G' can be constructed in $O(k^2 n + km)$ time and space if G' is represented by the adjacency lists.*

3.2 An example

We illustrate the construction of the auxiliary graphs by an example. Assume that a network G is defined by Fig. 1.

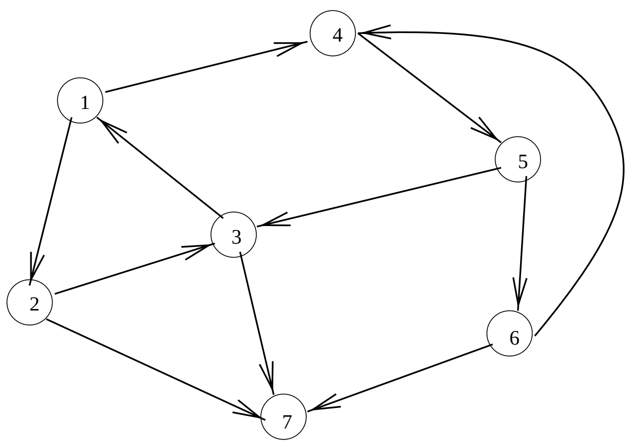

Figure 1: **The network $\mathbf{G} = (\mathbf{V}, \mathbf{E})$.**

There is a set $\Lambda = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$ of available wavelengths on G. The available wavelengths at each link of G are defined as follows.
$\Lambda(\langle 1, 2\rangle) = \{\lambda_1, \lambda_3\}$, $\Lambda(\langle 1, 4\rangle) = \{\lambda_1, \lambda_2, \lambda_4\}$,
$\Lambda(\langle 2, 3\rangle) = \{\lambda_1, \lambda_4\}$, $\Lambda(\langle 2, 7\rangle) = \{\lambda_1, \lambda_2, \lambda_3\}$,
$\Lambda(\langle 3, 1\rangle) = \{\lambda_2, \lambda_3\}$, $\Lambda(\langle 3, 7\rangle) = \{\lambda_3, \lambda_4\}$,
$\Lambda(\langle 4, 5\rangle) = \{\lambda_3\}$, $\Lambda(\langle 5, 3\rangle) = \{\lambda_2, \lambda_4\}$,
$\Lambda(\langle 5, 6\rangle) = \{\lambda_1, \lambda_3\}$, $\Lambda(\langle 6, 4\rangle) = \{\lambda_2, \lambda_3\}$,
$\Lambda(\langle 6, 7\rangle) = \{\lambda_2, \lambda_3, \lambda_4\}$.

Then, the auxiliary graph G_M of G is described as follows (see Fig. 2). Thus,
$\Lambda_{in}(G_M, 1) = \{\lambda_2, \lambda_3\}$, $\Lambda_{out}(G_M, 1) = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$,
$\Lambda_{in}(G_M, 2) = \{\lambda_1, \lambda_3\}$, $\Lambda_{out}(G_M, 2) = \{\lambda_1, \lambda_2, \lambda_4\}$,
$\Lambda_{in}(G_M, 3) = \{\lambda_1, \lambda_2, \lambda_4\}$, $\Lambda_{out}(G_M, 3) = \{\lambda_2, \lambda_3, \lambda_4\}$,
$\Lambda_{in}(G_M, 4) = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$, $\Lambda_{out}(G_M, 4) = \{\lambda_3\}$,
$\Lambda_{in}(G_M, 5) = \{\lambda_3\}$, $\Lambda_{out}(G_M, 5) = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$,
$\Lambda_{in}(G_M, 6) = \{\lambda_1, \lambda_3\}$, $\Lambda_{out}(G_M, 6) = \{\lambda_2, \lambda_3, \lambda_4\}$,
$\Lambda_{in}(G_M, 7) = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$, $\Lambda_{out}(G_M, 7) = \emptyset$.

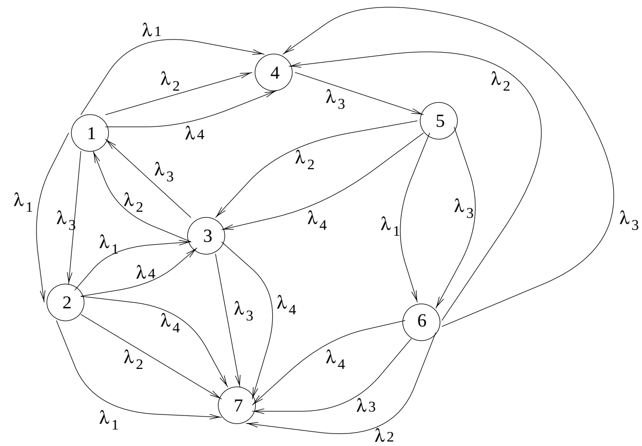

Figure 2: **The auxiliary graph $\mathbf{G_M} = (\mathbf{V_M}, \mathbf{E_M})$.**

Based on G_M, the bipartite graph G_v for every $v \in V_M = V$ is then constructed. Fig. 3 shows how to construct the bipartite graph $G_3 = (X_3, Y_3, E_3)$ at node 3. Every node labeled by "(v, λ_j)" represents the node obtained from node v of G_M and wavelength λ_j. From Fig. 3 we see that there is no link from a node in X_3 labeled by $(3, \lambda_2)$ to a node in Y_3 labeled by $(3, \lambda_3)$, which means that the wavelength conversion from λ_2 to λ_3 at node 3 is not allowed.

Assume that G_v for every $v \in V$ has been constructed, the construction of G' is easy. Fig. 4 shows a subgraph of G' induced by the nodes in G_1 and G_3. Note that there are two directed links $\langle u, v\rangle$ and $\langle p, q\rangle$ between the nodes in G_3 and G_1 in Fig. 4, which are obtained from the parallel links $\langle 3, 1\rangle$ of G_M labeled by λ_2 and λ_3 respectively. By the definition, $\{\langle u, v\rangle, \langle p, q\rangle\} \subset E_{org}$.

3.3 An efficient algorithm

Now we consider how to find an optimal semilightpath in G from s to t. First we construct a directed auxiliary graph $G_{s,t} =$
$(V' \cup \{s', t''\}, E' \cup \{\langle s', u\rangle \mid u \in Y_s\} \cup \{\langle v, t''\rangle \mid v \in X_t\})$, and assign the links $\langle s', u\rangle$ and $\langle v, t''\rangle$ weight zero. All other links in E' of $G_{s,t}$ are assigned the same weights as they have in G', where X_t and Y_s are the subsets of nodes in the bipartite graphs G_s and G_t. Obviously the number of nodes and links of $G_{s,t}$ are not more than $2kn + 2$ and $k^2 n + 2k + km$, respectively.

The construction of $G_{s,t}$ results in a straightforward 1-1 mapping between the shortest path in $G_{s,t}$ from s' to t'' and the optimal semilightpath of G from s to t. Let $\mathcal{P}_{s',t''}$ be a shortest path of $G_{s,t}$ from s' to

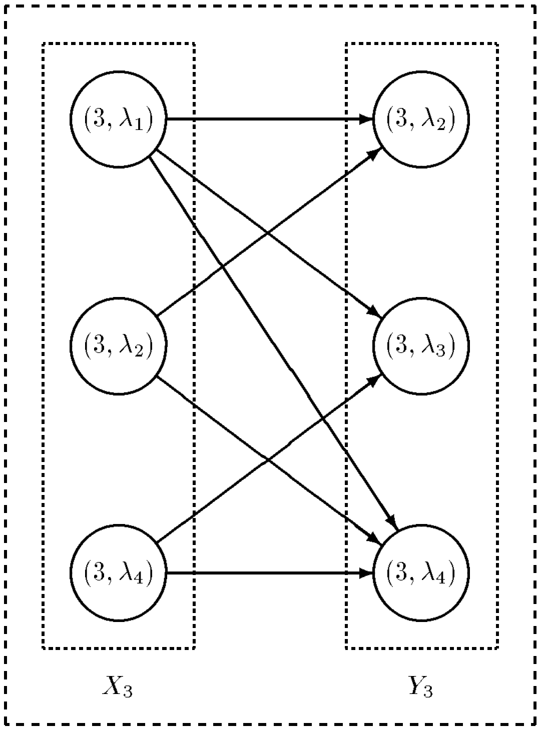

Figure 3: **The auxiliary graph $\mathbf{G_3 = (X_3, Y_3, E_3)}$.**

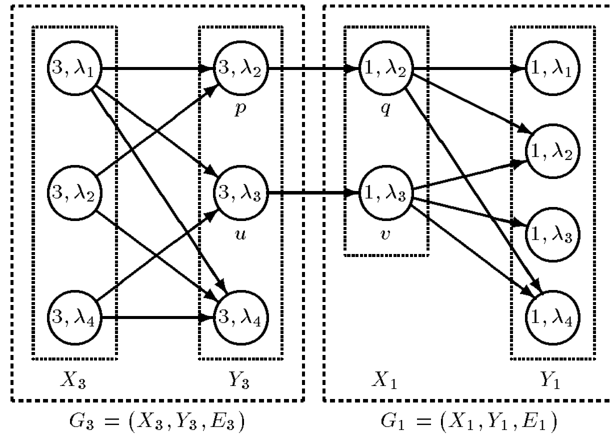

Figure 4: **A subgraph of $\mathbf{G'}$ induced by the nodes in $\mathbf{G_1}$ and $\mathbf{G_3}$.**

t'', and let $e'_1, e'_2, \ldots, e'_l$ be the link sequence of $\mathcal{P}_{s',t''}$. Then for every i, $1 \le i \le l$, if i is odd, $i \ne 1$ and $i \ne l$, and e'_i corresponds to a directed link e_i of G with weight $w(e_i, \lambda)$, then e_i is assigned with wavelength λ; otherwise, e'_i corresponds to a wavelength conversion at a node v from wavelength λ to wavelength λ', if the weight of e'_i is $c_v(\lambda, \lambda')$, i.e., set the switch at node v from λ to λ'. Hence, it remains to find a shortest path between a pair of nodes in a directed network, which has been well studied in the past.

Theorem 1 *Given a directed network $G(V, E)$ and a pair of nodes s and t, assume that each link e of G has been assigned with a set $\Lambda(e) \subseteq \Lambda$ of available wavelengths, and every node has been given the wavelength conversion function. There is an algorithm for finding an optimal semilightpath from s to t in G. The algorithm requires $O(k^2 n + km + kn\log(kn))$ time.*

Proof By the above discussion we first construct a directed weighted auxiliary graph $G_{s,t}$, which can be done in $O(k^2 n + km)$ time. Let $G_{s,t}$ be represented by adjacency lists. It is already known that the numbers of nodes and links of $G_{s,t}$ are not more than $2kn+2$ and $k^2 n + 2k + km$, respectively. Finding a shortest path in a graph H between two nodes takes $O(m' + n'\log n')$ time if the Fibonacci heap technique [7] is used (see, e.g., [6, p. 530]), where H has n' nodes and m' edges. Notice that the algorithm actually constructs a shortest tree rooted at the source node, i.e, the shortest paths from the root to all other nodes are constructed in this amount of time. So, there is an algorithm which can find the shortest path $\mathcal{P}_{s',t''}$ of $G_{s,t}$ in $O(k^2 n + km + kn\log(kn))$ time. $\Box$

Corollary 1 *Given a directed network $G(V, E)$ in which each link e of G has been assigned with a set $\Lambda(e)$ of available wavelengths, and every node has been given the wavelength conversion function, there is an algorithm for finding all pairs optimal semilightpaths of G. The algorithm requires $O(k^2 n^2 + kmn + kn^2\log(kn))$ time.*

Proof Let us recall the construction of $G_v(X_v, Y_v, E_v)$ for every $v \in V_M = V$. Now we construct another bipartite graph $G'_v = (X'_v, Y'_v, E'_v)$. $X'_v = X_v \cup \{v'\}$, $Y'_v = Y_v \cup \{v''\}$, and $E'_v = E_v \cup \{\langle v', u\rangle \mid u \in Y_v\} \cup \{\langle u, v''\rangle \mid u \in X_v\}$ and all links in $E'_v - E_v$ are assigned weight zero. Then an auxiliary directed weighted graph $G_{all} = (V_{all}, E_{all})$ can be constructed as follows. $V_{all} = \bigcup_{v\in V}(X'_v \cup Y'_v)$ and $E_{all} = \bigcup_{v\in V} E'_v \cup E_{org}$. Now it is clear that a shortest path of G_{all} from s' to t'' corresponds to an optimal semilightpath of G from s to t. There are $O(n^2)$ pairs of nodes in G. So, we need to construct n shortest trees in G_{all} rooted at v' for every $v \in V$. The construction of each such a tree takes $O(k^2 n + km + kn\log(kn))$ time if the algorithm due to Fredman and Tarjan [7] is employed because

$|V_{all}| = |V'| + 2n \leq 2kn + 2n = 2n(k+1)$ and $|E_{all}| = |E'| + \sum_{v \in V}(|X_v| + |Y_v|) \leq |E'| + 2kn \leq k^2n + km + 2kn$. There are n such trees. The corollary follows. $\square$

If n is quite large and the network is sparse with $k = o(n)$, our algorithm has much better performance than Algorithm CFZ. However our algorithm uses the Fibonacci heap data structure, which is relatively complicated to implement. If we want an easier algorithm for this problem with simple data structures, we can apply Dijkstra's algorithm for finding a shortest path directly. This can be implemented in $O(m' \log n')$ time in a sparse graph with n' nodes and m' links [6, p. 530]. Then, for our problem it takes $O(k^2 n \log(kn) + km \log(kn))$ time because the number of nodes and links in $G_{s,t}$ are $O(kn)$ and $O(k^2n + km)$ respectively. It is possible to use a better implementation of Dijkstra's algorithm (using balanced binary trees) for our problem which takes $O(k^2 n \log k + km \log(kn))$ time, by observing that $G_{s,t}$ is a special graph. The ensuing algorithm is slower than our first algorithm: it requires $O(k^2 n \log k + km \log(kn))$ time. As a compensation, it is easier to understand and implement in practice. We omit the details from this paper.

3.4 A distributed algorithm

Usually an optical network can be decomposed into two separate networks in functionality. One is called *data network* which is used to transfer large volume data such as image data and databases etc, by its optical fibers. The other one is called *control network* which is used for high level protocol controls such as finding an optimal semilightpath/lightpath and setting the optical switches along the path. Because the size of each message used for these controls is usually not large, people use electronic networks to implement high level protocol controls. Since we are dealing with large wide area networks, in practice it is not realistic to design a centralized algorithm as above to find an optimal semilightpath for the networks. Instead, a distributed algorithm seems more appropriate to the task. In most cases, if there is a request for establishing a lightpath/semilightpath, which happens quite often in on-line mode, the scheduler must respond immediately, and give a reject/accept answer. Therefore, a goal is to seek an efficient distributed algorithm on the network for finding an optimal semilightpath for such a request. Here we provide an answer.

What we do is to embed the ideal network $G_{s,t}$ into the physical network G first. We then simulate the network $G_{s,t}$ using G. Now we claim that the construction of $G_{s,t}$ has high locality, which is explained as follows. In the original network $G(V, E)$, we construct a weighted bipartite graph $G_v(X_v, Y_v, E_v)$ for every node v if $v \neq s$ and $v \neq t$. We construct a bipartite graph $G_s = (X_s \cup \{s'\}, Y_s, E_s \cup \{\langle s', u\rangle \mid u \in Y_s\})$ at node s, where all links in $\{\langle s', u\rangle \mid u \in Y_s\}$ are assigned weight zero. Similarly, the bipartite graph at node t can be constructed, $G_t = (X_t, Y_t \cup \{t''\}, E_t \cup \{\langle u, t''\rangle \mid u \in X_t\})$, where all links in $\{\langle u, t''\rangle \mid u \in X_t\}$ are assigned weight zero. Each physical link $e \in E$ of G serves as $|\Lambda(e)|$ links of $G_{s,t}$. As a result, $G_{s,t}$ is constructed and represented distributively, i.e., each node v of G holds all adjacency lists of nodes in G_v of $G_{s,t}$.

Then, the problem becomes finding a shortest path in $G_{s,t}$. Now we use G to simulate $G_{s,t}$, i.e., each node of G is actually a subgraph of $G_{s,t}$. Since the shortest path problem is a well studied problem in the distributed computing environments, there are many efficient algorithms for it, such as the algorithm by Chandy and Misra [3]. Therefore,

Theorem 2 *Given a directed network $G(V, E)$ and a pair of nodes s and t, assume that each link e of G has been assigned with a set $\Lambda(e)$ of available wavelengths, and every node has been given the wavelength conversion function. There is an efficient distributed algorithm for finding an optimal semilightpath from s to t. The communication complexity of the algorithm is $O(km)$ and the time complexity is $O(kn)$ on the traditional distributed computational model.*

Proof Since local computation is negligible in the distributed computational model, the construction of $G_{s,t}$ can be done in constant time. Then, use G to simulate $G_{s,t}$. That is, each node of G simulates a subgraph of $G_{s,t}$ with at most $2k+1$ nodes. It is already known that the communication and time complexities for finding a shortest path between two nodes in H are $O(m')$ and $O(n')$ [3], where H contains n' nodes and m' edges. Therefore, the communication and time complexities for our problem are $O(km)$ and $O(kn)$ because the links in $\bigcup_{v \in V} E_v$ in $G_{s,t}$ are *virtual links* which are located inside physical nodes of G. By the definition of this model, communications on these links are negligible. $\square$

Corollary 2 *Given a directed network $G(V, E)$, assume that each link e of G has been assigned a set $\Lambda(e)$ of available wavelengths, and every node has been given the wavelength conversion function. There is a distributed algorithm for finding all pairs optimal semilightpaths. The communication and the time complexities of the algorithm are $O(k^2n^2)$ and $O(k^2n^2)$ on the traditional distributed computational model.*

Proof We first construct the graph G_{all} which is defined in the proof of Corollary 1. The construction of G_{all} takes constant time because local computational time is negligible in the distributed computational model. We already know that a shortest path of G_{all} from s' to t'' corresponds to an optimal semilightpath of G from s to t. Therefore, finding the shortest paths in G_{all} between n^2 pairs of nodes corresponds to finding all pairs shortest paths of G. By Haldar's algorithm [9], finding all pairs shortest paths in a graph H with n' nodes can be done in $O(n'^2)$ time using $O(n'^2)$ messages. Following the same approach as in the proof of Theorem 2, we embed G_{all} in G, i.e., we use G to simulate G_{all}. It is clear that all pairs shortest paths of G_{all} can be found in $O(k^2n^2)$ time using $O(k^2n^2)$ messages, because every node of G holds a subgraph of G_{all} with at most $2k+2$ nodes. □

3.5 Comparison

We now compare our centralized algorithm with Algorithm CFZ [4] for the optimal semilightpath problem. Both algorithms are based on the decision tree model, in which only algebraic operations are allowed on the weights. We conclude that our algorithm outperform theirs in most cases, which includes all practical situations.

Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two algorithms for solving some problem, and let $T_{\mathcal{A}_1}$ and $T_{\mathcal{A}_2}$ be the execution times of $\mathcal{A}_1$ and $\mathcal{A}_2$. Then, we say that $\mathcal{A}_2$ improves $\mathcal{A}_1$ by a *factor* of $\Omega(\rho)$ if $T_{\mathcal{A}_1} = \Omega(\rho T_{\mathcal{A}_2})$.

Now let $T_{CFZ} = \Theta(kn^2 + k^2n)$ be the time complexity of their algorithm, and let T_c be the time complexity of the proposed algorithm using Fibonacci heaps. Then $T_c = O(k^2n + km + kn\log(kn))$ time.

If $k \geq n$, $T_c = T_{CFZ} = O(k^2n)$ since $m \leq dn \leq n^2$. This means that both algorithms have the same worst time complexity under this assumption (but this assumption is not realistic for large networks).

If $k < n$, $T_c = O(k^2n+km+kn\log(kn))$ and $T_{CFZ} = O(kn^2)$. Then

$$\begin{aligned} \frac{T_{CFZ}}{T_c} &= \frac{c'kn^2}{k^2n + km + kn\log(kn)} \\ &\geq \frac{c''n}{\max\{k, d, \log n\}} \end{aligned} \qquad (2)$$

where c' and c'' are constants and $m \leq dn$. We further analyze Inequality 2 to the following two cases: (i) if $\max\{k,d\} \leq c''' \log n$, then $T_{CFZ}/T_c = \Omega(n/\log n)$, which means that our algorithm has improved their algorithm by an $\Omega(n/\log n)$ factor; (ii) if $\max\{k,d\} \geq c''' \log n$, then $T_{CFZ}/T_c = \Omega(n/\max\{k,d\})$, which means that our algorithm has improved their algorithm by an $\Omega(n/\max\{k,d\})$ factor. Note that $d < n$, $k < n$, and d is usually a constant or a sublinear function of n which increases very slowly if G is a large, sparse wide area network. Overall, our algorithm has improved their algorithm by a factor of $\Omega(\frac{n}{\max\{k,d,\log n\}})$ in general. In particular, our algorithm improves on theirs a lot when k and m are relatively small but n is relatively large. For example, when $m = O(n)$ and $k = O(\log n)$, our algorithm runs in $O(n\log^2 n)$ time only, while their algorithm takes $O(n^2 \log n)$ time.

Thus our algorithm is never worse than Algorithm CFZ, and in all practical situations is much superior. Further, it is impossible in practice to run such a centralized algorithm for a very large wide area network, so a distributed version seems more appropriate. From the previous discussion, we know that our algorithm has high locality, so it can be implemented in the network in a distributed fashion. On the other hand it appears that Algorithm CFZ cannot be implemented in such a way since their auxiliary graph does not have suitable locality.

4 A Restricted Problem

In this section we analyze our algorithm for the optimal semilightpath problem under more restrictions. In the previous section we assume that, associated with each directed link e, there is a set $\Lambda(e)$ ($\subseteq \Lambda$) of wavelengths available. Though the number of wavelengths $|\Lambda| = k$ in a network may be quite large, the number of transmitters and the number of receivers (tuning) at every node usually are bounded, so is the number of wavelengths at every link. In Section 3 we assume every node can switch $O(k^2)$ pairs of wavelengths. In practice it is impossible for k to be $O(n)$ if the size n of the network is quite large. Based on this, in this section we make the reasonable assumption that the number of available wavelengths, $|\Lambda(e)|$, associated with each link e, is bounded by k_0, i.e., $|\Lambda(e)| \leq k_0$. We further assume that k_0 is either a constant or a function of n increasing very slowly, i.e., $k_0 = o(n)$ even if $k = \Omega(n)$. Under this new restriction, the aim is to find an optimal semilightpath in the network $G = (V, E)$ from s to t.

Following the definitions in Section 3, G_M now has $|V_M| = |V| = n$ nodes and $|E_M| = \sum_{e\in E} |\Lambda(e)| \leq mk_0$ links, and G_v has the following property.

Observation 4 *Let $G_v(X_v, Y_v, E_v)$ be the bipartite weighted graph defined as before. Then*
$|X_v| + |Y_v| = |\Lambda_{in}(G_M, v)| + |\Lambda_{out}(G_M, v)|$
$\leq d_{in}(G, v)k_0 + d_{out}(G, v)k_0 \leq 2dk_0$ *and*
$|E_v| \leq k_0^2 d_{in}(G, v) d_{out}(G, v) \leq d^2k_0^2$.

Recall that G' is obtained from G_v for all $v \in V_M = V$. Therefore,

Observation 5 *Let $G'(V', E')$ be the directed weighted graph defined before. Then*
$|V'| = \sum_{v \in V_M}(|X_v| + |Y_v|) \leq \sum_{e \in E} |\Lambda(e)| \leq mk_0$ *and*
$|E'| = \sum_{v \in V_M} |E_v| + |E_{org}|$
$\leq d^2nk_0^2 + m_1 \leq d^2nk_0^2 + mk_0$.

Finally $G_{s,t}$ can be constructed from G'. The number of nodes and number of links in $G_{s,t}$ are no more than $mk_0 + 2$ and $d^2nk_0^2 + mk_0 + 2dk_0$, respectively. Therefore,

Theorem 3 *Given a directed network $G(V, E)$ and a pair of nodes s and t, assume that each link e of G has been assigned a set $\Lambda(e)$ of available wavelengths and $|\Lambda(e)| \leq k_0$, and every node has been given the wavelength conversion function. There is an algorithm for finding an optimal semilightpath from s to t. The algorithm requires $O(d^2nk_0^2 + mk_0 + mk_0 \log(mk_0)) = O(d^2nk_0^2 + mk_0 \log(mk_0)) = O(d^2nk_0^2 + mk_0 \log n)$ time since $k_0 = o(n)$.*

By Theorem 3, we know that, even if the total number of wavelengths k in the network is quite large (e.g. $k > n$), the execution time of the algorithm only depends on the number of wavelengths k_0 at every link, the maximum degree d, and the number of links m in the network. That is, the execution time of the proposed algorithm is independent of k. As said before, we consider large wide area networks. In practice such networks are usually sparse (i.e. $m = O(n)$), d is fixed or a slowly increasing function of n (e.g., $d = O(\log n)$), and k_0 is also fixed or a slowly increasing function of n (e.g. $k_0 = O(\log n)$). Thus, the algorithm runs very fast in practice.

5 Conclusion

We have presented improved algorithms for quickly finding the optimal lightpath/semilightpath in wide area fiber optic networks. The lightpath/semilightpath obtained has minimum cost in terms of wavelength costs on the links and wavelength conversion costs at nodes. Compared with a previously known algorithm for the problem, our algorithms run faster in most cases. Importantly, our algorithms can be implemented well in a distributed computing environment. In addition, we also introduced a restricted optimal semilightpath problem, for which we presented an effective algorithm. The simplicity, fast running times, and high locality of the proposed algorithms make them competitive candidates for practical implementation in the emerging networks.

Acknowledgments

The first two authors were partially supported by the Australian Research Council.

References

[1] A. S. Acampora. "Partially reconfigurable multihop lightwave networks", In *GLOBECOM'90*, San Diego, Dec., 1990.

[2] A. Aggarwal, A. Bar-Noy, D. Coppersmith, R. Ramaswami, B. Schieber and M. Sudan. "Efficient routing in optical networks", *J. of the ACM* **46:**973–1001, 1996.

[3] K. M. Chandy and J. Misra. "Distributed computations on graphs: shortest path algorithms", *Commun. ACM* **25:**833–837, 1982.

[4] I. Chlamtac, A. Faragó, and T. Zhang. "Lightpath (wavelength) routing in large WDM networks", *IEEE Journal on Selected Areas in Communications* **14:**909–913, 1996.

[5] I. Chlamtac, A. Ganz, and G. Karmi. "Lightpath communications: a novel approach to high bandwidth optical WDN's", *IEEE Trans. Commun.* **40:**, 1992.

[6] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*, MIT Press, Cambridge, MA, 1989.

[7] M. L. Fredman and R. E. Tarjan. "Fibonacci heaps and their uses in improved network optimization algorithms", *J. ACM*, **34:**596–615, 1987.

[8] P. E. Green. *Fiber-Optic Communication Networks*, Prentice Hall, 1992.

[9] S. Haldar. "An 'all pairs shortest paths' distributed algorithm using $2n^2$ messages", *J. Algorithms*, **24:**20–36, 1997.

[10] J.-F P. Labourdette and A. S. Acampora. "A multichannel multihop local lightwave network", In *GLOBECOM'87*, Tokyo, Japan, Nov., 1987.

[11] A. D. McAulay. *Optical Computer Architectures*, John Wiley, 1991.

[12] R. Ramaswami. "Multi-wavelength lightwave networks for computer communication", *IEEE Communications Magazine* **31:**78–88, 1993.

[13] R. J. Vitter and D. H. C. Du. "Distributed computing with high-speed optical networks", *IEEE Computer* **26:**8–18, 1993.