Provisioning Algorithms for WDM Optical Networks

Murat Alanyali and Ender Ayanoglu

Bell Laboratories, Lucent Technologies
101 Crawfords Corner Road
Holmdel, NJ 07733-3030

Abstract

This paper concerns connection provisioning for optical networks employing wavelength division multiplexing. A heuristic algorithm is developed and numerically studied for routing and wavelength assignment of a set of static connection requests. The algorithm runs much faster than the optimum solution of this problem. An adaptation of the algorithm is proposed to design restorable networks which can handle a specified set of failures. The proposed algorithm is based on taking all failures into consideration simultaneously, and performs better than developing independent designs for each failure.

1 Introduction

Transport networks are wide area networks that provide connectivity for aggregated traffic streams. Modern transport networks increasingly employ *wavelength division multiplexing (WDM)* technology to utilize the vast transmission bandwidth of fiber. WDM is based on transmission of data over separate wavelength channels on each fiber. At the present time, WDM is mainly employed as a point-to-point transmission technology. In such networks, optical signals on each wavelength are converted to electrical signals at each network node. On the other hand, *WDM optical networking* technology, which has been developed within the last decade, and which is becoming commercially available [1, 2, 3], employs wavelengths on an end-to-end basis, without electrical conversion in the network.

Provisioning of a transport network refers to assigning network resources to a static traffic demand. Efficient provisioning is essential in minimizing the investment made on the network required to accommodate a given demand. In the context of WDM optical networks, provisioning means routing and wavelength selection for a set of end-to-end wavelength allocation demands, given a demand distribution and a network topology. Naturally, provisioning of WDM networks has been subject to considerable interest. This interest concentrates on roughly two categories of settings; the case of limited deployed fiber, where provisioning seeks to minimize the number of required wavelengths [4, 5], and the case of limited number of wavelengths per fiber, where provisioning seeks to minimize the amount of required fiber [6, 7].

In this paper we focus on heuristic methods for provisioning a static set of connections on a given WDM optical network topology. It is assumed that each connection requires a dedicated wavelength on each link of its path. We consider the setting where there is a fixed set of wavelengths available on each fiber, and therefore the connections are established at the expense of possibly multiple fibers on network links. Each fiber has a cost which reflects the fiber material, optical amplifiers, and the optical termination equipment at both ends of the link. The objective of provisioning is taken as the minimization of the total network cost.

Since transport networks are chartered to carry high volumes of traffic, network failures may have severe consequences. This imposes fault-tolerance as an essential feature for transport networks. Fault-tolerance refers to the ability of the network to reconfigure and reestablish communication upon failure, and is widely known as restoration. Restoration entails rerouting connections around failed components under a targeted time-to-restore. A network with restoration capability requires redundant capacity to be used in the case of failures, and a primary concern in designing such networks is to provide robustness with minimal redundancy. In principle, design methods devised for conventional, single-wavelength restorable networks [8, 9, 10] can be employed in WDM optical networks as well. Such designs prescribe switching all wavelengths in a fiber together in the case of failure. WDM optical networking, however, provides the capability to switch individual wavelengths, thereby offers a richer set of design methods [11, 12, 13].

The paper is comprised of two parts. The first part concerns provisioning in networks that do not account for restoration. Such a network is called a primary network, and the objective in primary-network design is to minimize the cost associated with the working fibers. This problem can be formulated as an integer linear program (ILP) in a straightforward manner, however the computational complexity of the ILP is prohibitive for a network whose size is not trivial. Therefore efficient heuristic solution methods are needed. We present such a heuristic algorithm that produces good solutions several orders of magnitude faster than general-purpose ILP packages. The second part of the paper involves an adaptation of this heuristic to provisioning in restorable networks. We consider precomputed restoration schemes where the reaction of the network to certain failures is computed in advance, so that upon each failure a certain reconfiguration is adopted. The heuristic algorithm entails determining all reconfigurations simultaneously, in a coordinated manner. This results in better designs compared to methods in which the configurations are developed independently for different failures.

The discussion in the paper is limited to wavelength-selective WDM optical networks: It is assumed that each connection occupies the same wavelength on its entire path. Nevertheless, the algorithm easily extends to networks in which an arbitrary subset of nodes have wavelength conversion capability.

The outline of the paper is as follows. The primary-network design problem is studied in Section 2. Essential definitions and a mathematical formulation of the problem are provided. Section 2.1 concentrates on a related but simpler problem, whose solution leads to a heuristic algorithm for the original problem. This algorithm is the subject of Section 2.2. Section 3 concentrates on the restorable-network design problem. The problem is formulated, and an adaptation of the heuristic algorithm is proposed as an efficient heuristic solution method. The performances of the proposed heuristics are observed numerically in Section 4. Finally, concluding remarks are presented in Section 5.

2 Primary-network design

We start the formulation of the primary-network design problem by providing some essential definitions which are used throughout the paper. In this paper a network is represented by an undirected, weighted graph $G = (N, E, D)$. Here N denotes a set of nodes, E denotes a set of links, and $D = (d(e) : e \in E)$ denotes a set of positive link weights. In the envisioned applications, the weight of a link represents the cost per fiber deployed on the link. The set of *wavelengths* that are available on each fiber is identified by W. Given a link e, a path p, and a wavelength w, the pair (e, w) is called a *lighthop*, and the pair (p, w) is called a *lightpath*. Let $U = N \times N$ denote the set of node pairs in the network. The *demand* for the network is an integer vector $\delta = (\delta(u) : u \in U)$ where $\delta(u)$ denotes the number of connections to be established between the pair of nodes in u. It is assumed that in the primary network each such connection can only be routed on a path from a prespecified set of *admissible paths*, which is identified by $P(u)$. If $p \in P(u)$ and $w \in W$, then we call the pair (p, w) an *admissible lightpath* for the node pair u.

An *assignment* is defined as a real vector $a = (a(p, w) : p \in P(u),\ u \in U,\ w \in W)$ such that $a \geq 0$ and $\sum_{p\in P(u)} \sum_{w\in W} a(p, w) = \delta(u)$ for each $u \in U$. Under assignment a the *load* at lighthop (e, w), $x(e, w)$, is expressed as

$$x(e, w) = \sum_{u\in U} \sum_{p\in P(u):e\in p} a(p, w). \tag{1}$$

If the vector a is an integer assignment then $a(p, w)$ represents the number of connections that are assigned to the lightpath (p, w). Each such connection occupies the wavelength w on each link of the path p. Therefore, if the integer assignment a is adopted, then the number of fibers required at link e is given by $\max_{w\in W} x(e, w)$. The *cost* corresponding to a, $J(a)$, is then defined as

$$J(a) = \sum_{e\in E} d(e) \max_{w\in W} x(e, w).$$

We can now formally state the design problem considered in this section. Given a network G, wavelength set W, demand δ, and admissible paths $(P(u) : u \in U)$, the *primary-network design problem*, P, consists of assigning an admissible lightpath to each connection so as to minimize the resulting cost. Namely,

$$P : \min\{J(a) : a \text{ is an integer assignment}\}.$$

The problem P can easily be formulated as an integer linear problem (ILP) as follows:

$$\begin{gathered} \min \textstyle\sum_{e\in E} d(e)\kappa(e) \\ \text{such that } x(e, w) \leq \kappa(e) \text{ for each } e \in E,\ w \in W, \\ \text{and } a \text{ is an integer assignment.} \end{gathered}$$

Here $(\kappa(e) : e \in E)$ and $(a(p, w) : p \in P(u),\ u \in U,\ w \in W)$ are variables, whereas $(x(e, w) : e \in E,\ w \in W)$ is determined by equality (1). Typical numbers of variables and constraints in the ILP formulation are listed in Table 1 for some mesh-like topologies of various sizes. General purpose codes for solving ILPs typically employ branch and bound techniques with linear programming relaxation. As indicated in Section 4, such solution methods generally entail a very high computational complexity, even for moderate size networks, particularly if there is a large number of connections to be established. In view of this, we focus on heuristic solution methods, and in the rest of this section introduce an efficient heuristic algorithm.

2.1 A motivating optimization problem

In this section we briefly digress from primary-network design, and introduce a collection of auxiliary optimization problems which are parametrized by a positive scalar α. These problems are related to the primary-network design problem P in that, as α grows larger, their cost functions converge to the function J. Proposition 1 provides a characterization of solutions to each auxiliary problem, which in turn inspires a heuristic solution method for problem P.

	number of nodes		
	9	32	88
number of variables	698	8,978	68,954
number of constraints	340	1,888	11,884

Table 1: The size of the ILP formulation of the primary-network design problem for several mesh-like networks. In each case the number of wavelengths is 8 and the number of admissible paths per node pair is 2.

We start with some definitions. Given a positive number α, let J^α denote a cost function such that for each assignment a

$$J^\alpha(a) = \alpha^{-1} \sum_{e \in E} d(e) \log \sum_{w \in W} \exp(\alpha x(e,w)).$$

Note that

$$J(a) \leq J^\alpha(a) \leq J(a) + \alpha^{-1} \sum_{e \in E} d(e) \log |W|,$$

thus $\lim_{\alpha \to \infty} J^\alpha(a) = J(a)$ uniformly for all assignments a. Finally, let the optimization problem P^α be defined as follows:

$$P^\alpha : \min\{J^\alpha(a) : \ a \text{ is a real assignment}\}.$$

Notice that in contrast to the problem P, solutions of P^α are not constrained to be integer-valued assignments.

We next study possible solutions of P^α. Towards this end, note the differentiability of J^α, and let $M^\alpha(p,w)$ denote its partial derivative with respect to $a(p,w)$. Namely, given an assignment a with corresponding load x,

$$M^\alpha(p,w) = \frac{\partial J^\alpha(a)}{\partial a(p,w)} = \sum_{e \in p} d(e) \frac{\exp(\alpha x(e,w))}{\sum_{w' \in W} \exp(\alpha x(e,w'))}.$$

If (p,w) and (p',w') are admissible lightpaths for a node pair u such that $a(p,w) > 0$ and $M^\alpha(p,w) > M^\alpha(p',w')$, then one can find a small, positive number σ such that decreasing $a(p,w)$ by σ and increasing $a(p',w')$ by σ results in another assignment whose cost is strictly less than $J^\alpha(a)$. Thus if an assignment a is optimal for problem P^α, then for each node pair u and each admissible lightpath (p,w) of u,

$$a(p,w) = 0 \quad \text{whenever} \quad M^\alpha(p,w) > m_u(a), \qquad (2)$$

where $m_u(a) = \min\{M^\alpha(p',w') : (p',w') \text{ admissible for } u\}$. The following proposition establishes that this condition is also sufficient for optimality in P^α.

Proposition 1 *There exists a solution to the problem P^α. An assignment a solves P^α if and only if condition (2) holds for each node pair u and each admissible lightpath (p,w) of u.*

Proposition 1 does not provide a constructive characterization for the solutions of problem P^α. However such solutions can be well approximated by employing steepest descent methods. In the context of optimal routing, such methods are referred to as flow deviation algorithms. In the present context, flow deviation entails identifying a lightpath that has the smallest partial derivative M^α, and obtaining a better assignment by increasing the assignment on that lightpath by an appropriate value. In particular one can employ a sequence of flow deviations to arrive at a real assignment that is near-optimal for problem P^α (for a sketch of proof, consult [14, Problem 5.31] together with Proposition 1). In the following section we use an adaptation of this idea to develop a heuristic, iterative procedure to obtain good *integer* assignments for the original problem P.

2.2 The heuristic design algorithm

In this section we return to the primary-network design problem, and describe a heuristic solution method. The method relies on a metric defined on the lightpaths of the network. Namely, given an assignment a with the corresponding load x, the *lightpath-metric* $M(p,w)$ of a lightpath (p,w) is defined as

$$M(p,w) = \sum_{e \in p} m(e,w),$$

where

$$m(e,w) = \begin{cases} d(e)/n_e(a) & \text{if } x(e,w) = \max_{w' \in W} x(e,w') \\ 0 & \text{else,} \end{cases}$$

and $n_e(a) = |\{\tilde{w} \in W : x(e,\tilde{w}) = \max_{w'} x(e,w')\}|$ is the number of wavelengths that have the maximum load on link e. Note that

$$M(p,w) = \lim_{\alpha \to \infty} M^\alpha(p,w).$$

To facilitate the description of the algorithm it is convenient to identify each connection in the network separately. Towards this end, for each node pair u, let $\Delta(u)$ denote the *set* of connections to be established between u. In particular $|\Delta(u)| = \delta(u)$. A *configuration* is defined as a mapping from the connections into the associated admissible lightpaths, and identifies a path and wavelength selection for each connection. Given a configuration, let $p(c)$ and $w(c)$ denote respectively the path and wavelength assigned to connection c. Also let the *cumulative metric* of the network denote the sum $\sum_{u \in U} \sum_{c \in \Delta(u)} M(p(c), w(c))$.

A configuration uniquely identifies an integer assignment a. We thus define the cost of a configuration as the cost $J(a)$ of the corresponding assignment. Let x denote the load corresponding to a. A straightforward calculation yields that

$$\sum_{u \in U} \sum_{c \in \Delta(u)} M(p(c), w(c)) \quad = \quad \sum_{e \in E} \sum_{w \in W} x(e,w) m(e,w)$$

$$
\begin{aligned}
&= \sum_{e \in E} d(e) \max_{w \in W} x(e,w) \\
&= J(a); \qquad (3)
\end{aligned}
$$

hence the cost of a configuration is given by the cumulative metric of the network. This observation, along with the flow deviation approach of Section 2.1, inspires a heuristic algorithm which we describe next.

The algorithm starts with an arbitrary initial configuration, and iteratively computes new configurations until a certain stopping criterion is fulfilled. An iteration involves one decision by each node pair. Namely, each node pair u first determines a connection $c \in \Delta(u)$ that maximizes the lightpath-metric $M(p(c), w(c))$ in the current configuration. A new configuration is then obtained by setting $(p(c), w(c)) = (p, w)$, where (p, w) is an admissible lightpath for u such that the lightpath-metric $M(p(c), w(c))$ is minimized under the resulting configuration. The flowchart of an iteration is given in Figure 1. The output of the algorithm is a configuration that achieves the smallest cost in the course of the iterations.

During the execution of the algorithm, node pairs greedily attempt to decrease the maximum lightpath-metric observed by their connections. While in view of identity (3) such an effort may be expected to decrease the cost, it may as well increase the cost by sufficiently increasing the lightpath-metrics observed by unmodified connections. Thus the cost is not necessarily monotone with respect to the iterations. However, increasing the capacity requirement at a link typically increases the number of lighthops (e, w) such that $m(e, w) = 0$, thereby may help the next iteration obtain a smaller cost. Intuitively speaking, the network cost is likely to have negative drift if it is too high. A typical variation of the network cost with iteration number is plotted in Figure 2. Here the initial configuration is also constructed with the same heuristic.

The computational complexity per iteration of the algorithm is linear in the total number of connections and in the total number of admissible lightpaths. As reported in Section 4, the algorithm typically produces good configurations in a small number of iterations, thus it provides a fast, suboptimal alternative to the integer programming approach.

3 Restorable-network design

This section concerns the provisioning of networks that have restoration capability. The aim of the provisioning methods considered here is to minimize the network cost while providing sufficient capacity to guarantee network robustness against a specified set of failures. We consider the case when a primary-network configuration is fixed in advance, and assume that upon each failure connections are sustained by adopting a predetermined reconfiguration. Two heuristic methods to obtain such reconfigurations are introduced next. The performances of these methods are numerically evaluated in the following section.

We start with a number of definitions which are helpful in the formulation of the design problem. Suppose that a primary-network configuration is given, so that under normal operation conditions (i.e. no failures) the lightpath assignment of each connection is known. A *failure scenario* is defined as a collection of failed network components. Here a network component may denote a link or a node. The set of connections that are allowed to be reconfigured upon a failure scenario f is referred to as the *impact set* of the failure scenario, and is denoted by I_f. This set is a design variable, however it necessarily includes the connections that utilize at least one of the failed components in the primary-network configuration. Larger impact sets yield more efficient designs at the expense of larger restoration time and complexity. Upon failure scenario f, each connection c in the impact set I_f needs to be reassigned a lightpath from the set of *restoration paths* for c, denoted by $P_f(c)$. Naturally, the paths in the set $P_f(c)$ may not utilize any of the components in the failure scenario f. This set also takes on different forms for link-based and path-based restoration schemes, as explained in Section 4. Finally, a *reconfiguration* for the failure scenario f refers to a lightpath assignment in which the connections in the impact set I_f are assigned restoration paths, whereas the remaining connections retain the same lightpaths as in the primary-network configuration.

Each reconfiguration identifies an integer vector on the lightpaths of the network. We now provide a description of this vector, which will be useful in the mathematical formulation of the problem. Consider a reconfiguration for failure scenario f, and let $a_f(p, w)$ denote the number of connections assigned to lightpath (p, w). Also let $b_f(p, w)$ be the number of connections that are assigned the lightpath (p, w) in the primary-network configuration, but do not belong to the impact set I_f. Note that each such connection needs to be assigned the same lightpath in the reconfiguration, therefore $a_f(p, w) \geq b_f(p, w)$. For each node pair u, let $P_f(u) = \cup_{c \in \Delta(u) \cap I_f} P_f(c)$ denote the restoration paths associated with u. The total number of connections on these paths is no less than the number of reconfigured connections, therefore $\sum_{p \in P_f(u)} \sum_{w \in W} a_f(p, w) \geq |\Delta(u) \cap I_f|$ for $u \in U$. Finally, the reconfiguration should satisfy the demand, and thus $\sum_{p \in P(u) \cup P_f(u)} \sum_{w \in W} a_f(p, w) = \delta(u)$. An integer vector which satisfies the above conditions is called a *reassignment for failure scenario* f. Note also that a reconfiguration can easily be constructed from such a vector.

Design of a restorable network entails determining reassignments for a collection of envisioned failure scenarios. Let F be such a collection. A set $A_F = (a_f : f \in F)$ of one reassignment for each failure scenario in F is called a *restoration program* for F. A restoration program identifies a reconfiguration to be adopted upon each failure scenario. Let x denote the load corresponding to the primary-network configuration, and for each $f \in F$, let x_f denote the load under reassignment a_f. For each lighthop (e, w), define $x^*(e,w) = \max\{x(e,w), \max_{f\in F} x_f(e,w)\}$. Note that if restoration program A_F is adopted, then the capacity requirement at a link e is $\max_{w\in W} x^*(e,w)$, so that all connections are guaranteed connectivity under at most one failure scenario from F. The cost of the restoration program A_F is now defined as

$$J^*(A_F) = \sum_{e\in E} d(e) \max_{w\in W} x^*(e,w).$$

Given a primary-network configuration, a collection of failure scenarios F, impact sets $(I_f : f \in F)$, and restoration paths $(P_f(c) : f \in F, c \in I_f)$, we define the restorable-network design problem, R, as follows:

$$R : \min\{J^*(A_F) : A_F \text{ is a restoration program}\}.$$

The problem R admits the following ILP formulation.

$$\begin{gathered}\min \textstyle\sum_{e\in E} d(e)\kappa(e)\\ \text{such that } x(e,w) \le \kappa(e) \text{ for each } e \in E,\ w \in W,\\ x_f(e,w) \le \kappa(e) \text{ for each } e \in E,\ w \in W,\ f \in F,\\ a_f \text{ is an integer reassignment for each } f \in F.\end{gathered}$$

In addition to the size of the network, the complexity of the above ILP depends on the number of failure scenarios and the size of the associated impact sets. In typical applications each link failure is considered as a possible failure scenario, and the size of the ILP is large enough to necessitate fast and suboptimal heuristic solution methods. Two such methods are considered next, and their performances are compared in the following section.

3.1 Method of independent designs

A straightforward heuristic solution for problem R entails decoupling the problem into $|F|$ independent network design problems, one for each failure scenario $f \in F$. In particular, for each failure scenario f, one adopts a reconfiguration that would be optimal in the case $F = \{f\}$. Namely, the *method of independent designs* prescribes the restoration program $(a_f^* : f \in F)$, where a_f^* denotes a solution for the following ILP:

$$\begin{gathered}\min \textstyle\sum_{e\in E} d(e)\kappa(e)\\ \text{such that } x(e,w) \le \kappa(e) \text{ for each } e \in E,\ w \in W,\\ x_f(e,w) \le \kappa(e) \text{ for each } e \in E,\ w \in W,\\ a_f \text{ is an integer reassignment for } f.\end{gathered}$$

If the size of each impact set is small relative to the total number of connections, then the method of independent designs results in significant reduction in computational complexity. However, since the reassignments $(a_f^* : f \in F)$ are obtained in an oblivious fashion, this method may lead to restoration programs with high redundant capacity and thus high cost. To illustrate this on an example, consider the topology of Figure 3 where F consists of the three single-link failures. Assume that the number of wavelengths is 8, and 24 connections are to be established between the two nodes, so that an optimal primary-network design yields one fiber per link. For definitiveness, assume also that the impact set of a failure is the set of connections on the associated link. The method of independent designs may conceivably result in one redundant fiber on each link, whereas it actually suffices to have one redundant fiber on only two of these links. This observation motivates coordinated heuristic design methods, one of which is the subject of the following subsection.

3.2 Heuristic algorithm for coordinated designs

An adaptation of the primary-network design algorithm of Section 2.2 provides an approximate solution method for the restorable-network design problem. Description of the method relies on a metric defined for each lightpath in the network. Namely, given a restoration scheme A_F, the metric $M^*(p,w)$ of a lightpath (p,w) is defined as

$$M^*(p,w) = \sum_{e\in p} m^*(e,w),$$

where

$$m^*(e,w) = \begin{cases} d(e)/n_e^*(A_F) & \text{if } x^*(e,w) = \max_{w'\in W} x^*(e,w') \\ 0 & \text{else,}\end{cases} \tag{4}$$

and $n_e^*(A_F) = |\{\tilde{w} \in W : x^*(e,\tilde{w}) = \max_{w'} x^*(e,w')\}|$. Note that $n_e^*(A_F)$ is the number of wavelengths on link e that would become fully loaded under some failure scenario $f \in F$, provided that the restoration program A_F is adopted.

We now describe the algorithm. The algorithm maintains one reconfiguration for each failure scenario. The pair $(p_f(c), w_f(c))$ denotes the lightpath assignment of connection c in the current reconfiguration for failure scenario f. All reconfigurations are initially set equal to the primary-network configuration. The algorithm then iteratively computes a sequence of restoration programs, until a certain stopping criterion is satisfied. An iteration of the algorithm involves consideration of failure scenarios in a particular order. For each failure scenario f, one decision is executed by each node pair: Namely, node pair u identifies a connection $c \in \Delta(u) \cap I_f$ that maximizes the metric $M^*(p_f(c), w_f(c))$, and then reroutes that connection by choosing a restoration path in $P_f(c)$ and a wavelength in W so that $M^*(p_f(c), w_f(c))$ is minimized under the resulting restoration program. The flowchart of the algorithm is given in Figure 4. The output of the algorithm is a restoration program that achieves the smallest cost in the course of the iterations. Note that the metric of a lightpath is determined by all assignments in the current restoration program, in turn developments of distinct reconfigurations are coupled with each other.

At any stage of the algorithm, the quantity $\max_{x\in W} x^*(e,w)$ denotes the capacity requirement of link

e under the current restoration program. By definition (4) a lighthop contributes to a lightpath metric only if there is a failure scenario under which the lighthop becomes fully loaded. Therefore, when rerouting a connection, the algorithm prefers to avoid such lighthops if possible, thereby does not promote increasing the capacity requirement of the network. Furthermore, if a lighthop is fully loaded under some failure scenario, then its contribution to the lightpath metric is inversely proportional to the number of such lighthops on the same link; thus the lightpath selection also promotes high wavelength utilization. Note that in the example of Figure 3 the algorithm finds the optimal solution in one iteration. On the other hand, one can imagine cases in which the rerouting decision taken by the algorithm entails a capacity increase at some links, hence increase of cost, although this is not absolutely necessary. Such decisions typically increase the number of lighthops (e, w) such that $m^*(e, w) = 0$, thereby may help the next rerouting decision result in a smaller cost. The algorithm typically performs significantly better than the method of independent designs, as can be observed by the numerical results given in the following section.

4 A numerical study

This section focuses on a numerical study of the heuristic algorithms introduced in Sections 2 and 3. The heuristic primary-network design algorithm of Section 2.2 is compared with the optimal solution of the design problem, as well as a commercially available, general purpose heuristic algorithm for solving linear integer problems. Finally the two restorable-network design algorithms of Section 3 are studied under several restoration schemes.

The numerical results reported in this section are based on simulations that are conducted on the network topology of Figure 5. This topology is comprised of 32 nodes and 50 links. The weight of each link is taken as the distance of the link, hence the objective of the design problem is to minimize the total fiber-miles in the network. The number of admissible paths is identical for each node pair, and is denoted by k. The set of admissible paths is taken as the first k shortest paths.

The optimal cost and the cost obtained by the heuristic primary-network design algorithm of Section 2.2 are given in Table 2, for several values of k. In all cases the heuristic solutions are obtained within 200 iterations, and the optimal solutions are calculated via the CPLEX software package[1]. Table 2.a summarizes our observations in the case when a total of 200 connections are established and the number of wavelengths is 8: For $k = 1$ the optimal solution was obtained in 355 (cpu) seconds on a Sparc20 workstation. For $k = 2$ and $k = 3$ the optimal solution could not be obtained within a week; therefore only upper and lower bounds, provided by the same package, are provided for these cases. For $k = 2$ an integer assignment with cost 30,160 was obtained

[1]CPLEX Optimization Inc., NV, USA.

	Optimal		Heuristic	
	cost	time	cost	time
$k = 1$	34,970	355 sec	34,970	8 sec
$k = 2$	30,160$\geq$ $\geq$ 29,300	~hours	31,240	16 sec
$k = 3$	30,160$\geq$ $\geq$ 27,160	~day	31,460	24 sec
$k = \infty$	–	–	30,460	420 sec

(a)

	Optimal		Heuristic	
	cost	time	cost	time
$k = 1$	23,617	1400 sec	23,617	19 sec
$k = 2$	20,677$\geq$ $\geq$ 15,370	~hours	20,056	39 sec
$k = 3$	20,004$\geq$ $\geq$ 15,370	~day	19,864	62 sec
$k = \infty$	–	–	17,935	4760 sec

(b)

	Optimal		Heuristic	
	cost	time	cost	time
$k = 1$	35,033	10 hours	35,033	128 sec
$k = 2$	$\geq$ 22,230	–	30,619	270 sec
$k = 3$	$\geq$ 22,230	–	30,801	432 sec
$k = \infty$	–	–	30,659	~day

(c)

Table 2: Costs obtained by the optimal and heuristic algorithms when (a) 200 connections are established with 8 wavelengths, (b) 200 connections are established with 16 wavelengths, and (c) 800 connections are established with 32 wavelengths. The heuristic algorithm runs much faster than the optimal algorithm, as explained in the text.

in a few hours, and for $k = 3$ an integer assignment with cost 44,335 was obtained in a day. Since the integer assignment obtained for the case $k = 2$ is also admissible for the case $k = 3$, the value 30,160 is registered as an upper bound for the latter case. The corresponding heuristic solutions were obtained in roughly $k \times 8$ (cpu) seconds. Similar observations were made about establishing 200 connections with 16 wavelengths, as illustrated in Table 2.b. When a larger problem with 800 connections and 32 wavelengths was considered, the CPLEX algorithm computed the optimal assignment in about 10 hours for $k = 1$, however for larger values of k it failed to generate any integer assignments within a week. In contrast, the heuristic algorithm generated solutions in roughly $k \times 130$ (cpu) seconds, as indicated in Table 2.c. The lower bounds of Tables 2.b and 2.c are obtained via the LP relaxation of the network design problem.

The value of the optimal solution is minimal in the case $k = \infty$, when any path connecting a given pair of nodes is admissible for that pair. While this case entails a high complexity for the ILP formulation, it can be handled efficiently by a simple variation of the heuristic algorithm. In particular each selected connection is first removed from the network, then an admissible lightpath that has the minimum metric is found via Dijkstra's shortest path algorithm, and finally the connection is rerouted on that lightpath. The last rows in Tables 2.a, 2.b, and 2.c give the values obtained by this approach.

In the heuristic algorithm, a lighthop contributes to a lightpath-metric only if it achieves the maximum load on the associated link. One can thus imagine configurations in which connections are routed over a large number of hops, yet it is undesirable for any one connection to change its lightpath. Such situations are likely to limit the performance of the algorithm if the number of admissible paths per node pair is large so that some paths are considerably longer than others. To avoid such high-cost fixed points it is reasonable to modify the metric M by setting

$$m(e,w) = \begin{cases} d(e)/n_e(a) & \text{if } x(e,w) = \max_{w' \in W} x(e,w') \\ \epsilon & \text{else,} \end{cases}$$

for a small positive ϵ, or to replace M by M^α for a suitably large value of α. In the simulations the described modifications provided up to 10% cost improvement, particularly in the case $k = \infty$.

We now consider the merits of the primary-network design algorithm relative to a general purpose heuristic solution method that is provided in the CPLEX package. The CPLEX heuristic is based on pre-assigning integer values to a number of variables, using a rounding argument on a solution of the LP relaxation of the problem. A reduced ILP of a manageable size is thereby obtained. The major drawback of this approach is that it may not lead to integer assignments. This issue arises consistently in large problems. Here we concentrate on a range of the total number of connections for which the CPLEX heuristic produced integer assignments. The resulting costs and time complexities of the two heuristics are plotted in Figure 6. In the simulations $k = 2$, the number of wavelengths is 8, and the data are obtained by averaging over 10 different traffic demands.

In the rest of this section we focus on the design of restorable networks. Three different restoration schemes are considered for these networks: In the *full-reconfiguration* scheme all connections are allowed to be reconfigured upon a failure, whereas in the *path-based* and *link-based* restoration schemes a connection can be reconfigured only if it uses failed components in the primary-network configuration. In full-reconfiguration and path-based schemes a reconfigured connection is assigned to one of the k shortest paths in the faulty network. In the link-based scheme such a connection retains the functional segment of its original lightpath, and bypasses each failed network component by using one of the k appropriate shortest paths.

We now provide a numerical comparison of the two restorable-network design algorithms introduced in Section 3. We take $k = 2$, and consider the two problems of provisioning 200 connections with 8 wavelengths, and provisioning 800 connections with 32 wavelengths. The set of failure scenarios, F, is taken to be the set of all single-link failures, and the three restoration schemes of the previous paragraph are considered. The heuristic algorithm of Section 2.2 is employed to obtain the primary-network configuration, as well as each individual reconfiguration in the method of independent designs. The ratio of the redundant fiber-miles to the working fiber-miles obtained in each case is given in Table 3. Note that in each of the settings (a) and (b), the primary-network design, hence the working fiber-miles, is identical for all schemes and algorithms. The full-reconfiguration and link-based schemes required respectively the least and most fiber-miles, and the coordinated heuristic resulted in roughly 20% savings in spare capacity in each restoration scheme.

	Independent	Coordinated
Full-reconfiguration	0.92	0.70
Path-based	1.01	0.87
Link-based	1.43	1.23

(a)

	Independent	Coordinated
Full-reconfiguration	0.84	0.66
Path-based	0.91	0.74
Link-based	1.56	1.28

(b)

Table 3: Total redundant capacity (in fiber-miles), normalized by dividing with the total working capacity (in fiber-miles), under the considered restoration schemes and design algorithms. (a) 200 connections with 8 wavelengths, (b) 800 connections with 32 wavelengths.

5 Conclusion

This paper concerned two heuristic methods for routing and wavelength assignment of static connections in WDM optical networks. The first method is an iterative algorithm which is suitable for networks that do not account for failures. The algorithm is based on greedy decisions by the connections to decrease a certain metric whose minimum value corresponds to an optimal assignment. The computational complexity per iteration of the algorithm is low, and it is shown numerically on a mesh-like topology that the algorithm generates efficient solutions in a reasonable number of iterations. The second method concerns the design of fault-tolerant networks, and is obtained by an adaptation of the first method.

The method entails coordinated planning of several failure scenarios, and a numerical study shows that it provides more efficient designs than those obtained by considering the failures independently.

Acknowledgement. Work supported partially by DARPA via the MONET (Multiwavelength Optical Networking) Consortium of Lucent Technologies, AT&T, Bellcore, Bell Atlantic, Bell South, and SBC, with the participation of NRL and NSA.

References

[1] S. B. Alexander, R. S. Bondurant, D. Byrone, V. W. S. Chan, S. G. Finn, R. Gallager, B. S. Glance, H. A. Haus, P. Humblet, R. Jain, I. P. Kaminow, M. Karol, R. S. Kennedy, A. Kirby, H. Q. Le, A. A. M. Saleh, B. A. Schofield, J. H. Shapiro, N. K. Shankaranarayanan, R. E. Thomas, R. C. Williamson, R. W. Wilson, "A precompetetive consortium on wide-band all-optical networks," *Journal of Lightwave Technology*, Vol. 11, pp. 714–735, May 1993.

[2] G. K. Chang, G. Ellinas, J. K. Gamelin, M. Z. Iqbal, and C. A. Brackett, "Multiwavelength reconfigurable WDM/ATM/SONET network testbed," *Journal of Lightwave Technology*, Vol. 14, pp. 1320–1340, June 1996.

[3] R. E. Wagner, R. C. Alferness, A. A. M. Saleh, and M. S. Goodman, "MONET: Multiwavelength optical networking," *IEEE Journal of Lightwave Technology*, Vol 14, pp. 1349–1355, June 1996.

[4] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: An approach to high bandwidth optical WAN's," *IEEE Transactions on Communications*, Vol. 40, No. 7, pp. 1171–1182, July 1992.

[5] N. Nagatsu, Y. Hamazumi, and K. Sato, "Number of wavelengths required for constructing large-scale optical path networks," *Electronics and Communications in Japan*, Part 1, Vol 78, No 9, pp. 1–11, September 1995.

[6] N. Nagatsu and K. Sato, "Optical path accommodation design enabling cross-connect system scale evaluation," *IEICE Trans. Commun.*, Vol. E78-B, No. 9, pp. 1339–1343, September 1995.

[7] G. Jeong and E. Ayanoglu, "Comparison of wavelength-interchanging and wavelength-selective cross-connects in multiwavelength all-optical networks," *Proceedings of IEEE INFOCOM'96*, pp. 156–163, March 1996.

[8] R. Dighe, Q. Ren, and B. Sengupta, "A link based alternative routing scheme for network restoration under failure," *Proceedings of GLOBECOM'95*, pp. 2118–2123, November 1995.

[9] W. D. Grover, T. D. Bilodeau, and B. D. Vanables, "Near optimal spare capacity planning in a mesh restorable network," *Proceedings of GLOBECOM'91*, pp. 2007–2011, December 1991.

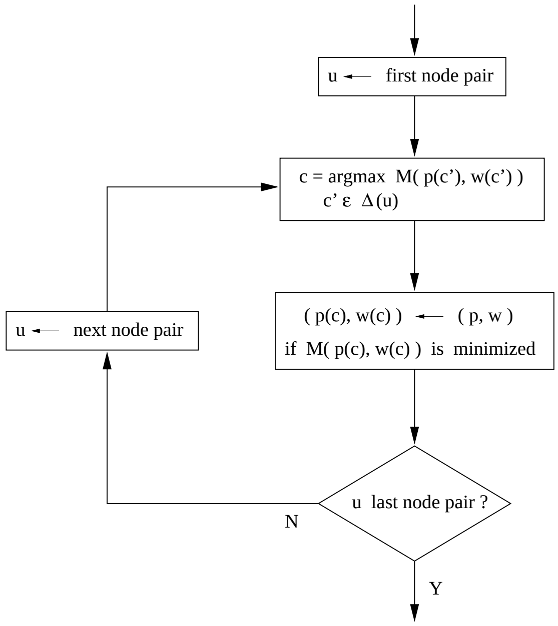

Figure 1: Flowchart of an iteration of the primary-network design algorithm.

[10] B. D. Venables, W. D. Grover, and M. H. MacGregor, "Two strategies for spare capacity placement in mesh restorable networks," *Proceedings of ICC'93*, pp. 267–271, May 1993.

[11] N. Nagatsu, S. Okamoto, and K. Sato, "Optical path cross-connect scale evaluation using path accommodation design for restricted wavelength multiplexing," *IEEE JSAC*, Vol. 14, No. 5, pp. 893–901, June 1996.

[12] K. Sato and N. Nagatsu, "Failure restoration in photonic transport networks using optical paths," *Proceedings of OFC'96*, pp. 215–216, March 1996.

[13] L. Wuttisittikulkij and M. J. O'Mahony, "Use of spare wavelengths for traffic restoration in multi-wavelength transport network," *Proceedings of ICC'95*, pp. 1778–1782, June 1995.

[14] D. Bertsekas and R. Gallager, *Data Networks*, Prentice-Hall, New Jersey, 1992.

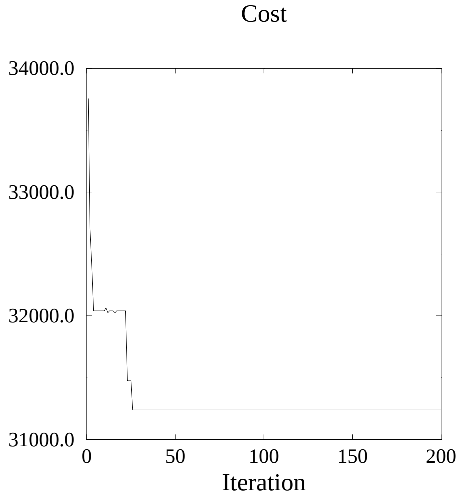

Figure 2: Typical variation of the cost with respect to the iterations for the heuristic primary-network design algorithm. The initial configuration is also constructed via the same heuristic, therefore the iterations start with an already small cost.

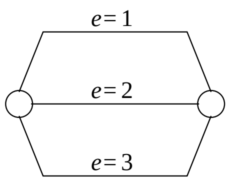

Figure 3: Sample topology to illustrate the inefficiency of the method of independent designs. It is assumed that $W = 8$, and 24 connections are to be established.

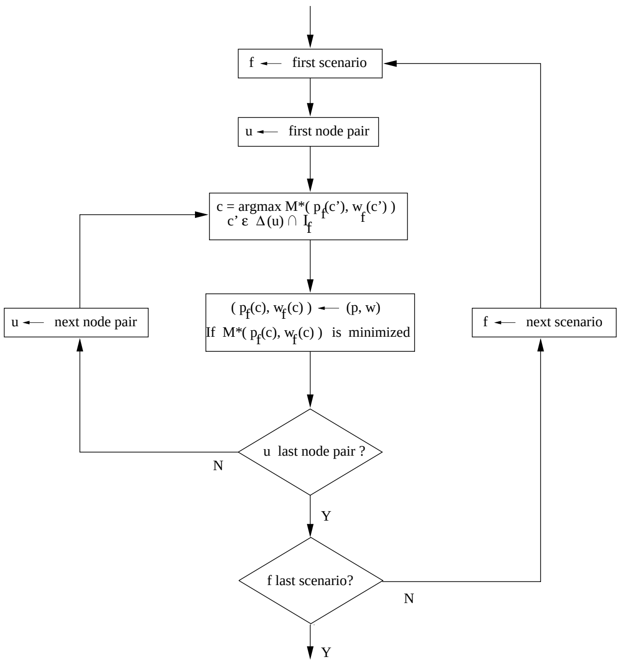

Figure 4: Flowchart of an iteration of the restorable-network design algorithm.

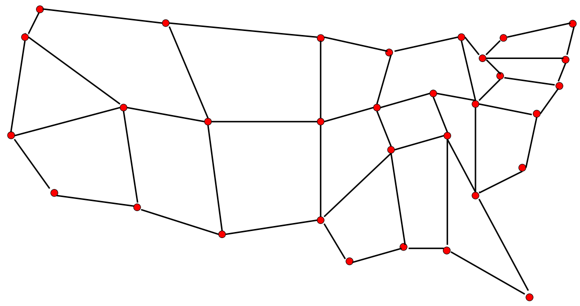

Figure 5: The network topology considered in the numerical study.

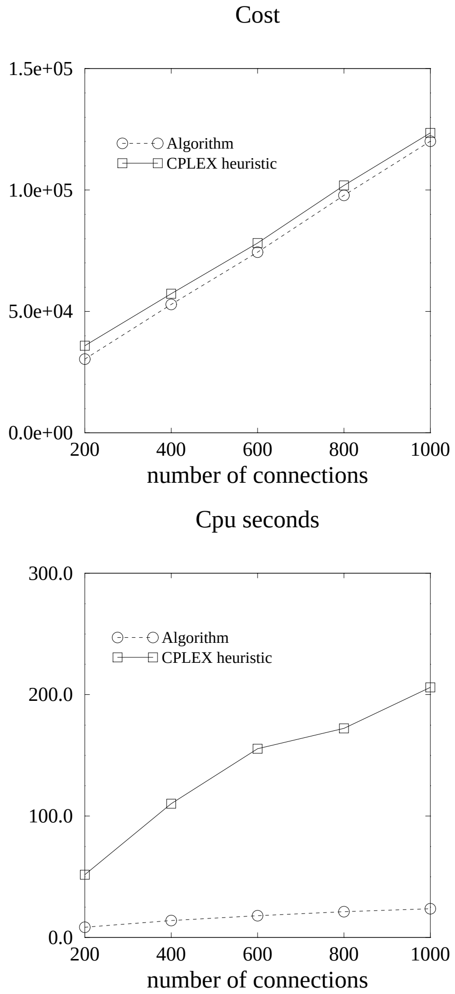

Figure 6: A comparison of the proposed algorithm and the CPLEX heuristic for primary-network design.