Benefits of Partial Reconfigurability in Circuit-Switched WDM Networks

James Martin

University of Cambridge

Abstract

We describe and study *partially reconfigurable* circuit-switched networks. These are dynamic networks in which partial, but not complete, flexibility of resource assignment is retained during the lifetime of a connection. In particular we analyse two examples of partially reconfigurable WDM networks with flexibility of wavelength assignment. In the first example, each connection in the network requires a main lightpath (whose wavelength must remain fixed) and a standby lightpath (whose wavelength may change). In the second example, the network contains two classes of connections with different service levels, one class reconfigurable and the other not. We propose two contrasting reconfiguration algorithms, and demonstrate that even a small amount of reconfigurability can yield large performance gains.

1 Introduction

Consider a circuit-switched WDM network. Each link in the network may be considered as a collection of channels, each one identified with a particular wavelength. A connection using a set of links needs a channel on each link, and in the absence of wavelength conversion, the connection must use the same wavelength on each link. (This is referred to as the "wavelength continuity constraint"; each connection needs to be carried on a single "lightpath").

Both *dynamic* and *static* networks have been extensively studied [8]. In a static network [2], the set of required connections is completely known in advance, so that some sort of global "offline" optimisation may be performed to decide the route and/or wavelength to assign to each connection. In a dynamic network [7], on the other hand,

Email: jbm@statslab.cam.ac.uk. Work funded by Marconi plc under the project *Optimum Network Architectures*. Marconi contact: Derek Stagg (derek.stagg@marconi.com)

connections arrive and depart over time, and allocation of resources must be carried out "online".

Resource allocation problems analogous to those for a static network arises when the connections in a dynamic network are *reconfigurable*; that is, if the choice of route and/or wavelength may be changed while the connection is underway. Such reconfigurability will typically improve the performance of the network, in the sense that blocking will be less frequent (or that fewer resources are needed to attain a given level of blocking). On the other hand, reconfiguration can incur cost or impair the quality of the connections offered by the network, for example if a change of route or wavelength requires a short interruption to the connection.

We consider *partially reconfigurable* networks, for which the resource allocation problems lie somewhere between those for the static and dynamic networks described above. Partial reconfigurability may arise in many ways, for example:

(1) Some connections in the network are reconfigurable while others are not. For example, different connections in the network may have different priorities or different quality of service guarantees; only those with lower priority, or with a less stringent quality of service guarantee, may be reconfigured.

(2) Each connection in the network may be partially but not completely reconfigurable. For example:

   (a) Each connection in the network requires a *main path* and a *protection path* (typically, disjoint from the main path). The protection path can be reconfigured during the lifetime of the connection, but the main path cannot.

   (b) The network has flexibility both of routing and of wavelength assignment; reconfiguration of the route of a connection is impossible while reconfiguration of the wavelength (which may require less complicated synchronisation and signalling between nodes) is possible.

(3) Connection requests arrive (and must be definitively accepted or rejected) some time in advance of the beginning of the connection itself (e.g. requests for leased lines). Once a connection is accepted, capacity must be guaranteed for it, but resources need not be assigned until the connection itself starts. In effect, the connection is reconfigurable between acceptance and the start of transmission, and unreconfigurable thereafter.

(4) More generally, individual connections may move between "active" and "idle" states; only connections which are currently in an idle state may be reconfigured.

In this study we analyse the performance of partially reconfigurable networks. (The performance may for example be measured by the *blocking probability*, i.e. the long-run average proportion of connection requests refused). This paper examines specifically networks with fixed routing, concentrating on the wavelength assignment problem alone. (In this situation both the static and dynamic unreconfigurable case have been extensively studied – see for example [8, 2, 7] and their references). A feature of the fixed-routing case which we exploit in this paper is that good performance bounds are available which allow us to assess objectively the effectiveness of the algorithms under study; these bounds are obtained by analysing corresponding networks with full wavelength conversion. A second paper [5] will examine networks with choice of both routing and wavelength assignment.

We examine in particular cases (1) and (2)(a) above. We study both a regular network (a square lattice-torus topology) and a real topology, at a range of traffic levels, and we examine a variety of assignment and reconfiguration methods.

As one might expect from previous work on static networks, e.g. [2], we find that *complete* reconfigurability (corresponding essentially to offline wavelength assignment) provides considerable performance gain – the performance of the network becomes very close indeed to that of the corresponding fully wavelength-convertible network. We find that *partial* reconfigurability is also able to provide a large proportion of this performance gain. To yield the maximum gain, reasonably complicated reconfiguration methods seem to be necessary; but we also find that more simple methods – which are typically less disruptive to the network and which are also more scalable (i.e. could more practically be implemented in a large network without central control) – can also provide considerable gain.

Such assessments of the gains provided from partial reconfigurability do not seem to have appeared previously in the literature. An exception is [1], where a situation similar to (2)(a) is analysed for an incremental network. However, the study there is limited to a particular class of reconfiguration algorithms which are shown in fact to provide no performance gain over a non-reconfigurable network.

2 Network models

2.1 Basic model

We use the following simple model of a WDM network without wavelength conversion. The network consists of a set of links; each link consists of a single fibre, and may be considered as a collection of channels, with a wavelength associated to each channel. (All links contain the same set of wavelengths). A connection using a route consisting of one or more links requires a channel on each link of the route, and must use the same wavelength on each link.

We study two particular scenarios. The first is a network with protection paths. Each request for a connection requires a *main path* and a *protection path* (which will be link disjoint); it is not necessary for both paths to use the same wavelength. The request is accepted only if both paths can be established. During the lifetime of the connection, the wavelength used by the main path is fixed, but that used by the protection path may be changed.

The second scenario is a network with two different types of connection: reconfigurable, whose wavelength may be altered, and unreconfigurable, whose wavelength must remain the same.

We measure the performance of the network by its *blocking probability*, i.e. the long-run proportion of connection requests refused. Note that connection blocking in such a network may occur for two reasons. Either the capacity of the network is insufficient, and blocking would equally occur if the network were equipped with full wavelength conversion; or alternatively the overall capacity of each link is sufficient, but blocking occurs due to the wavelength continuity constraint.

2.2 Networks with complete wavelength conversion

We also consider an analogous model of a network with complete wavelength conversion at each node (essentially this is the case of a simple circuit-switched loss network [4]). A connection using a route requires a channel on each link of the route, but the wavelength used on each link is not important; as a consequence, reconfigurability of wavelength allocation is not an issue.

Now the only cause of connection blocking is lack of capacity. We will use this model as a benchmark against which to compare the performance of networks *without* wavelength conversion, but with partial or full reconfigurability.

3 Assignment methods

When a connection request arrives, a wavelength may already be free on the desired route; in this case, no reconfiguration is necessary in order to accept the connection. If more than one wavelength is free, a choice needs to be made. We consider two strategies for wavelength assignment (both previously considered for unreconfigurable networks):

3.1 First Fit

We label the set of wavelengths by $1, 2, \ldots, W$ in some arbitrary but consistent order, where W is the total number of wavelengths available on each link.

Under *first fit*, a connection requiring a certain route is assigned to the lowest numbered wavelength out of those which are free on all the links of the route. This attractively simple method works well and is easy to implement in a distributed fashion, although it is generally outperformed by more complicated methods which take account more fully of the current state of the network, such as MΣ or RCL (described below).

3.2 Max-Sum (MΣ) and Relative Capacity Loss (RCL)

Max-Sum (MΣ) [6] and Relative Capacity Loss (RCL) [9] are wavelength assignment methods which aim to choose, from those wavelengths available, the wavelength which maximises the value of some function representing the remaining network capacity.

Let ψ represent the state of the network. The *link capacity* $r(\psi, l, w)$ on link l and wavelength w in state ψ is defined as the number of fibres on link l on which wavelength w is unused. (In a network where each link has a single fibre, this number must be 0 or 1).

For a path p consisting of a set of links, the *path capacity* $r(\psi, p)$ of p in state ψ is defined as the maximum number of lightpaths that could simulataneously be set up along p; that is,

$$R(\psi, p) = \sum_{w=1}^{W} \min_{l \in p} r(\psi, l, w). \tag{3.1}$$

When a connection request arrives, MΣ assigns to it, from those wavelengths available, the wavelength w which maximises the quantity

$$\sum_{p \in P} R(\psi(w), p), \tag{3.2}$$

where P is the set of potential paths that may be requested by connections, and $\psi(w)$ is the network state that will result from assigning the connection to wavelength w.

Weighted MΣ then arises if (3.2) is replaced by

$$\sum_{p \in P} \lambda_p R(\psi(w), p), \tag{3.3}$$

where λ_p represents the known or estimated arrival rate of connections requiring path p.

Note that maximising (3.3) is equivalent to minimising

$$\sum_{p \in P} \lambda_p \left[R(\psi, p) - R(\psi(w), p) \right], \tag{3.4}$$

where ψ is the current network state. We can introduce greater sensitivity to paths whose remaining capacity is already low by replacing (3.4) by

$$\sum_{p \in P} \lambda_p \frac{[R(\psi, p) - R(\psi(w), p)]}{R(\psi, p)}; \tag{3.5}$$

this yields an assignment strategy similar to RCL [9]. We use this method (which we loosely refer to as RCL) in some of the simulations reported in Section 6.

Assignment strategies such as MΣ and RCL involve a larger computation overload than, say, first fit, but do improve performance. Generally they would be implemented in a centralised way, although in [8] a method for implementing a version of RCL in a distributed way is proposed.

4 Reconfiguration methods

When a request for a new connection arrives in the network for which no lightpath can immediately be assigned, we will attempt to create one by changing the wavelengths allocated to existing reconfigurable connections.

We propose and study two contrasting reconfiguration methods; the first is less powerful than the second, but also simpler, and has various operational advantages which we discuss below.

4.1 Displacement

We first describe a reconfiguration method that we refer to as *displacement*.

Suppose a connection arrives requiring a path p, and that no single wavelength is currently available on every link of p. We consider the wavelengths $1, 2, \ldots, W$ in turn. A wavelength w is a candidate for assignment to the new connection if the only connections blocking the path p on w are reconfigurable. (A connection *blocks* p on w if it uses at least one of the links of p on w). Suppose this is the case, and let the reconfigurable connections be c_1, $c_2, \ldots, c_n$. For each connection c_i, we try to find a wavelength $w_i \neq w$ such that the links required by c_i are all currently unused on wavelength c_i; (if there is more than one candidate for w_i, we can for example follow the "first-fit" principle of Section 3.1 and choose the lowest one).

Displacements of this kind constitute only a small number of the possible reconfigurations; hence there are many situations in which no such displacement may be found (and the new connection may be rejected) even when a feasible reconfiguration exists which would allow the connection to be taken; thus the displacement procedure is less powerful than, for example, the repacking procedures described below.

However, it has many practical advantages:

- The calculations involved are very simple and so introduce little computational overhead;
- The procedure can easily be carried out "locally", i.e. in a distributed way, and hence scales well as the size of the network grows;
- The number of connections that need to be reconfigured is typically fairly small, reducing the disruption to the network;
- When a connection is reconfigured, the new lightpath can be set up before the old one is torn down; this can reduce or remove the impairment to service involved in reconfiguration.

4.2 Repacking and Graph Colouring

We now move on to a more complex reconfiguration procedure, which is a natural extension of the methods used to assign wavelengths in static networks.

Suppose a connection arrives to which we are unable to assign a wavelength. We now construct an *auxiliary graph* G for the network; the wavelength assignment problem for the network becomes a colouring problem for the graph.

The vertices of G are of two kinds. The vertices $v_1, v_2, \ldots, v_n$ represent reconfigurable connections already present in the network, together with the connection which has just arrived and has not yet been assigned. In addition we add vertices $u_1, u_2, \ldots, u_W$ which represent the available wavelengths.

An (undirected) edge exists between the two vertices v_i and v_j in G if the paths of the connections which they represent intersect (i.e. share at least one link). An edge between vertex v_i and vertex u_w exists if the path of connection v_i is blocked on wavelength w by a fixed connection.

We colour node u_w with colour w. The question of whether we can find a reconfiguration of the network allowing the new connection to be accepted is then the same as the question of whether we can colour the vertices $v_1, \ldots, v_n$ of the graph with the colours $1, \ldots, w$ in such a way that no two vertices joined by an edge are coloured with the same colour.

Such colouring problems are typically hard to solve definitively. Indeed, the problem of deciding whether such a set of connections can be assigned using w or fewer wavelengths is NP-complete in general as the size of the problem grows in a suitable sense.

Good approximate methods exist however. In particular, we consider *sequential* algorithms. This means that we re-order the vertices $v_1, \ldots, v_n$ in some way as $v'_1, \ldots, v'_n$, and then assign colours/wavelengths to them in sequence according to first-fit.

The following ordering procedure is known as "smallest last" and performs well in general (see [8] for fuller details): let v'_n be the uncoloured vertex with smallest degree in G, and then recursively for $i = 1, 2, \ldots, n-1$ let v'_{n-i} be the uncoloured vertex with smallest degree in the reduced graph $G \setminus \{v'_n, v'_{n-1}, \ldots v'_{n-i+1}\}$ (that is, the graph G with the vertices $v'_n, v'_{n-1}, \ldots v'_{n-i+1}$ removed). Ties may be broken at random, for example.

We will find that the corresponding reconfiguration procedure works very well in practice. Reconfigurations found by such methods tend to involve reassigning a large proportion of current connections however; the computational load involved in finding them is also rather high, and the calculations need to be performed centrally.

5 Toplogies

For our simulations we used two different topologies:

5.1 Lattice torus

We considered a 7x7 lattice torus with one fibre on each link and 24 wavelengths on each fibre. The nodes can be represented by points (x, y), $0 \leq x, y \leq 6$. There are links from (x, y) to the four nodes $(x \pm 1, y)$ and $(x, y \pm 1)$ where addition and subtraction is mod 7. In all there are 49 nodes and hence 1176 node-pairs. We considered connection demands distributed equally over these node-pairs, arriving as a Poisson process and with independent exponential durations.

Fixed routing was employed. The shape of the routes for various possible relationships of the endpoints is indicated in Figure 5.1. Main paths are shown as solid lines, protection paths as dashed lines. "L-shaped" paths were chosen in order to maximise as far as possible the average *interference length*. This measures, loosely speaking, the extent to which two paths that share a link are likely to share several links. Barry and Humblet [3] note that increasing the interference length tends to reduce wavelength "fragmentation", and so reduce blocking caused by the wavelength continuity constraint.

5.2 Realistic topology

The second topology, which we refer to as "Network A", is a representation of part of a European national backbone network. The 13-node, 18-link topology is shown in Figure 5.2. We simulate this network only in the situation where both main and protection paths are used; main paths are routed along the shortest path (according to the distances shown in Figure 5.2) and protection paths along the shortest remaining

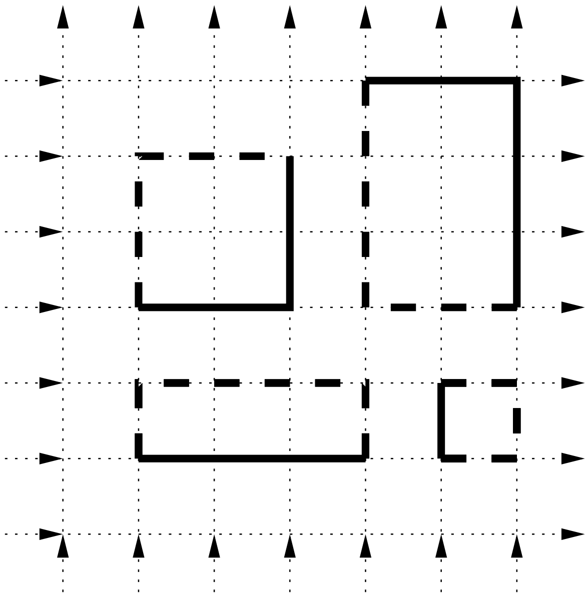

Figure 5.1: 7x7 lattice-torus topology (Section 5.1), with examples of main and standby paths.

node- and link-disjoint path. We again take 24 wavelengths per fibre. Connections have independent exponential holding times, and arrival streams are Poisson with rates chosen according to the average traffic demands given in the table in Figure 5.2.

6 Results and Discussion

6.1 Description of Simulations

We present the results of simulations to estimate the blocking probability, at a range of traffic loads, for networks with and without protection paths.

The first-fit assignment algorithm (Section 3.1) is combined with the displacement reconfiguration algorithm (Section 4.1), while RCL (Section 3.2) is combined with repacking using smallest-last (Section 4.2).

For networks with protection paths, we compare the following cases: networks where no reconfiguration is allowed; networks where protection paths are reconfigurable but main paths are not; networks where all paths are reconfigurable; and networks employing full wavelength conversion. The simulations cover the following cases:

- lattice torus, assignment by first-fit and reconfiguration by displacement (Figure 6.1);

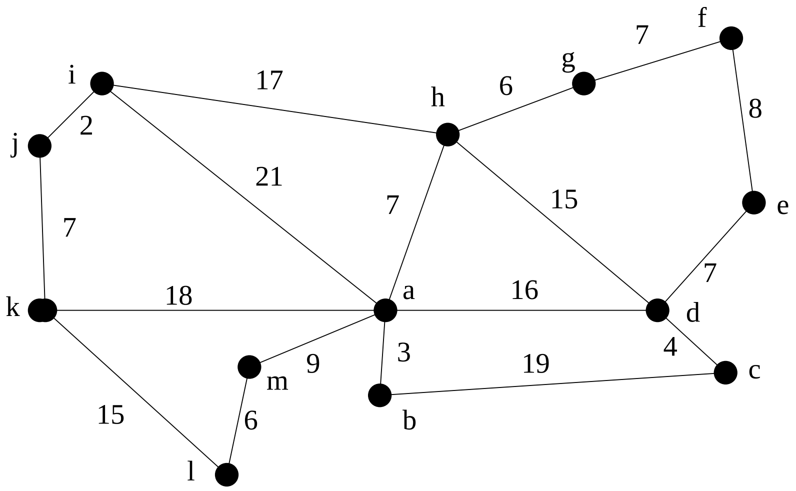

Figure 5.2: Network A (Section 5.2), with relative link distances (above), and average node-pair traffic demands (below)

Node	b	c	d	e	f	g	h	i	j	k	l	m
a	2	2	16	1	0	0	9	16	16	9	9	1
b	-	0	0	0	0	0	2	1	1	0	0	0
c	-	-	2	0	0	0	0	2	0	0	0	0
d	-	-	-	8	0	0	0	16	16	0	4	0
e	-	-	-	-	0	1	8	1	0	4	0	0
f	-	-	-	-	-	1	0	0	0	0	0	0
g	-	-	-	-	-	-	0	0	0	0	0	0
h	-	-	-	-	-	-	-	8	0	0	0	0
i	-	-	-	-	-	-	-	-	17	0	0	0
j	-	-	-	-	-	-	-	-	-	8	4	0
k	-	-	-	-	-	-	-	-	-	-	8	0
l	-	-	-	-	-	-	-	-	-	-	-	1

- lattice torus, assignment by RCL and reconfiguration using smallest-last repacking (Figure 6.2);
- network A, assignment by first-fit and reconfiguration by displacement (Figure 6.3).

For networks without protection paths, we vary the proportion of connections in the network which are reconfigurable, from 0 (no reconfiguration), to 1 (complete reconfigurability), and compare these also with the case of a network with full wavelength conversion. The simulations cover the following cases:

- lattice torus, first-fit with displacement (Figure 6.4);
- lattice torus, RCL with smallest-last repacking (Figure 6.5).

In each set of simulations we vary the average network "load", i.e. the ratio of the offered traffic arriving (taking into account the number of links required and the length of the required connection) to the network capacity available. For the regular network, every link is equivalent and the overall load is equivalent to the load on any given link. For the real network, there is considerable variation between the links and the "worst" link has a much higher load than the network as a whole.

Graphs and tables of the blocking probabilities are shown in Figures 6.1-6.5. In the graphs, the blocking probability is plotted on a log scale, either against average network load or against the proportion of reconfigurable connections.

6.2 Discussion

The performance gain achieved by the introduction of complete wavelength conversion into a non-reconfigurable network can be large – for example, a reduction in the blocking probability by a factor of more than 50 can be seen in the final column of the table in Figure 6.1. We find that *complete* reconfigurability using the graph-colouring algorithm of Section 4.2 can yield almost all the performance gain associated with wavelength conversion – see the lowest two rows of the tables in Figures 6.2 and 6.5. Thus the introduction of wavelength conversion into a completely reconfigurable network would provide only a very slight further performance gain.

We see also that *partial* reconfigurability can provide an important performance gain. The effect is noticeable at all traffic levels, and particularly pronounced when the load is fairly low.

For example, in the lattice torus with protection paths and a load of 0.4, allowing the reconfiguration of protection paths reduces the blocking probability by a factor of around 16, from 0.007142 to 0.0004169; the introduction of complete wavelength conversion would further reduce the blocking probability by a factor of only around 1.4, to 0.0002937 (see the 4th column of the table in Figure 6.2).

Similarly, for a lattice torus without protection paths at a load of 0.3, we see that when a proportion of 0.3 of the connections are reconfigurable, the blocking probability reduces by a factor of around 4.5, from 1.415×10^{-5} to 3.010×10^{-6}; introducing complete wavelength conversion would further reduce the blocking probability by a factor of less than 2, to 1.656×10^{-6} (see the last column of Figure 6.5).

For the networks employing assignment by first-fit, notice that (since the assignment strategy is less sophisticated) the wavelength conversion gain is much greater; we see that much of this gain can be realised by partial reconfiguration using only rather simple methods such as "displacement". For the lattice torus with protection paths at a load of 0.3, the wavelength conversion gain is a factor of around 53; the reconfiguration of protection paths alone, by displacement, already yields a reduction by a factor of 20. (See the final column of the table in Figure 6.2). Similar effects are seen for the network without protection paths in Figure 6.5.

In network A, there is a large disparity between the average load on different links. This reduces the importance of wavelength blocking relative to capacity blocking, and hence the wavelength conversion gain is already relatively small. The gain achievable from reconfiguration is therefore also small, but we see in Figure 6.3 that at all load levels, a large proportion of this gain is already provided by the reconfiguration of protection paths according to the displacement method.

7 Conclusions

We have described and studied *partially reconfigurable* circuit-switched networks, whose performance (as measured for example by blocking probability) lies between that of static and dynamic networks, and whose operation and analysis combines elements of both.

We have in particular presented results for the wavelength assignment and reassignment problem in such networks. We examined the cases of networks where protection paths, but not main paths, are reconfigurable, and networks where two classes of connections exist with different service levels, one class reconfigurable and the other not.

We have presented two contrasting reconfiguration algorithms. When the more sophisticated reconfiguration algorithm (based on graph colouring methods) is employed, full reconfigurability provides essentially all the attainable performance gain (in that the network performs as well as an equivalent network with full wavelength conversion). Partial reconfigurability provides a large part of this gain, especially at

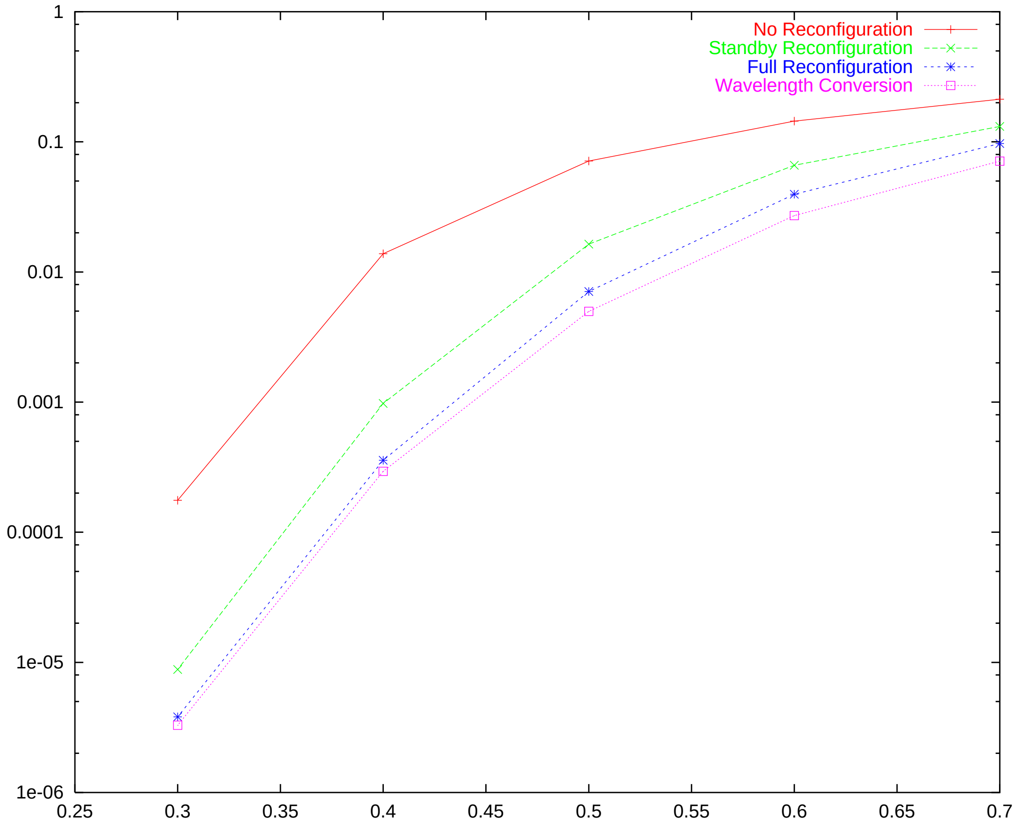

Figure 6.1: 7x7 lattice torus with protection paths. Wavelength assignment using *first-fit* and reconfiguration using *displacement*, for the following situations: networks with no reconfiguration; networks where protection paths only are reconfigurable; fully reconfigurable networks; networks with complete wavelength conversion. Blocking probability on a log scale is plotted against average network load.

Load:	0.7	0.6	0.5	0.4	0.3
None	0.2123	0.1443	0.07122	0.01380	1.758×10^{-4}
Standby	0.1313	0.06609	0.01636	0.0009759	8.820×10^{-6}
Full	0.09701	0.03961	0.007076	0.0003576	3.814×10^{-6}
WC	0.07102	0.02711	0.004974	0.0002937	3.287×10^{-6}

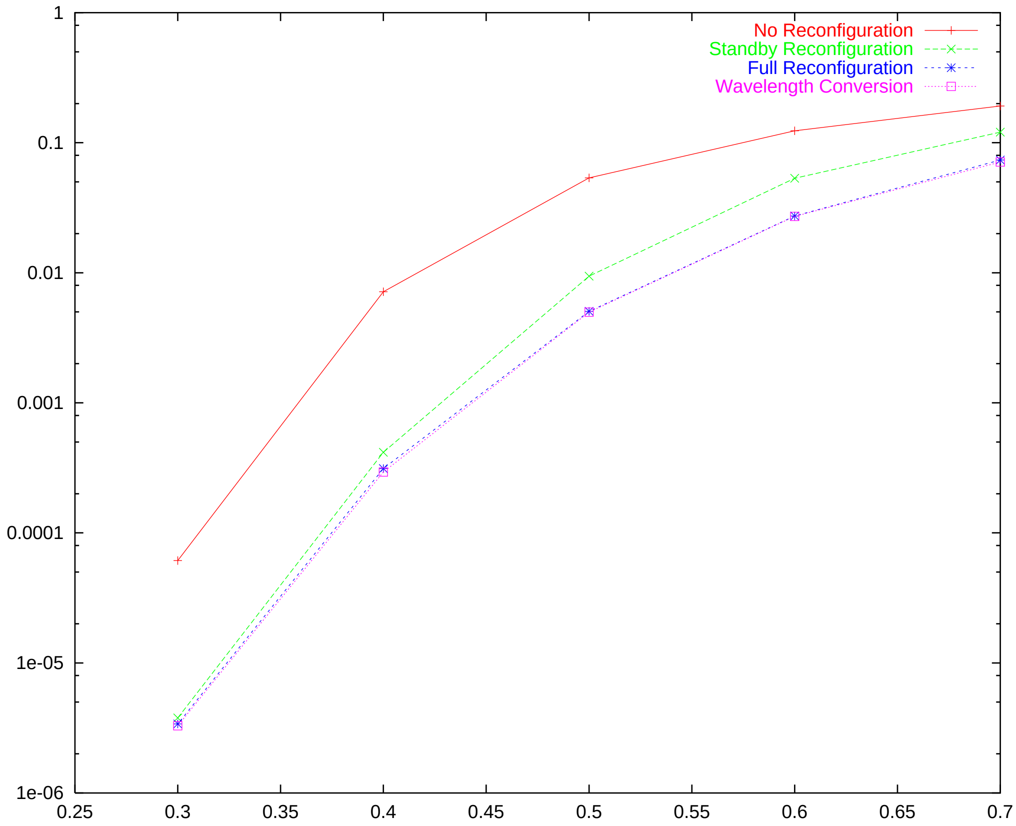

Figure 6.2: 7x7 lattice torus with protection paths. As for Figure 6.1, but with wavelength assignment using RCL and reconfiguration using smallest-last repacking.

Load	0.7	0.6	0.5	0.4	0.3
None	0.1917	0.1236	0.05369	0.007142	6.198×10^{-5}
Standby	0.1205	0.05324	0.009410	0.0004169	4.110×10^{-6}
Full	0.07365	0.02733	0.005051	0.0003124	3.388×10^{-6}
WC	0.07102	0.02711	0.004974	0.0002937	3.287×10^{-6}

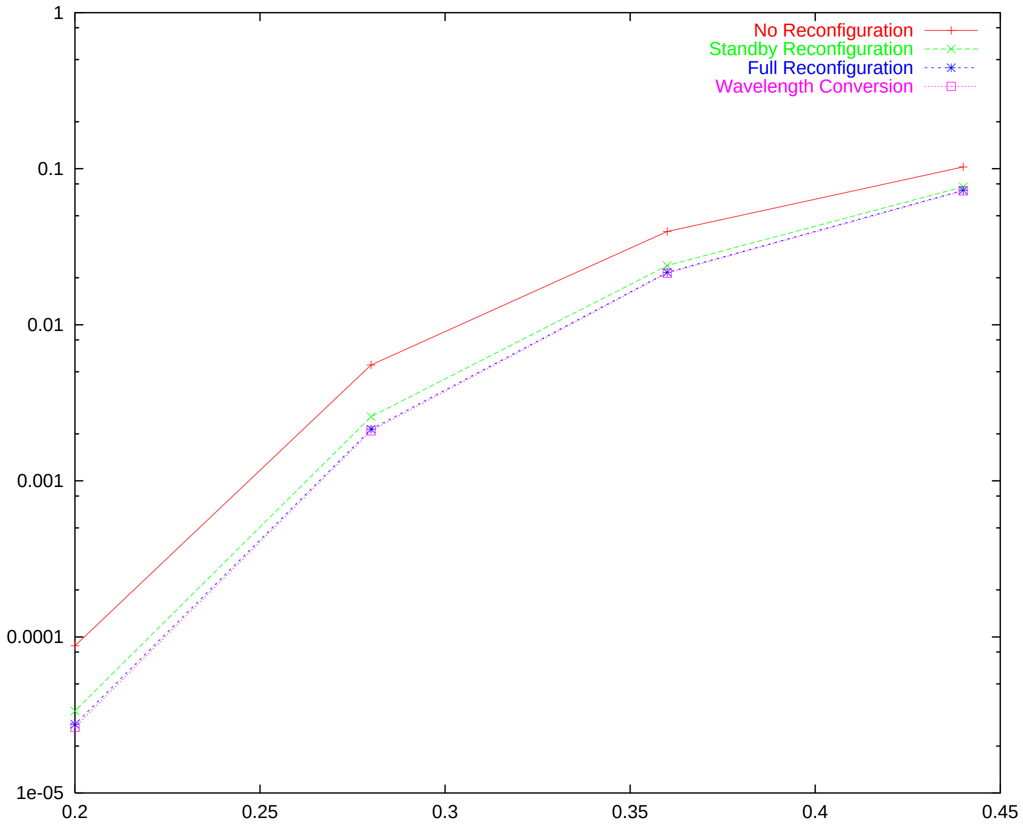

Figure 6.3: Network A with protection paths. Wavelength allocation by *first-fit* and reconfiguration by *displacement*.

Load	0.44	0.36	0.28	0.2
None	0.1027	0.03964	0.005532	8.767×10^{-5}
Standby	0.07659	0.02394	0.002576	3.355×10^{-5}
Full	0.07283	0.02167	0.002142	2.754×10^{-5}
WC	0.07219	0.02144	0.002092	2.636×10^{-5}

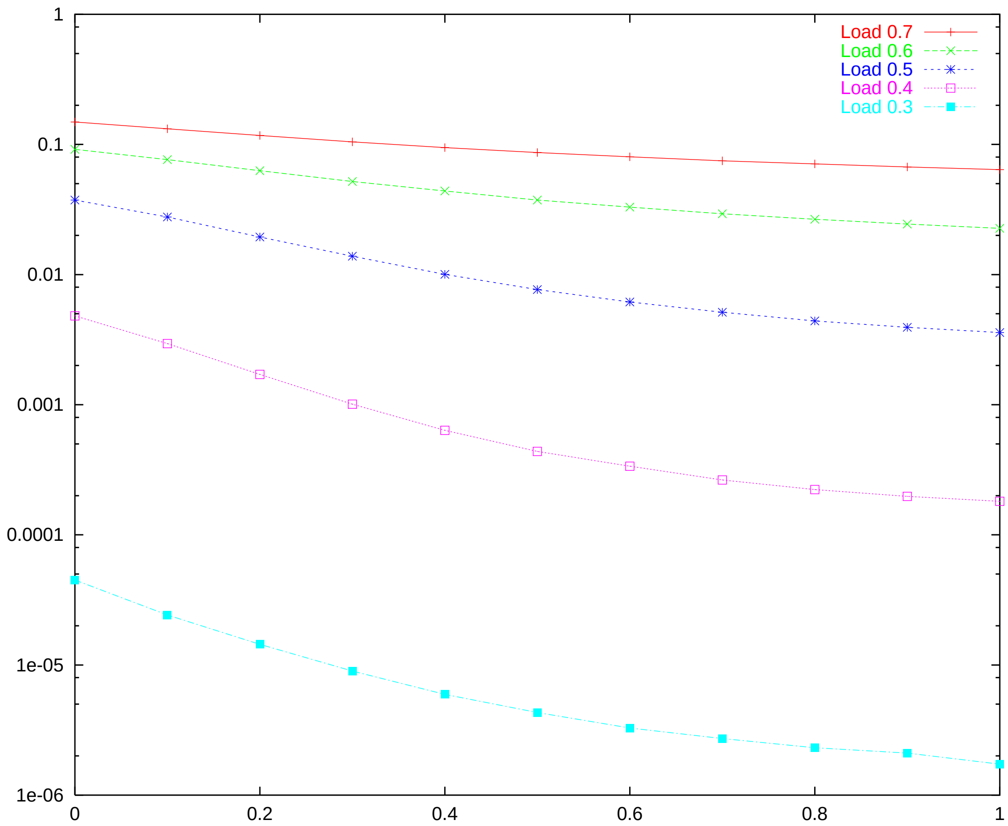

Figure 6.4: 7x7 lattice torus without protection paths. First-fit with displacement. Average network load varying from 0.3 to 0.7, and proportion of reconfigurable connections varying from 0 to 1; the bottom line of the table gives the analogous values for a network with complete wavelength conversion.

Load	0.7	0.6	0.5	0.4	0.3
0	0.1488	0.09179	0.03743	0.004819	4.494×10^{-5}
0.1	0.1319	0.07650	0.02768	0.002945	2.420×10^{-5}
0.2	0.1171	0.06289	0.01946	0.001709	1.442×10^{-5}
0.3	0.1047	0.05195	0.01384	0.001010	8.980×10^{-6}
0.4	0.09455	0.04391	0.01005	0.0006348	5.940×10^{-6}
0.5	0.08665	0.03739	0.007662	0.0004379	4.316×10^{-6}
0.6	0.08031	0.03298	0.006153	0.0003369	3.289×10^{-6}
0.7	0.07481	0.02934	0.005132	0.0002640	2.724×10^{-6}
0.8	0.07093	0.02660	0.004407	0.0002227	2.312×10^{-6}
0.9	0.06717	0.02447	0.003934	0.0001977	2.106×10^{-6}
1	0.06404	0.02265	0.003587	0.0001808	1.730×10^{-6}
WC	0.04544	0.01508	0.002466	0.0001398	1.656×10^{-6}

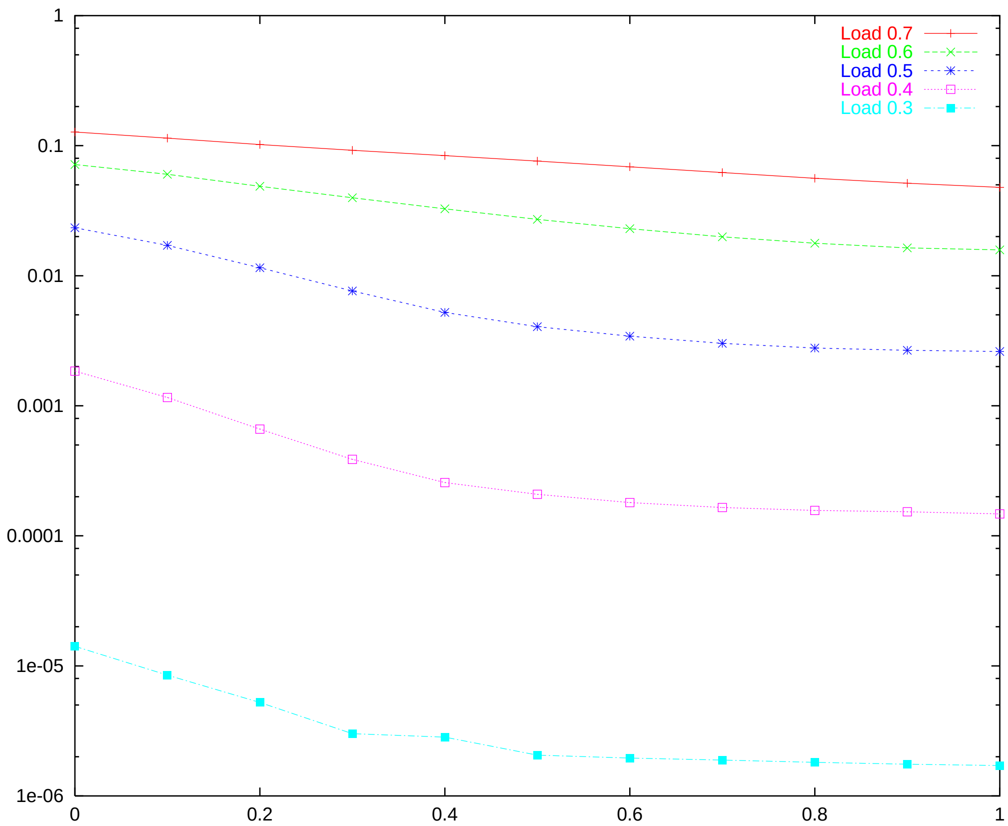

Figure 6.5: 7x7 lattice torus without protection paths. As Figure 6.4, but now using RCL with smallest-last repacking.

Load	0.7	0.6	0.5	0.4	0.3
0	0.1274	0.07162	0.02340	0.001846	1.415×10^{-5}
0.1	0.1143	0.06018	0.01711	0.001158	8.500×10^{-6}
0.2	0.1020	0.04873	0.01150	0.0006613	5.230×10^{-6}
0.3	0.09218	0.03979	0.007637	0.0003871	3.010×10^{-6}
0.4	0.08389	0.03266	0.005222	0.0002567	2.830×10^{-6}
0.5	0.07606	0.02712	0.004059	0.0002087	2.060×10^{-6}
0.6	0.06881	0.02297	0.003431	0.0001802	1.955×10^{-6}
0.7	0.06218	0.01993	0.003021	0.0001650	1.888×10^{-6}
0.8	0.05618	0.01779	0.002780	0.0001566	1.815×10^{-6}
0.9	0.05141	0.01635	0.002670	0.0001530	1.752×10^{-6}
1	0.04782	0.01581	0.002617	0.0001475	1.713×10^{-6}
WC	0.04544	0.01508	0.002466	0.0001398	1.656×10^{-6}

low load levels, where only a small amount of reconfigurability yields large gains.

The simpler reconfiguration method, which may be more appropriate in various practical situations because of its lower computational overhead and better scalability, and because of the lower disruption caused to the operation of the network, is also shown to provide considerable performance gains.

References

[1] Anand, V. and Qiao, C., (2001) Static versus dynamic establishment of protection paths in WDM networks. *J. High Speed Networks* **10**, 317–327.

[2] Baroni, S. and Bayvel, P., (1997) Wavelength Requirements in Arbitrarily Connected Wavelength-Routed Optical Networks. *J. Lightwave Tech.* **15**, 242–251.

[3] Barry, R. A. and Humblet, P. A., (1996) Models of Blocking Probability in All-Optical Networks With and Without Wavelength Changers. *IEEE JSAC* **14**, 858–867.

[4] Kelly, F. P., (1991) Loss Networks. *Annals of Applied Probability* **1**, 319–378.

[5] Martin, J. B., (2002) Partial Reconfigurability in Circuit-Switched WDM Networks with Routing Choice. Preprint.

[6] Subramaniam, S. and Barry, R. A., (1997) Wavelength Assignment in Fixed Routing WDM Networks. *IEEE International Conf. Commun.* pages 406–410.

[7] Yates, J. M., Rumsewicz, M. P. and Lacey, J. P. R., (1999) Wavelength Converters in Dynamically Reconfigurable WDM Networks. *IEEE Communications Surveys* .

[8] Zang, H., Jue, J. P. and Mukherjee, B., (2000) A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks. *Optical Networks Magazine* **1**.

[9] Zhang, X. and Qiao, C., (1998) Wavelength assignment for dynamic traffic in multi-fiber WDM networks. In *Proceedings of the 7th International Conference on Computer Communications and Networks*, pages 479–485. Lafayette, LA.