Lightpath Rerouting in Wavelength-Routed WDM Networks

Xiaowen Chu,[1,*] Hao Yin,[2] and Xiang-yang Li[3]

[1]*Department of Computer Science, Hong Kong Baptist University, Hong Kong, China*

[2]*Department of Computer Science and Technology, Tsinghua University, Beijing, China*

[3] *Department of Computer Science, Illinois Institute of Technology, USA*

[*]*Corresponding author: chxw@comp.hkbu.edu.hk*

Rerouting is a viable and cost-effective approach to decrease the blocking probability in legacy circuit-switched networks. We study lightpath rerouting in optical WDM networks in this paper. First, we investigate two different lightpath rerouting strategies, namely, *passive rerouting* and *intentional rerouting*. Passive rerouting means the rerouting of existing lightpaths to accommodate new lightpath requests which will otherwise be blocked. Intentional rerouting is to intentionally reroute existing lightpaths during their life period so as to achieve a better load balancing. Second, we investigate the *hybrid rerouting* scheme which combines passive rerouting and intentional rerouting. Through extensive simulation studies, we draw the following conclusions: 1) when there is wavelength conversion, passive rerouting works much better than intentional rerouting; and hybrid rerouting can only improve the performance over passive rerouting slightly; 2) when there is no wavelength conversion, a naïve-wavelength-retuning algorithm can achieve the most benefit of passive rerouting while path-adjusting does not help any further; however, the hybrid rerouting scheme can improve the blocking performance significantly.

OCIS codes: 060.1155, 060.4265.

1. Introduction

After entering the twenty-first century, we are witnessing another storm of Internet explosion: the number of Internet users keeps on growing drastically; more and more optical fibers are planted underground or submarine; new killer applications proliferate rapidly, such as P2P file sharing and P2P video streaming; new high-speed access network technologies begin to replace the old ADSL and Cable-Modem (Fiber-to-the-home has already become very popular in Japan, Korea, and Hong Kong). Today's P2P applications (e.g., BitTorrent) can easily eat up a 10Mbps or even 100Mbps Ethernet connection; and lots of users are using such software more than ten hours per day. Lots of ISPs are starting to complain about the shortage of backbone bandwidth. All of the above factors will boost the deployment of Wavelength Division Multiplexing (WDM) technology.

WDM has been studied for more than two decades, and it is widely used for point-to-point long-distance communications nowadays. However, a more promising technique is the wavelength-routed all-optical WDM networks [1]. A wavelength-routed all-optical WDM network consists of optical wavelength routing nodes called *wavelength routers* interconnected by optical fiber links. Each fiber link can support a number of wavelength channels by using WDM. A *lightpath*, i.e., an optical communication path, is established between the source node and the destination node upon receiving a connect request from a client [2]. Today, a single lightpath can carry about 40Gbps of data traffic, and its holding period is usually very long (e.g., weeks to months) as compared with the circuit holding time in telephone network. To establish a lightpath, it is normally required that the same wavelength channel be allocated on all the links along the route. This limitation is known as the *wavelength continuity constraint*, which makes the wavelength-routed WDM networks different from the traditional circuit-switched telephone networks.

In this paper, we are mainly concerning with the dynamic traffic model, in which lightpath connection requests arrive over time dynamically and each lightpath has a random holding time. Each lightpath needs to be set up by determining a route across the network connecting the source to the destination, and allocating a free wavelength channel along the route. Some of the lightpath requests could be blocked if there is currently no common free wavelength along the route. One of the primary design objectives of wavelength-routed all-optical WDM networks is to minimize the blocking probability. To achieve this goal, lots of efforts have been made in the literature, mainly in two different directions:

(1) *Wavelength conversion* [3-7]: Wavelength conversion means changing the optical signal from one wavelength to another. It can eliminate the wavelength continuity constraint and thus improve the blocking performance. In this paper, we only consider two cases: no wavelength conversion, and full wavelength conversion. Under full wavelength conversion, a lightpath can be setup on a path if every link on the path has at least one free wavelength channel.

(2) *Routing and Wavelength Assignment (RWA) algorithms* [8-17]: Existing research results have shown that adaptive routing algorithms can usually achieve better performance than static routing algorithms. In adaptive routing algorithms, a set of candidate routes are pre-calculated for each source-destination node pair. The focus is to choose the “best” route for the lightpath connection request based on the information of the network status, such as the traffic load distribution. Once the lightpath has been established, its physical route and its wavelength are not allowed to be changed.

In this paper, we study a third direction of improving the blocking performance, i.e., by using *lightpath rerouting*. Normally, once a lightpath has been setup, its physical path and its wavelength will not change. Lightpath rerouting means the action of changing the physical path and/or the wavelength(s) of an established lightpath. There could be different reasons to perform lightpath rerouting. For example, if a lightpath is corrupted because of link failure or node failure, it has to be rerouted to another physical path. But in this paper, we do not consider network failure; instead, we are more interested in studying the benefit of lightpath rerouting in terms of decreasing the connection blocking probability.

We propose three different rerouting strategies in this paper. The first one is called *passive rerouting*. In passive rerouting scheme, once a lightpath request cannot be satisfied by the current network, we try to reroute some existing lightpaths such that the new lightpath request can be accepted. It is straightforward that passive rerouting can decrease the overall blocking probability. An example of passive rerouting is shown in Fig. 1. In Fig. 1 (a), assume lightpath L_{12} is originally setup between node 1 and node 2 using the blue wavelength, and lightpath L_{23} is setup between node 2 and node 3 using the red wavelength. Due to the wavelength continuity constraint, a new lightpath request between node 1 and node 3 has to be blocked. However, in Fig. 1 (b) where passive rerouting is allowed, we can first reroute lightpath L_{23} from the red wavelength to blue wavelength (using the same link); afterwards we can successfully setup the lightpath L_{13} between node 1 and node 3 using the red wavelength. In this example, the physical path of lightpath L_{23} is not changed; instead, only its wavelength is changed. This is referred to as *wavelength-retuning* [18]. In case the physical path of an existing lightpath is changed, we call it *path-adjusting*. An example of path-adjusting is shown in Fig. 2. In this example, we assume each link only supports one wavelength channel. In Fig. 2(a), a lightpath has been setup though path *c-e-d*. If a new request between *a* and *d* arrives, this request will be blocked because link *e-d* has no free wavelength. By path-adjusting, we can shift the lightpath *c-e-d* to *c-f-d*, and then the request between *a* and *b* can be accepted by building a lightpath *a-e-d-b*, as shown in Fig. 2(b).

The second rerouting strategy is called *intentional rerouting*, by which existing lightpaths are dynamically rerouted to a more suitable physical path to achieve a better load balancing during its whole life period. The main supporting argument of such rerouting strategy is that, a lightpath usually holds for a very long period. When it is setup at the very beginning, it could select a good path at that time; however, the network traffic distribution is changing with time, and it is possible that the selected path may not always be a good choice. It is therefore possible to improve the overall blocking performance by dynamically changing the physical paths of existing lightpaths. An example of intentional rerouting is shown in Fig. 3. In Fig. 3 (a), the lightpath between node pair *(s, t)* is originally setup on path *s-a-b-t* with a hop-length of 3. Maybe after a while, it is possible to move the lightpath to the shorter path *s-c-t*, as shown in Fig. 3 (b). It is obvious that intentional rerouting can utilize wavelength resources more efficiently.

The third rerouting strategy is the combination of passive routing and intentional rerouting, which is referred to as *hybrid rerouting*.

All the three routing strategies are studied under two cases: no wavelength conversion, and full wavelength conversion.

The remainder of this paper is organized as follows. Some related work in the literature is reviewed in Section 2. Then we discuss our passive rerouting algorithms, intentional rerouting algorithms, and hybrid rerouting algorithms in Section 3, 4, 5, respectively. Our simulation results and analysis are presented in Section 6. Finally, Section 7 concludes the paper.

2. Related Work

The concept of rerouting is originally introduced in the design of circuit-switched telephone networks [19, 20]. It has also been applied to optical WDM networks recently [18, 21-28]. Rerouting is simply the action of switching an active *circuit* (or *virtual path* in ATM network, *lightpath* in WDM network) from one path to another path without changing the source and destination. A comprehensive survey of rerouting techniques can be found in [29]. An analysis of rerouting in circuit-switched network is given in [30], which studies symmetrical fully connected networks.

In [18, 24], rerouting is used as an approach to alleviate the effects of wavelength continuity constraint when there is no wavelength conversion. The basic idea is that, once a new lightpath request cannot be setup directly, the system will try to reroute some existing lightpaths to create a wavelength-continuous route so as to accommodate the new lightpath request. Specifically, a rerouting scheme called *move-to-vacant wavelength-retuning* (MTV-WR) was proposed by the authors, which has a very short lightpath disruption period. *Move-to-vacant* means rerouting a lightpath to a free path without affecting other established lightpaths. *Wavelength-retuning* means retuning the wavelength of a lightpath without changing its physical path. In MTV-WR, a few existing lightpaths may be reassigned to different wavelengths without changing their physical paths, in order to accommodate a new lightpath request which will be otherwise blocked. By simulation studies, MTV-WR has been shown to reduce the blocking probability by 30% on average. The main concern of their rerouting algorithm is to minimize the lightpath disruption time. The time complexity of the algorithm is $O(N^3W + N^2W^2)$ where N is the number of nodes and W is the number of wavelength channels per link. In [25], the authors have developed a sophisticated time optimal rerouting algorithm aiming to further shorten the disruption time. The time complexity of their rerouting algorithm is $O(N^2W)$. In the above rerouting schemes, the average number of rerouted lightpaths per rerouting case is more than one and increases as the request arrival rate increases.

Passive rerouting in WDM networks with sparse wavelength conversion has been studied in [26]. In sparse wavelength conversion WDM networks, only some nodes in the network can do wavelength conversion [6]. The author extends the work of [24, 25] to semi-lightpath routing/rerouting. However, only wavelength-retuning is studied. In [28], a departure-triggered rerouting strategy has been proposed. It performs rerouting at the time when an existing lightpath is due for departure. After releasing a lightpath *D*, it tries to reroute some other lightpath *P* whose source node and destination node are both on the path of lightpath *D*, in case wavelength resources can be saved by this rerouting action.

In [22], the authors propose an *intentional rerouting* scheme in wavelength routed WDM networks. The basic idea is to intentionally reroute existing lightpaths to some vacant routes if better load balancing can be achieved. They propose a new adaptive routing scheme called Dynamic Least Congested Routing (DLCR) which dynamically switches existing lightpaths to the least congested routes. The blocking performance of DLCR has been shown to outperform that of the Shortest Path Routing (SPR), Fixed-Alternate Routing (FAR) [12] and Least Congested Routing (LCR) [9] algorithms. However, DLCR only considers the *k*-shortest paths and it is too aggressive to reroute a lightpath.

Our work differs from previous work in the following aspects:

(1) In passive rerouting with full wavelength conversion, we propose a heuristic move-to-vacant one-path-adjusting algorithm (MTV-OPA) with polynomial time complexity. Our

simulation result illustrates that the MTV-PA algorithm can improve the blocking performance significantly.

(2) In passive rerouting without wavelength conversion, we propose a move-to-vacant naïve-wavelength-retuning (MTV-NWR) algorithm and also a path-adjusting algorithm. We show that our simple NWR algorithm can achieve very good performance, while the path-adjusting scheme can only provide very marginal improvement.

(3) We propose a generalized timer-based intentional rerouting framework which introduces a threshold parameter when making a rerouting decision.

(4) We study a hybrid rerouting scheme which combines passive rerouting and intentional rerouting.

3. Passive Lightpath Rerouting

In this section, we propose a set of passive lightpath rerouting algorithms. In the case of full wavelength conversion, wavelength-retuning does not make sense any more because there is no wavelength continuity constraint. We first describe a basic routing algorithm without passive rerouting; then we propose a passive rerouting algorithm which performs path-adjusting to accommodate the otherwise blocked new arrivals. When there is no wavelength conversion, we also first describe a basic routing algorithm without passive rerouting; then we propose an efficient passive rerouting algorithm using only wavelength-retuning; we further propose a passive rerouting algorithm which can perform path-adjusting in case wavelength-retuning does not help.

The physical topology of a wavelength-routed WDM network is represented by an undirected graph $G = (V, E)$ where V is the set of wavelength routers and E is the set of fiber links. Each fiber link can support a set of W wavelengths, denoted by $\Lambda = \{\lambda_1, \lambda_2, \cdots, \lambda_W\}$. With full wavelength conversion, a lightpath between node pair (s,t) is denoted by $p(s,t)$. When there is no wavelength conversion, a lightpath between node pair (s,t) using wavelength i is denoted by $p_i(s,t)$. The set of all current existing lightpaths is denoted by L, and we denote the number of currently existing lightpaths as $|L|$.

3.A. With Full Wavelength Conversion

If full wavelength conversion is provided, a lightpath can be setup if all the links along the path have at least one free wavelength. Since there is no wavelength continuity constraint, wavelength-retuning becomes meaningless. In fact, the main purpose of wavelength-retuning in the case of no conversion is to achieve a close performance to that of full wavelength conversion [18, 24, 25].

In order to show the benefit of rerouting, we first describe a basic routing algorithm called Shortest Available Path Routing (SAPR) without using rerouting. This algorithm is an extension of the FAR algorithm [12]: upon arrival of a lightpath request, the shortest available path is selected to setup the lightpath.

Algorithm 1: SAPR-FC

---

1) For each connection request on node pair (s,t), we use *Breadth First Search* (BFS) to find the shortest available path between node s and t.

2) If there is no such a route with free wavelength channels, block the connection request.

Because the running time of BFS is $O(|V|+|E|)$, the time complexity of SAPR under full conversion is $O(|V|+|E|)$.

With full wavelength conversion, the topology can be simplified by a weighted undirected graph $G = (V, E, w)$, where w represents the weight function mapping from link e to the number of free wavelength channels on it. If a lightpath request cannot be setup by the above SAPR algorithm, it is possible to adjust some established lightpaths in order to accommodate the new request. In order to minimize the lightpath disruption time, we limit that only one lightpath is allowed to be rerouted. Therefore we name the algorithm Move-to-Vacant One-Path-Adjusting (MTV-OPA). It is worthy to point out that, the "release" operation is only performed in the algorithm, which does not mean to release the physical lightpath. The same rule applies to the "restore" operation in Step 4 and Step 5.

Algorithm 2: MTV-OPA-FC

---

1) For each connection request on node pair (s,t), we first use BFS to find the shortest available route between node s and t. If there is no path with free wavelength channels, goto Step (2) to try to reroute one established path to accept this connection request:
2) If L is empty, block this request; else goto Step (3).
3) If L is not empty, select one existing lightpath from set L, say $p(s',t')$. Release it in graph G, and remove it from L.
4) For both nodes (s,t) and (s',t'), solve the maximum-flow problem with multiple sources and sinks. By adding a supersource s_0 and supersink t_0, the maximum-flow problem with

multiple sources and sinks can be reduced to an ordinary maximum-flow problem. If the value of the maximum flow is larger than 1, goto Step (5). Otherwise, restore the original path $p(s',t')$ in graph *G* and goto Step (2).

5) On the induced subgraph G' where each link's flow is non-zero, use BFS to find the shortest path for (s',t'). If such path cannot be found, restore the original path $p(s',t')$ in graph *G* and goto Step (2). Otherwise, update graph *G* and goto Step (6).

6) On graph *G*, use BFS to find the shortest available path for the node pair (s,t). If there does not exist such a path, goto Step (2).

7) Perform path-adjusting on $p(s',t')$. Afterwards, we setup the new lightpath request for (s,t).

The running time of the maximum-flow problem is cubic in the number of nodes, i.e., $O(|V|^3)$, by using the relabel-to-front algorithm [31]. Therefore the time complexity of the above MTV-OPA rerouting algorithm is $O(|L||V|^3)$.

It is a natural idea that an algorithm may reroute a set of established lightpaths to accept the new connection request. Unfortunately it is an NP-Hard problem, since it is a general case of maximum disjoint connecting paths problem [32]. It is also not practical to disrupt lots of existing lightpaths without breaking the Move-to-Vacant rule in order to accept a new lightpath request.

3.B. No Wavelength Conversion

When there is no wavelength conversion, we can have two different approaches of passive rerouting: wavelength-retuning and path-adjusting.

We first present a basic routing algorithm without rerouting, for performance comparison purpose.

Algorithm 3: SAPR-NC

---

1) Given the topology $G=(V,E)$, we first reproduce W copies of G (referred to as sub-graph), labeled by $\{G_1, G_2, \cdots, G_W\}$, where G_i denotes the topology for wavelength channel i. Due to the wavelength continuity constraint, a lightpath can only be setup within some sub-graph G_i.
2) Then, for each connection request on node pair (s,t), we use BFS to find the shortest available route between node s and t from the graph set $\{G_1, G_2, \cdots, G_W\}$. If there exist a tie, break it by choosing the path with smaller wavelength index, i.e., to use the first-fit wavelength assignment policy [17].
3) In case there is not such a path in all the W sub-graphs, the lightpath request has to be blocked.

For each sub-graph G_i, the running time of BFS is $O(|V|+|E|)$. Since the algorithm needs to search all the W sub-graphs, its time complexity is $O(W(|V|+|E|))$.

Next, we propose a passive rerouting algorithm using only wavelength-retuning. Different from the algorithms in [24, 25], we allow retuning only one lightpath to accommodate the new request, such that the algorithm is very simple and fast to be implemented, and its performance is almost the same of those in [24, 25]. The algorithm is named as Move-to-vacant Naïve-Wavelength-Retuning (MTV-NWR).

Algorithm 4: MTV-NWR

---

1) For each connection request on node pair (s,t), we first call the SAPR-NC algorithm. If the request cannot be accepted, goto Step (2) to try to reroute one established path using wavelength-retuning to accept the connection request:

2) If L is empty, block it; else goto Step (3).

3) If L is not empty, select one existing lightpath from set L, say $p_i(s',t')$. Release it, and remove it from L.

4) Using BFS to search the shortest path $p_i'(s,t)$ in graph G_i. If such a path exists, goto Step (5); otherwise, restore the original path $p_i(s',t')$ and goto Step (2).

5) From the graph set $\{G_1, G_2, \cdots, G_W\} \setminus \{G_i\}$, use BFS to find the shortest available path for node pair (s',t'). If success, goto Step (6). In case there is a tie, break it by choosing the smallest wavelength index. If such path cannot be found, release path $p_i'(s,t)$ and restore the original path $p_i(s',t')$, and then goto Step (2).

6) Perform wavelength-retuning on $p_i(s',t')$, i.e., retune the wavelength from *i* to *j*. Afterwards, we setup the new lightpath request using $p_i'(s,t)$.

The running time of checking one adjustment is $O(W|V|)$. Therefore the time complexity of the above algorithm is $O(W|L||V|)$.

It is obvious that wavelength-retuning cannot always help. Therefore we apply the previous MTV-OPA algorithm here, which tries to move the physical path of some established lightpath to accept a new request, in case MTV-NWR algorithm fails.

Algorithm 5: MTV-OPA

---

1) For each connection request on node pair (s,t), we first call shortest path routing algorithm to process it. If the request can not be accepted, goto Step 2 to try to readjust one established path to accept the connection request on node pair (s,t):
2) If *L* is empty, block it; else goto Step 3.
3) If *L* is not empty, select one existing lightpath from set *L*, say $p_i(s',t')$. Release it, and remove it from *L*.
4) If by BFS, one shortest path for node pair $p_i(s,t)$ can be found on sub-graph G_i, goto Step 5; else restore the original path $p_i(s',t')$ and goto Step 2.
5) From the sub-graph set $\{G_1, G_2, \cdots, G_W\} \setminus \{G_i\}$, use BFS to find the shortest available path for node pair (s',t'). If success, goto Step 6. In case there is a tie, break it by choosing the path with the smallest wavelength channel index. If such path cannot be found, release path $p_i(s,t)$ and goto Step 2.
6) Assume the shortest path found in Step 5 is $p_j(s',t')$. Reroute the lightpath $p_i(s',t')$ to $p_j(s',t')$. Afterwards, we setup the new lightpath request using $p_i(s,t)$.

The running time of adjustment once is $O(W(|V|+|E|))$. The time complexity of one-wavelength-path-adjustable rerouting is therefore $O(W|L|(|V|+|E|))$.

4. Intentional Lightpath Rerouting

In intentional lightpath rerouting, an established lightpath dynamically adjusts its physical path according to some pre-defined criteria. One example is to switch the lightpath from a longer path to a shorter one; another example is to switch the lightpath from an over-loaded path to a slightly-loaded path. In this section, we propose a general framework of Timer-based Intentional Rerouting algorithm. It can work for both no wavelength conversion and full wavelength conversion.

Algorithm 6: Timer-based Intentional Rerouting (TBIR)

---

1) For each node pair (s,t), we first pre-calculate the k-shortest paths, denoted by $p^1(s,t)$, $p^2(s,t)$, …, $p^k(s,t)$. Each path $p^i(s,t)$ is associated with a weight value $w^i(s,t)$ which can be calculated by a pre-defined weight function. The design of the weight function could be very complicated. Usually it should consider lots of factors, e.g., the path hop-length, and the free wavelength distributions. We assume that a large weight value means a good candidate path. We also define a positive number ***ts*** as a threshold to control the rerouting behavior.

2) For each connection request on node pair (s,t), we call the SAPR algorithm to setup the lightpath request. It is possible that the selected path is different from any of the k-shortest paths. For each established lightpath, we associate a timer at the source node to trigger the intentional rerouting routine. The timer will fire after a time period named *Reroute Time Interval* (RTI).

3) Once the timer of any lightpath $p(s,t)$ fires, perform the following routine:

---

(1) Calculate the weight value of the current path $p(s,t)$, denoted by $w(s,t)$.

(2) Calculate $w^i(s,t)$ for $i = 1, 2, \ldots, k$.

(3) If $w(s,t) > \max\{w^i(s,t) : i = 1,2,\cdots,k\} - ts$, it is not necessary to do wavelength rerouting, goto Step (4); else, assume $w^j(s,t) = \max\{ w^i(s,t) : i = 1,2,\cdots,k \}$, reroute the lightpath from $p(s,t)$ to $p^j(s,t)$ using the smallest available wavelength index as follows:

a. Setup a lightpath between (s,t) on path $p^j(s,t)$;

b. Switch the optical signal from the original lightpath $p(s,t)$ to the new one on $p^j(s,t)$;

c. Release the old lightpath on $p(s,t)$.

(4) Reset the timer.

The above algorithm is trying to reroute an exiting lightpath to one of the *k*-shortest path with the highest weight value and the difference between the weight values is beyond the pre-defined threshold. The algorithm has the following three advantages: First, it is simple to be implemented since the lightpaths are always rerouted to vacant paths. Rerouting of a lightpath does not affect other existing lightpaths. Second, the computational requirement is very low, i.e., to calculate the weigh values for a set of pre-calculated paths. Third, the lightpath disruption time is minimized to the physical limitation of switching the optical signal from one lightpath to another, since the data transmission is preserved on the old lightpath during the setup of the new one. The value of RTI has a great impact on the achievable performance improvement. A smaller value of RTI can lead to more chances of lightpath rerouting, which means better load balancing. But if the value of RTI is too small, the signaling overhead will increase. A simple heuristic is to set RTI as *T*/*A* where *T* is the average lightpath holding time and *A* is the traffic load in Erlang. By doing so, the

TBIR algorithm will be able to catch up with the changing of network status caused by lightpath arrivals and departures.

5. Hybrid Lightpath Rerouting

Passive rerouting and intentional rerouting are two quite different rerouting strategies. It is very interesting to see whether a combination of these two rerouting strategies can further improve the blocking performance. Therefore, in this section, we propose two hybrid rerouting (HR) algorithms, for full wavelength conversion and no wavelength conversion, respectively.

Algorithm 7: HR-FC

Part 1:

Upon arrival of a lightpath request on node pair (s,t), call the MTV-OPA algorithm.

If success, associate a timer with the lightpath; If failed, the request is blocked.

Part 2:

Run the TBIR algorithm for every established lightpath.

Algorithm 8: HR-NC

Part 1:

Upon arrival of a lightpat request on node pair (s,t), call the MTV-NWR algorithm.

If success, associate a timer with the lightpath; If fail, the request is blocked.

Part 2:

Run the TBIR algorithm for every established lightpath.

6. Performance Evaluation

The blocking performances of all the rerouting algorithms are evaluated by extensive simulation studies on a set of network topologies. Since we are only interested in the steady-state blocking probability, the data of initial transient period in each simulation are discarded. For each data point, 20 independent simulations are conducted and the 95% confidence interval of the blocking probability is estimated. The running time of each simulation is set to be long enough for achieving a small width of confidence interval. For instance, when the blocking probability is at the order of 10^{-5}, about 100 million lightpath requests are generated in just one simulation; while for a higher blocking probability such as 10^{-2}, 1 million lightpath requests per simulation are good enough to generate precise estimations of the blocking probabilities.

In our simulations, the lightpath requests arrive to the network following a Poisson process; and each node pair has the same lightpath request arrival rate. The lightpath holding time is exponentially distributed with a unit time. Each fiber link can support 40 bi-directional wavelength channels.

Due to the space limitation, we only present the simulation results of the following three topologies: 21-node ARPA-2 network (Fig. 4 (a)), 14-node NSFNET (Fig. 4(b)), and 25-node mesh-torus network (Fig. 4 (c)). The specific TBIR algorithm we are using in the simulation is based on the least-loaded principle: (1) $w(s,t)$ is the number of free wavelengths of path $p(s,t)$; (2) k = 2, which means that we use 2 candidate paths for each node pair; (3) ts = 2, which means that intentional rerouting is performed only if the new path has at least two more free wavelength channels than the present path.

6.A. Discussion on the Value of RTI

As we mentioned in Section 4, the value of RTI has some impact on the blocking performance. We conduct a set of simulations using NSFNET topology, with different values of RTI. Fig. 5(a) shows the blocking probabilities for different values of RTI, and Fig. 5(b) shows the average number of reroutings per lightpath for different values of RTI. We can see that smaller value of RTI leads to more times of lightpath rerouting, which results in a better load balance and hence a smaller blocking probability. When the value of RTI approaches *T/A*, the performance improvement by decreasing RTI value becomes smaller and smaller. It is worthwhile to point out that rerouting is not a frequent operation. In the whole lifetime of a lightpath, on average there is no more than one rerouting action to be performed.

6.B. Performance Evaluation: With Full Wavelength Conversion

The blocking performances of different routing algorithms in ARPA-2, NSFNET, Mesh-torus are shown in Fig. 6, Fig. 7, Fig. 8, respectively. In all three topologies, we notice that DLCR performs better than LCR algorithm, due to the intentional rerouting. TBIR further improves the performance over DLCR. Recall that DLCR only considers the *k*-shortest paths but TBIR tries all the possible paths when accepting a new request. The magnitude of performance improvement depends on the network topology and also the traffic load. The performance of SAPR depends on the network topology and traffic load a lot. In ARPA-2, the sparsest topology among the three, SAPR does not perform well when the traffic load is low, but it can even outperform TBIR when the traffic is beyond 160 Erlangs. It is well-known that dynamic routing algorithms are not suitable for very sparse topologies, such as ring, when wavelength conversion is available [10]. The main reason is that in a sparse topology, the secondary path is usually much longer than the primary path. Dynamic routing algorithms (including the TBIR) may use too many secondary paths, which could decrease the resource utilization ratio. For example, in ARPA-2 topology, the average length of secondary paths is 2.94 hops longer than the average length of primary paths. As a comparison, the counterpart in NSFNET and Mesh-torus is 1.55 hops and 0.53 hops, respectively. In NSFNET and Mesh-torus, TBIR algorithm performs much better than SAPR algorithm in all traffic loads that we have simulated. The performance of MTV-OPA algorithm is very promising. In a reasonable range of blocking probability (<10%), we can conclude that MTV-OPA is an effective approach to improve the blocking performance. Our last observation is that, the performance of HR algorithm is only slightly better than that of MTV-OPA when the traffic load is low. Once the traffic load is beyond some threshold, the performance of HR becomes a little worse than that of MTV-OPA. This can be explained as follows. When the traffic load is low, the network is not fully utilized; intentionally rerouting some existing lightpaths to lightly loaded paths can be positive to the decrease of blocking probability. But if the traffic load is high, the network is already heavily utilized. With full wavelength conversion, considering a pair of link-disjoint paths between a node pair, the probability that the longer path has more free wavelengths than the shorter path is not negligible; hence intentional rerouting could choose the lightly loaded but longer path, which has a negative effect on the blocking probability.

6.C. Performance Evaluation: No Wavelength Conversion

In the case of no wavelength conversion, we first show that wavelength-retuning can achieve the most benefit of passive rerouting. Note that, in MTV-OPA algorithm, upon arrival a lightpath request, we actually first call the SAPR algorithm. If it fails, we denote such a lightpath request "failed request". We then call MTV-NWR algorithm; if it fails again, we try MTV-OPA algorithm. In Fig. 9, we show the percentage of (1) the number of requests that failed by SAPR but succeeded by MTV-NWR; (2) the number of requests that failed by SAPR and MTV-NWR but succeeded by MTV-OPA; (3) the blocked lightpaths; over the total number of failed requests. It is shown that 96% of the failed requests can be accepted by simple MTV-NWR algorithm; and only 1% of the failed requests can be further accepted by MTV-OPA algorithm. It is possible to accept more lightpath requests if we remove the One-Path-Adjusting constraint, i.e., by shuffling all existing lightpaths. However, doing so could disrupt lots of existing lightpaths because it may not be possible to reschedule the lightpaths without breaking the Move-to-Vacant rule.

We conduct simulations to compare the blocking performance of six different routing algorithms without wavelength conversion, namely, LCR, DLCR, SAPR, TBIR, MTV-NWR, and HR. The simulation results in ARPA-2, NSFNET, and Mesh-torus are shown in Fig. 10, Fig. 11, Fig. 12, respectively. In all curves, heavier traffic leads to higher blocking probability. In all three topologies, LCR algorithm performs the worst. DLCR performs better than LCR; SAPR performs better than DLCR; TBIR performs better than DLCR; MTV-NWR performs better than TBIR. Recall that in the case of full wavelength conversion, HR does not perform as well as MTV-OPA. But without wavelength conversion, HR performs the best among the six algorithms. The reason is that, without wavelength conversion, the probability that a longer path has more common free wavelength than a shorter path becomes very small. Hence the intentional rerouting seldom reroutes a lightpath from a shorter path to a longer path. Lots of rerouting actions are due to the fact that some lightpaths are accepted using a longer path at the very beginning. Intentional rerouting provides an opportunity to shift them to better paths.

7. Conclusion

In this paper, we have studied three different rerouting strategies, namely, passive rerouting, intentional rerouting, and hybrid rerouting. Our main conclusions are as follows:

When there is full wavelength conversion, path-adjusting is the only way of passive rerouting. In all the topologies we have investigated, passive rerouting outperforms intentional rerouting a lot. The benefit of rerouting is more significant in dense networks, such a mesh-torus. However, the combination of passive rerouting and intentional rerouting does not offer any advantage over pure passive rerouting.

When there is no wavelength conversion, wavelength-retuning is an efficient way to improve the blocking performance. Path-adjusting can only improve the performance of wavelength-retuning very marginally. Intentional rerouting can also improve the blocking performance, but not as notable as wavelength-retuning. The performance of hybrid rerouting is very promising, which is different from the case of full wavelength conversion.

References

[1] R. Ramaswami and K. Sivarajan, Optical Networks, a Practical Perspective, 2nd Edition, Morgan Kaufmann Publishers, 2002.

[2] 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.

[3] M. Kovacevic and A. Acampora, “Benefits of Wavelength Translation in All-Optical Clear-Channel Networks,” IEEE Journal on Selected Areas in Communications, vol. 14, no. 5, pp. 868-880, June 1996.

[4] H. Q. Ngo, D. Pan, and Y. Yang, "Optical Switching Networks with Minimum Number of Limited Range Wavelength Converters," IEEE/ACM Transactions on Networking, vol. 15, no. 4, pp. 969-979, August 2007.

[5] B. Ramamurthy and B. Mukherjee, "Wavelength Conversion in WDM Networking," IEEE Journal on Selected Areas in Communications, Vol. 16, No. 7, pages 1061-1073, September 1998.

[6] S. Subramaniam, M. Azizoglu, and A. K. Somani, "All-Optical Networks with Sparse Wavelength Conversion," IEEE/ACM Transactions on Networking, vol. 4, no. 4, pp. 544-557, August 1996.

[7] J. Yates, J. Lacey, D. Everitt, and M. Summerfield, "Limited Range Wavelength Translation in All-Optical Networks," in Proceedings of IEEE INFOCOM'96, pp. 954–961, 1996.

[8] N. M. Bhide, K. M. Sivalingam, and T. Fabry-Asztalos, "Routing Mechanisms Employing Adaptive Weight Functions for Shortest Path Rounting in Optical WDM Networks", Photonic Network Communications, vol. 3, no. 3, pp. 227-236, July 2001.

[9] K. Chan and T. P. Yum, "Analysis of Least Congested Path Routing in WDM Lightwave Networks," in Proceedings of IEEE INFOCOM '94, pp. 962-969, 1994.

[10] X. Chu and B. Li, "Dynamic Routing and Wavelength Assignment in the Presence of Wavelength Conversion for All-Optical Networks," IEEE/ACM Transactions on Networking, Vol. 12, No. 3, pp. 704-715, June 2005.

[11] O. Gerstel, S. Kutten, R. Ramaswami, and G. Sasaki, "Worst-case Analysis of Dynamic Wavelength Allocation in Optical Networks," IEEE/ACM Transactions Networking, vol. 7, no. 6, pp. 8333-8456, December 1999.

[12] H. Harai, M. Murata, and H. Miyahara, "Performance of Alternate Routing Methods in All-Optical Switching Networks," in Proceedings of IEEE INFOCOM '97, pp. 517-525, April 1997.

[13] L. Li and A. K. Somani, "Dynamic Wavelength Routing Using Congestion and Neighborhood Information," IEEE/ACM Transactions on Networking, vol. 7, no. 5, pp. 779-786, October 1999.

[14] A. Mokhtar and M. Azizoğlu, "Adaptive Wavelength Routing in All-Optical Networks," IEEE/ACM Transactions on Networking, vol. 6 no. 2, pp. 197-206, April 1998.

[15] R. Ramamurthy and B. Mukherjee, "Fixed-Alternate Routing and Wavelength Conversion in Wavelength-Routed Optical Networks," IEEE/ACM Transactions on Networking, vol. 10, no. 3, pp. 351-367, June 2002.

[16] R. Ramaswami and K. Sivarajan, "Routing and Wavelength Assignment in All-Optical Networks," IEEE/ACM Transactions on Networking, vol. 3, no. 5, pp. 489-500, October 1995.

[17] Y. Zhu, George N. Rouskas, and H. G. Perros, "A Comparison of Allocation Policies in Wavelength Routing Networks," Photonic Network Communications, vol. 2, no. 3, pp. 267-295, August 2000.

[18] K. C. Lee and V. O. K. Li, "A Circuit Rerouting Algorithm for All-Optical Wide-Area Networks," in Proceedings of IEEE INFOCOM'94, pp. 954-961, 1994.

[19] M. H. Ackroyd, "Call Repacking in Connecting Networks," IEEE Transactions on Communications, vol. COM-27, no. 3, pp. 589-591, March 1979.

[20] A. Girard and S. Hurtubise, "Dynamic Routing and Call Repacking in Circuit-Switched Networks," IEEE Transactions on Communications, vol. COM-31, no. 12, pp. 1290-1294, December 1983.

[21] A. Caprara, G. F. Italiano, G. Mohan, A. Panconesi, and A. Srinivasan, "Wavelength Rerouting in Optical Networks, or the Venetian Routing Problem," Journal of Algorithms, vol. 45, pp. 93-125, 2002.

[22] X. Chu and J. Liu, "DLCR: a New Adaptive Routing Schemes in WDM Mesh Networks," in Proceedings of IEEE ICC 2005, pp. 1797-1801, May 2005.

[23] R. Datta, S. Ghose, and I. Sengupta, "A Rerouting Technique with Minimum Traffic Disruption for Dynamic Traffic in WDM Networks," in proceedings of IEEE International Conference on Networks (ICON), pp. 425-430, 2003.

[24] K. C. Lee and V. O. K. Li, “A Wavelength Rerouting Algorithm in Wide-Area All-Optical Networks,” IEEE/OSA Journal of Lightwave Technology, vol. 14, no. 6, pp. 1218-1229, June 1996.

[25] G. Mohan and C. Siva Ram Murthy, “A Time Optimal Wavelength Rerouting Algorithm for Dynamic Traffic in WDM Networks,” IEEE/OSA Journal of Lightwave Technology, vol. 17, no. 3, pp. 406-417, March 1999.

[26] G. Xue, “Optimal Lightpath Routing and Rerouting in WDM Networks,” in proceedings of IEEE Globecom’01, pp. 2124-2128, 2001.

[27] W. Yao and B. Ramamurthy, “Rerouting Schemes for Dynamic Traffic Grooming in Optical WDM Mesh Networks,” in Proceedings of IEEE GLOBECOM’04, pp. 1793-1797, 2004.

[28] J. Zheng, B. Zhang, and H. T. Mouftah, “A Departure-Triggered Rerouting Mechanism for Wavelength-Routed WDM Networks,” in proceedings of International Symposium on Communication Systems, Networks and Digital Signal Processing, pp. 340-343, July 2002.

[29] E. W. M. Wong, A. K. M. Chan, and T. S. P. Yum, “A Taxonomy of Rerouting in Circuit-Switched Networks,” IEEE Communications Magazine, pp. 116-122, November 1999.

[30] E. W. M. Wong, A. K. M. Chan, and T. S. P. Yum, “Analysis of Rerouting in Circuit-Switched Networks,” IEEE/ACM Transactions on Networking, vol. 8, no. 3, pp. 419-427, June 2000.

[31] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, Second Edition, MIT Press, 2001.

[32] A. Srinivasan, “Improved Approximations for Edge-disjoint Paths, Unsplittable Flow, and Related Routing Problems,” in Proceedings of IEEE Symposium on Foundations of Computer Science’97, IEEE Computer Society, pp. 416-425, 1997.

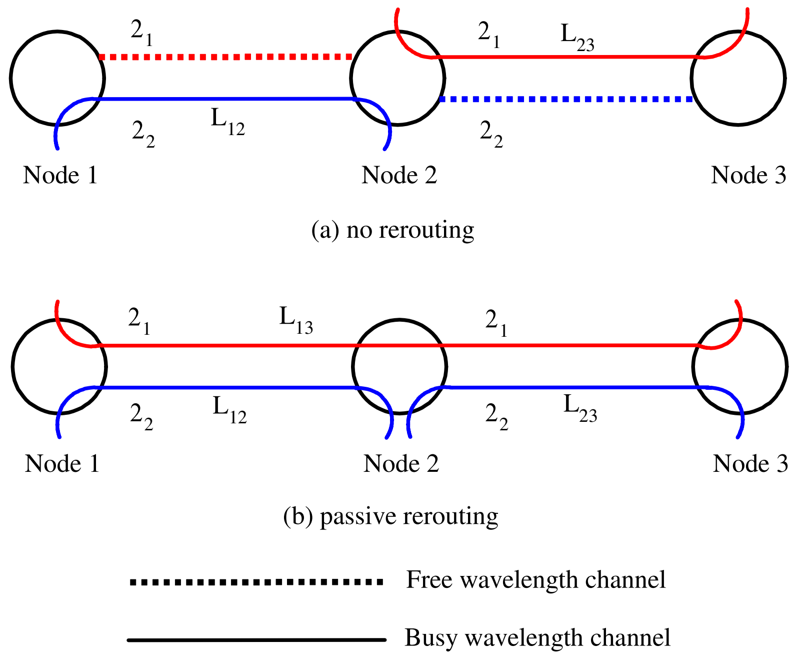


Fig. 1. Example of passive rerouting (wavelength-retuning)

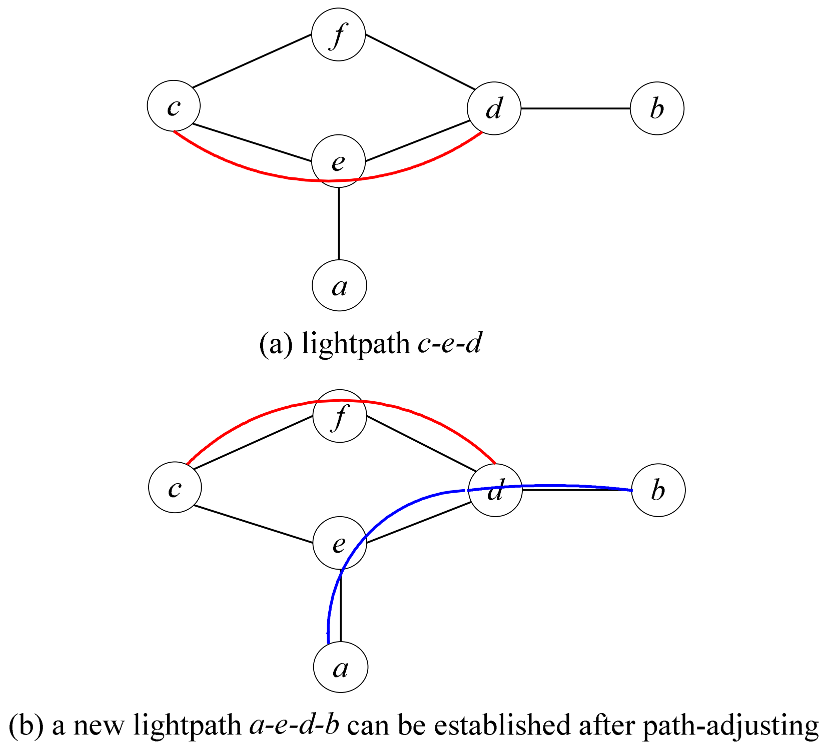
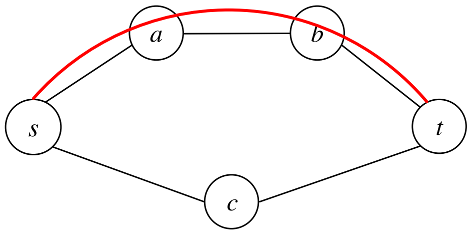
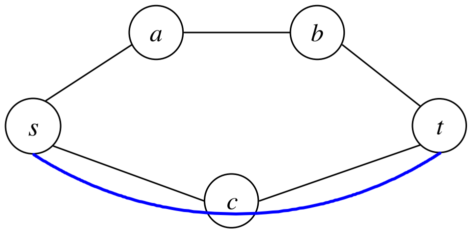


Fig. 2. Example of passive rerouting (path-adjusting)

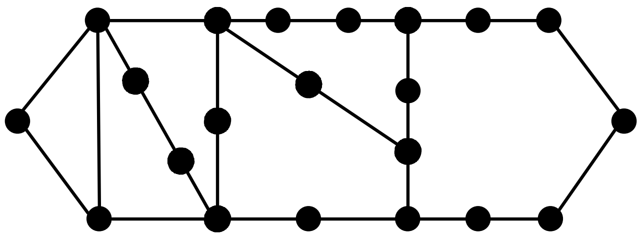


(a) lightpath *s*-*a*-*b*-*t*

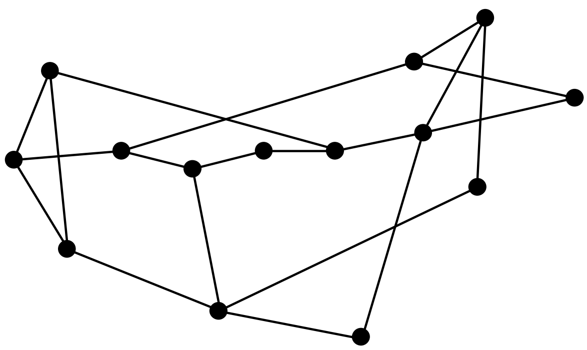


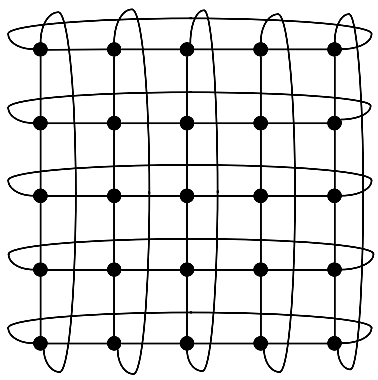
(b) lightpath *s*-*c*-*t*

Fig. 3. Example of intentional rerouting



(a) 21-node ARPA-2

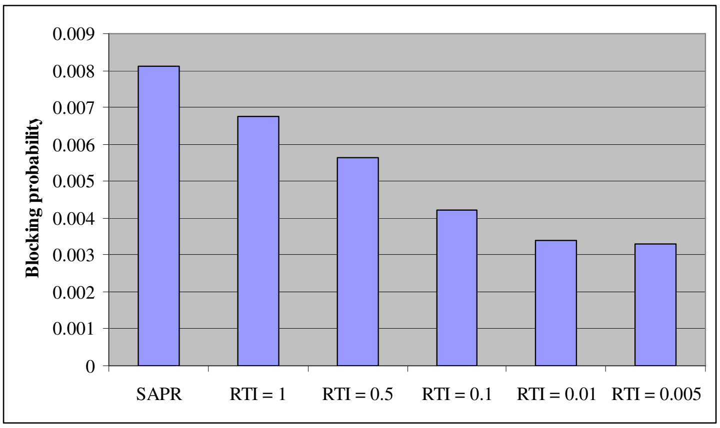




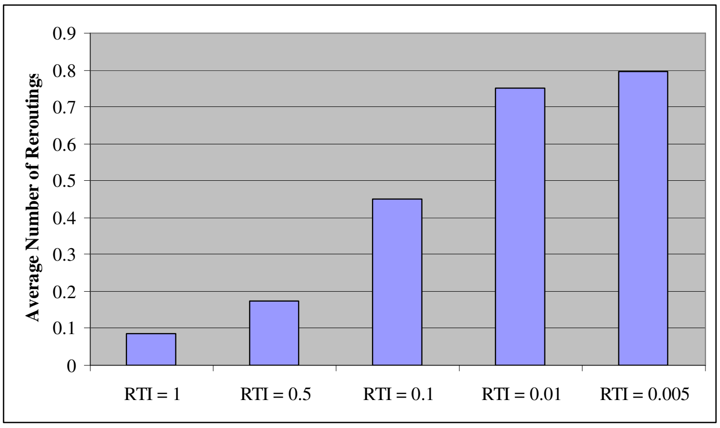
(b) 14-node NSFNET

(c) 25-node mesh-torus

Fig. 4. Network Topologies



(a) Blocking Probability versus Reroute Time Interval



(b) Average Number of Reroutings versus Reroute Time Interval

Fig. 5. NSFNET, no wavelength conversion, W = 40, total traffic load = 220 Erlangs

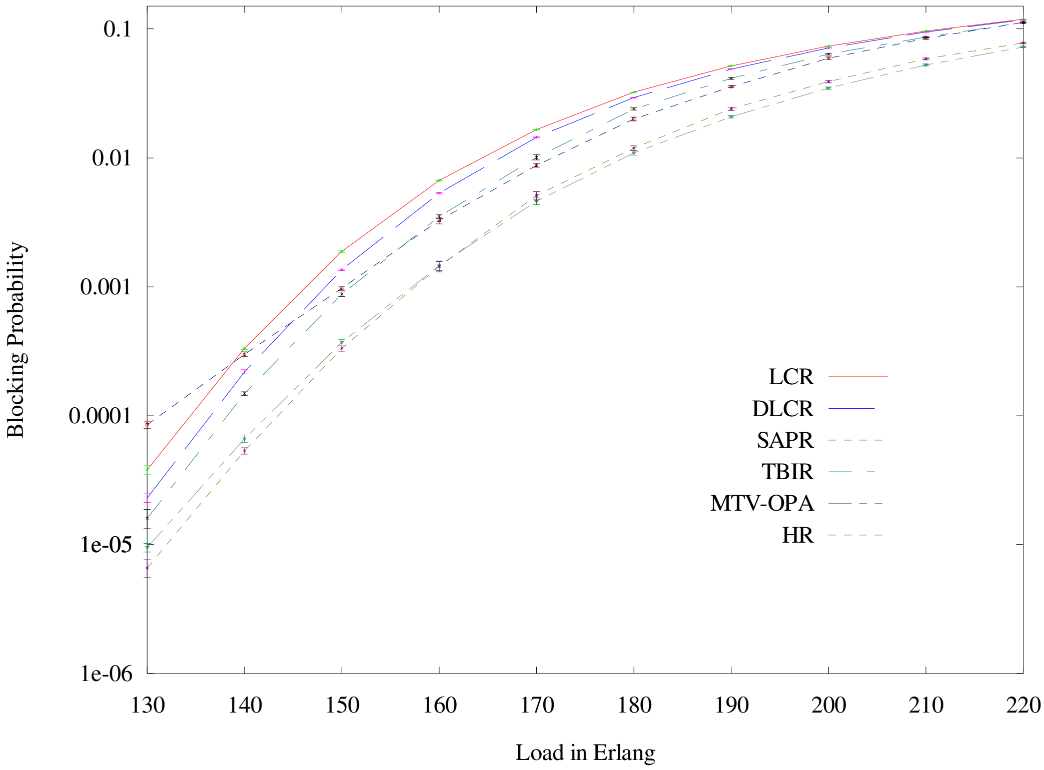


Fig. 6. Blocking performance of different rerouting schemes in ARPA-2, with full wavelength conversion

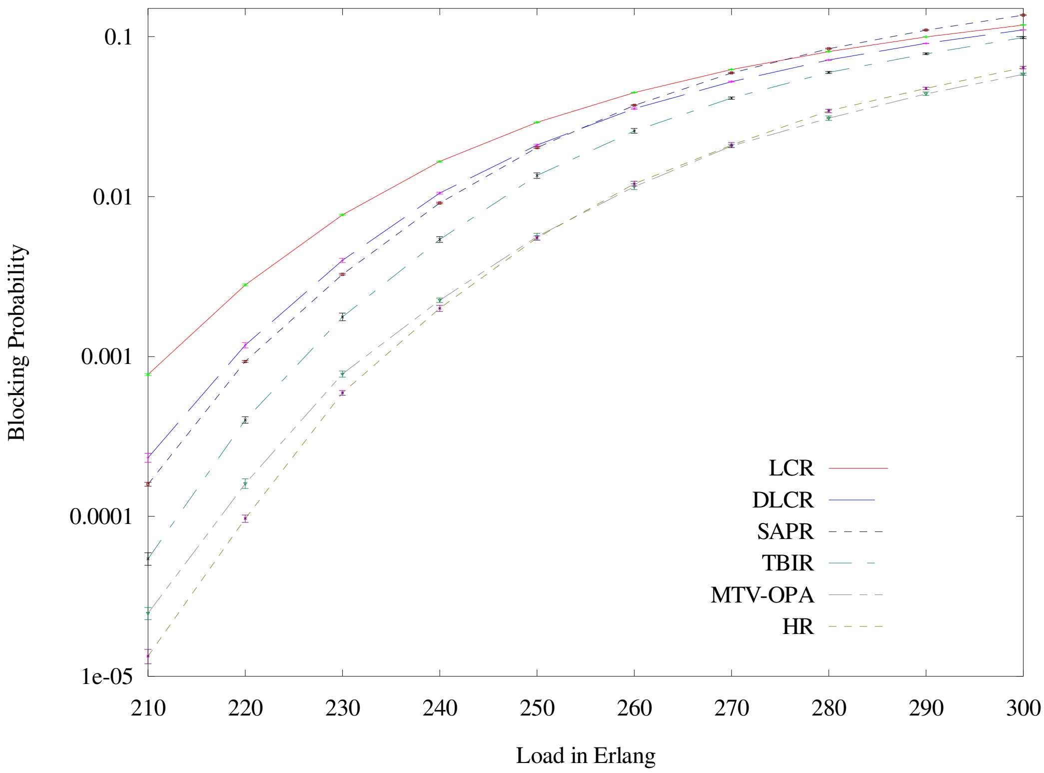


Fig. 7. Blocking performance of different rerouting schemes in NSFNET, with full wavelength conversion

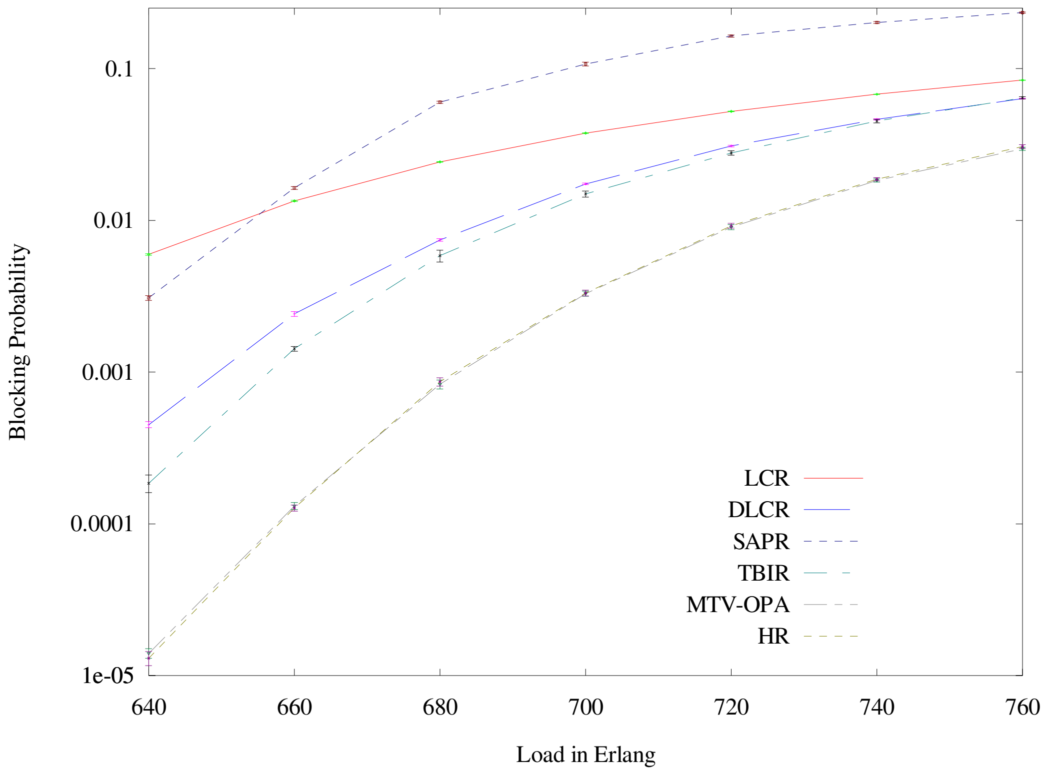


Fig. 8. Blocking performance of different rerouting schemes in Mesh-torus, with full wavelength conversion

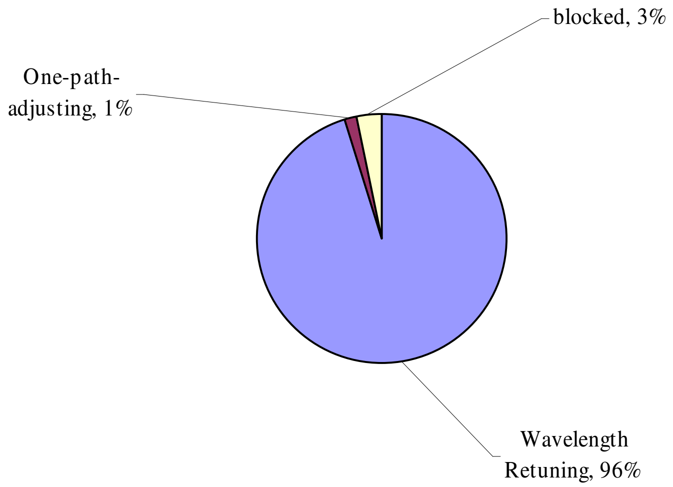


Fig. 9. Effect of MTV-NWR and MTV-OPA, no wavelength conversion

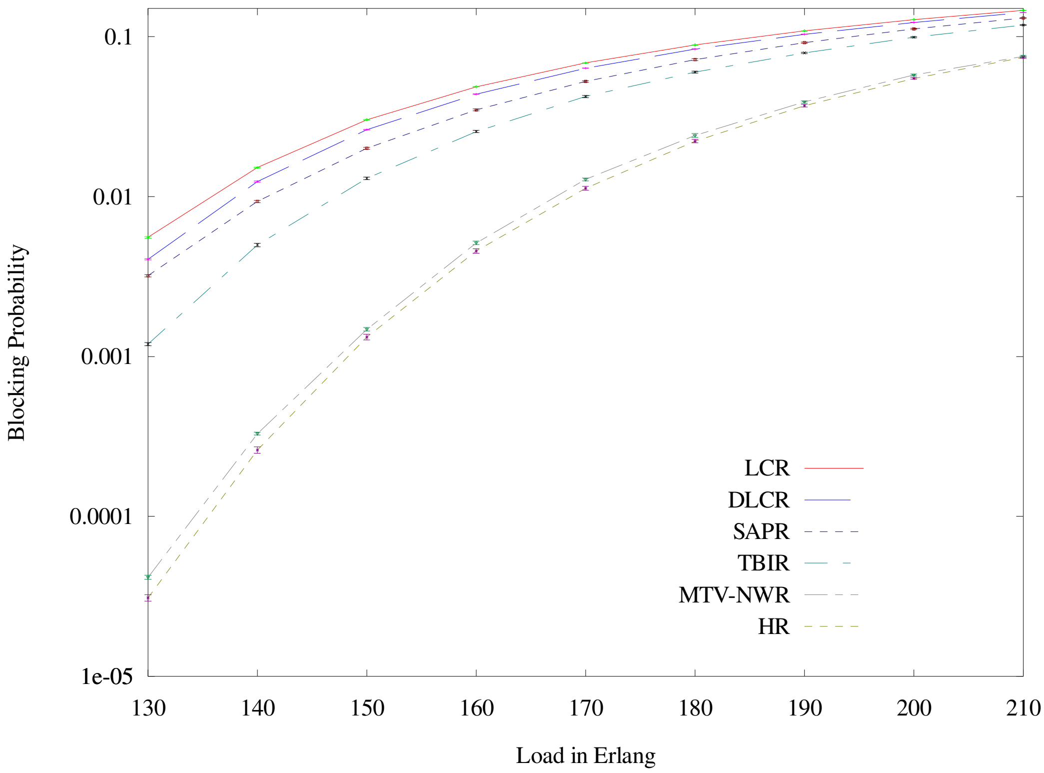


Fig. 10. Blocking performance of different rerouting schemes in ARPA-2, no wavelength conversion

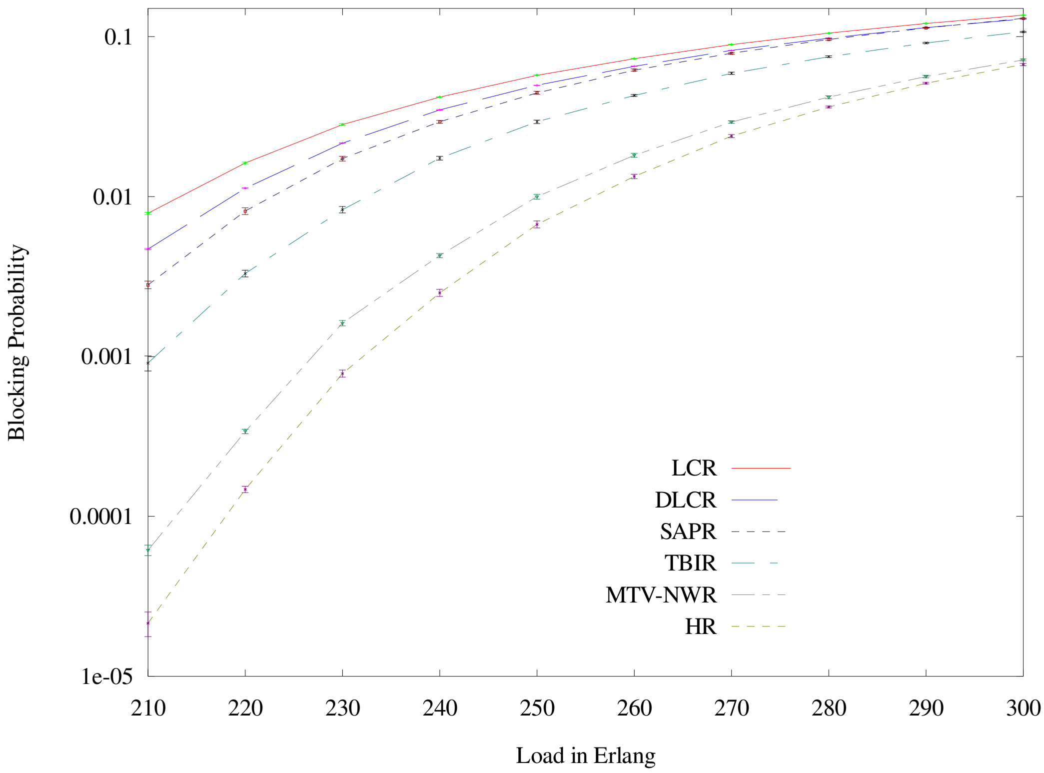


Fig. 11. Blocking performance of different rerouting schemes in NSFNET, no wavelength conversion

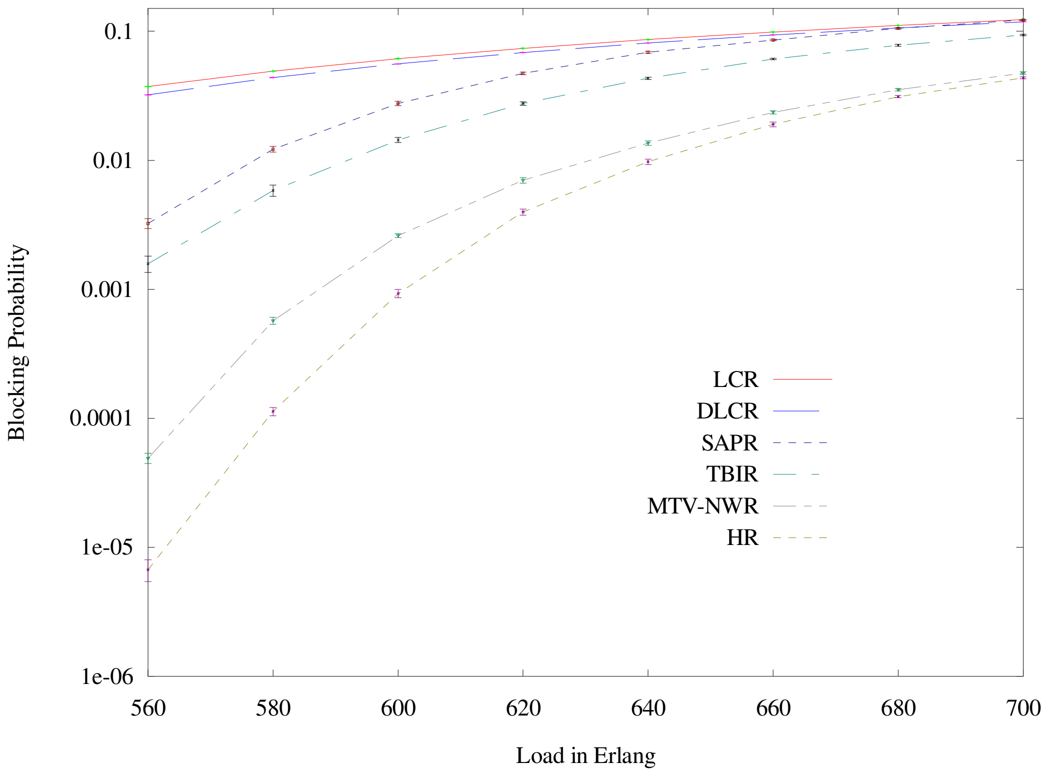


Fig. 12. Blocking performance of different rerouting schemes in Mesh-torus, no wavelength conversion