Constraint Satisfaction in Optical Routing for Passive Wavelength-Routed Networks

Dhritiman Banerjee and Jeremy Frank

Department of Computer Science
University of California, Davis
Davis, CA 95616
{*banerjed,frank*}*@cs.ucdavis.edu*

Abstract. A wavelength-routed, optical network employs all-optical channels *(lightpaths)* on multiple wavelengths to establish a rearrangeable interconnection pattern *(virtual topology)* for transport of data. A lightpath may span multiple fiber-links, and may be routed optically at an intermediate node without undergoing electronic conversion. We examine the problem of establishing a set of lightpaths in an optical network, which employs a passive wavelength routing device called a Latin Router (LR). Latin Routers are attractive for optical network design because of their fault-tolerance and low cost, but make traditional routing algorithms difficult to implement due to the complexity of the constraints they impose on legitimate routes and colors. We employ a local search algorithm to search the space of virtual topologies in order to satisfy a maximum number of given lightpath requests. We use the same algorithm to maximize the number of single-hop connections for a given network. We show that the algorithm can satisfy a high percentage of lightpaths under low to moderate network loads. Experiments reveal that we can establish $O(N)$ lightpaths in an optical network with N nodes. We believe that our work is the first known attempt at designing optical wide-area networks using Latin Routers.

Keywords: wavelength routing, latin routers, local search, optical networks

1 Introduction

A lightpath is used in a wavelength-routed, optical network to establish high-speed, all-optical, channels which can span multiple fiber links without undergoing electronic processing at the intermediate nodes of the network. For example, in Fig. 1, optical lightpath LP4 is established directly connecting nodes 4 and 2 through an all-optical channel. In the absence of wavelength conversion devices at the intermediate nodes of the network, a lightpath will be on the same wavelength on all the fiber-links through which it traverses; the lightpath will be switched optically at the intermediate nodes, e.g., lightpath LP4 is optically switched at node 5 and node 1 before it finally terminates at node 2. A lightpath is typically a unidirectional channel of communication, i.e., a lightpath from node 4 to node 2 does not necessarily mean that there will be a lightpath from node 2 to node 4.

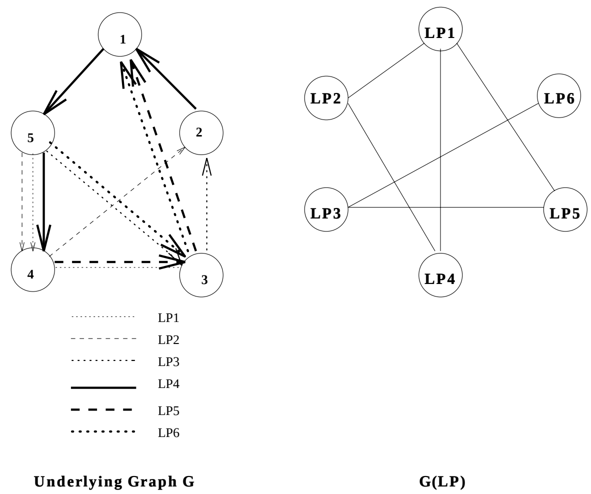

Fig. 1. From G to $G(VC)$.

Using all-optical lightpaths in the network architecture considerably reduces the processing time at intermediate switching nodes, by optically switching forwarded traffic. If two lightpaths traverse one or more common fiber links, the lightpaths must necessarily be operated on different wavelengths. For example, in figure 1 lightpaths LP1 and LP4 traverse a common fiber 4-5, and hence should be on different wavelengths. Typically the number of wavelengths available in the network is fixed at some maximum number, and is limited by the technology used to build the network.

A routing node in this network employs an optical component for wavelength routing of all-optical lightpaths. We use a Latin Router as a passive wavelength router in the optical component of the router, because of its low cost and robustness. A $K \times K$ Latin Router (LR) (shown in Fig. 2) provides complete connectivity between every input and output port, by passively routing K^2 optical connections on K wavelengths. A certain router, called the *Shift Latin Router* (SLR), has a fixed cyclical-permutation-based interconnection pattern between its input and output ports, e.g., wavelength k on input port i is always routed to the same wavelength on output port $(i+k)$ $modulo$ K, $\forall i, k \in [0, 1, \ldots, N-1]$. A particular feature of the Latin Router is that the number of wavelengths supported in the router is equal to the size of the Latin Router.

A common problem in optical network design is: given a physical network topology and a numbers of available wavelengths (hereafter called colors), can we establish a given set of lightpaths? Requested lightpaths are given as source-destination pairs of nodes in the underlying physical topology. Previously pro-

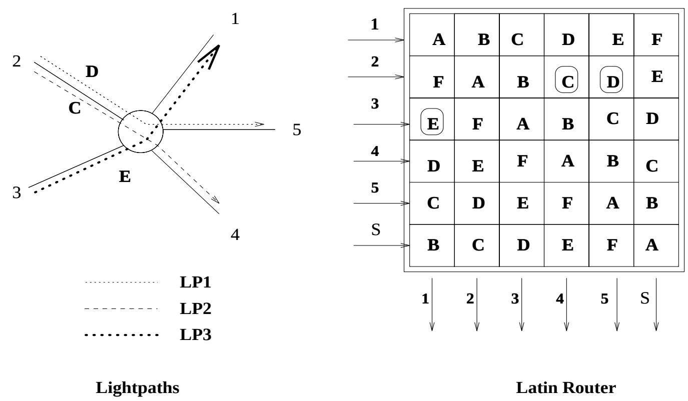

Fig. 2. Latin Square Router.

posed algorithms typically solve this problem in two distinct phases: *routing* and *coloring*. The routing algorithm uses traditional shortest-path-based algorithms to generate a set of routes for each lightpath. Each lightpath is then assigned a color, such that no two lightpaths passing through a common link are assigned the same color. Most existing network designs are based on the Wavelength Routing Switch (WRS), which is a reconfigurable router allowing any wavelength to be switched from any input port to any output port, and hence does not impose any restrictions on the routing algorithm [BM95]. The advantage of this scheme is that routing and coloring are both well-studied problems, and therefore a large number of existing techniques can be employed to solve the problem.

While the above techniques work well in WRS-based networks, we must find different solutions for LR-based networks. In particular, Shift Latin Routers make it difficult to separate the algorithmic process into routing and coloring stages. The constraints that are imposed by a Shift Latin Router on the wavelength assigned to a lightpath needs to be accommodated by the routing algorithm, otherwise unacceptably small numbers of lightpaths will have colorings which obey those constraints. Modifying the routing algorithms to minimize the impact of those constraints is difficult and substantially complicates the routing process (this is explained later in the Appendix). Is is important to develop schemes to handle the constraints that these routers impose on optical routing.

An alternative approach to constraint satisfaction is *local search* which has proven successful at rapidly solving a variety of constraint problems [SLM92, MJPL92]. Local search makes small changes to a complete assignment of variables in a constraint problem in order to improve the quality of the solution. We have devised an algorithm called the Local-search Optical Network Configuration Algorithm (LONCA[1]) which routes static requests for an optical network built using Latin Routers. LONCA addresses two problems, which are as follows.

– Establish the maximum number of a set of requested lightpaths in a network.

[1] A *lonka* is an Indian chili pepper.

– Establish the maximum number of single-hop[2] connections in a network.

LONCA operates by performing local search on the space of *virtual topologies* (i.e. possible interconnections of routers) to find such topologies which satisfy the maximum number of lightpath requests or which establishes the maximum number of single-hop in an optical network.

It is not possible to set up a complete graph as a virtual topology by establishing lightpaths to provide single-hop connectivity between every pair of nodes. In this case, traffic might need to go through multiple lightpaths, while undergoing electronic switching at the endpoints of adjacent lightpaths. Minimizing the average number of optical hops that traffic has to traverse in the network to reach from a source node to a destination node is a related optimization problem. LONCA does not handle this problem directly, but we investigate the virtual topologies maximizing the number of single-hop connections to see how many optical hops are necessary to establish all connections in the network.

In §2 we provide the problem formulation, and in the Appendix we describe the constraints imposed by the physical network topology and the routers used on the lightpath establishment problem. We discuss traditional constraint solving techniques, which separate routing and coloring, in §3. In §4, we discuss LONCA, which is based on the well-known local search paradigm. We present simulation results for this algorithm on randomly generated problem instances in §5. Our results suggest that LONCA can establish $O(N)$ lightpaths in a network on N nodes. This can help in drastically reducing the average hop distance in the network. In §6 we conclude and discuss future work.

2 Routing in Optical Networks

We discuss the problem formulation, and briefly mention the traditional routing algorithms used to satisfy a set of lightpath requests. In the Appendix, we discuss how different parameters in the underlying graph and the chosen router can affect the constraint-satisfaction problem.

2.1 Problem Formulation

The input to the problem is a graph $G =< V, E >$, $|V| = N$, a number of available wavelengths k and a set of i requested lightpaths $C = \{s_i, d_i\}$. The number of wavelengths available restricts the number of lightpaths which can traverse a single link. Lightpaths are directed; in other words, we may desire to have a directed lightpath from s_i to d_j, without a lightpath from d_j to s_i. Traditionally we employ a *routing* algorithm which takes the list $C = \{s_i, d_i\}$ and produces a set of uncolored routes $LP = \{lp_{i1}, lp_{i2}...lp_{ij}\}$ such that $\forall i,\ lp_{i1} = s_i \bigwedge lp_{ij} = d_i$. Two lightpaths lp_a, lp_b can't have the same *color* if they share an edge $(i, j) \in G$. This motivates a graph coloring problem where each connection

[2] A single-hop connection is a lightpath connecting two nodes of the network, such that they can communicate with each other in one optical hop

is isomorphic to a node of an auxiliary graph; two nodes have an edge between them in the auxiliary graph if their corresponding lightpaths share an edge in the underlying graph G. We shall refer to this auxiliary graph as the graph *induced* by LP and denote it as $G(LP)$. Fig. 1 shows an example of constructing $G(LP)$ from G. We observe, for example, that lp_1, lp_2 and lp_4 all use edge $4-5$ in the graph, and so they form a 3-clique in $G(LP)$.

Given N nodes in G, $G(LP)$ has $|LP| < N(N-1)$ nodes. The number of edges in $G(LP)$ is dependent on the routing of the lightpaths. Intuitively, shorter routes for lightpaths result in the fewer edges in $G(LP)$, because fewer lightpaths share the fiber links. The network topology, and the size of the router used, have also been shown to impact lightpath routing [BH95].

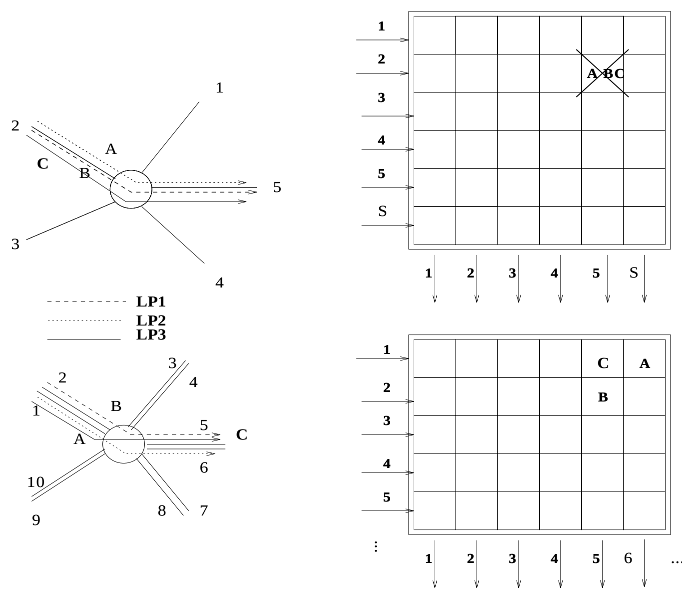

Fig. 3. Lightpaths in a Multigraph.

2.2 Physical Network

The characteristics of the network topology represented by G, and the router used at the network nodes, influences the solution to the problem. It is clear that the more edges G has, the less constrained the graph coloring problem is likely to be, because there will be fewer lightpaths sharing a fiber, thus decreasing the number of edges in $G(LP)$ If the underlying graph is a *multigraph*, i.e., there are multiple fibers connecting adjacent nodes in the physical topology, then more connections can pass between two heavily congested nodes without an increase in the number of colors the fiber must support. In Fig. 3 we see 3 lightpaths passing from edge 2 to edge 5. In a simple graph this would result in a 3-clique in $G(LP)$; however, clever selection of the edges the lightpath uses can result in a less restrictive constraint graph. We discuss the impact of fiber multiplicity on lightpath routing in §3.

2.3 Optical Wavelength Routers

Optical networks can be built using many types of optical routers. The different capabilities in these routers impose different constraints on legal routes and have an impact on the constraint problem instance. A WRS [BM95] can perform arbitrary routing of wavelengths; it allows multiple wavelengths to be optically switched from any input fiber to any output fiber, as long as multiple lightpaths on the same wavelength don't need to be switched onto the same output fiber; however WRSs are costly to build.

The routing pattern in an *Arbitrary Latin Router* (ALR) is based on the Latin Square; for a router of size K the routing pattern consists of a $K \times K$ table. Each fiber attached to a node in G is attached to a single row (input port) and a single column (output port) of the table, and table entry (i, j) corresponds to the wavelength switched from input port i to output port j as shown in figure 2. The routing table has the property that no color appears twice in the same row or in the same column of the table, and only one color occupies each entry of the table. These two properties ensure that there is only a single wavelength which is switched from any input port to any output port, and that lightpaths switched onto an output port are distinct colors. Rows and columns which have no edge attached to them may be used as *access ports*, and are used to terminate lightpaths originating or ending at the node. Access ports are labeled with an S in figure 2. We do not know of any existing prototypes of Arbitrary Latin Routers.

A more restricted form of Latin Routers is called a *Shift Latin Router* (SLR). This router has the additional property that the wavelengths appear in increasing order in the top row of the table, and each subsequent row is the previous row rotated by one table cell; formally, wavelength k on input port i is always routed to the same wavelength on output port $(i+k)$ *modulo* K, $\forall i, k \in [0, 1, \ldots, N-1]$. Such a router appears in Fig. 2. We have used letters to denote wavelengths in order to avoid confusion with the labels for the incoming edges. In this figure, we observe that lp_3 is routed from input edge three to output edge one, and is assigned wavelength E.

Latin Routers impose a variety of different constraints on both routing and coloring. Traditional routing techniques can handle some of the constraints for ALRs [CB95], but SLRs add immense complexity to the problem, requiring intricate routing algorithms and the generation of polynomially many constraints even before coloring begins. This is because there are $O(NK^4)$ coloring constraints (N being the number of nodes, and $K \times K$ being the size of each router) which are imposed by the static routing property of the Latin Router, in addition to the coloring constraints imposed by two lightpaths sharing the same fiber. The reader interested in more details is referred to the Appendix.

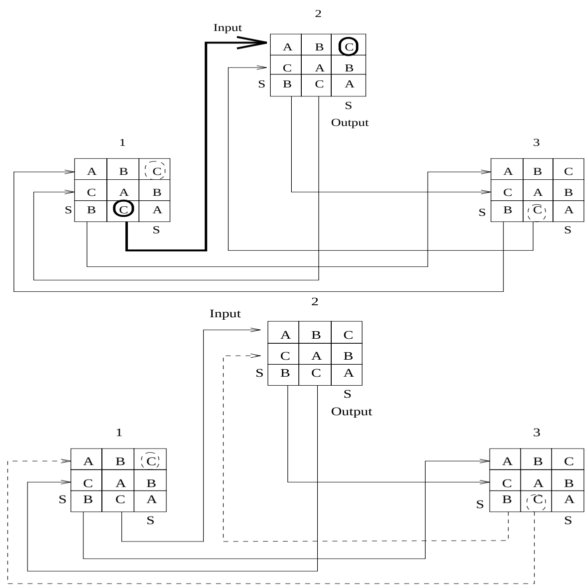

Fig. 4. Interconnection using Latin Squares.

In figure 4 we show an example of a 3 node network composed of 3x3 Latin Routers. Each undirected edge of the graph is represented by 2 directed edges between the routers. As before, an S by the input port (row) denotes the port at which connections originate at the node, while an S at the output port (column) denotes connections terminating at the node. Let us suppose we want a lightpath established between nodes 1 and 2. Input port 3 of node 1 is the designated port where connections originate, and the edge from output port 2 runs from node 1 to node 2. This edge enters port 1 of node 2, and output port 3 of node 2 terminates this lightpath. In order for this to be a valid lightpath, table entry (3,2) of node 1 and table entry (1,3) of node 2 must have the same wavelength; if we look at the circled table entries we see that both entries have wavelength C. So this edge denotes a lightpath. If we want to establish a lightpath from 3 to 1, however, we can't take the edge from 3 to 1 since entry (3,1) of node 3 is B and entry (1,3) of node 1 is C. In fact, we can't establish a lightpath from node 3 to node 1 in this configuration at all.

3 Previous Work

We briefly discuss previous work on algorithms designed to route and color lightpaths on optical networks. In most proposed algorithms, routing and coloring are treated as separate phases of the algorithm [BM95, RS94, ZA94]. We first discuss algorithms for Wavelength Routing Switches, and then for Arbitrary Latin Routers.

3.1 Lightpath establishment for WRS-based Networks

[BM95] present an analysis of routing and coloring techniques for establishing lightpaths using a WRS. Since the WRS imposes no additional constraints on coloring, the routing algorithm should provide a graph coloring problem instance which is as easy to solve as possible. One approach is to find a set of p short routes for each lightpath and pick the route which adds the fewest edges to $G(LP)$. A modification of Dijkstra's algorithm can be used to compute the p-shortest paths for each connection.

Once $G(LP)$ is created, any coloring algorithm can be used to color the resulting graph. On-line coloring algorithms have been used [BM95], although other fast algorithms such as min-conflicts can also be used [MJPL92]. An interesting special case of the problem occurs when the underlying network topology is a single cycle. Clearly the shortest path in such a network is an arc of length less than or equal to N/2. We see that the resulting constraint graph is a *circular arc graph*. While coloring such graphs is still known to be $\mathcal{NP}$-Hard, [Tuc75] gives an upper bound on the number of colors required by such graphs and a multi-commodity flow algorithm which solves the problem; [MIR93] gives on-line algorithms which approximate the number of required colors to within a constant factor.

3.2 Lightpath Establishment for ALR-based Networks

[CB95] present a discussion of routing in networks using Arbitrary Latin Routers. Their approach is to use p-shortest path as described above to find available routes for each connection. The algorithm selects the route which minimally constrains routers, and then assigns a wavelength which minimally constraints those routers. This algorithm was developed for ALRs and does not address the question of routing in networks composed of SLRs. To our knowledge no previous work has devised an optical routing algorithm for networks using SLRs.

4 Satisfying Connections Using Local Search

Traditional constraint satisfaction techniques use routing to generate an easy coloring problem, then satisfy the coloring constraints. There are polynomially many coloring constraints related to the configuration of the SLRs, and it is difficult to write routing algorithms which result in easy coloring problems due to the intricate nature of these constraints.

Local search has been successful in finding satisfying assignments for solvable K-SAT problems [SLM92] and graph coloring problems [MJPL92]. While it does not guarantee a solution, in practice local search algorithms tend to solve constraint problems with solutions very quickly. Local search algorithms examine small changes to a complete assignment of variables in a constraint problem and select one of the changes which improves the number of satisfied constraints the most. This procedure is sometimes referred to as *gradient ascent* or *hill-climbing*.

Since complete search of all virtual topologies appears to be costly, we felt that local search was an attractive alternative.

We decided to use local search to find a virtual topology which would satisfy the most lightpath requests; in effect, we merge the routing and coloring stages into the same algorithm. To do so, we make small changes in the virtual topology by changing the connections between edges in the graph and ports on the SLRs, thereby changing the mapping of colors onto edges. For instance, in figure 4, if we wanted a lightpath between nodes 3 and 1 and the nodes were connected in the fashion indicated, we would be unable to satisfy this request. However, if we were to change the edges assigned to the output ports of node 3 so that the edge from node 3 to node 1 was connected to output port 2, then the wavelength at entry (3,2) of node 3 and the wavelength at entry (1,3) of table 1 (corresponding to the termination of a connection) are both C, indicating that this is now a valid lightpath. To make this change, we would switch the edge running from node 3 to node 2 to output port 1 on node 3. Swapping pairs of input port or output port locations defines a natural neighborhood for local search.

We present the Local-Search Optical Network Configuration Algorithm (LONCA) in figure 5. Making a change to a single router is simply a matter of swapping the position of 2 of the edges attached to either input or output ports of a Latin Router. If the router size is K then there are at most $K^2 - K$ row swaps and column swaps possible for each router in G. These local changes to the network configuration allow us to move through the space of all virtual topologies. At each iteration of the local search algorithm, we select the edge swap which increases the number of satisfied lightpath requests the most; if there are several such changes we choose any one among them at random. For a single iteration, we examine swaps in only one router due to the cost of analyzing the updates and the large number of swaps which must be examined.

```
procedure LONCA(G, VC, RouterSize, MaxSwaps)
  connect all routers in G at random
  for i=1 to MaxSwaps
    pick a router at random
    for each row and column swap
      evaluate the number of connections satisfied
      update set of best swaps
    end for
    pick one of the best swaps and do it
    if all connections satisfied exit
  end for
end
```

Fig. 5. LONCA Algorithm Sketch.

A related problem to that of satisfying requested connections is that of maximizing the number of connections established in the network. In some cases network designers may not have a clear set of requests in mind, and may try to maximize the number of lightpath that can be established in the network, in order to minimize the average hop distance in the network. We use LONCA with a complete graph as the requested virtual topology to solve this problem.

5 Empirical Results

We tested LONCA on physical networks of different sizes and with differing numbers of requested connections. In each case we generated physical networks in the following way: we required each network to be a biconnected graph with average degree varying uniformly between two and seven, hence the number of edges in the networks were $4.5N$. These assumptions are based on characteristics of present-day fiber networks. Lightpath requests were generated by selecting source-destination pairs of nodes chosen uniformly at random without replacement. For these experiments we assume that all routers are of the same size K and that the multiplicity m of each edge in the graph is the same.

5.1 Satisfying Requested Lightpaths

Our first set of experiments was designed to analyze LONCAs ability to satisfying a set of requested connections. We generated 100 sets of connections for each of 5 different physical topologies, each consisting of 50 nodes. We analyzed networks with two different configurations: $m = 1, K = 8$ and $m = 2, K = 15$. We chose the router size such that a node in the graph with maximum degree would be guaranteed at least one access port for termination of lightpaths; hence $K = 7m + 1$. Moreover, Latin Router prototypes of sizes 8 and 15 have been reported in the literature [DEK91]. We ran LONCA once per set of connections with MaxSwaps set to 500 since our initial tests indicated that after 500 swaps LONCA was unable to improve the number of requests satisfied. We generated numbers of connections ranging from $2N - 10N$ and computed the proportion of connections satisfied. We present the results in Fig. 6. Each line of the plot indicates the performance of differing numbers of connections for the same physical network.

We observe that the proportion of established connections is higher for a fiber multiplicity of two than for a multiplicity of one. We observe, as expected, that in both cases the percentage of lightpaths satisfied decreases as the number of requests increases. However, the degradation in number of requests satisfied is relatively slow, with over 70% of 500 requests satisfied for a router size of 15 and a fiber multiplicity of two.

There are likely to be connections which we cannot be established due to violations of the Arbitrary Latin Router constraint. For example, if 10 requested lightpaths originate at a node with a physical degree of seven and a single source port, we can only set up seven of these connections. We were able to analyze those lightpath requests which were not satisfied with respect to the number

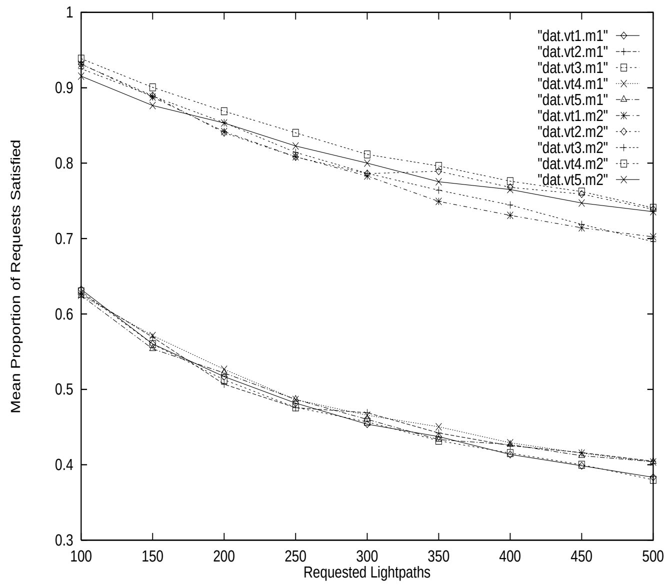

Fig. 6. Mean number of requested connections established.

of lightpath requests for the source and destination nodes. We found that for $m = 1$, only a small fraction of lightpath request failures were due to source or destination port overload, and that for $m = 2$ no lightpath request failures were caused by this problem.

5.2 Maximizing Single-Hop Connections

In our next experiment we ran LONCA on 100 different networks of sizes 50-100 incremented by 10. Our objective was to maximize the number of single-hop connections established in the given networks; in these experiments we asked LONCA to attempt to establish all $N(N-1)$ directed connections possible in the network. We ran LONCA 10 times per network configuration (to average out the randomizing effect of the local search algorithm) again using two network configurations: $m = 1, K = 8$ and $m = 2, K = 15$ with MaxSwaps set at 500. Fig. 7 shows the scaling in the number of connections we were able to establish. We see that LONCA is able to set up about $11.4N$ connections on N nodes when $m = 1$ and $22.8N$ connections when $m = 2$; we conjecture that LONCA can establish about $11.4mN$ connections in a network with multiplicity m, but we need to run more experiments to verify this claim.

We examined the resulting networks to determine how many nodes could be reached in one or two optical hops. We report on the results obtained for 100 node networks; again we tested the case for 100 different topologies with $m = 1, K = 8$ and $m = 2, K = 15$. When $m = 1$ LONCA was able to establish an average of 74% of all $N^2 - N$ connections in one or two optical hops. If m

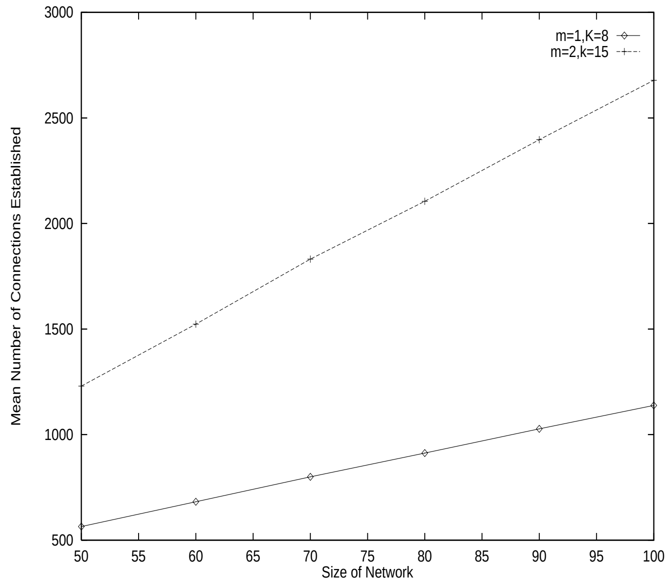

Fig. 7. Mean number of single-hop connections established.

increases to two, LONCA can establish an average of 99% of the connections (Fig. 8).

6 Conclusions and Future Work

We have presented and analyzed LONCA, a local search algorithm designed to perform routing of static lightpaths in all-optical networks. We show that LONCA is able to satisfy a high proportion of lightpath requests effectively in networks with high edge multiplicity, for a large number of requested connections. We also showed that LONCA can establish $O(N)$ lightpaths in a network of size N, and can effectively connect every node in a 100 node network with nearly every other node simultaneously in one or two optical hops, using a router size of 15×15, and fiber multiplicity of two.

m	Mean	Sdev
1	74.07%	1.43%
2	99.28%	0.23%

Fig. 8. Percentage of Connections Established in 2 Hops.

We were disappointed that we could not effectively analyze the reason for LONCA's inability to establish lightpaths. Clearly, understanding the problem in establishing lightpaths will help us create more effective algorithms for establishing lightpaths in the future. We plan to study theoretical upper and lower bounds on the number of lightpaths that can be established in networks of Shift Latin Routers.

There are many variants of local search algorithms which improve performance. Most of these variants force more vigorous exploration of the assignment space by promoting changes to the assignment involving frequently unsatisfied constraints or frequently ignored variables. We hope to investigate the impact of these improvements on LONCA's performance.

We have only tested physical networks of one type in our experiments. We hope to continue experimenting with both more sparse and more dense networks to analyze LONCA's ability to satisfy constraints. We have suggested that when a network is a Hamiltonian Cycle that we may be able to devise better routing algorithms and coloring algorithms. Other special case networks such as regular graphs are also worth examining.

A major drawback to our experiments is the selection of sets of requests to examine. In the analysis section we mention that, even before routing, we can guarantee some connections will not be established due to excessive load at either the source or the destination. We therefore wish to examine a somewhat different problem; given a model of generating requests for lightpaths, how do we build inexpensive networks using SLRs which perform within some specified tolerance?

References

[BH95] R. A. Barry and P. A. Humblet. Models of blocking probability in all optical networks with and without wavelength changers. *Proceedings of IEEE Infocom*, pages 402–411, June 1995.

[BM95] D. Banerjee and B. Mukherjee. Practical approaches for routing and wavelength assignments in large all optical wavelength routed networks. *IEEE Special Issue JSAC/JLT on Optical Networks*, 1995.

[CB95] C. Chien and S. Banerjee. Optical switch configuration and lightpath assignment in wavelength routing multihop lightwave networks. *Proceedings of the 14th IEEE Infocom*, 1995.

[CPS90] O. Chien, H. Plugfelder, and J. Smith, editors. *Quasigroups and Loops: Theory and Applications*. Helderman Verlag, 1990.

[DEK91] C. Dragone, C. A. Edwards, and R. C. Kistler. Integrated optics nxn multiplexer on silicon. *IEEE Photonics Technology Letters*, 3(10):896–898, 1991.

[MIR93] M. Marathe, H. B. Hunt III, and S. S. Ravi. Efficient approximation algorithms for domatic partition and on-line coloring of circular arc graphs. *5th International Symposium on Algorithms and Information*, 1993.

[MJPL92] S. Minton, M. Johnston, A. Phillips, and P. Laird. Minimizing conflicts: A heuristic repair method for constraint satisfaction and scheduling problems. *Artificial Intelligence*, 1992.

[RS94] R. Ramaswami and K. Sivarajan. Optical routing and wavelength assignment in all optical networks. *Proceedings of IEEE Infocom*, pages 970–983, 1994.

[SLM92] B. Selman, H. Levesque, and D. Mitchell. A new method for solving hard satisfiability problems. *Proceedings of the 10th National Conference on Artificial Intelligence*, 1992.

[Tuc75] A. Tucker. Coloring a family of circular arcs. *SIAM Journal of Applied Mathematics*, 1975.

[ZA94] Z. Zhang and A. Acampora. A heuristic wavelength assignment algorithm for multihop wdm networks with wavelength routing and wavelength reuse. *Proceedings of IEEE Infocom*, pages 534–543, 1994.

A Appendix: The Constrainedness of Latin Routers

In this section we discuss the constraints Latin Routers impose upon constraint satisfaction in more detail. One of the features of a Latin Router is that the number of wavelengths supported by Latin Router is equal to the size of the Latin Router. It is not known how to interconnect a network of nodes employing Latin Routers of different sizes (as this would bring a mismatch in the number of wavelengths supported in different parts of the network). Therefore, we assume that the sizes of the Latin Router used is the same for every node in the network. The router size K must be larger than the maximum physical degree in the physical topology, i.e., $K > \Delta$. In addition, we notice that if the degree of a node is much less than K, there may be a large number of free ports in the node. Also, since the number of colors supported is equal to the dimension of the router, we must increase the router dimension if we need more colors to color the constraint graph, subject to limitations imposed by the maximum router size. Finally, since one table entry switches an incoming connection on one edge out to another edge and each table entry only switches one color, we know that a demand to switch two colors from one incoming edge to the same outgoing edge can't be satisfied. This imposes restrictions on the constraint graph which are known at routing time; namely two or more lightpaths can't share two adjacent edges in G. Fig. 3 shows this situation: we cannot route two virtual circuits from edge 2 to edge 5 using a Latin Router. This restriction also has an impact on the number of virtual circuits originating and terminating at a node; if there are s source ports and the degree of the node in the physical network is d then only sd connections can originate or terminate at the node. We refer to this as the bypass restriction in an LR-based network.

A.1 Routing for Arbitrary Latin Routers

We notice that the only additional constraint that an Arbitrary Latin Router imposes on establishing connections is due to bypass restrictions; there are no additional coloring constraints. This implies that if we can manage to route without violating the bypass restrictions, we can utilize existing coloring algorithms to assign wavelengths to the lightpaths.

This problem is alleviated by increasing the size of the router and by increasing the fiber multiplicity of edges in the network; two routes can now be established on different fibers but routing connections to the same node of the network. Fig. 3 shows how increasing the multiplicity alleviates the problem in Latin Routers. If the multiplicity is μ we see that we can route μ^2 connections sharing two adjacent edges; the ease in restrictions extends to routes to and from the source ports as well. With reference to the special case when the underlying network is a single Hamiltonian Cycle we observe that if Latin Routers are to be used, we must increase the multiplicity of edges in the physical network. We observe if the maximum clique in $G(LP)$ has size L then the multiplicity of edges must be $> \sqrt{L}$, resulting in router sizes of at least $\Delta\sqrt{L}$.

A.2 Routing for Shift Latin Routers

The Shift Latin Router is a more restricted form of Latin Routers. We can characterize the form this router must take if we label the available wavelengths as integers from 1 to K. If we examine any four table entries (i,a),(i,b), (j,a) and (j,b) and refer to the color of entry (x,y) as c_{xy} then $c_{ia} - c_{ib} + c_{jb} - c_{ja} \bmod K = 0$. The number of Latin Squares with this property is many times fewer than the number of arbitrary Latin Squares [CPS90]. This restriction turns out to be quite difficult to accommodate in lightpath establishment. Routing and Wavelength Assignment algorithms determine the sequence of edges used by lightpaths and then assign these paths non-conflicting colors. However, in this case we need to make certain that all the color entries obey the above restrictions; in addition to encoding the normal constraints we must also generate and accommodate $O(NK^4)$ additional constraints imposed by the special structure of the Shift Latin Router. To see that this is the correct number of constraints, notice we select 2 entries from each of 2 rows: the total number is $K(K-1)^2$. Since there are N routers we have a total of $O(NK^4)$. The heuristics we designed to account for all of these intricacies were highly complicated and expensive to run. Further, these constraints are highly restrictive on the space of solutions: of the k^4 colorings on 4 variables, a Latin Router constraint on these 4 variables leaves only $k(k-1)^2$ colorings remaining, which is incredibly restrictive compared to the edge constraints on 2 variables, each of which leave $\frac{k^2-k}{k^2}$ colorings.