Abstract of dissertation entitled 'Performance Optimization of WDM Lightwave Networks'
submitted by Chan Kam Wing
for the degree of Master of Information Technology
at The Hong Kong Polytechnic University in April, 1999.

In this dissertation, we consider percentages of improvement of WDM multihop lightwave network for non-uniform traffic. Our goal is to maximize the throughput of a store-and-forward network with infinite buffer in the nodes. The objective is the same as minimizing the largest link flow of the network. To achieve the goal, our work includes two parts. In the first part, we consider optimal node assignment problem (ONAP) for regular topologies. By optimal assigning the network nodes to network locations, we expect to obtain the minimum value of the largest link flow of the network. We solve the problem by formulating it as a quadratic assignment problem (QAP). By applying simulated annealing algorithm, we obtain a significant reduction in the largest link flow of the network. We study the percentage of improvement of four different regular topologies for four non-uniform traffic patterns. The results show that percentage of improvement is largely dependent on the number of routing paths between nodes and also the matching between topological structure and traffic pattern.

In the second part, we study the link assignment problem (OLAP) for arbitrary topology. Through the optimal node assignment problem in the first part, we have conjectured that network throughput can be increased if there is a matching between traffic pattern and topology. We study the conjecture by formulating an optimal link assignment problem (OLAP) which studies the percentage of improvement of the largest link flow of the network between a randomly chosen topology and an optimal topology. By optimal assigning the logical links between nodes, an optimal topology with a minimal largest link flow is found. We solve the problem by formulating it as a Quadratic Assignment Problem. By applying simulated annealing algorithm, we show that the problem is *NP*-Hard. The largest link flow of the network decreases as the network size increases. Results also show that the optimal link assignment approach can have a large reduction to the largest link flow for four non-uniform store-and-forward traffic patterns.

ACKNOWLEDGMENTS

I would like to thank my supervisor, Dr. Rocky Chang, for his patient guidance, continued support and encouragement. I have learnt a lot and it is my pleasure to work with him. I also want to express my thankful to Dr. K. W. Hung for being my co-supervisor.

Contents

LIST OF FIGURES

List of Tables

Chapter 1

INTRODUCTION

1.1 Optical Networks

Since the discovery of optical fiber, high bandwidth and high data rate communication was shown to be the future of telecommunication. The low-loss bandwidth of an optical fiber is about 28 to 30THz for single mode [2]. Moreover there is abundant raw material of the fiber and also the manufacturing cost is low. Along with the rapid growth of Internet and high demand of multimedia applications, optical network should be the mainstream of high bandwidth networking infrastructure. Examples of high bandwidth applications include high-definition television and video-on-demand and videoconference.

Basically, there are two types of optical fiber, single mode and multimode. Multimode fiber propagates signal at various angles. This is not desirable in communication network because optical power is usually distributed somewhat uniformly over the modes, and the overall attenuation will be higher than the single-mode fiber. In addition, since modes propagate at different angles, their group velocities are different, and they are therefore delayed with respect to each other at the fiber output [14]. It causes problem on filtering the modulated light field on each mode. So, in general, single mode fiber is preferred to multimode fiber in optical telecommunication networks due to higher bandwidth, lower intrinsic losses, compatible with integrated optics and lower price [1].

Optical network can also be divided into two types. They are single-channel and multi-channel optical networks. Single-channel optical networks only use up a narrow portion of the low-loss bandwidth for data transmission. Its data rate is limited by the speed of electronic circuits. Multi-channel optical networks allow several light sources having disjoint spectral bands to transmit signals through different wavelength channels in the same fiber simultaneously. Each channel can be operated at a peak electronic speed, say a few Gb/s. As a single fiber can accommodate up to 10^4 electronic-grade channels [3], different wavelengths can use one of the channels for transmission. It is called as Wavelength Division Multiplexing (WDM). Nevertheless, innovative parallelism and concurrency mechanisms should be employed in order to put the huge optical bandwidth into a single fiber [3].

One of the innovative applications of WDM is DataCycle^{TM}. It is a database system developed by Bellcore. A conventional database system cannot serve an unlimited number of users' requests because of limitation of its processing and also the I/O power which is difficult to increase. As most database systems serve only read-oriented transactions, DataCycle can provide very high query throughput by pumping data sequentially and repeatedly to the fibers so that database applications can select the required information from the fiber. It is feasible because the large bandwidth of optical network and low propagation delay.

1.2 Wavelength Division Multiplexed Lightwave Networks

There are several methodologies for accessing the huge optical channels. They include Time Division Multiple Access (TDMA), Code Division Multiple Access (CDMA) and Wavelength Division Multiple Access (WDMA).

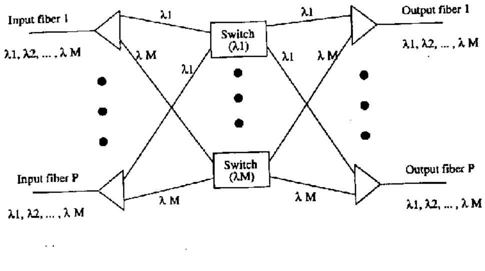
TDMA is a well-known technique for point-to-point digital telecommunication. A number of nodes share a common broadcast channel in a round-robin fashion [4]. There are two types of TDMA systems, synchronous and asynchronous. In synchronous system, each node is assigned an allotted time-slot for data transmission even if it is idle. In asynchronous system, on the other hand, does not waste any time on idle nodes, an active node can continue transmission if other nodes are idle. However, in any case, TDMA requires high-speed synchronization and therefore not suitable for high-speed network communication.

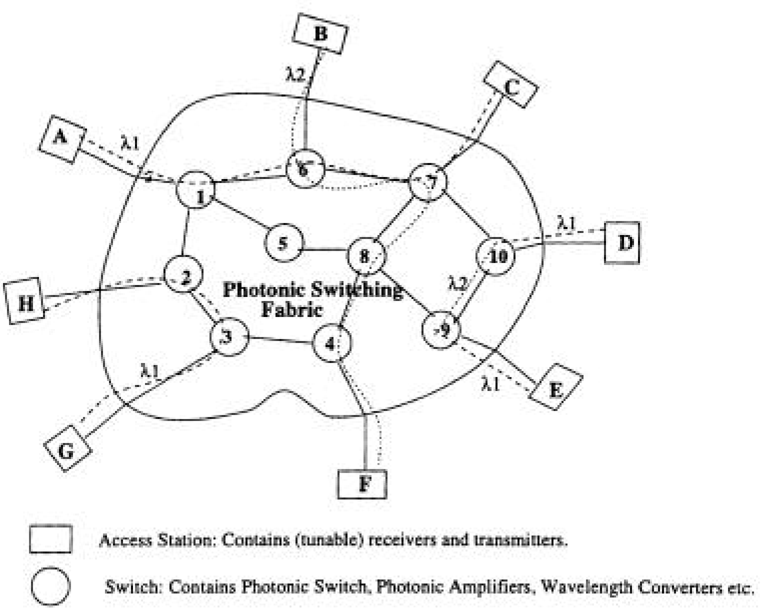
In CDMA, all users are allowed to access the entire bandwidth simultaneously. In order to distinguish individual transmissions, each user is assigned a unique code to encode user's data so that only the receiver that is tuned to the same code can interpret the data. Therefore, CDMA is ideal for security transmission. Besides, CDMA allows theoretically unlimited users to share the entire bandwidth and it is less susceptible from interference from other transmitting nodes. On the other hand, CDMA requires a special power control scheme as the power from closer transmitters will be higher than the power from more distant transmitters. Moreover, if the bit rate of data rate is too high, physical components may not be able to handle a very high data rate [4].

WDMA divides the bandwidth of the low-loss region (1.3 to 1.6 m) of the fiber into a number of concurrent channels [6]. Each channel only allows a wavelength to be transmitted. As every node in an optical network is equipped with a few number of transmitters (laser) and receivers (filters), a node can send data to other nodes if one of its transmitter is tuned to the same wavelength of receiver of another node. In WDM optical networks, there are two types of proposed architecture. They are wavelength routing optical network and broadcast-and-select optical network [4].

(a)

(b)





Data source: [4].

Figure 1.1: (a) A physical structure of wavelength routed optical WDM network that allows wavelength reuse. (b) A P x P reconfigurable router with M wavelengths. The router can switch each wavelength at its input ports independently from its output ports.

1.2.1 Wavelength Routed Optical WDM Network

A wavelength routed optical WDM network is shown in Figure 1.1. Each node is equipped with a few numbers of transmitters and receivers, both of which may be tunable. Ideally, two nodes communicate with each other if there is a lightpath between them. A lightpath is an all-optical communication channel without any intermediate electronics between two nodes in the network, and it may be span more than one fiber link [4]. The end-nodes of the lightpath access the lightpath with transmitters and receivers that are tuned to the wavelength on which the lightpath operates. If there are N nodes in the network, it requires $N(N-1)$ lightpaths to make all nodes communication exclusively. However, in reality, it may not be possible to establish so many lightpaths for a network with medium size. Actually if a message cannot be sent using a single lightpath, it should be switched from one lightpath to another one through intermediate node which should have wavelength conversion device to make optic-electronic and electronic-optic conversion. The most important advantage of wavelength routed optical WDM Network over broadcast network is that the wavelengths of the former can be reused in different parts of the network infrastructure [7]. These properties makes wavelength routed optical WDM network suitable for Wide Area Networks (WANs). However, once a connectivity diagram is built for a given traffic, it is not feasible to change with a given set of wavelength allocations and routes, as opposed to broadcast network.

1.2.2 Broadcast-and-select WDM optical Network

In local or metropolitan WDM optical network, the network nodes are commonly constructed with a broadcast-and-select methodology. A node in a broadcast-and-select network sends message through one of its transmitter to every node directly. The physical topology of the network may be star, bus or tree. Figure 1.2 shows an example of broadcast-and-select WDM network through a star coupler. As message is sent through one of the available wavelength in the network, transmitter and receiver should be in the same wavelength in order to have communication. Therefore, broadcast-and-select WDM networks require some form of tuning at either or both the source and/or the destination. This technique can be implemented using either a tunable transmitter/receiver or multiple receivers/transmitters. The four commonly accepted configuration are fixed transmitters and fixed receivers, tunable transmitters and fixed receivers, fixed transmitters and tunable receivers, and tunable transmitters and tunable receivers [15]. Due to the simplicity of broadcast-and-select WDM network, some of the WDM system have been developed such as IBM's Rainbow, Columbia's TeraNet, and Stanford's STARNET [4].

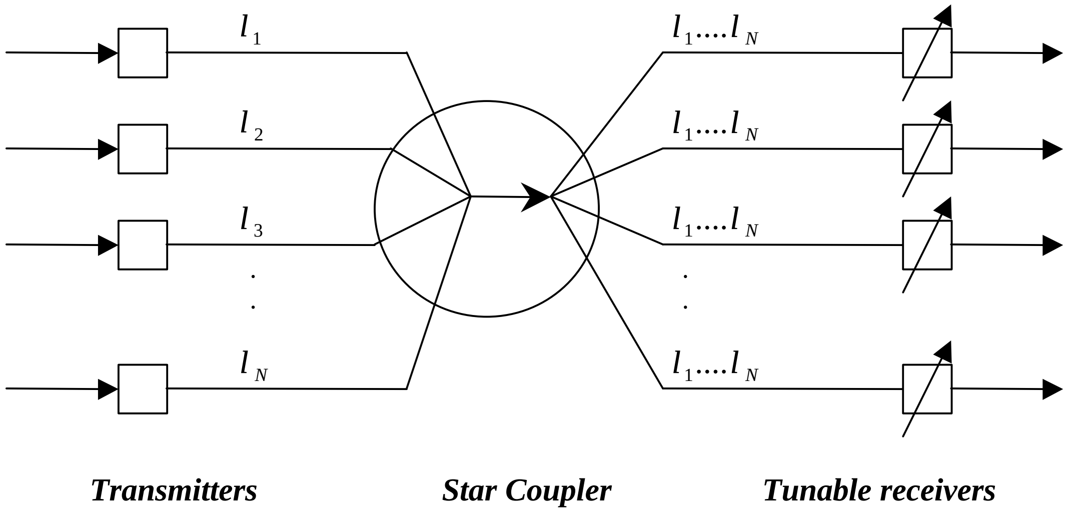


Figure 1.2: A broadcast-and-select WDM network with wavelengths $l_1....l_N$ with a star coupler.

The broadcast-and-select WDM network is recognized to be successful in local area and metropolitan networks but not in wide area networks. This is due to the two problems. The first one is that signals from a node should be broadcast evenly to every node. A very strong transmission power is required if the size of the network is large, say 1000 nodes. The second problem is that in order to have all transmitters transmit signals to receivers, a delicate wavelength should be assigned to each transmitter. It is a serious problem if the network size is large as current technology may only allow hundred of wavelength transmitted in a single fiber. However, the technology required implements a broadcast-and-select WDM network is much simpler than that of a wavelength routed WDM network. It makes broadcast-and-select WDM network become an important choice of local area network.

In fact broadcast-and-select WDM network can be divided into two types. They are single-hop network and multihop network. Single-hop network requires a dynamic coordination between nodes. For a packet transmission to occur, one of the transmitters of the sending node and one of the receivers of the destination node must be tuned to the same wavelength for the duration of the packet's transmission. This allows a direct communication without concern any intermediate nodes. However, it requires a dedicated channel for controlling and scheduling before message is sent. As there is a few number of transmitters and receivers in a node, it is important that transmitters and receivers must be able to tune to different channels quickly so that packets may be sent or received in quick succession. Currently, the tuning time for transceivers is relatively long comparing to packet transmission times, and also the tunable range of these transceivers is small [4]. These deficiencies greatly limit the development of single-hop network.

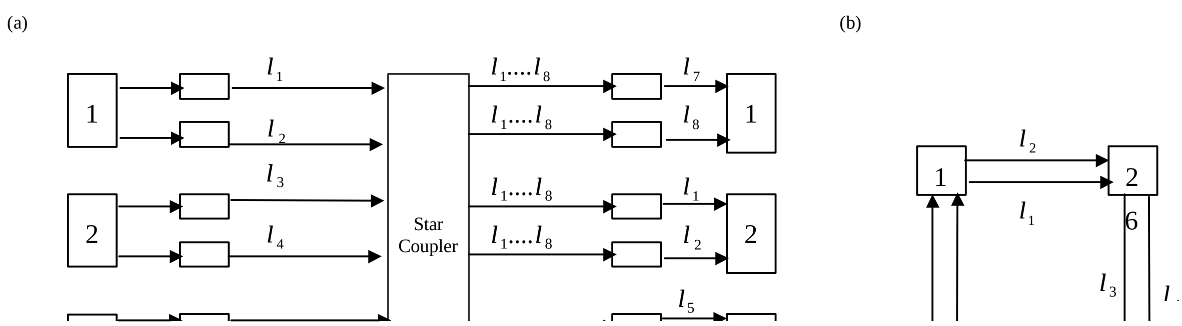
Figure 1.3: (a) The physical topology of a four nodes multihop broadcast-and-select network (b) The logical topology of the network

In a multihop network, each node is assigned one or more channels to its transmitters and receivers that can be tuned slowly. Usually, the assignment is rarely to be changed except for improving network performance. Connectivity between any arbitrary two nodes is achieved by having all nodes acting as intermediate routing nodes. If the communication between two nodes cannot be made by a single lightpath, it should have optic-electronic and electronic-optic conversion through intermediate routing nodes so that a packet sent out on one of the sender's transmit channels finally gets to the destination's receive channels. There are two advantages of multihop network over single-hop network. Firstly, multihop does not require pre-transmission coordination between nodes before transmission. Secondly, the channels being assigned to a node's transmitter and receiver is relatively static and this is an important factor in high-speed communication.

The physical topology of the multihop network may be quite difference to its logical topology. Figure 1.3 shows an example of four nodes connected with a star coupler and there are two transceivers in each node. We see that the logical topology is a ring topology that is totally difference to its physical topology. In fact, the physical topology concerns the physical fiber links between nodes and the star coupler while the logical topology determines how the network nodes optically connect to each other. The logical topology can be either regular or arbitrary. Examples of regular topologies are Ring, ShuffleNet, Manhattan Street Network (MSN) and Hypercube.

1.3 Review on Optimization in WDM Lightwave Network

The ability of separation between physical and logical topologies in multihop network plays a vital role in operation performance improvement. In reality, the transmitters or/and receivers in the multihop network can be tuned to a limited number of wavelengths slowly. By assigning different wavelengths to the transmitters and receivers of every node, different logical topology of the WDM network would be realized accordingly. The change of logical topology, in fact, can be either regular or arbitrary. In the reconfiguration of a regular topology, only the locations of the nodes in the topology are interchanged. For example, if the channels of the transceivers of nodes 3 and 4 of Figure 1.3 are interchanged, the logical structure would be changed as in Figure 1.4 but retaining a Ring topology. Arbitrary topology optimization generally addresses the optimality criterion directly, but the routing complexity is much higher than that of a regular topology because they lack any structural connectivity pattern. A sophisticated algorithm should be employed to find out the routing pattern such as Dijkstra's Algorithm, Bellman-Ford algorithm [13]. An example of arbitrary topology optimization of Figure 1.3 is shown in Figure 1.5 in which the logical links between node 2 and node 3, node 3 and node 4, and node 4 and node 1 are replaced by the logical links node 2 and node 4, node 3 and node 1, and node 4 and node 2 respectively.



l_5

l_7

l_8

Figure 1.4: Reconfiguration of physical topology with the same regular logical topology.

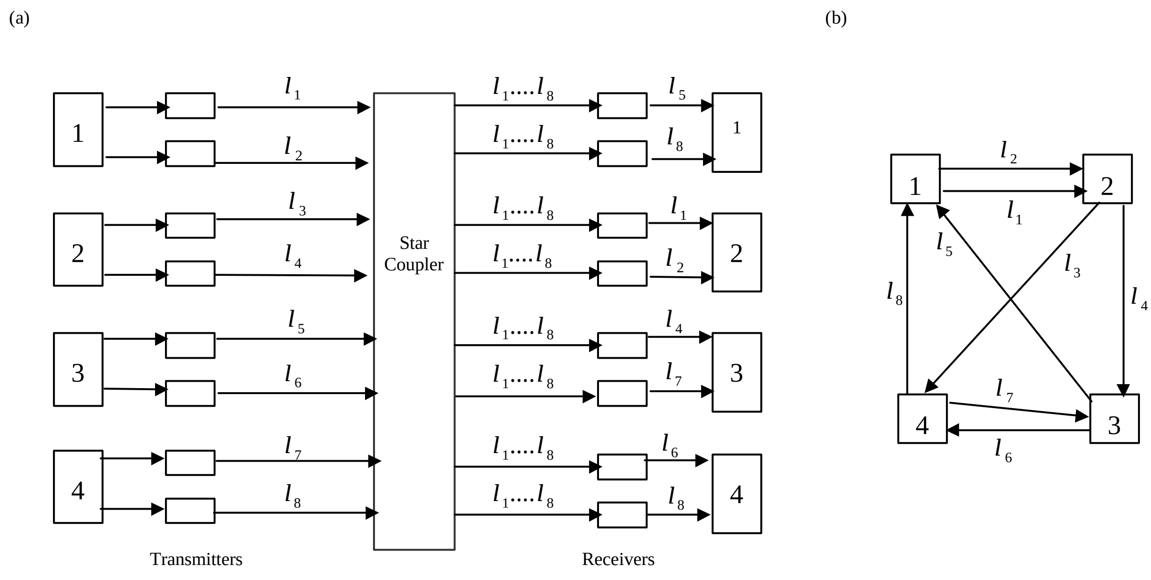


Figure 1.5: Reconfiguration in an arbitrary logical topology

In this dissertation, reconfiguration on both regular and arbitrary topologies will also be considered. For regular topology, we will studied the topological structure of Ring, ShuffleNet, Manhattan Street Network (MSN) and bi-directional MSN (Torus) network. An example of 8 nodes of these topologies is shown in Figure 1.6. The diameter and the average hop distance of the topologies are listed in Table 1.1 [8]. Two nodes are at a hop distance of *h* if the shortest path between them requires *h* hops. The maximum hop distance between any two nodes is referred to as the structure's diameter. In general, a small average hop distance and small diameter are desirable.

Topology	Diameter	Average hop distance ($\overline{h}$)
Ring (N nodes)	$\frac{N-1}{2}$ if N is odd $\frac{N}{2}$ if N is even	$\frac{N-1}{2}$ if N is odd $\frac{N}{2}$ if N is even
ShuffleNet (p,k)	2k – 1	$\frac{kp^k(p-1)(3k-1)-2k(p^k-1)}{2(p-1)(kp^k-1)}$
MSN (m ×n)	$\frac{m+n}{2}+1$ if $\frac{m}{2}$ and $\frac{n}{2}$ are odd $\frac{m+n}{2}$ otherwise	$\frac{\frac{N}{4}(n+m+4)-m-4}{N-1}$ if $\frac{m}{2}$ is odd, $\frac{n}{2}$ is even $\frac{\frac{N}{4}(n+m+4)-4}{N-1}$ if $\frac{m}{2}$ and $\frac{n}{2}$ are even $\frac{\frac{N}{4}(n+m+4)-m-n-2}{N-1}$ if $\frac{m}{2}$ and $\frac{n}{2}$ are odd
Torus (N nodes)	$\sqrt{N}-1$ if N is odd $\sqrt{N}$ if N is even	$\frac{\sqrt{N}}{2}$ if N is odd $\frac{N^{3/2}}{2(N-1)}$ if N is even

Data source: [8]

Table 1.1: Diameter and average hop distance of topologies of Ring, ShuffleNet, MSN, Torus.

The Ring network configuration generally exhibits a great survivability to faults if bi-directional ring architecture is implemented. Basically, a source node routes its packets to a destined node should follow a shortest path which is the path passing through the smallest number of intermediate routing nodes. If the *N*-nodes Ring are labeled sequentially from 0 to *N*-1, the shortest path for a packet from node *i* destined to node *j* would be routed to node (*i*+1) mod *N* if (*N*+*j*-*i*) mod *N* < $\frac{N}{2}$, or routed to node (*N*+*i*-1) mod *N* otherwise [8]. It is noticed that there is only one shortest path between the source node *i* and the destination node *j*, except the node *j*, which has two shortest paths, is the farthest node from node *i* and *N* is even.

A (*p*,*k*) ShuffleNet consists of $N = kp^k$ nodes arranged in *k* columns of p^k nodes each, where *p* is the number of transceivers per node. The nodal connectivity between adjacant columns is a *p*-shuffle with the k^{th} column connected to the first column. A network node is identified by (*c*, *r*) where c is the node's column coordinate (from 0 to *k*-1) and r is the row coordinate using base *p* digits written as $r = r_{k-1}r_{k-2}...r_0 \ (0 \le r_i \le p-1)$. In order to route packet through the path having the smallest number of hops, the number of columns *D* between the source (c^s, r^s) and the destination (c^d, r^d) is:

$$D = \begin{cases} (k + c^d - c^s) \bmod k \\ k \end{cases}$$

if $c^d \quad c^s$
if $c^d = c^s$

The packet will be routed to the intermediate routing node from the source would be

$$\left((c^s + 1) \bmod k, r^s_{k-2} r^s_{d-3} \cdots r^s_0 r^d_{D-1}\right)$$

where r^d_{D-1} and r^s_{k-2} denote the (D-1)th digit of the row address of the destination node and the (k-2)th digit of the row address of the source node respectively. From Figure 1.6, it is noticed that there exists more than one shortest path from a source node to a destination node if the hop number is comparison large. For example, node 1 can send packets to node 7 through either the intermediate nodes 5 and 2, or the intermediate nodes 6 and 4. Comparison with Ring topology, ShuffleNet cannot have arbitrary number of nodes because of its shuffle structure. Its network size can only be expanded by increasing p or/and k in a symmetric (p, k) ShuffleNet. However, a large p involves increasing the number of transceiver in the nodes and increasing the cost. Another suggestion is to expand a (p, k) ShuffleNet into a (p, k+1) ShuffleNet and using intermediate node for routing.

An ($N \times M$) Manhattan Street Network (MSN) is a regular mesh structure of degree 2 with its opposite sides connected to form a torus. Unidirectional communication links connect its nodes into N rows and M columns, with adjacent row links and column links alternating in direction [13]. The N and M should be an even number. Since the MSN is highly modular and the wrap-around links allow the extremes of the network to be connected, each node in the network sees itself lying in the center of the network. As a result the routing algorithm is applicable to every node. Similar to ShuffleNet, every node in an MSN is addressed by (r, c), where r is the row number and c is the column number starting from 0. Given a node address in (r, c), we can easily determine its two outgoing links to the nodes in the MSN network. For example, if the addresses of the next node along the same row is (r, c_{next}) and the next node along the same column is (r_{next}, c), we have

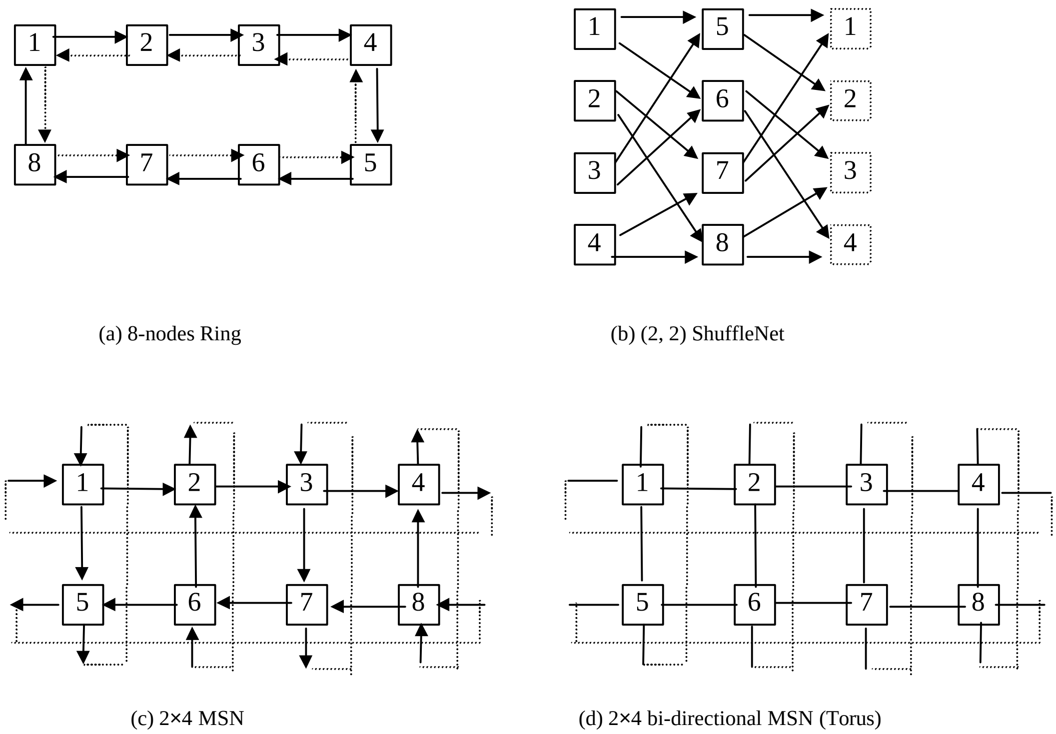
$$r_{\text{next}} = (M + r + d_r) \bmod M$$

where $d_r = 1$ if c is even and $d_r = -1$ if c is odd.

$$c_{next} = (N + c + d_c) \bmod N$$

where $d_c = 1$ if r is even and $d_c = -1$ if r is odd.

Similar to ShuffleNet, MSN is also not flexible to have arbitrary number of node due to its topological structure. However, the limitation is not as severe as ShuffleNet. For example, in the network size between 16 and 64, MSN allows node number to be 16, 24, 32, 36, 40, 48, 56 and 64, while ShuffleNet only allows node number to be 24 and 64.



(a) 8-nodes Ring

(b) (2, 2) ShuffleNet

(c) 2×4 MSN

(d) 2×4 bi-directional MSN (Torus)

Figure 1.6: Regular topologies of Ring, ShuffleNet, MSN and bi-directional MSN

Torus is a bi-directional MSN and there are 4 transceivers in each node. It doubles the nodal degree compared of the other three topologies. So the links shown in its logical diagram are bi-directional. In a ($N \times M$) Torus, each node is addressed by (r, c), where r is the row number and c is the column number starting from 0. Its four links directly connect the nodes with an addresses of ((r + 1) mod N, c), (r, (c + 1) mod M), ((N + r – 1) mod N, c) and (r, (M + c – 1) mod M).

1.4 Organization and Contribution of this Work

In this dissertation, we focus on performance optimization on store-and-forward multihop networks with infinite buffer on regular topologies and arbitrary topologies under non-uniform traffic patterns. Siu and Chang formulated an optimal node assignment problem as a quadratic assignment problem [8]. The objective of the problem was to find an optimal node assignment so that the average hop distance of the multihop networks is a minimum. They employed simulated annealing algorithm to solve the problem and showed that the problem is in fact *NP*-Hard and the percentage of improvement decreases when the network size increases. Our work is an extension of [8]. We first consider the same four regular topologies under the same non-uniform traffic patterns. However, our objective is to find an optimal node assignment so that the throughput of the networks is a maximum. Next we study the reconfiguration problem on network topology under non-uniform traffic in order to increase the network throughput.

In the first problem, we study the node assignment problem for regular topology with non-uniform traffic pattern. For a particular traffic pattern, we consider the way of finding an optimal assignment of nodes to network locations so that the throughput of the network is the maximal or the maximum link flow is the minimal. We formulate the optimal node assignment problem (ONAP) as a quadratic assignment problem and we show that some of the topologies exhibit asymptotic behavior in which the difference between the maximum link flow in a network with an optimal node assignment and the one with a random node assignment decreases as the network size increases. For the topologies not exhibit asymptotic behavior, they still exhibit a significant of percentage of improvement on decreasing maximum link flow. We study the problem with four different topologies and four traffic patterns. The Ring network shows the minimum percentage of improvement and it also exhibits asymptotic property.

Contrasting to the optimal node assignment problem for regular topologies, next we study the topology assignment problem with non-uniform traffic pattern. From the study of Siu and Chang in [8], they showed that a matching between topological structure and traffic pattern increases the network improvement. Our work is to explore the effect of topological structure with non-uniform traffic pattern. We consider the problem of finding an optimal assignment of link to node transceivers such that the throughput of the network is the maximal or maximum link flow is the minimal. We formulate the optimal link assignment problem (OLAP) as a quadratic assignment problem and we show that the problem is an *NP*-Hard problem. We show that the OLAP exhibits asymptotic behavior. The result shows a high improvement in small network size. For large networks, the improvement is only minimal. We also study the problem for four non-uniform traffic patterns. However our interpretation to the problem can be applied to any traffic patterns.

The organization of the rest of the dissertation is as follows. In Chapter 2, we study the optimal node assignment problem. In Chapter 3, we consider the optimal link assignment problem and compare the improvement with ONAP. Finally, we give a conclusion of the dissertation and discuss further research direction.

Chapter 2

THROUGHPUT OPTIMIZATION IN WDM LIGHTWAVE NETWORKS FOR REGULAR TOPOLOGIES

2.1 Introduction

In this chapter, we consider the performance optimization for WDM lightwave networks on regular topologies under non-uniform traffic. From the previous chapter, we know that the physical and logical topologies can be totally difference in a multihop broadcast-and-select network. As the routing through intermediate nodes involve optic-electronic and electronic-optic conversion, the propagation delay in the intermediate node is significance rather than if there has a direct lightpath between the source node and destination node. In a regular topology, the location of nodes in the network is fixed. We can assign the nodes to the locations such that the pair of nodes with a heavy traffic can be connected directly with a lightpath (or single hop). However, the lightpaths from a node to other nodes are limited by the available wavelengths and transceivers in the network, it would be recently a hot topic to find out the best assignment of the network nodes to the locations of a given regular topology under a given traffic pattern.

There are literatures exploiting the optimization on WDM based network. Generally, optimization can be done on Delay-based or Flow-based. The Delay-based optimization targets to find out an optimal virtual topology such that the average network-wide packet delay is the minimum. The packet delay has two components. The first is due to the propagation delays encounter by the packets as it hops from the source through intermediate nodes to the destination. The second is due to queuing in the buffer at the intermediate nodes. In a high-speed network, the channel capacity is quite large and the link utilization are expected in the range from light-to-moderate, the queuing delay component can be ignored comparing to the propagation delay component. Previous study formulate an optimal node assignment problem as a Quadratic Assignment Problem (QAP) to find out an optimal node arrangement in regular topologies such that the average hop distance in the network is the minimal. On the other hand, the queuing delay would be the most significance factor to be considered when the traffic in the network is scaled up to saturation.

In Flow-based optimization, which is the focus of this work, is to maximize the scale factor by which the traffic in the network can be scaled up, i.e. to allow a maximum capacity upgrade for future traffic demands [13]. We focus on maximize the throughput of the network by minimize the largest link flow in the network. Based on the traffic loading in the network, we consider the problem to assign N nodes to N node locations such that the network performance optimized. We refer this problem as Optimal Node Assignment Problem (ONAP). By using tunable transceivers in the network, the logical topology can be changed dynamically. The reconfiguration from the current configuration to a new logical topology should be controlled by some protocols so that the interruption from reconfiguration is minimized. One of the protocols using branch-exchange operations is shown in [17].

There are four regular topologies considered in this work. They are Ring topology, ShuffleNet topology, MSN topology and bi-directional MSN topology (Torus). However, the generalization of our study can be applied to any regular topology. The ONAP for any regular topology is known to be *NP*-Hard problem for shortest path optimization by formulating the ONAP as a Quadratic Assignment Problem [8]. One important property of QAP is the asymptotic property, which states that the difference between the optimal solution and the worst solution decreases as the problem size increases. Some of our experimental results agree with the prediction.

We choose the four topologies because MSN and Shufflenet are the most common topologies that were studied by literatures. Ring topology is flexible to add new node but less flexibility on routing. Bi-directional MSN (Torus) and MSN have the same topological structure but the Torus is twice the number of transceivers (nodal degree) than that of MSN. In this Chapter, we assume that each node is equipped with two transceivers for the topologies of Ring, Shufflenet and MSN, and the effect of topological structure is investigated.

Our problem is to assign the nodes to the locations in multihop networks and computes the percentage of improvement on throughput between random assignment and optimal node assignment. It should be noticed that it does not need any optimization if the traffic from every node is uniform. Therefore, we study four different non-uniform traffic patterns and determine the percentage of improvement to the four regular topologies. The four non-uniform traffic patterns are random traffic, ring traffic, clustered traffic and centralized traffic. Instead of using ONAP on comparing the effectiveness of heuristic algorithms to solve the problem for a specific topology [8,17], we use Simulated Annealing algorithm to find out the percentage of improvement from the effect of the topologies, the traffic patterns and the network size.

The rest of the chapter is organized as follows. We formulate the ONAP as a QAP in Section 2.2. In Section 2.3, we apply the Simulated Annealing Algorithm to solve the QAP with different topologies and traffic matrices. Then we present and discuss our experimental results in Section 2.4.

2.2 Formation of Optimal Node Assignment Problem with QAP

A Quadratic Assignment Problem (QAP) is known to be *NP*-Hard problem [10]. It can be applied to facility location problems, building layout problems, backboard wiring problems, control panel problems and suburban land-use planning problems [9]. Usually, the objective in these problems can be either to minimize time, cost, walking distance or length of wire. In this section, we will show that the ONAP can be solved by formulating a QAP and applied for any regular topology. Our objective is to assign N entities to N mutually exclusive locations to minimize a total quadratic interaction cost [12].

It is well known that the QAP exhibits asymptotic property, which states the difference between the optimal solution and worst solution tends to zero as the size of problem tends to be infinity. In our problem, the objective function of the ONAP is to seek the assignment of *N* network nodes to *N* network locations in order to maximize network throughput by minimize the maximum link flow. We define the following matrices for any arbitrary number of nodes *N*, which are indexed as 1, 2, . . . , *N*, and *N* node locations.

1. Each node has 2 transmitters and 2 receivers.

2. The capacity of each WDM channel is *C* units (say bps). The traffic matrix is given by [$_{sd}$], where $_{sd}$ is the rate of traffic flow, which can be measured in terms of bits/second, generated from source node *s* to destination node *d* for *s*, *d* = 1, 2, . . . , *N*. We further assume $_{ss} = 0$ and is the sum of all $_{sd}$.

3. The flow in link *ij* is denoted by f_{ij} , while the $_{sd}$ traffic flowing through link *ij* is denoted by f_{ij}^{sd} . $f_{ij}^{sd} = 1$ if $_{sd}$ flows through the link *ij* and $f_{ij}^{sd} = 0$ otherwise.

4. Let l_{ij} represents the link matrix of logical connections from network node in location *i* to network node in location *j*. Without loss of generalities, l_{ij} is restricted to take value in {0, 1}, thereby allowing at most one directed link from one station to another. So, $l_{ij} = 1$ if the node in location *i* is logically connected to location *j* and $l_{ij} = 0$ otherwise.

5. Assignment matrix $X = [x_{sk}]$, where $x_{sk} = 1$ if node *s* is assigned to location *k* and $x_{sk} = 0$ otherwise.

The traffic matrix $_{sd}$ and the link matrix l_{ij} are input to the problem directly and the matrix f_{ij}^{sd} is determined from route matrix r_{sd} that is predefined by the Dijkstra's algorithm which is stated in Appendix A. The route matrix indicates the next intermediate node for the packet from source node *s* to destination node *d* for *s*, *d* = 1, 2, . . . , *N* so that the hop distance should be the shortest. In case of there exists two intermediate nodes that can route the packet with the same shortest distance, one of them is chosen randomly. Assignment matrix is a decision variable, which determines the node assignment upon solving the problem. The objective function of the ONAP is then given by (2.1) to (2.4).

$$\min\left(\max_{(i,j)} f_{ij}\right) \quad \text{for } i, j = 1, 2, \ldots, N$$

$$\text{or } \min\left(\max_{(i,j)}\left(\sum_{s=1}^{N}\sum_{d=1}^{N}\sum_{h=1}^{N}\sum_{k=1}^{N} g_{sd} f_{ij}^{sd} l_{ij} x_{sh} x_{dk}\right)\right) \quad \text{for } i, j = 1, 2, \ldots, N \qquad (2.1)$$

subject to $\sum_{s=1}^{N} x_{sh} = 1$ for $h = 1, 2, \dots, N$ (2.2)

$$\sum_{k=1}^{N} x_{ik} = 1 \quad \text{for } i = 1, 2, \dots, N \qquad (2.3)$$

$$x_{ik} \in \{0,1\} \quad \text{for } i, k = 1, 2, \dots, N \qquad (2.4)$$

where $\left(\max_{(i,j)}\left(\sum_{s=1}^{N}\sum_{d=1}^{N}\sum_{h=1}^{N}\sum_{k=1}^{N} g_{sd} f_{ij}^{sd} l_{ij} x_{sh} x_{dk}\right)\right)$ is the throughput of the link with the maximum network throughput among the network. We denote the throughput of the link as *El*. The constraints of (2.2) and (2.3) ensure that exactly one node is assigned to exactly one location. The objective function in (2.1) ensures that the routing algorithm employed (not necessarily a shortest-path algorithm) always follow the same routing path from the source location *s* to destined location *d*, i.e. the same links are passed through for the traffic from the location *s* to location *d*. In order to maximize the network throughput, we should decrease the delay on nodal processing time, the transmission delay, the propagation delay and the queuing delay. The nodal processing delay and transmission delay can usually be considered as constant. The propagation delay is important when the traffic is low. Queuing delay becomes the dominant factor to be considered to scale-up the throughput in the high-speed networks in order to avoid saturation in any link. Thus our objective function optimizes the network throughput performance by minimizing the largest flow over the links in the network [6]. Other possible objective functions, on the other hand, only consider the propagation delay and the end-to-end packet delay, neglecting the queuing delay component of the total delay [6, 7].

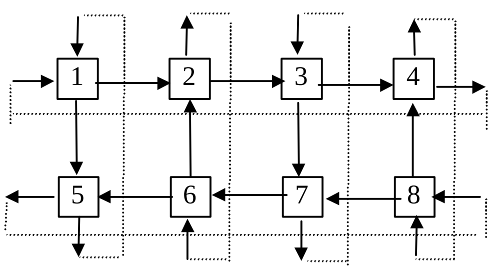
It should be pointed out that the QAP formulation for the ONAP is very general and does not imply the locations of the nodes. The QAP formulation should have the asymptotic property if the following criteria are fulfilled:

1. The entries in the traffic matrix are mutually independent.

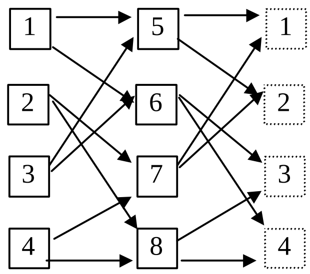
2. The entries in the route matrix are mutually independent.

3. The entries in the traffic matrix and that in the route matrix are mutually independent.

We assume the above criteria are satisfied and applied to any regular and even arbitrary topologies. As the topology information is effectively embedded in the route matrix by specifying that if the traffic between a pair of nodes passes through a link of two nodes. Basically, we employ a shortest–path routing algorithm, i.e. the hop distance between the source node *s* and the destination node *d* should be the minimum. It is very clear to see that the shortest-path routing can have different routing path between the node *s* and node *d* provided that the different routing paths have the same shortest hop distance. In this case, we randomly select the routing path. Once the routing path is chosen, the routing path would be the same for the node *s* and node *d* throughout the simulation process. In Figure 2.1, we show the topologies of an 2 ×4 MSN and an (2,2) ShuffleNet, and their route matrices respectively.



(a) 2×4 MSN



(b) (2, 2) ShuffleNet

To location

location	1	2	3	4	5	6	7	8
1	1	1	2	2	1	5	2	5
2	3	2	2	3	6	2	3	6
3	4	4	3	3	7	7	3	4
4	4	1	8	4	1	8	8	4
5	5	1	8	8	5	1	8	5
6	5	6	2	2	6	6	2	5
7	3	6	7	3	6	7	7	3
8	4	7	7	8	7	7	8	8

Intermediate node to be routed

(a) Route matrix of the 2×4 MSN

To location

From location	1	2	3	4	5	6	7	8
1	1	5	6	6	1	1	5	6
2	7	2	8	8	7	7	2	2
3	5	5	3	6	3	3	6	5
4	7	7	8	4	7	7	4	4
5	5	5	1	2	5	1	2	2
6	3	4	6	6	3	6	4	4
7	7	7	2	1	1	1	7	2
8	3	4	8	8	3	3	4	8

Intermediate node to be routed

(b) Route matrix of the (2, 2) ShuffleNet

Figure 2.1: Topologies of 2×4 MSN and (2, 2) ShuffleNet and their route matrices.

2.3 Simulated Annealing Algorithm

In this section, we mainly propose a solution for the above ONAP. Many theoretical and practical combinatorial optimization problems belong to the class of *NP*-complete problem [4]. The number of possible combinations grows exponentially with the problem size. It cannot be solved simply with a polynomial solution. Some literatures suggest solving such problem with an improvement method [7, 12].

Many heuristic algorithms have been proposed to solve QAP. Most of the heuristic algorithms for the QAP lies into the category of improvement methods and iterative algorithms that are most commonly used. However, the result from the iterative algorithm may not give better result because solution from the algorithm may fall into local optimum. Other algorithms such as limited enumeration method and simulation method may express a better result but their computational complexities are much higher. In this paper, we will employ simulated annealing algorithm to generate optimal node assignments for regular topologies. The algorithm is described in Figure 2.2. The simulated annealing algorithm has been found to overcome the problem of local optimality occurred in the iterative algorithms and to provide good solutions for complex optimization problems.

The simulated annealing algorithm is based on a model of Boltzmann machine. In the simulated annealing process, the algorithm starts with an initial random configuration for the virtual topology. Node-exchange operations are then used to arrive at neighboring configurations. In a node-exchange operation, adjacent nodes in the virtual topology are swapped and examined for the change of cost (lower maximum link flow). The swap of nodes gives better results than the current solution is accepted automatically. Solutions, which are worse than the current one, are accepted with a certain probability which is determined by a system control parameter. The probability decreases as the algorithm progresses in time so as to simulate the “cooling” process associated with annealing.

In order to simulate the annealing process, our program consists of two loops of iterations. The outer loop is executed with an initial “temperature”. The inner loop is used to compute cost improvement by swapping the locations of two nodes. We let *n(i)* be the location of node *i* for *i* = 1, 2, …, *N* and $_{ij}$ be the reduction in *El* if *n(i)* and *n(j)* are swapped. Similar to the iterative algorithm, their locations are swapped if $_{ij} > 0$ as there is a cost improvement after nodes swapping. However, the simulated annealing approach also allow location swap for two nodes even the cost of swapping of the nodes is not improved in order to avoid the algorithm from trapping into a local optimal point. The subsequent iteration in the outer loop starts with a reduced temperature from a cooling factor to the previous temperature. This cooling process continues until a certain stopping condition is met when the cost cannot be further reduced in a successive number of attempts.

Here, we adopt several guidelines established in the earlier studies of the simulated annealing algorithms to determine the initial temperature, cooling rates, and set of scheduled temperature [7]. We set the *MaxAttempt* to 10×*N* and the *MaxMove* to *N*, and the cooling rate to 0.8. The initial temperature *T* is computed based on the following formula : $T = \frac{-\overline{\Delta_+}}{\ln c}$, where is the desired probability that nodes swap will be accepted for an initial solution and $\overline{\Delta_+}$ is the average change of $_{ij}$ for those node swaps with $_{ij} > 0$ for the initial solution. By setting the value of to 0.6, we compute $\overline{\Delta_+}$ by taking randomly a number of neighbors of the initial solution where $_{ij} > 0$ and computing the average change of cost.

1. Generate a random node assignment.
2. Initialize the temperature *T*.
3. While (no_of_attempts < max_attempt) do

    While (no_of_moves $\leq$ max_move) do

        Randomly select two neighboring nodes *i* and *j*

        Compute $_{ij}$ if *n(i)* and *n(j)* are swapped.

        If $_{ij}$ < 0, swap the node locations; no_of_attempts = 0;

            Increment no_of_moves

        If $_{ij} \geq 0$, swap the node locations and increment no_of_moves

            with probability $e^{-\Delta_{ij}/T}$; increment no_of_attempts

    $T = T \times a$

Figure 2.2: Simulated annealing algorithm for the ONAP [8]

2.4 Numerical Results and Discussion

2.4.1 Numerical Results

In this section, we would like to study the percentage of improvement in the network throughput (or minimizing the maximum flow on any link) by applying the simulated annealing algorithm to the regular topologies of Ring, ShuffleNet, Manhattan Street Network (MSN) and bi-directional MSN (Torus). The nodes in the topologies of Ring, ShuffleNet and MSN possess two transceivers (nodal degree) while there are four transceivers for the nodes of Torus. The Ring, MSN and ShuffleNet are selected to compare the difference in the percentage of improvement when the structure of the topology is totally difference but with the same nodal degree. By applying different traffic patterns to each topology, we can investigate the effect to the percentage of improvement on the throughput from the topological structure, traffic patterns, nodal degree and the network sizes.

There are four types of traffic pattern under our investigation. They are random traffic, ring traffic, clustered traffic and centralized traffic patterns. Figure 2.3 shows examples the matrix of the traffic patterns for network size equal to 8. Each entry in the random traffic matrix $_{sd}$, except $s = d$, is generated independent to other entries from the random numbers ranging from 1 to 20. Meanwhile, the entries in the other three traffic matrices take the number either from a high traffic intensity or a low traffic intensity. The entries belonging to the high traffic intensity is generated from a random numbers between 12 and 20 and each such entry is also generated independent of other entries, whereas the entries belonging to the low traffic intensity is generated from a random numbers between 1 and 7. In the ring traffic pattern, the entries $g_{s(s+1) \bmod N}$ are generated from the high traffic intensity and the rest are come from the low traffic intensity. The entries in the clustered traffic can be divided into two parts that are intra-cluster and inter-cluster. The entries belonging to the intra-cluster are generated from the high traffic intensity and the ones belonging to the inter-cluster are generated from the low traffic intensity. In the centralized traffic matrices, the traffic from and to a particular node, e.g. node of server, is heavy and so its entries from the higher traffic intensity but the rest from the low traffic intensity. The particular node is chosen if the node number is equal to (N+1) / 2 when the N is odd and the node number is equal to (N / 2 + 1) when the node number is even.

There are twenty-five samples are generated for each type of traffic patterns so that the fluctuation of percentage of improvement can be evened out. In order to investigate the effect of topological structure to the percentage of improvement, we use the same seed to generate the same set of random assignment for each regular topology. The network size being studied is range from 16 nodes to 160 nodes. Due to the restriction on the structure of a (p,k) ShuffleNet, the data being collected from ShuffleNet is fewer than other topological structures. In our simulation experiments, the value of p is fixed to 2, i.e. the transceiver is 2, and the value of k starting from 3 (network size = 24) to 5 (network size = 160). We apply the simulated annealing algorithm on each random node assignment and applying the objective function of (2.1). To compute the resulted percentage of improvement (PI), we denote $E(l)_{ra}$ and $E(l)_{oa}$ as the largest link flow from a random node assignment and from an optimal node assignment respectively. The PI for the ONAP is compared to that of the random node assignment and given by

$$PI = \frac{E(l)_{ra} - E(l)_{oa}}{E(l)_{ra}} \times 100\% \qquad (2.5)$$

From node \ To node	1	2	3	4	5	6	7	8
1	0	9	17	4	2	18	20	17
2	13	0	4	11	6	3	13	11
3	4	14	0	16	10	14	8	18
4	15	16	8	0	9	6	15	2
5	2	19	4	9	0	13	1	19
6	2	16	11	10	14	0	18	18
7	18	4	14	16	12	3	0	16
8	11	16	12	10	9	6	18	0

(a) Random Traffic Pattern

From node \ To node	1	2	3	4	5	6	7	8
1	0	18	4	4	6	3	4	3
2	6	0	19	4	1	2	2	4
3	3	6	0	18	1	7	2	5
4	4	2	7	0	16	1	1	3
5	5	4	3	3	0	20	3	4
6	5	5	7	4	2	0	18	4
7	1	3	3	3	2	7	0	13
8	17	2	2	7	2	3	6	0

(b) Ring Traffic Pattern

From \ To node	1	2	3	4	5	6	7	8
1	0	17	18	20	4	4	6	3

From \ To node	1	2	3	4	5	6	7	8
1	0	2	4	4	13	4	3	8

node								
2	13	0	16	14	8	8	6	2
3	19	16	0	19	2	2	4	3
4	12	16	16	0	6	2	6	1
5	7	2	5	4	0	12	14	16
6	1	1	3	5	20	0	12	13
7	8	6	7	5	20	14	0	20
8	3	8	4	5	17	19	15	0

(c) Clustered Traffic Pattern

node								
2	8	0	2	2	19	4	1	2
3	2	4	0	6	18	2	6	1
4	7	2	5	0	12	2	7	1
5	16	18	13	13	0	14	14	16
6	1	1	3	5	20	0	3	3
7	8	6	7	5	20	3	0	4
8	5	5	7	4	19	2	8	0

(d) Centralized Traffic Pattern

Figure 2.3: Example of traffic matrices of Random, Ring, Clustered and Centralized for 8-nodes network

We calculate the cost reduction by averaging the *PI* from the twenty-five samples and then find out the standard deviation and also the confidence interval of the samples. A probability of 0.95 is adopted to calculate the upper and lower limits of the samples. The graphical results of the samples are shown in Figures 2.4 to 2.11. The interpretation for the results will be given in the next section.

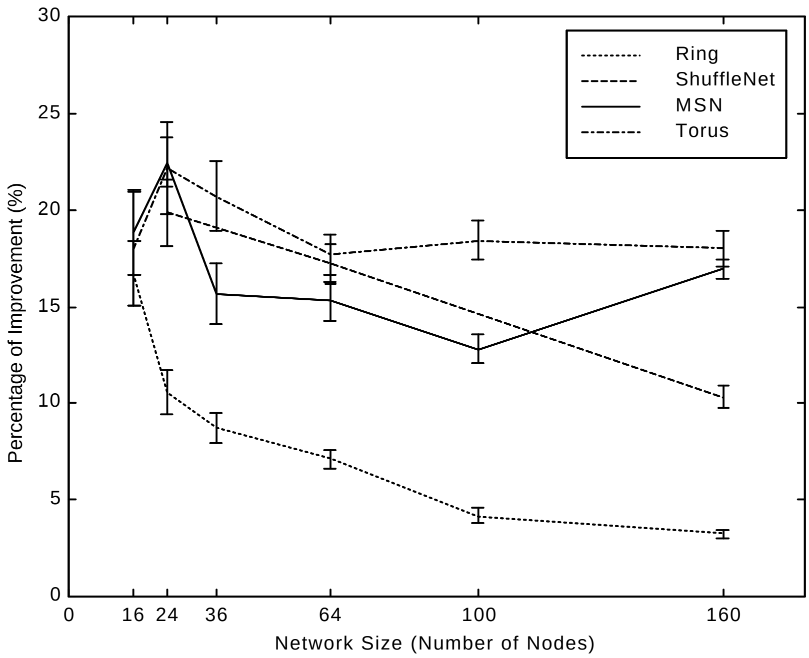


Figure 2.4: Performance of the simulated annealing algorithm on different topologies with random traffic pattern..

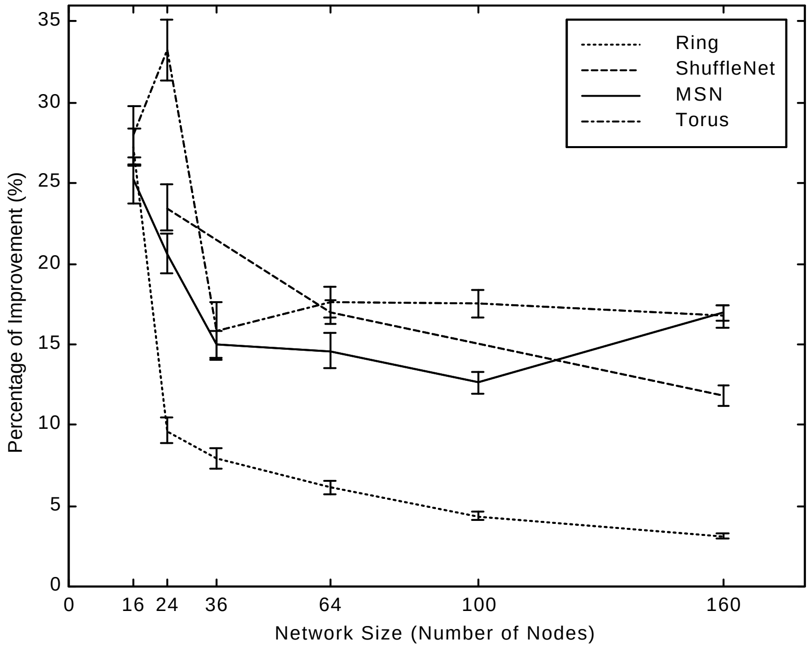


Figure 2.5: Performance of the simulated annealing algorithm on different topologies with ring traffic pattern.

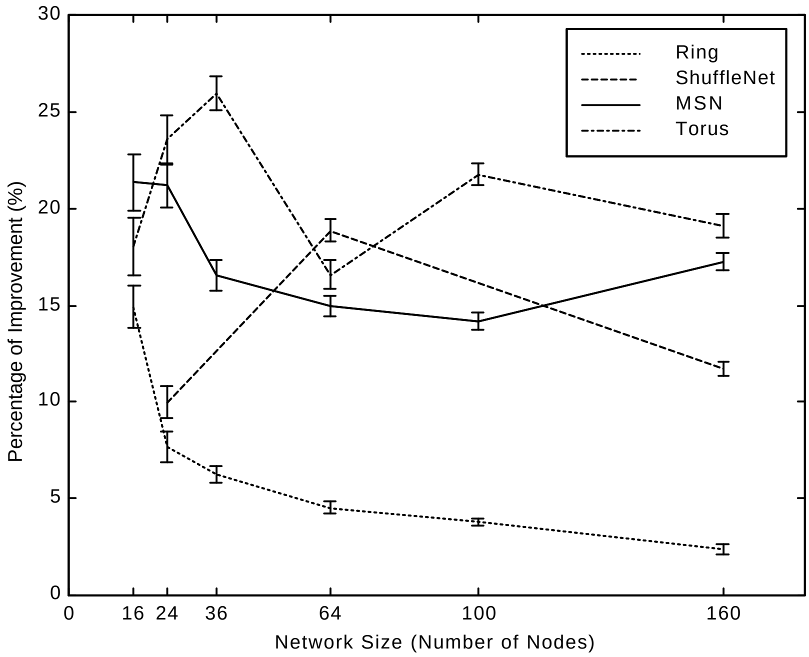


Figure 2.6: Performance of the simulated annealing algorithm on different topologies with clustered traffic pattern.

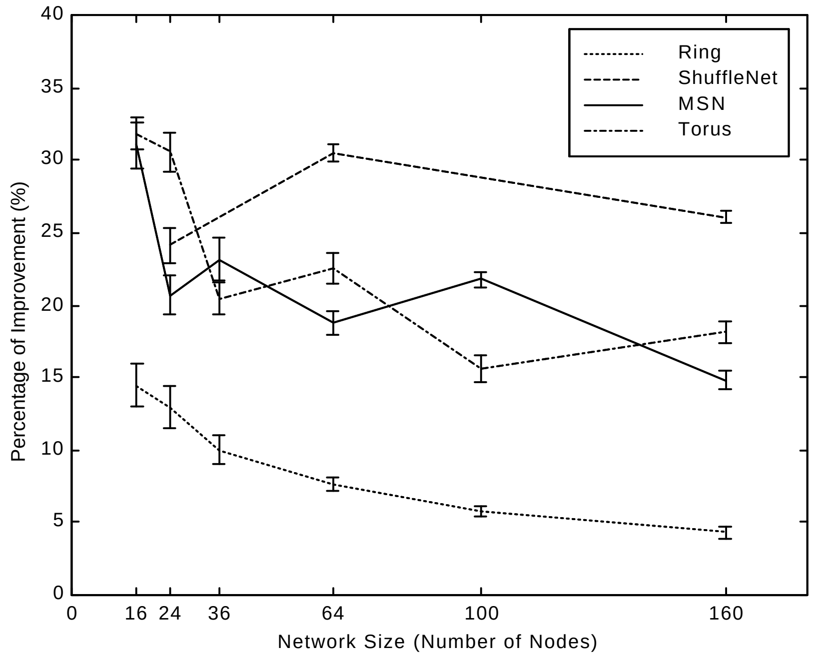


Figure 2.7: Performance of the simulated annealing algorithm on different topologies with centralized traffic pattern

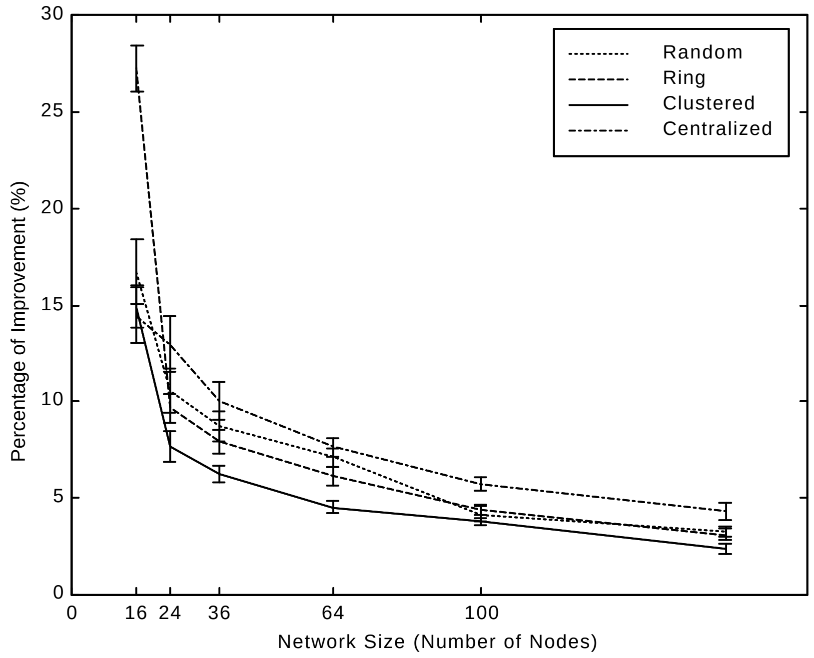


Figure 2.8: Performance of the simulated annealing algorithm on different traffic patterns with Ring topology.

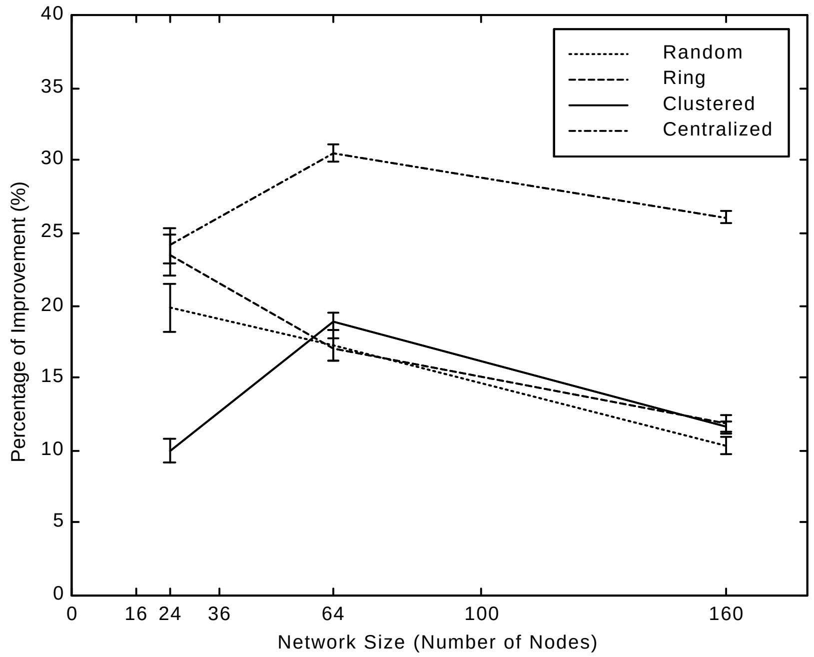


Figure 2.9: Performance of the simulated annealing algorithm on different traffic patterns with ShuffleNet topology.

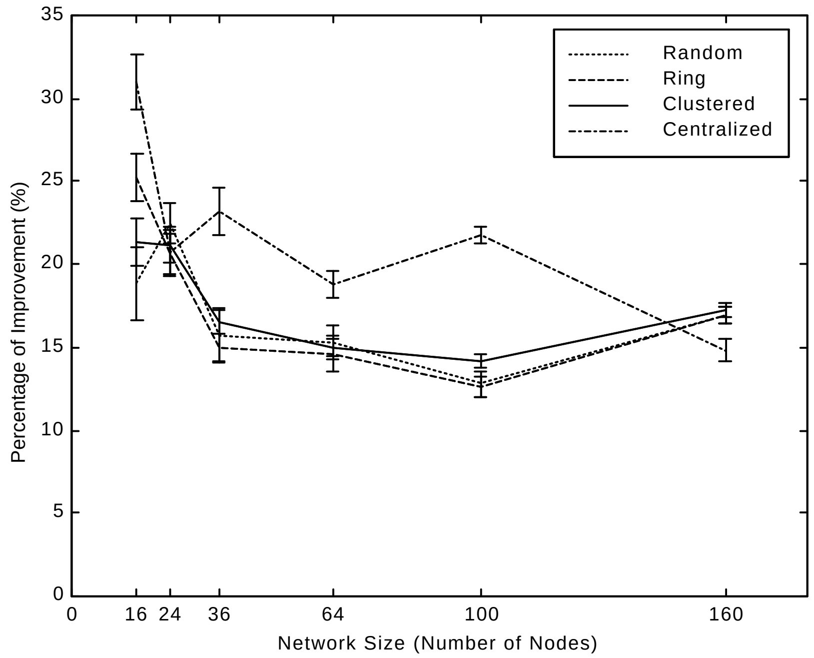


Figure 2.10: Performance of the simulated annealing algorithm on different traffic patterns with Manhattan Street Network (MSN) topology.

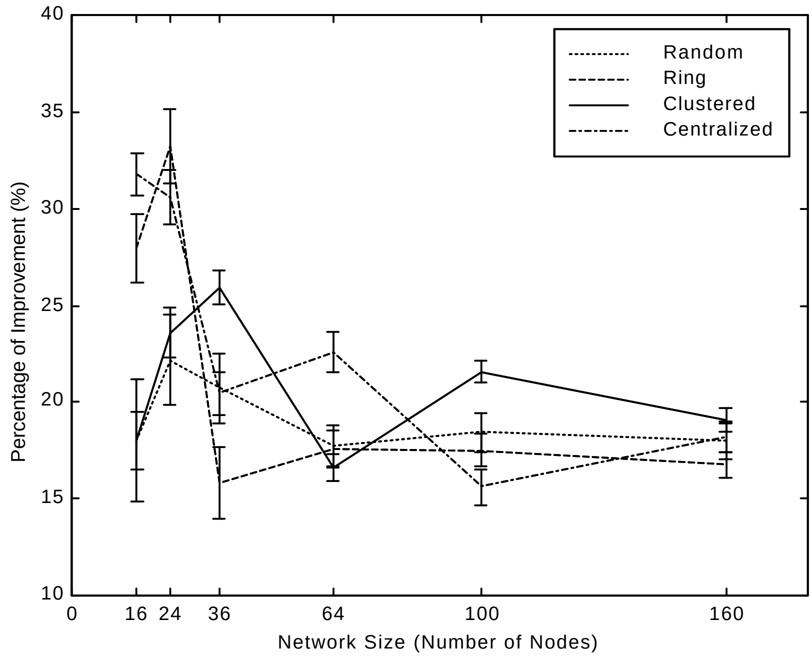


Figure 2.11: Performance of the simulated annealing algorithm on different traffic patterns with Torus topology.

From the initial interpretation of the experimental results, the followings are observed:

1. (Topology) Ring topology has the lowest *PI* among the four topologies in each traffic pattern.
2. (Traffic Pattern) The *PI* obtained for the centralized traffic pattern is generally higher than for other three traffic patterns for all four topologies.
3. (Number of transceivers) Torus generally has a better improvement than MSN (except for N = 36 and N = 100 in centralized traffic pattern).
4. (Network size) By applying a 95% confidence interval, it agrees in general that the variation of *PI* of the samples decreases as the network size increases.
5. (Network size) The *PIs* for the Ring topology decrease with network size for all traffic patterns. Moreover, the *PIs* tend to converge to a same value of *PI* when N is sufficiently large. On the other hand, except for ShuffleNet in random and ring traffic patterns, the value of *PI* do not decline when the network grows.

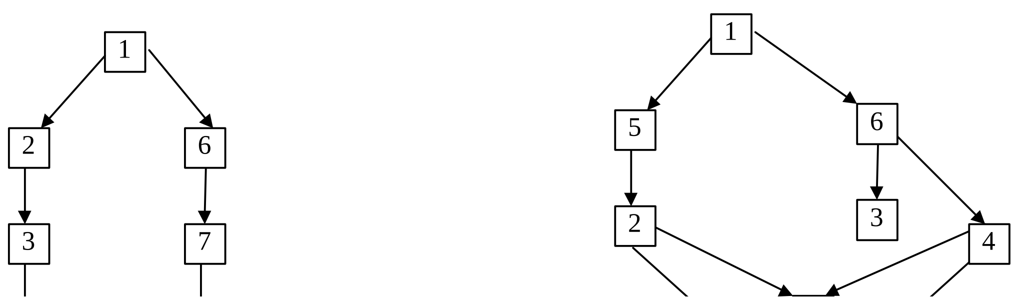
2.4.2 Discussion

In this section, we are going to discuss the observation of the aforementioned experimental results and give insight into the effectiveness of the optimal node assignment under different situations. To maximize the throughput in a network, it is ideal that the traffic of the network can be distributed evenly in each network links. Similarly, the effort on optimization would be wasted if the traffic between any node were the same. Actually, the ideal case would rarely be met and the traffic between nodes should be uneven and change in time. However, the distribution of traffic may retain a similar pattern, say a ring pattern, in a long period, say few minutes. Therefore our experimental result is worthy in real situation.

First, it is helpful to understand how the network performance can be improved through node assignment. A general idea is to exploit the variance in the traffic intensity and the traffic flow on each logical link of a topology. In general, $E(l)$ can be further reduced by assigning a relatively small hop distance (in the distance matrix) to a relatively high traffic intensity entry (in the traffic matrix) by making a corresponding node assignment [8]. Similarly, it is helpful to assign a relative large hop distance to a relative low traffic intensity entry. In order to minimize the largest traffic flow among the links in a network, it is ideal to distribute the network traffic along the links evenly but still follow the shortest path between two nodes to minimize their propagation delay. As the idea can be employed in many heuristic algorithms. Therefore, we expect that the following explanation for the numerical results is independent of the choice of heuristic algorithm as long as the algorithm gives a reasonably good solution to the ONAP.

A. Effect of topology

From the Figures 2.4 to 2.6, ring topology exhibits the lowest *PI* among the topologies with same nodal degree. We can explain this observation in the following. With the same traffic matrix among the topologies, one major factor affecting the *PI* is the number of the routing path between nodes. It is the requirement that the routing path between nodes should follow the shortest hop distance between them. If there are more than one routing path with an equal shortest hop distance between any two nodes, the traffic of the two nodes is randomly directed to one of the paths. Among the four regular topologies that we have studied, it is found that the Ring topology has the smallest deviation paths between two locations with the same shortest hop distance. We illustrate the routing diagrams in Figure 2.12 and the routing paths in Table 2.1 for 8-nodes network from the node in location 1 to other nodes in different locations for the four regular topologies. A key observation from the Figure and the Table that except the node in location 5, Ring topology only posses one routing path from the node in location 1 to other nodes in different locations. Other topologies have more than one locations possess two or more routing paths between nodes. The situation is the same even in a large network size. So we conclude that the topologies having more routing paths between their nodes have a higher *PI*. As the locations in the four topologies are symmetric, every node feels itself to be lying in the center of the network. The routing paths for every node of a regular topology should have a similar routing structure.



(a)

Destination node	Routing path (source node → intermediate nodes → destination node)
2	1 → 2
3	1 → 2 → 3
4	1 → 2 → 3 → 4
5	1 → 2 → 3 → 4 → 5
	1 → 8 → 7 → 6 → 5
6	1 → 8 → 7 → 6
7	1 → 8 → 7
8	1 → 8

(b)

Destination node	Routing path (source node → intermediate nodes → destination node)
2	1 → 5 → 2
3	1 → 6 → 3
4	1 → 6 → 4
5	1 → 5
6	1 → 6
7	1 → 5 → 2 → 7
	1 → 6 → 4 → 7
8	1 → 5 → 2 → 8
	1 → 6 → 4 → 8

(c)

Destination node	Routing path (source node → intermediate nodes → destination node)
2	1→ 2
3	1 → 2 → 3
4	1 → 2 → 3 → 4
	1 → 5 → 8 → 4
5	1 → 5
6	1 → 2 → 6
7	1 → 2 → 3 → 7
	1 → 5 → 8 → 7
8	1 → 5 → 8

(d)

Destination node	Routing path (source node → intermediate nodes → destination node)
2	1 → 2
3	1 → 2 → 3
	1 → 4 → 3
4	1 → 4
5	1 → 5
6	1 → 2 → 6
	1 → 5 → 6
7	1 → 2 → 3 → 7
	1 → 2 → 6 → 7
	1 → 4 → 3 → 7
	1 → 4 → 8 → 7
	1 → 5 → 6 → 7
	1 → 5 → 8 → 7
8	1 → 5 → 8
	1 → 4 → 8

Table 2.1: Routing paths from node in location 1 to other locations from a 8-nodes network for (a) Ring topology, (b) ShuffleNet topology, (c) MSN topology and (d) Torus topology.

We formulate the ONAP as a Quadratic Assignment Problem which is well-known to be *NP*-Hard. We conjecture that the *PI* decreases as the network size grows. However, only the Ring topology exhibits such situation. Others topologies exhibit a higher *PI* when the size of the network increases. For example, in Figure 2.9, the 64-nodes ShuffleNet have a higher *PI* (= 30.48%) than the *PI* (= 24.15%) from the 24-nodes ShuffleNet. We explain the phenomenon by the fact that there are more routing paths of the nodes from a 64-nodes network than that from a 24-nodes network. As the improvement is largely dependent on the routing paths between the nodes and the routing paths increases as the network size increases, the percentage of improvement from the topologies (ShuffleNet, MSN and Torus) may not exhibit a monotonic decreasing feature with the grow of the network. We illustrate the conjecture by finding the mean and standard deviation of the number of routing paths between the nodes for 8-nodes and 24-

nodes ShuffleNet in Table 2.2. We see that the mean number of routing path and the standard deviation of number of routing paths increase as the network size increases.

Size of ShuffleNet	Mean number of routing paths	Standard deviation of number of routing paths
8-nodes	1.28	0.45
24-nodes	1.78	1.10

Table 2.2: Mean and standard deviation of the routing paths between nodes for 8-nodes and 24-nodes ShuffleNet.

B. Effect of Traffic Pattern

From Figures 2.8 to 2.11, the *PI* in general is higher for the centralized traffic pattern for all topologies. Given a network topology, the *PI* of different traffic patterns is large be affected by the following two factors:

(i) The distribution of traffic of the nodes. It is observed that the higher the traffic distributed unevenly, the larger the *PI* will be. To illustrate the observation, we calculate the standard deviation of the traffic density for 64-nodes and 100-nodes networks. It is shown in Table 2.3 and 2.4 respectively. We observe that the percentage of improvement increases as the standard deviation of the traffic density increases. Similarly, the standard deviation increases as the difference between the average hop distance of the random assignment and that of the optimal assignment increases. As our target is to minimize the maximum link flow among the logical network, it should be no change in the maximum link flow regardless of the node assignment if there is a uniform traffic among the nodes. We observe that among the four traffic patterns, the standard deviation of the centralized traffic pattern is the highest. Hence, we expect the percentage of improvement of the centralized traffic pattern should be the highest. However, even though the standard deviation of the random traffic pattern is more than twice the values from the ring traffic pattern and the clustered traffic pattern, the percentage of improvement of the random traffic pattern does not exhibit a higher value than that from other two traffic patterns. In this case, the second factor in the following may explain the observation.

Traffic Pattern	SD_{in}	SD_{out}	$\overline{E(l)}_{ra}$	$\overline{E(l)}_{oa}$	*PI* (%)
Random	46.11	43.72	2586	2186	15.49
Ring	16.12	15.80	1027	871.1	15.18
Clustered	19.26	18.41	2483	2105	15.23
Centralized	91.86	89.84	1269	1036	18.38

SD_{in}= Standard Deviation of in-traffic, SD_{out} = Standard Deviation of out-traffic

Table 2.3: Standard deviation of the in-traffic and out-traffic from the nodes of 64-nodes MSN

Traffic Pattern	**SD_{in}**	**SD_{out}**	$\overline{E(l)}_{ra}$	$\overline{E(l)}_{oa}$	***PI* (%)**
Random	57.13	56.28	4557	3980	12.66
Ring	20.05	19.59	1793	1564	12.77
Clustered	25.31	23.45	4496	3864	14.05
Centralized	115.00	115.30	2328	1822	21.73

SD_{in}= Standard Deviation of in-traffic, SD_{out} = Standard Deviation of out-traffic

Table 2.4: Standard deviation of the in-traffic and out-traffic from the nodes of 100-nodes MSN

(ii) The second factor is the matching of the traffic pattern with the topology. We expect that the maximum link flow will be largely minimized if the nodes with heavy traffic can have a lightpath between them. This is the case when the traffic pattern matches with the logical topology of the network, the improvement gain will be high. For the traffics with the same standard deviation on traffic density, the traffic pattern becomes the determining factor for the improvement. We illustrate this in the following example.

There are three traffic matrices shown in Figure 2.13. The standard deviations of traffic density of these matrices are the same but the patterns differ. For example, Figure 2.13 (a) shows a traffic matrix which is generated according to the connectivity of 8-nodes MSN. Traffic from node *i* to *j* is 10 if distance from node *i* to node *j* is 1 and the traffic is 1 otherwise. We can easily obtain a significant improvement if the nodes are arranged according to the traffic pattern. In the case of regular topology, we obtain a significant improvement when the traffic pattern matches with the logical topology. Figure 2.13 (b) shows a traffic matrix that is slightly different from (a), we observe there is a drop in the improvement already. Figure 2.13 (c) shows a traffic matrix that has a random traffic pattern. The improvement is clearly less than that of the matrices in Figure 2.13 (a) and (b). So, we conclude that the matching between the traffic pattern and the topology plays a dominant factor on improvement if the standard deviation of the traffic density is the same. This can also explains the low improvement of the random traffic pattern, which in fact is the second highest standard deviation on traffic density among the four traffic patterns.

0	10	1	1	10	1	1	1

0	10	1	1	1	1	1	10

0	1	1	10	1	1	1	10

1	0	10	1	1	10	1	1
1	1	0	10	1	1	10	1
10	1	1	0	1	1	1	10
10	1	1	1	0	1	1	10
1	10	1	1	10	0	1	1
1	1	10	1	1	10	0	1
1	1	1	10	1	1	10	0

E(l) of Random Assignment = 38

E(l) of Optimized Assignment = 19

Percentage of Improvement = 50 %

(a)

1	0	10	1	1	10	1	1
1	1	0	10	1	1	10	1
10	1	1	0	1	1	1	10
10	1	1	10	0	1	1	1
1	10	1	1	10	0	1	1
1	1	1	1	10	10	0	1
1	1	10	1	1	1	10	0

E(l) of Random Assignment = 47

E(l) of Optimized Assignment = 27

Percentage of Improvement = 42.55%

(b)

10	0	10	1	1	1	1	1
1	1	0	1	1	10	10	1
1	10	1	0	10	1	1	1
1	1	1	10	0	1	10	1
10	1	10	1	1	0	1	1
1	1	1	1	10	1	0	10
1	10	1	1	1	10	1	0

E(l) of Random Assignment = 38

E(l) of Optimized Assignment = 27

Percentage of Improvement = 28.95%

(c)

Figure 2.13: Examples of different traffic matrices of 8-nodes MSN. The standard deviations of the traffic density are the same.

C. Effect of Number of Transceivers

From the observation 3, Torus exhibits a higher *PI* than MSN. As the topological structure of the two topologies is the same, the nodal degree (the number of transceivers) plays an important factor on the number of routing paths between the nodes. As comparison with MSN in Table 2.1, Torus has more routing paths from a location to other locations. With the explanation of (B), Torus should posses a higher *PI* than that of MSN.

D. Effect of Network Size

It is well known that the QAP exhibits asymptotic property which states that the difference between the worst solution and the optimal solution tends to zero as the problem size tends to infinity [11]. The asymptotic property was proved under the following conditions: (1) the entries in the traffic matrix are mutually independent, (2) the entries in the route matrix are mutually independent, and (3) the entries in the traffic matrix and that in the route matrix are mutually independent. However, the entries in the route matrix for the ONAP are clearly not independent. The entries in clustered, ring and centralized traffic matrices are also not independent.

Since the requirements for the QAP's asymptotic property cannot be satisfied totally by the traffic matrix and also the route matrix where the number of routing paths between nodes changes with the network size by the explanation of (B), the *PI* does not decrease monotonically with the increase of network size except the Ring topology and ShuffleNet topology in ring and random traffic patterns. However, we still observe a significant percentage of improvement with the QAP approach.

From the observation 4, the confidence interval decreases when the network size increases. Confidence interval is well known to be a statistical method to find out the variation of the sample data. For the samples under a traffic pattern, a variation on the improvement of the samples to a topology was mainly caused by the different traffic intensity of the samples. Under a certain traffic pattern, there is a high value of confidence interval in a small network size means that the variation of traffic intensity on the nodes causes a significant effect to the improvement. As the network size increases, the value of confidence interval decreases as the variation of traffic intensity on the nodes decreases. Moreover, the phenomenon is also caused by the asymptotic property of the problem. Among the four traffic patterns, the confidence interval for the random traffic always maintain the largest value except the Ring topology where the random traffic is the second highest. These are shown on Figure 2.14 to 2.17. Since the traffic from the random traffic pattern does not maintain a particular traffic pattern as compared with other three traffic patterns, a higher value on confidence interval from the random traffic pattern implies that a particular traffic pattern can limit the variation on the *PI*.

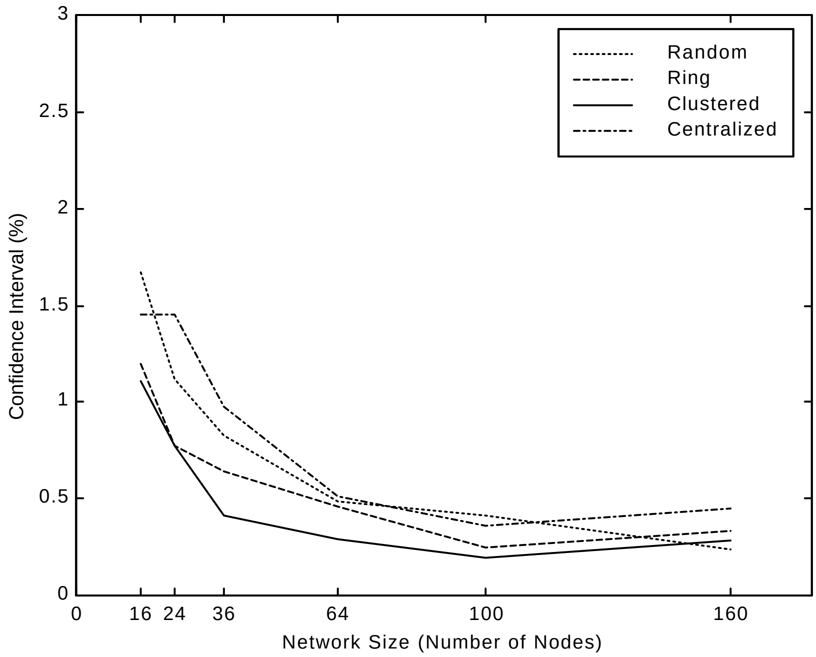


Figure 2.14: Confidence Interval for different traffic patterns with Ring topology.

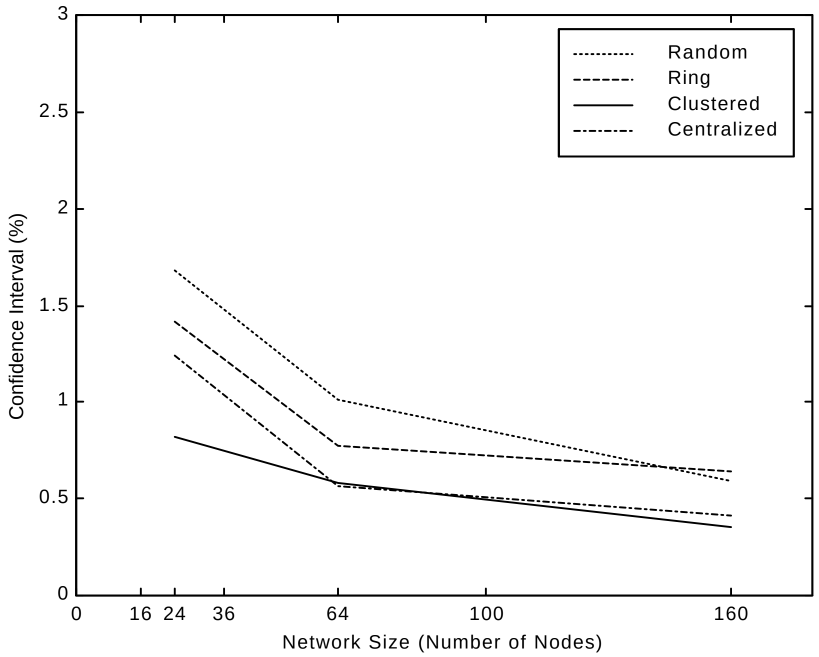


Figure 2.15: Confidence Interval for different traffic patterns with ShuffleNet topology.

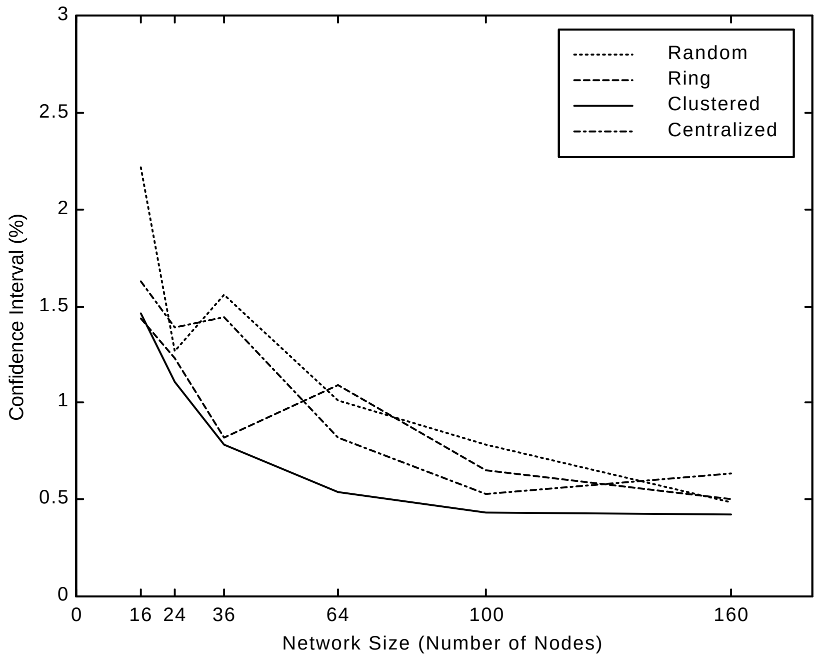


Figure 2.16: Confidence Interval for different traffic patterns with MSN topology.

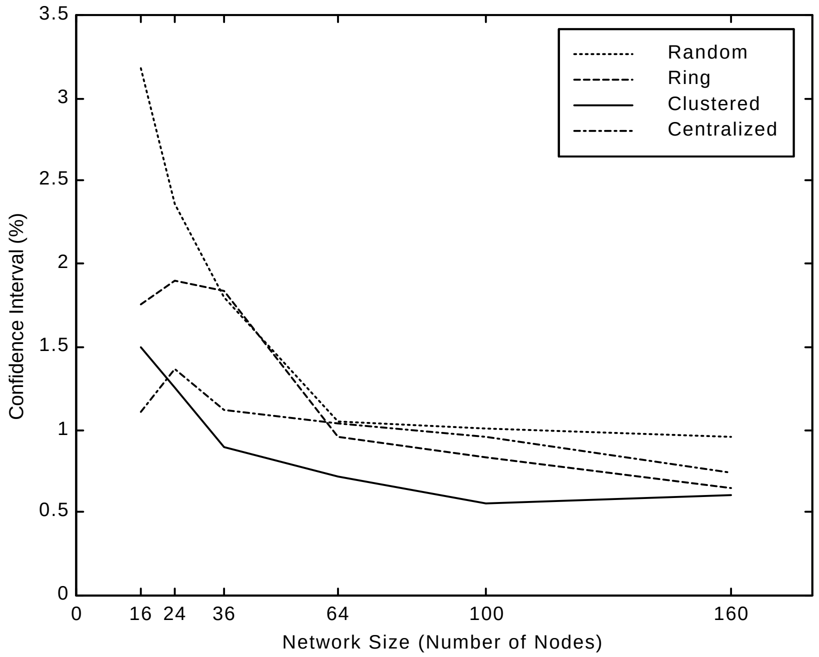


Figure 2.17: Confidence Interval for different traffic patterns with Torus topology.

Chapter 3

THROUGHPUT OPTIMIZATION IN WDM LIGHTWAVE NETWROKS FOR ARBITRARY TOPOLOGY

3.1 Introduction

In the previous chapter, we have shown that the network throughput of regular topologies can be increased by optimal node assignment. Among the four regular topologies, ring topology is found to have the lowest improvement. It is explained that there is only one routing path between source node and destination node except the farthest destination node and the network size is an even number. In other regular topologies, the routing path between source node and destination node increases when the network size increases. Apart from the routing paths between the source node and destination node, we also show that the matching between the topological structure and traffic pattern plays an important role on increasing throughput.

In this Chapter, we address the relationship between topological structure with traffic pattern. We assumed a full *WDM* with all nodes now be possible to set up lightpaths between any two nodes. Similar to the previous Chapter, we assume that each node is equipped with two transceivers so that all graphs of degree two are candidates for possible virtual topologies. Besides, we also assume that all lightpaths are routed over the shortest path on the physical topology. Our problem is to assign the $2N$ node links to $2N$ node transceivers in multihop arbitrary topology network and computes the percentage of improvement on largest link flow between random link assignment and optimal link assignment. We formulate the problem as an Optimal Link Assignment Problem (OLAP) as contrast to ONAP in Chapter 2. Our objective is to maximize the throughput of the network by minimizing the largest link flow in the networks so that the throughput of the networks is the maximal. We solve the OLAP by formulating the OLAP as a Quadratic Assignment Problem (QAP). We will finally prove that the OLAP for arbitrary topology is indeed an *NP*-Hard problem and the QAP have the asymptotic behavior [10] which states that any solution is almost as good as an optimal solution in a large network.

Because the optimal solution is the same as the worst solution when the traffic intensity on every node is uniform, we focus the optimization on four non-uniform traffic patterns and determine their percentage of improvement. The four traffic patterns are random traffic pattern, ring traffic pattern, clustered traffic pattern and centralized traffic pattern. We determine the percentage of improvement for the network size of 16, 24, 36, 64, 100 and 160 nodes. Because this kind of problem is so complicate and cannot be solved in polynomial time, the simulated annealing algorithm will again be used to solve the problem by swapping the links between transceivers in order to minimizing the large link flow. An example is shown in Figure 3.1 where 4-nodes network originally formatted as (a), then we get (b) when there is swapping of the link from node 2 to node 1 with the link from node 3 to node 4.

(a) (b)

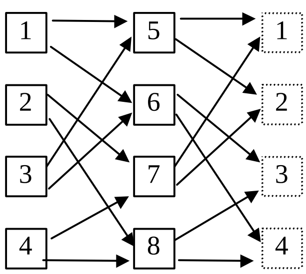
Figure 3.1: Example of arbitrary topology for 4-nodes network.

The rest of the chapter is organized as follows. In Section 3.2, we introduce the formulation of the OLAP as a QAP. In Section 3.3, we present and discuss our experimental results. Then we make a comparison between the results that we get from regular topology and arbitrary topology in Section 3.4. Finally, we conclude this work in Section 3.5.

3.2 Problem Statement

Arbitrary topology optimization is also called as full WDM [4]. It is much more complicate than the regular topology optimization as the arbitrary topology lacks the information of node locations that should be determined through simulation process. Some arbitrary topologies for 8-nodes network are illustrated in Figure 3.2. Although Figure 3.2 (a) and (b) are regular topologies of ShuffleNet and MSN, they are also members of arbitrary topology. In order to reduce the complication on the OLAP and any disconnection of networks, the following assumptions are enforced in the OLAP.

(a) ShuffleNet



1. Each node has two transmitters and two receivers.

2. The transmitters of a node does not allow to connect back to itself.

3. There are at most one lightpath connected between two nodes.

We need the assumption (1) because the condition is more complicate and will involve a long computational time on the simulation program if the transceiver of the nodes has more than two. Besides, the restriction can allow us to compare the results of improvement from ShuffleNet, MSN and Ring in Chapter 2 where the three topologies have two transceivers in each node. We need the assumption (2) because the condition should be avoided in real situation. The assumption (3) is enfored because intermediate node only route the incoming traffic through one of the shortest path which is determined after the arbitrary topology is formed. If two nodes have more than one lightpath connected to each other, one of the lightpath would be idled throughout the optimization process. Indeed we can recompute the routing path during the transmittion of each package but the computation will be so complicate and will involve a long computational time and not subitable in practical situation.

4. The capacity of each WDM channel is C units (say bps). The traffic matrix is given by [$_{sd}$], where $_{sd}$ is the rate of traffic flow, which can be measured in terms of bits/second, generated from source node s to destination node d for s, d = 1, 2, . . . , N. We further assume $_{ss}$ = 0 and is the sum of all $_{sd}$.

5. The flow in link ij is denoted by f_{ij} , while the $_{sd}$ traffic flowing through link ij is denoted by f_{ij}^{sd} . f_{ij}^{sd} = 1 if $_{sd}$ flows through the link ij and f_{ij}^{sd} = 0 otherwise.

6. Let l_{ij} represents the link matrix of logical connections from network node in location i to network node in location j. Without loss of generalities, l_{ij} is restricted to take value in {0, 1}, thereby allowing at most one directed link from one station to another. So, l_{ij} = 1 if the node in location i is logically connected to location j and l_{ij} = 0 otherwise.

7. Assignment matrix $X = [x_{sk}]$, where x_{sk} = 1 if a transmitter of node s is assigned a link which directly connects to a receiver of node k and x_{sk} = 0 otherwise.

The traffic matrix $_{sd}$ and the link matrix l_{ij} are input to the problem directly and the matrix f_{ij}^{sd} is determined from route matrix r_{sd} that is predefined by the Dijkstra's algorithm.. The route matrix indicates the next intermediate node for the packet from source node s to destination node d for s, d = 1, 2, . . . , N so that the hop distance should be the shortest. In case of there exists two intermediate nodes that can route the packet with the same shortest distance, one of them is chosen randomly. Assignment matrix is a decision variable, which determines the link assignment upon solving the problem. The objective function of the OLAP is then given by (3.1) to (3.4).

$$\min\left(\max_{(i,j)} f_{ij}\right) \qquad \text{for } i, j = 1, 2, \ldots, N$$

$$\text{or } \min\left(\max_{(i,j)}\left(\sum_{s=1}^{N}\sum_{d=1}^{N}\sum_{h=1}^{N}\sum_{k=1}^{N} g_{sd} f_{ij}^{sd} l_{ij} x_{sh} x_{dk}\right)\right) \quad \text{for } i, j = 1, 2, \ldots, N \qquad (3.1)$$

$$\text{subject to} \quad \sum_{s=1}^{N} x_{sh} = 1 \qquad \text{for } h = 1, 2, \ldots, N \qquad (3.2)$$

$$\sum_{k=1}^{N} x_{ik} = 1 \qquad \text{for } i = 1, 2, \ldots, N \qquad (3.3)$$

$$x_{ik} \in \{0,1\} \qquad \text{for } i, k = 1, 2, \ldots, N \qquad (3.4)$$

The objective function $\left(\max_{(i,j)}\left(\sum_{s=1}^{N}\sum_{d=1}^{N}\sum_{h=1}^{N}\sum_{k=1}^{N} g_{sd} f_{ij}^{sd} l_{ij} x_{sh} x_{dk}\right)\right)$ is the throughput of the link with the maximum network throughput among the network. We denote the throughput of the link as *El*. The constraints of (3.2) and (3.3) ensure that the transmitter of a node connect exactly one receiver of another node.

Simulated annealing algorithm is known to be a heuristic algorithm that the solution from the algorithm can give a better result but the computing is more complicate. The algorithm applied in our problem is described in Figure 3.3. To compute the resulted percentage of improvement (*PI*), we denote $E(l)_{ra}$ and $E(l)_{oa}$ as the maximum link flow from a random link assignment and from an optimal link assignment respectively. The *PI* for the OLAP is computed by the difference of $E(l)_{ra}$ and $E(l)_{oa}$ and is given by

$$PI = \frac{E(l)_{ra} - E(l)_{oa}}{E(l)_{ra}} \times 100\% \qquad (3.1)$$

Let $l(i)$ be the link of two nodes for all i = 1,2, …, 2*N*.
Let $_{ij}$ be the reduction in the cost if links *i* and *j* are swapped.

1. Generate a random topology.
2. If not all nodes in a network; go to step 1.
3. Initialize the temperature *T*.
4. While (no_of_attempts < max_attempt) do
   - While (no_of_moves $\leq$ max_move) do
     - Randomly select two neighboring links *i* and *j*
     - Compute $_{ij}$ if *l(i)* and *l(j)* are swapped.
       - If $_{ij} < 0$, swap the links; no_of_attempts = 0; Increment no_of_moves
     - If $_{ij} \geq 0$, swap the links and increment no_of_moves with probability $e^{-\Delta_{ij}/T}$; increment no_of_attempts
   - $T = T \times a$

Figure 3.3: A simulated annealing algorithm for the OLAP with arbitrary topology with two transceivers in each node.

Similar to the previous Chapter, we solve the QAP with simulated annealing algorithm with the same parameters. The *MaxAttempt* is set to 10 ×*N* and the *MaxMove* to *N* where *N* is the network size. The cooling rate is set to 0.8 and the initial temperature *T* is computed based on the formula : $T = \frac{-\overline{\Delta_+}}{\ln c}$, where is the desired probability that two links swap will be accepted for an initial solution and $\overline{\Delta_+}$ is the average change of $_{ij}$ for those links swaps with $_{ij} > 0$ for the initial solution. By setting the value of to 0.6, we compute $\overline{\Delta_+}$ by taking randomly a number of neighbors of the initial solution where $_{ij} > 0$ and computing the average change of cost. To even out the fluctuation of samples, we apply fifty samples for each type of traffic patterns and calculate the average of the *PI*. In order to find out the variation of the *PI* of the samples, 95% of confidence interval is found for each traffic pattern.

3.3 Shortest Path Search Strategy

Shortest (and longest) path problems are one of the most important problems in order to determine an arbitrary topology. In this section we will illustrate the shortest path search strategy to determine the routing path of the network nodes. In fact the strategy is based on the concept of the Dijkstra's algorithm which is described in Appendix A. We apply the strategy to find the routing path of the four nodes with two transceivers each in Figure 3.4.

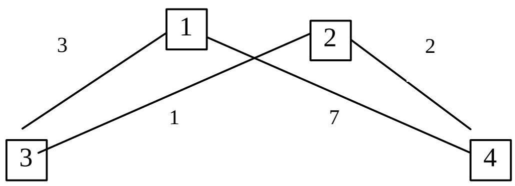


Figure 3.4: Example of 4-nodes network.

Let L_{ij} be the distance from node *i* node *j* and $L_{ii} = 0$,. If the distance is undefine, we denote as -1. PL_i is the list of node finding the routing path from *i*.

Step 1: We first find the shortest distance start from source node 1 to other nodes.

$L_{12} = -1$, $L_{13} = 3$, $L_{14} = 7$ $PL_1 = \{1\}$

As L_{13} is the minimum, the next starting node would be from node 3 and routing path to node 3 would be 1 → 3.

$L_{12} = \min(-1, L_{13} + L_{32} = 3+1 = 4)$, $L_{14} = 7$ $PL_1 = \{1, 3\}$

As L_{12} is the minimum, the next starting node would be from node 2 and routing path to node 2 would be 1 → 3 → 2

L_{14} = min(7, $L_{13} + L_{32} + L_{24}$ = 3+1+2=6), L_{14} = 7 PL_1 ={1, 3, 2}

As L_{12} is the minimum, the next starting node would be from node 2 and routing path to node 4 would be 1 → 3 → 2 → 4. Therefore we have found all the routing paths from node 1 to other nodes.

Step 2: We repeat the step 1 by finding the shortest distance start from source node 2 to other nodes. Indeed the step 1 is repeated by changing the start node to other source node until all of the routing paths in the network have found.

Finally, we get the routing paths for the nodes in the network. In this example, we shows the routing path from the source node 1 to other nodes in Table 3.1.

Source node	Destination node	Routing path
1	2	1 → 3 → 2
	3	1 → 3
	4	1 → 3 → 2 → 4

Table 3.1: Routing path from source node 1 to other nodes.

3.4 Numerical Results and Discussion

3.4.1 Numerical Results

In this section, we will interpret the numerical result finding through our simulation program. The results are shown in Figure 3.5. We will interpret the result according to the traffic patterns and the network sizes. The observation of the result is as following.

1. (Traffic pattern) The *PI* attained for the centralized traffic pattern is generally higher than the other three traffic patterns, which have a *PI* similarly.

2. (Network size) The *PI* is monotonically decreases with network size (except for N = 16 for centralized traffic pattern). It follows that apart from the centralized traffic pattern, other traffic patterns converge to the same value of *PI* when N is sufficiently large.

3. (Network size) By applying a 95% confidence interval, in general, the variation of *PI* of the samples decreases as the network size increases.

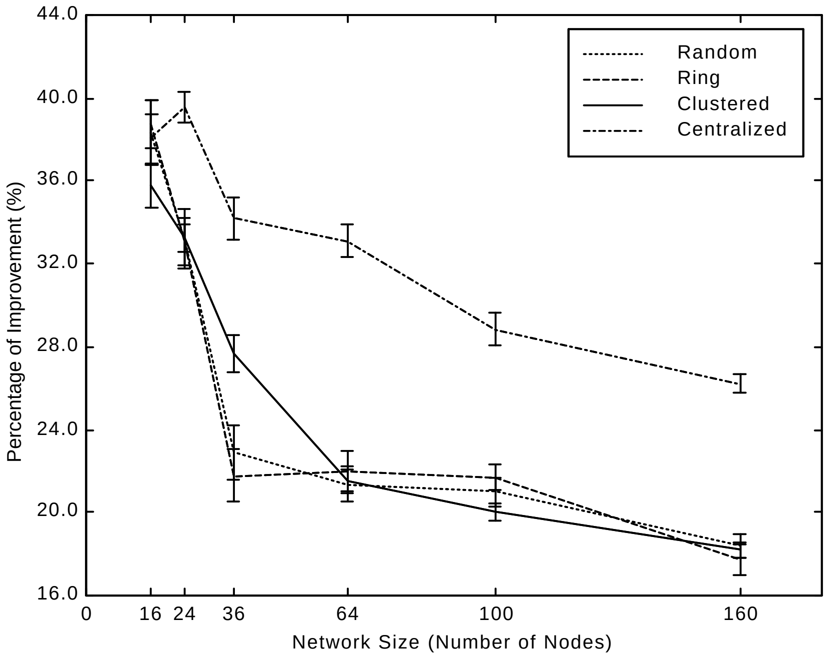


Figure 3.5: Performance of the simulated annealing algorithm of the simulated annealing algorithm on different traffic patterns.

3.4.2 Discussion

In the following discussions, we refer to the numerical results in Figure 3.5. We will especially concentrate on the effect on *PI* from the traffic pattern and the network size.

A. The Effect of Traffic Pattern

In the Chapter 2, we said that the *PI* is largely dependent on (i) the traffic density of the nodes and also (ii) the matching between traffic pattern and topology. If there is a large uneven traffic density among the nodes, the value of *PI* for the traffic should be high. In the Table 3.2 and 3.3, we determine the standard deviation of traffic density of 64-nodes and 100-nodes networks. The centralized traffic pattern exhibits the highest standard deviation among the four traffic patterns and also the *PI* for the traffic pattern is the highest. It is by no mean that only centralized traffic pattern suitable for optimization. The result only implies that a higher standard deviation on traffic density, a higher improvement will be awarded through optimization.

Traffic Pattern	**SD$_{in}$**	**SD$_{out}$**	$\overline{E(l)}_{ra}$	$\overline{E(l)}_{oa}$	***PI* (%)**
Random	46.11	43.72	2586	2186	21.37

Ring	16.12	15.80	1027	871.1	22.00
Clustered	19.26	18.41	2483	2105	21.48
Centralized	91.86	89.84	1269	1036	33.12

SD_{in}= Standard Deviation of in-traffic, SD_{out} = Standard Deviation of out-traffic

Table 3.2: Standard deviation of the in-traffic and out-traffic from the nodes of 64-nodes network.

Traffic Pattern	**SD_{in}**	**SD_{out}**	$\overline{E(l)}_{ra}$	$\overline{E(l)}_{oa}$	***PI* (%)**
Random	57.13	56.28	4557	3980	20.97
Ring	20.05	19.59	1793	1564	21.68
Clustered	25.31	23.45	4496	3864	20.04
Centralized	115.00	115.30	2328	1822	28.81

SD_{in}= Standard Deviation of in-traffic, SD_{out} = Standard Deviation of out-traffic

Table 3.3: Standard deviation of the in-traffic and out-traffic from the nodes of 100-nodes network.

For the second factor, the *PI* is high if there is a matching between traffic pattern and topology. Intuitively, arbitrary topology optimization can form any topology to match with underlying traffic pattern so that a high *PI* can be awarded. Nevertheless, some traffic patterns cannot be matched by a topology due to limitation of node's transceiver and also available lightpaths. Among the four traffic patterns, ring and clustered traffic patterns can be matched by suitable arrangement of node links. Centralized traffic patterns will match no topology because there are only 2 transceivers in each node but the traffic only concentrates on a server node. From the numeric result, we find that the first factor is more important than the second factor as the *PI* from centralized traffic pattern is the highest.

B. Effect of Network Size

From the Figure 3.5, the numeric result shows that the QAP formulation for ONAP satisfies the asymptotic property. The asymptotic property was proved under the following conditions: (1) the entries in the traffic matrix are mutually independent, (2) the entries in the route matrix are mutually independent, and (3) the entries in the traffic matrix and the entries in the route matrix are mutually independent. Though the entries in the route matrix are not mutually independent, the Figure 3.5 shows that our QAP formulation is also *NP*-Hard. Apart from the centralized traffic pattern, the *PIs* from the other three traffic patterns tend to be the same as the network size is very large. Besides, the *PI* decreases rapidly as the network size increases from 24-nodes to 64-nodes.

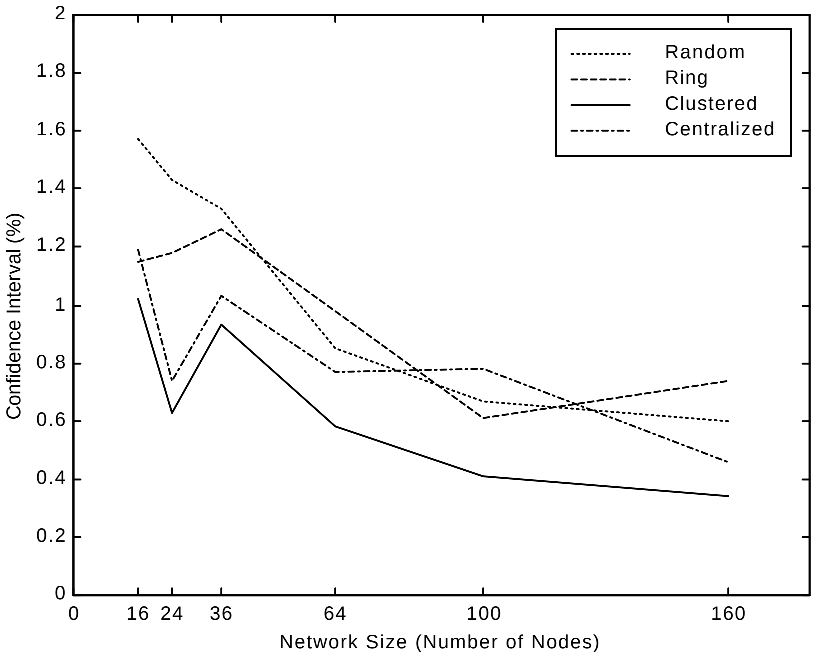


Figure 3.6: Confidence Interval for different traffic patterns in optimization with arbitrary topology.

From the observation 3, the confidence interval of improvement decreases when the network size increases. In Chapter 2, a high value of confidence interval for regular topology is owing to the variation on the traffic intensity of the samples of the traffic patterns. The variation of confidence interval with the network size is shown in Figure 3.6. We see that in small network size (N = 16, 24), the confidence interval for random traffic pattern is the highest. As explained in the previous Chapter, a predefined traffic pattern will limit the variation of the improvement. However, the variation decreases as the network size increases due to the asymptotic property of the problem as well as the variation of traffic density on network node decreases.

3.5 Comparison between Regular Topologies and Arbitrary Topology

The aim of this report is concentrated on the percentage of improvement on regular and arbitrary topologies under different traffic patterns in order to find out some properties between topological structure and traffic patterns. In this section, we would like to make a comparison between regular topology and arbitrary topology under various traffic patterns, network size and nodal degree. Graphical representation for their percentage of improvement under the four non-uniform traffic patterns is given in Figure 3.7 to 3.10. A discussion for the numerical results is as follows.

1. With the same nodal degree of the network nodes, the value of *PI* from arbitrary

topology can obtain a higher value than the four regular topologies. It is under our expectation and conjecture. In a specific traffic pattern, the value of *PI* is dependent on the number of routing paths between the nodes and also the matching between the traffic pattern with the topological structure. In fact a regular topology can be a member of the arbitrary topology. Therefore, the *PI* for arbitrary topology is expected to have a better value on *PI* especially in a small network size.

2. Apart from the clustered traffic pattern with N = 100 and N = 160, the *PIs* from arbitrary topology (nodal degree = 2) have a higher value than Torus (nodal degree = 4). It does not mean that a free topological structure is more important than the number of transceiver in the nodes. In a case that the number of transceiver of the nodes is enough to form all lightpaths communication between them. The performance should be the best because there is only one hop distance between the nodes. In our case, a free topological structure with nodal degree two is seen to be more effective than a regular topology with twice the nodal degree. Even in the clustered traffic pattern with N = 100 and N = 160 where Torus has a higher *PI*, the difference on *PI* between arbitrary topology and Torus is only a small value.

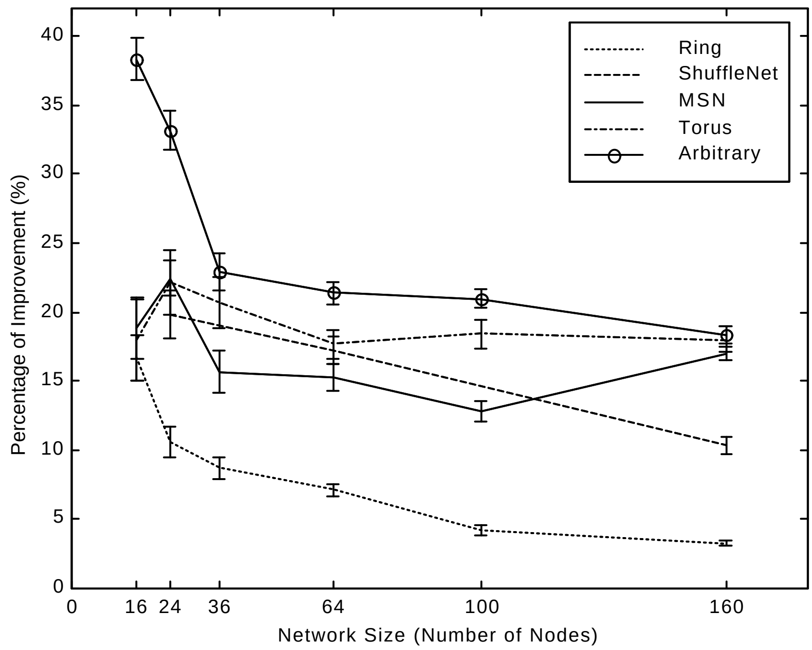


Figure 3.7: Performance of the simulated annealing algorithm on regular and arbitrary topologies with random traffic pattern.

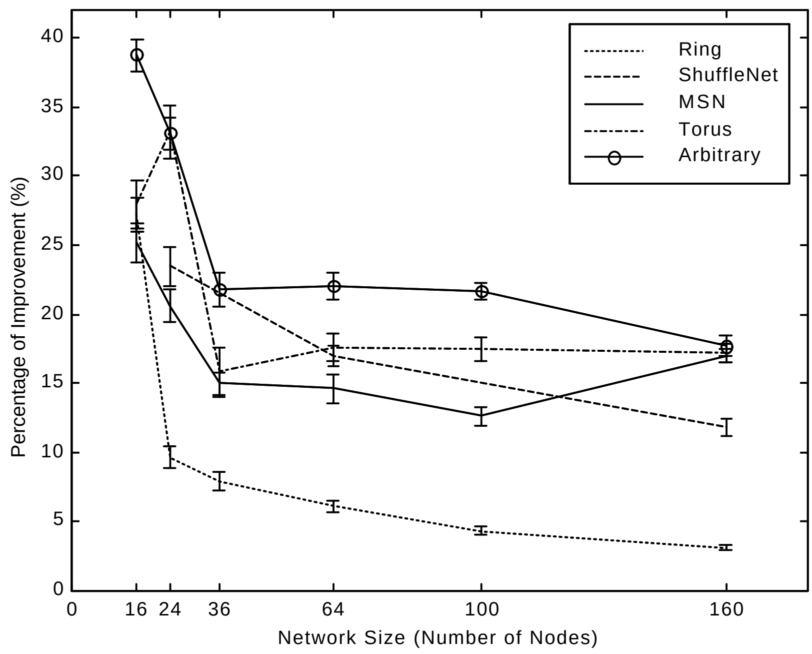


Figure 3.8: Performance of the simulated annealing algorithm on regular and arbitrary topologies with ring traffic pattern.

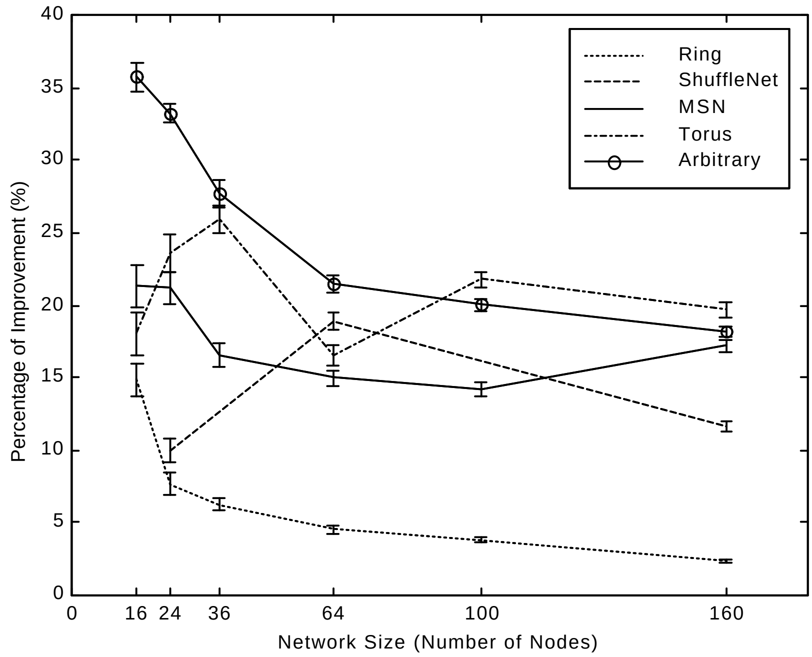


Figure 3.9: Performance of the simulated annealing algorithm on regular and arbitrary topologies with clustered traffic pattern.

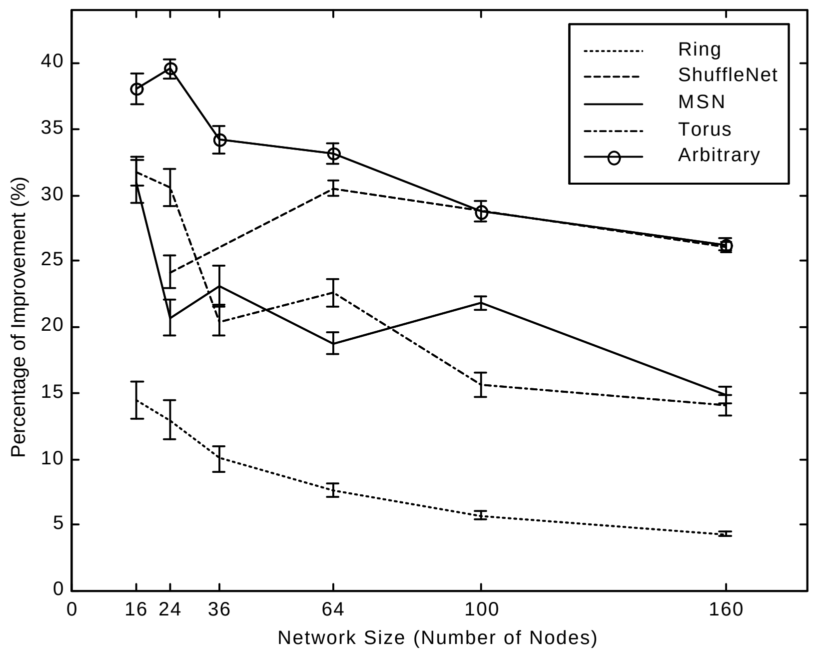


Figure 3.10: Performance of the simulated annealing algorithm on regular and arbitrary topologies with centralized traffic pattern.

Chapter 4

CONCLUSIONS

4.1 Conclusion of this Work

This dissertation is concerned with improving the performance of optical WDM network with regular and arbitrary topologies. The work for the dissertation includes (1) formulation of the optimal node assignment problem as a Quadratic Assignment Problem for regular topologies in order to increase network throughput. (2) Formulation of the optimal link assignment problem as a Quadratic Assignment Problem for arbitrary topology in order to increase network throughput and (3) develop simulation programs to find out optimal solution

The first contribution is in the study of the Optimal Node Assignment problem (ONAP). We study the ONAP for regular network topology and also arbitrary network topology. The objective of the problem is to maximize the network throughput by minimizing the largest link flow in the networks. We use the definition defined in [8] to formulate the problems as quadratic assignment problems (QAP) and establish the *NP*-Hardness of the problem. We find that the routing paths between the nodes increases as the network size increases which in term reducing the effect of asymptotic behavior of the problem. Nevertheless, the result shows a significant improvement on the network throughput from the optimal node assignment.

In arbitrary topology, we show that the decrease in the improvement as the network size increases is due to the asymptotic property of the problem. We make use the concept of ONAP to formulate the OLAP as a quadratic assignment problem (QAP). The result shows that the problem is in fact *NP*-Hard and has a higher improvement in small to moderate network size.

The third contribution is writing simulation program to solve the ONAP and OLAP and verify the theoretical prediction through simulated network environment. Simulation programs are usually used to simulate practical situation and prove the theoretical idea before applied in practical environment, e.g. nuclear explosion. Our simulation programs are shown to be an efficient mean on finding optimal assignment problems especially on arbitrary topology.

4.2 Further Directions

We present several research directions here in order to enhance the quality of the optimal assignment problems.

Our experimental results show a significant improvement on the performance through the QAP formulation though some of the regular topologies do not exhibit an obvious asymptotic property. Since solving a QAP is so complicate and also time consuming especially in arbitrary topology, further research may be considered to simplify the QAP such that the QAP can be solved in polynomial time [5]. On the other hand, if the network traffic pattern is static, the heuristic algorithm can be improved by combining simulated annealing algorithm with flow deviation algorithm as proposed in [13] to solve optimization problem in NSFNET T1 backbone.

Further research may also be considered to compliance the percentage of improvement with throughput optimization and shortest path optimization [8]. When the network traffic is small to moderate, shortest path optimization is considered to be more important [8]. When the traffic in a network is heavy, throughput optimization is more important since it can effectively decrease link saturation due to heavy traffic in the network. An algorithm can be developed to make use of different kind of optimization strategies according to the loading on the network.

Appendix A

DIJKSTRA'S ALGORITHM FOR SHORTEST PATH

Start Algorithm Dijkstra [$G = (V, E)$, $V = \{1, 2, \ldots, n\}$, l_{ij} for all (i, j) in E]

Given a connected graph $G = (V, E)$ with vertices 1, 2, …, n and edges (i, j) having lengths $l_{ij} > 0$, this algorithm determines the lengths of shortest paths from vertex 1 to the vertices 2, …, n.

> INPUT: Number of vertices n, edges (i, j) and lengths l_{ij}
> OUTPUT: Lengths L_{ij} of shortest paths $1 \rightarrow j$, $j = 2, \ldots, n$
> Where PL (a permanent label) = length L_v of a shortest path $1 \rightarrow v$
> TL (a temporary label) = upper bound
> PL = the sets of vertices with a permanent label
> TL = the sets of vertices with a temporary label

1. Initial step
   Vertex 1 gets PL: $L_1 = 0$.
   Vertex j (= 2, …, n) gets TL: $\tilde{L}_{1j}$ $(= \infty$ if there is no edge (1, j) in G).
   Set PL = {1}, TL = {2, 3, . . ., n}.
2. Fixing a permanent label
   Find a k in TL for which $\tilde{L}_k$ is minimum, set $L_k = \tilde{L}_k$. Take the smallest k if there are several. Delete k from TL and include it in PL.
   If TL = (that is, TL is empty) then
   OUTPUT $L2, \ldots, Ln$. Stop
   Else continue (i.e., go to Step 3).
3. Updating tentative labels
   For all j in TL , set $\tilde{L}_j = \min_k \{\tilde{L}_j, \tilde{L}_k + l_{kj}\}$.
   Go to Step 2.

End of Dijkstra

The algorithm has an initial step in which vertex 1 gets the permanent label $L_1 = 0$ and the other vertices get temporary labels, and then the algorithm alternates between Step 2 and 3. In Step 2 the idea is to pick k "minimally". In Step 3 the idea is that the upper bounds will in general improve (decrease) and must be updated accordingly [18].

BIBLIOGRAPHY

[1] Jean-Pierre Laude, Wavelength Division Multiplexing, Prentice Hall International Series in Optoelectronics, pp. 3-14, 1993.

[2] Biswanath Mukherjee, "WDM-Based Local Lighwave Networks – Part I: Single-Hop Systems", IEEE Network, vol. 6, pp. 12-26, May 1992.

[3] Biswanath Mukherjee, "WDM-Based Local Lightwave Networks – Part II: Multihop Systems", IEEE Network, pp. 20-32, July 1992.

[4] Biswanath Mukherjee, "Optical Communication Networks", McGraw-Hill Series on Computer Communications, 1997.

[5] P. Pardalos, F. Rendl and H. Wokowicz, "The Quadratic Assignment Problem: A Survey and Recent Developments", Proceedings of the DIMACS Workshop on Quadratic Assignment Problems, Vol. 16 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science, pp. 1-41, American Mathematical Society, 1994.

[6] C. Brackett, "Dense Wavelength Division Multiplexing Networks: Principle and Applications" IEEE Journal on Selected Areas in Communications, vol. 8, pp. 948-964, Aug. 1990.

[7] Jean-Francois P. Labourdette, "Traffic Optimization and Reconfiguration Management of Multiwavelength Multihop Broadcast Lightwave Networks", Computer Networks and ISDN Systems, pp. 981-998, 1998.

[8] F. Siu, K. C. Chang, "Optimal Node Assignment in Reconfigurable WDM Lightwave Networks with Regular Virtual Topology", Computer and Communications, ISCC '98. Proceedings. Third IEEE Symposium on, pp. 375-379, 1998.

[9] Wu-Ji Li and J. MacGregor Smith, "An Algorithm for Quadratic Assignment Problems", European Journal of Operational Research 81, pp. 205-216, 1995.

[10] Sahni. S. and Gonzalez, T., "P-complete approximation problems", Journal of the ACM 23, pp. 555-565, 1976.

[11] W. Rhee, "A Note on Asymptotic Properties of the Quadratic Assignment Problem", Operations Research Letters, pp. 197-200, 1989.

[12] G. Finke, R. E. Burkard, and F. Rendl, "Quadratic Assignment Problems", Annals of discrete Mathematics 31, pp. 61-82, 1987.

[13] B. Mukherjee, D. Banerjee, S. Ramamurthy and A. Mukherjee, "Some Principles for Designing a Wide-Area WDM Optical Network", IEEE/ACM Transactions on Networking, vol. 4, No. 5, Oct. 1996.

[14] Robert M. Gagliardi and Sherman Karp, Optical Communications, John Wiley & Sons, Inc., 2nd Edition, pp. 21-27, 1995.

[15] Jobn M. Senior, Michael R Handley, Mark S. Leeson, "Developments in Wavelength Division Multiple Access Networking", IEEE Communications Magazine, vol. 36, pp. 28-36, Dec. 1998.

[16] Jadranka Skorin-Kapov, "On Throughput Optimization in Optical Networks With Regular and Arbitrary Topologies", Journal of Computing and Information Technology, 1997.

[17] Jean-Francois P. Labourdette, Anthony S. Acampora and George W. Hart, "Reconfiguration Algorithms for Rearrangeable Lightwave Networks", IEEE INFOCOM '92, Eleventh Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 3, pp. 2205-2214, 1992.

[18] Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons, sixth ed., Chapter 22, pp. 1133-1152, 1988.