On the Application of the Blocking Island Paradigm in All-Optical Networks

Ding Zhemin and Mounir Hamdi, *Member, IEEE*

***Abstract*—In this paper, we investigate the problem of routing and wavelength assignment as well as the problem of the placement of wavelength converters in all-optical networks. In particular, we present a general framework, based on the blocking island (BI) paradigm, to illustrate how it can be used to solve these problems in a unified way. We first give a brief introduction about the BI graph network model, and then use this model to derive simple and general algorithms that can be used in various applications in optical networks. We discuss the implementation issues of our algorithms and present simulation results to evaluate and compare our solutions with other heuristic algorithms under both static and dynamic traffic assumptions.**

***Index Terms*—Blocking island (BI) paradigm, routing, wavelength assignment, wavelength converters.**

I. INTRODUCTION

WAVELENGTH-division multiplexing (WDM) is widely regarded as the key technology for the next generation Internet. Current WDM systems offer 8–32 wavelengths at 2.5–10 Gb/s/wavelength, approaching 1 Tb/s capacity, while research-level systems already exceed multiple terabits in a single fiber [1]. The flexibility of WDM optical networks has been achieved using wavelength routing, which allows operators to provide network node pairs with end-to-end optical channels, known as lightpaths. However, there are many challenges that need to be addressed before this WDM technology becomes truly mature. One of these challenges is the optical network resource allocation and management (ONRAM), which covers many different issues such as routing and wavelength assignment (RWA), traffic grooming, and load balancing, to name a few. This paper focuses on the introduction and the application of a general framework that addresses many issues in the ONRAM problem in a unified way. This framework is based on a clustering scheme called the blocking island (BI) [2]. The BI is an abstraction technique of available resources in a network, especially bandwidth. It can efficiently plan the allocation of network resources to connection requests and provide a novel way of identifying the bottlenecks in a network.

In this paper, we first review the basic idea of the BI and the blocking island graph (BIG) network model. In Section IV, we introduce a general algorithm, BI-RWA, that can solve the RWA problem under different assumptions: static or dynamic traffic, and single or multiple fiber links between node pairs. In Section V, we further apply the BI paradigm to solve the problem of placement of wavelength converters in all optical networks. Simulation results are presented in Section VI. Section VII concludes the paper.

II. BLOCKING ISLAND (BI) PARADIGM

In this section, we assume all the network requests are unicast traffic and the only quality of service (QoS) parameter taken into account is bandwidth. The network physical topology consists of m nodes arbitrarily connected by n bidirectional links. We depict it by a network graph $G = (V, L)$ as shown in Fig. 1, where $|V| = m, |L| = n$. A request is defined by a triple, $d_u = (x_u, y_u, \beta_u)$, where x_u and y_u are distinct nodes of the network and β_u is the bandwidth requirement.

Introduced by Frei and Faltings [2], the BI provides an efficient way of abstracting resource (especially bandwidth) availability in a communication network into different levels. The BI clusters segments of the network according to the bandwidth availability. A β BI for a node x is the set of all nodes of the network that can be reached from x using links with at least β available bandwidth. For example, Fig. 1 shows a 40 BI for node V_1.

β BI has some very useful properties. Below we list a few without proof (for a proof, please refer to [2]).

Unicity: There is one and only one β BI for a node. Thus, if S is the β BI for a node, S is the β BI for every node in S.

Route Existence: Given a request $d_u = (x_u, y_u, \beta_u)$, it can be satisfied if and only if the nodes x_u and y_u are in the same β_u BI. For example, in Fig. 1, a request (V_1, V_9, 40) cannot be satisfied because V_1 and V_9 are not in the same 40 BI, while a request (V_1, V_4, 40) can be satisfied because V_1 and V_4 are in the same 40 BI.

In other words, to check for a route existence between two nodes (given a required bandwidth, β), all we need to do is to check whether the two nodes belong to the same β BI or not.

Inclusion: If $b_i < b_j$, the b_j BI for a node is a subset of the b_i BI for the same node.

Partition: β BI induces a partition of nodes in a network.

Using the concept of β BI, we can construct a recursive decomposition of BIGs in decreasing order of β's, e.g., $\beta_1 > \beta_2 \cdots > \beta_n$. We call this layered structure of BIGs a blocking island hierarchy (BIH). We give an example of BIH in Fig. 2. On the bottom of the hierarchy there is a 0 BI abstracting the

---

Paper approved by W. C. Kwong, the Editor for Optical Communications of the IEEE Communications Society. Manuscript received June 4, 2002; revised January 3, 2003 and April 22, 2003. This work was supported in part by the UGC-Hong Kong under Grant AoE/E-01/99.

The authors are with the Department of Computer Science, Hong Kong University of Science and Technology, Kowloon, Hong Kong (e-mail: hamdi@cs.ust.hk).

Digital Object Identifier 10.1109/TCOMM.2003.818088

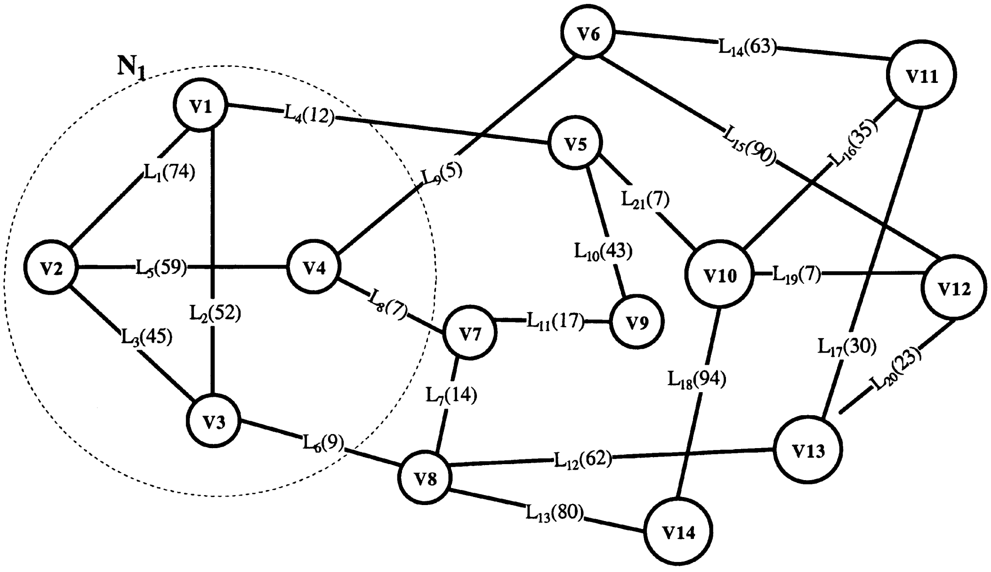

Fig. 1. A network topology (NSFNet) where the available bandwidth on a link is given in brackets. $L_n(x)$ refers to link n which has x available bandwidth. $N_1 = (V_1, V_2, V_3, V_4)$ is 40 BI for node V_1.

smallest resource requirement. The whole network is abstracted into a single node, since the network is connected.

Given a request $d_u = (x_u, y_u, \beta_u)$, based on the β_u BIG, we immediately know whether the request can be satisfied or not by using the "routing existence" property. It may be argued that a link-state routing protocol and Dijkstra's algorithm are also capable of checking the route existence. However, one of the key requirements of resource allocation in communications systems is the ability of responding very quickly to the question: Can I have a route between A and B with a bandwidth X? Thanks to the route existence property of the BI paradigm, unlike the link-state routing, this question can be answered without having to compute a route.

With this abstraction technique, instead of studying the whole network topology, we focus our attention only on a small segment. A β BIG allows us to get a clear picture about the load as nodes and links with enough resources are hidden behind an abstract node. In particular, network bottlenecks are identified by the interlinks between the BIs.

After the allocation of a request, it is possible that some BIs in the BIH have to be split, for there is not enough bandwidth left. For example, in Fig. 1, if we assign a route $V_1 -- > V_3 -- > V_2$ with 40 bandwidth, the 40 BI N_1 will be split into two 40 BIs: (V_1, V_2, V_4) and (V_3). This splitting means that some requests that can be satisfied before the allocation of the route cannot be satisfied anymore. Based on analysis of the consequences that a given route has on the BIH, a routing heuristic called "minimal splitting" (MS) is proposed. The difference between this heuristic and others is that it tries to find a route which does not provoke a split in the BIH. If the splitting is unavoidable, we would rather select a route which incurs the fewest splittings, since the more splittings, the worse the situation gets in terms of future requests. Using the MS heuristic, first, we take the shortest route that does not affect the BIH. Second, if there is no such route, we take the route that causes the fewest splittings. The MS heuristic has a very good load balancing effect and the implicit objective of this heuristic is to reserve the largest possible resources possible for the future requests.

III. BIG NETWORK MODEL

Define a network topology $G(V, L, W)$ for a given WDM optical network, where V is the set of nodes, L is the set of bidirectional links, and W is the set of wavelengths per fiber link. Assume this is a single-fiber network without wavelength converters, then the set of wavelengths on each fiber link is the same. Each connection request needs to be allocated along a route and assigned one wavelength. In particular, the network can be abstracted into $|W|$ BIGs. Each BIG starts with one BI representing one of $|W|$ wavelengths and having the same topology as the original WDM optical network. Hence, the BIG network model BIG $(m_1, m_2, \ldots, m_{|W|})$ can be obtained from a given network topology G as follows. The topology of G is replicated $|W|$ times denoted by $m_1, m_2, .., m_{|W|}$. Each BIG m_i is composed of one BI representing a wavelength and the link capacity is one.

It is obvious that this BIG network model is a simplified BIG. All the properties such as unicity, partition, and route existence still hold.

For a general network, the implementation of the MS heuristic is difficult and time consuming, since all routes must be computed in order to determine which one satisfies the requirement best. On the other hand, the proposed BIG network model is a simplified BIG, hence, the available bandwidth on each link is either 1 or 0 (which means there is only one BIH level). We also propose to approximate this heuristic by combining it with another heuristic (in our case, we use the shortest path heuristic).

1) Compute K different routes according to the shortest path heuristic (K-alternate shortest paths). In our case, the "k different routes" is not a set of edge-disjoint shortest

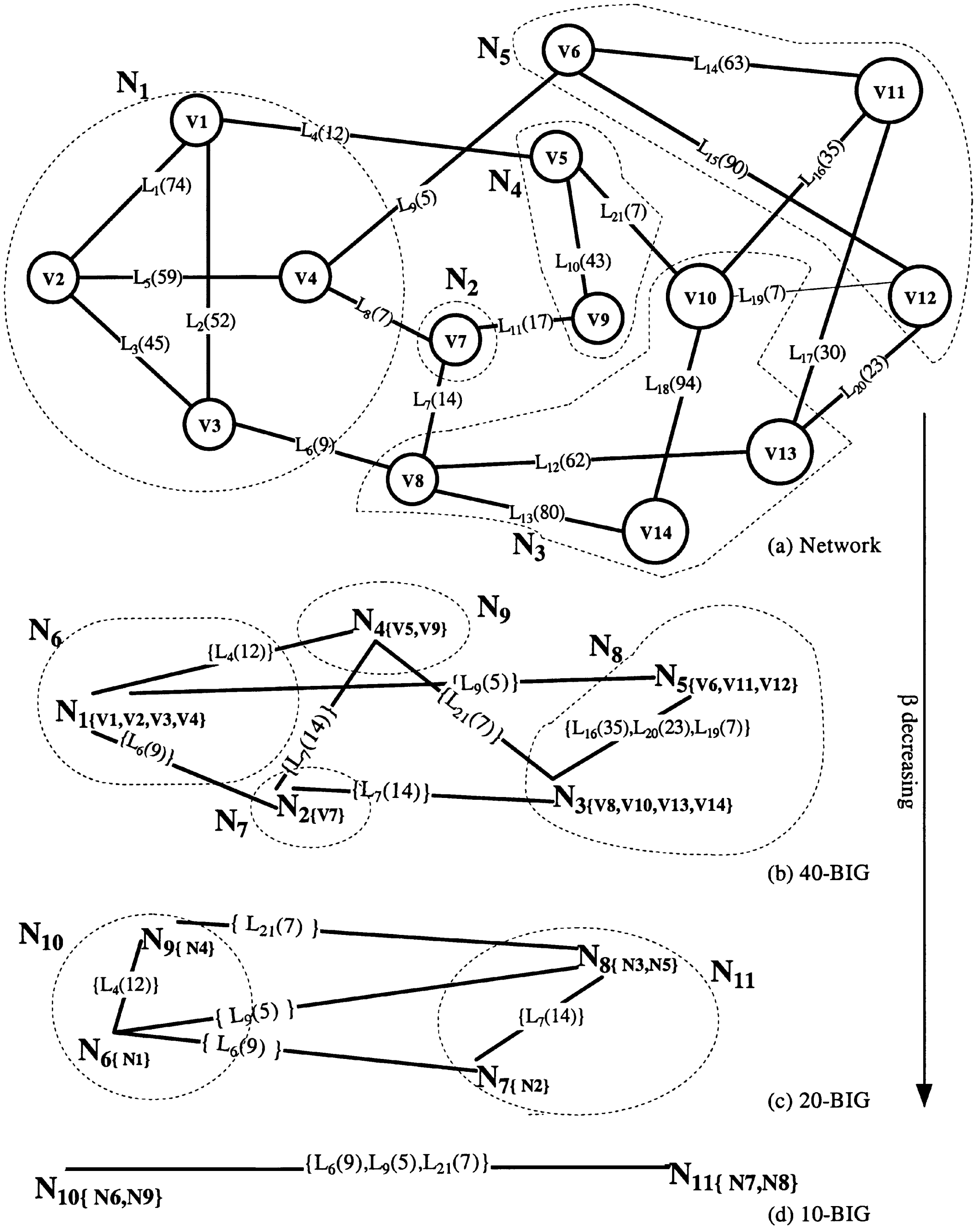

Fig. 2. BIH for bandwidth requirement (40, 20, 10).

paths. The paths in the set may share same links, but there is at least one different link between any two paths of this set.

2) Order them according to the minimal splitting criterion.
3) If the routes have the same minimal splitting number, we use some other heuristics (such as the most-loaded-link heuristic) or select the route randomly.

IV. BI-RWA ALGORITHM

In a wavelength-routed WDM network, a lightpath (e.g., wavelength continuous path without processing in the intermediate nodes) is first established between two network nodes before communication takes place. A lightpath may span multiple fiber links and must occupy the same wavelength on all the fiber links it traverses if there are no wavelength converters. This property is known as the wavelength continuity constraint. In order to satisfy a lightpath request in a wavelength-routed WDM network, we not only need to consider routing, but the wavelength selection as well. Given a set of connection requests, the problem of setting up a lightpath by routing and assigning a wavelength to each connection is called the routing and wavelength assignment (RWA) problem. In this section, we propose a new RWA algorithm using the BI paradigm. The algorithm proposed can be applied to any WDM network with an arbitrary topology.

Briefly, we first transform the network topology into a BIG network model. Then we use the route existence property to decide if the request(s) can be satisfied or not by checking whether the two end nodes are in the same BI of at least one BIG. If they are satisfied, we do the routing and wavelength selection and several heuristics are employed to get the "best" one.

Before describing the algorithm, some concepts need to be explained. *The splitting number* for a route is equal to the number of BIs that will be newly generated if the route is removed from the current BI. *The most-loaded link* for a route means that in a route, there is a link on which the most wavelengths have been used. Also, here we assume traffic is static. Our goal is to maximize the number of accepted requests given a fixed number of wavelengths per fiber link.

```
Static BI-RWA
Input:
 A set of connection requests $D$
Output:
 Whether the network could satisfy all the requests or not
Description:
1: Transform the network topology into a BIG network model.
2: Order all the connection requests in decreasing length of their minimum number of hops (MNH) distance (MNH distance is calculated using any shortest-path algorithm, e.g., Dijkstra. Paths with equal lengths are ordered randomly.).
3: Select an unallocated request $d$, $D = D - d$. If the request set $D$ is empty then go to step 7.
4: Check route existence. If all the requests exist, assign the request $d$ to each possible wavelength BIG and calculate $K$ alternate shortest paths. If they do not, go to step 7.
5: Route and wavelength selection. Now we have a set of candidate routes in different BIGs. Compute the splitting number and the most-loaded link for each route. Find one with the minimum splitting number. If the minimum splitting number is the same, locate the route whose most-loaded link has the most number of available wavelengths. If there are still several routes, pick up one with the shortest MNH distance.
6: Get the route and corresponding wavelength. Reconstruct the BIGs.
7: If the request set is empty, output the result; otherwise, output can not be satisfied.
```

In order to get a more optimal result, a backtracking scheme is added to the algorithm (please refer to **Algorithm 1**) as long as time is allowed. In step 4, if all the requests cannot be satisfied individually, instead of going to step 7, we backtrack to the previous request and try another of K alternate routes. Notice this algorithm may not find a solution even if one exists, since it looks at K shortest paths only. In step 5, several heuristics are employed to get the "best" route and wavelength. The first heuristic tries to keep the integrity of the BIs intact. It is equivalent to keeping the connectivity of each node pair, since the link capacity is one. The second heuristic is to reduce the resource consumption by picking up the shortest path. The third heuristic balances the network load by adjusting the most-loaded links (most number of wavelengths in the link has been used).

If the requests arrive dynamically, we need to do a few modifications to the original algorithm (please refer to **Algorithm 2**). For example, we can not order the requests and the backtracking scheme is impossible.

```
Dynamic BI-RWA
Input:
 Dynamic traffic requests
Output:
 Block or accept the request
Description:
1: Transform the network topology into a BIG network model.
2: A connection request $d$ arrives. Based on the arriving time of the request, reconstruct the BIG.
3: Check the request $d$ using the route existence property in each BI. If it exists in some BI, calculate $K$ alternate shortest paths; if it does not exist in any BI, it is blocked.
4: Route and wavelength selection (the same as the static case).
5: Allocate the request and go to step 2.
6: Get the route and corresponding wavelength. Reconstruct the BIGs.
7: If the request set is empty, output the result; otherwise, output can not be satisfied.
```

If there are multiple fibers per link in the network graph, we adopt the node architecture proposed in [5]. Since an M-fiber W-wavelength network is functionally equivalent to an MW-wavelength network with partial wavelength conversion of degree M, we only need to modify the link capacity between each node pair. In this case, the link capacity is equal to the number of fiber links. The rest of the algorithm can still apply to the new graph.

If there are wavelength converters in the network graph, assuming we know the placement of converters and the conversion range of those converters, we simply replace the original BIG with the modified BIG as the initial input graph. We modify the original BIG by adding virtual links to represent the increasing connectivity between each node pair. The weight of virtual links is zero.

For the static traffic, usually time is not a big concern, so here we only discuss the time complexity of the dynamic RWA algorithm. The most common operation in the dynamic RWA is the BI construction. The β BI for a given node x of a network can be obtained with a simple greedy algorithm. Starting with an initial set x, we recursively add every node to the set if the node can be reached from any other node in the set by a link that has at least β available bandwidth. In the worst case, this construction process will examine all links. Therefore, the β BI construction process is linear in $O(n)$, where n is the number of links in the network ($n = |L|$). If the request cannot be satisfied, it will be determined immediately by using the route existence property. The computation time, in this case, is only the time of reconstructing the BIG, which is $O(|W|mn)$, where m is the number of nodes and n is the number of links in the network ($m = |V|$, $n = |L|$). $|W|$ is the number of wavelength in the network. If the request can be satisfied, the running time is equal to the combination of 1) reconstruction time; 2) K alternate shortest paths; 3) route and wavelength selection; and 4) assign route and wavelength and reconstruction time. That is, $O(|W|mn) + K|W| * O(nlg(m)) + K * O(|W|mn) + O(|W|mn)$, where K is a constant and $|W|$ is a constant. So the running time is linear in $O(mn)$.

V. PLACEMENT OF WAVELENGTH CONVERTERS

In simple WDM networks, a connection must be established along a route using a common wavelength on all of the links making up the route. This constraint may be removed by the introduction of wavelength converters, which are devices that take the data modulated on an incoming wavelength, and transfer it to a different outgoing wavelength. Obviously, wavelength converters improve the network blocking performance. Ideally, each node in the network is able to remove the wavelength constraint completely. However, because of the expensive hardware cost and node complexity, we usually only have a limited number of converters. As a result, an important problem arises. Given a limited number of converters, how do we place them in the network so that maximum network performance improvement is achieved?

There are two cases of wavelength conversion. 1) Complete conversion. In this case, any wavelength can be converted into any other wavelength and such wavelength converters exist in every node. 2) Limited number of converters and limited range of conversion. This means only part of the network nodes have wavelength converters and those wavelength converters may only have a limited range of conversion. The limited range of conversion means either it can only translate limited incoming wavelengths or the translation capacity is limited. In the second case, we can have three network scenarios: 1) a limited number of nodes are provided with full range convertibility; 2) converters with limited range of wavelength conversion are placed at all nodes; and 3) converters with limited range of wavelength conversion are placed at a subset of nodes.

The placement of limited-range wavelength converters at a subset of nodes is an NP-complete problem in an arbitrary WDM mesh network [3]. It was shown that an appropriate placement of limited-range wavelength converters could result in reduced blocking probability and low distortion of optical signals. Lee and Li [3] proposed a shortest-path routing algorithm to reduce the number of converters. The node configuration they employ is called "share per node," and they assume every node is equipped with the same and limited number of full-wavelength converters (FWCs). Notice the concept of FWC is different from our wavelength converter. FWC can only convert one incoming wavelength to any outgoing wavelength. So if a node is provided with full wavelength convertibility, the number of FWCs needed is equal to the total number of outgoing channels of that node. Based on the concept proposed in [3], Xiao and Leung [11] improve the result by using a simulation-based optimization approach. To the best of our knowledge, this allocation requires the smallest number of FWCs to achieve a given blocking probability. [3] and [11] mainly focus on the type-b wavelength converter placement problem. In terms of type-a and c wavelength converter placement problems, the benefits of using wavelength converters in wavelength routed all-optical networks have been studied in [4] and [12]–[15] under various assumptions. Usually, the analytical models are derived from simple topologies and algorithms are proposed under statistical independence assumptions. Although good performance can be obtained, those algorithms are restricted to the specific cases and independence assumptions. Wan *et al.* [17] and Subramanian *et al.* [16] consider the optimal placement of wavelength converters. Wan shows the optimal placement is tractable in topologies like trees and trees of rings. Subramanian considers the placement of wavelength converters on a path assuming link-load independence.

In this paper, we propose a heuristic algorithm based on the BI paradigm to place a limited number of converters at a subset of nodes in arbitrary network topologies. We adopt the simulation-based optimization approach, in which we first collect the utilizing statistics of each node, and then perform the optimization of the allocation of wavelength converters. In the simulation, we show, by optimizing the placement of a limited number of wavelength converters, the blocking performance is very close to that of a network with full wavelength conversion at every node. We also compare our algorithm with the best existing allocation. The results demonstrate that our algorithm can greatly reduce the overall blocking probability.

The basic idea of our algorithm is simple: Try to find the most congested nodes and place converters on them. Since the BI paradigm balances the load in the whole network by keeping the integrity of the BIs intact, we could easily decide the bottleneck links using the BIH. We also record the utilization statistics for each node through processing the incoming traffic generated by computer simulation. Based on the statistics of each node and the bottle neck links in the BIH, we place the wavelength converters.

A. Static Traffic

If the incoming traffic of computer simulation is static, we first assume full conversion at any node. This means there is no wavelength assignment problem. We treat the network as one BI with the link capacity equal to the number of wavelengths. Since the traffic is static, we know all the requests in advance. According to the bandwidth requirements, we build the BIH. We also record the utilization statistics for each node. After processing all the traffic requests, we check the BIH and utilization statistics to identify bottleneck links and nodes with a high volume of traffic. Then we place wavelength converters at those nodes. Please refer to **Algorithm 3** for the detailed algorithm.

B. Dynamic Traffic

We deal with the dynamic traffic case as follows. We first need to obtain certain network statistics of the arbitrary WDM network by simulation. Every time a connection request arrives, we reconstruct the BIH and record the bottleneck links. We also record the call duration statistics for each node. That is, for each transmission, how long the corresponding nodes are occupied. After testing enough requests, we calculate the *tightness* of each link and call duration statistics for each node. For any link L, we define the number of times a link is a bottleneck link by L and the total number of connection requests by N. Then

$$\text{Tightness of a link } L = \frac{BL}{N}.$$

Order all the links in decreasing value of tightness and order all the nodes in decreasing value of call duration statistics. The first link in the list with the highest call duration statistics has the highest priority to put a converter on one of its two nodes. The second link has the second highest priority, and so on.

```
Static Converter Placement
Input:
 A set of static traffic requests and $N$ converters
Output:
 The placement of those $N$ converters
Description:
1: Transform the network into a network without wavelength constraint.
2: Build the BIH based on the bandwidth requirement.
3: Order traffic requests by decreasing length of MNH distance.
4: Select an unallocated traffic request and route it using the lowest level heuristic [2]. The principle is to route a request in the lowest $\beta$ BI, where $\beta$ BI is the highest bandwidth requirement BI that accommodates the endpoints of the request. If the request can not be routed, record blocking information.
5: Update BIH.
6: If the request set is empty, go to step 7; otherwise, go to step 4.
7: Check the utilization statistics for each node to order nodes in the decreasing order of traffic volume. Also check the BIH to identify the most congested links (bottleneck links).
8: Using the congested links and the statistics of traffic volume to decide $N$ most congested nodes and place converters on them.
```

VI. NUMERICAL RESULTS

Simulations have been carried out to examine the performance of placement of wavelength converters and the performance of the BI-RWA algorithms, using the NSFNet with 14 nodes and 21 links. The placement problem is also studied in a randomly generated topology [5], shown in Fig. 3, with 15 nodes and 29 links. We use the same dynamic traffic generator model employed in [5]. Calls (requests) arrive at each node according to an independent Poisson process with arrival rate α. An arriving session is equally likely to be delivered to any node in the network. The session holding time is exponentially distributed with mean $1/\mu$. Thus, the load per $s-d$ node pair is $\rho = \alpha/N(N-1)\mu$, where N is the number of nodes in the network. Note that a node may engage in multiple sessions and several sessions may be simultaneously conducted between an $s-d$ node pair. In our simulation, extensive tests are carried out to ensure a steady state is reached.

A. Wavelength Converter Placement

The placement of wavelength converters (WCs) is evaluated using the NSFNet as well. We first identify the most congested links with uniform traffic. Then we order the nodes according to the bottleneck links. In our case, we select the five most congested nodes to place wavelength converters: V_1; V_4; V_8; V_9; and V_{10}. Those five nodes are then evaluated in Fig. 4 with dynamic traffic, where ten wavelengths for each fiber are considered. In the dynamic case, we place FWCs onto those nodes to evaluate the call-blocking probability. Results show a strong correlation between the degree of a node and its transit traffic.

The assumption made for the simulation in Fig. 4 is that the incoming dynamic traffic has a "steady" pattern. In our case, we assume that the statistics can be obtained by running the simulation with the uniform traffic. It is very difficult to identify both the optimal number and location of WCs to minimize the total cost of the network. From a more practical perspective, we simply let our "cost" decide how many WCs we can have and then try to optimize the location instead.

Fig. 4 shows the benefit of using a limited number of WCs, which can achieve a lower blocking probability (compared with the case with no WCs) at a lower hardware cost (compared with FWCs) by optimizing the location in the network. In Fig. 4, we can see that at lower loads, the blocking probability with WCs is significantly lower, while at higher loads, the network

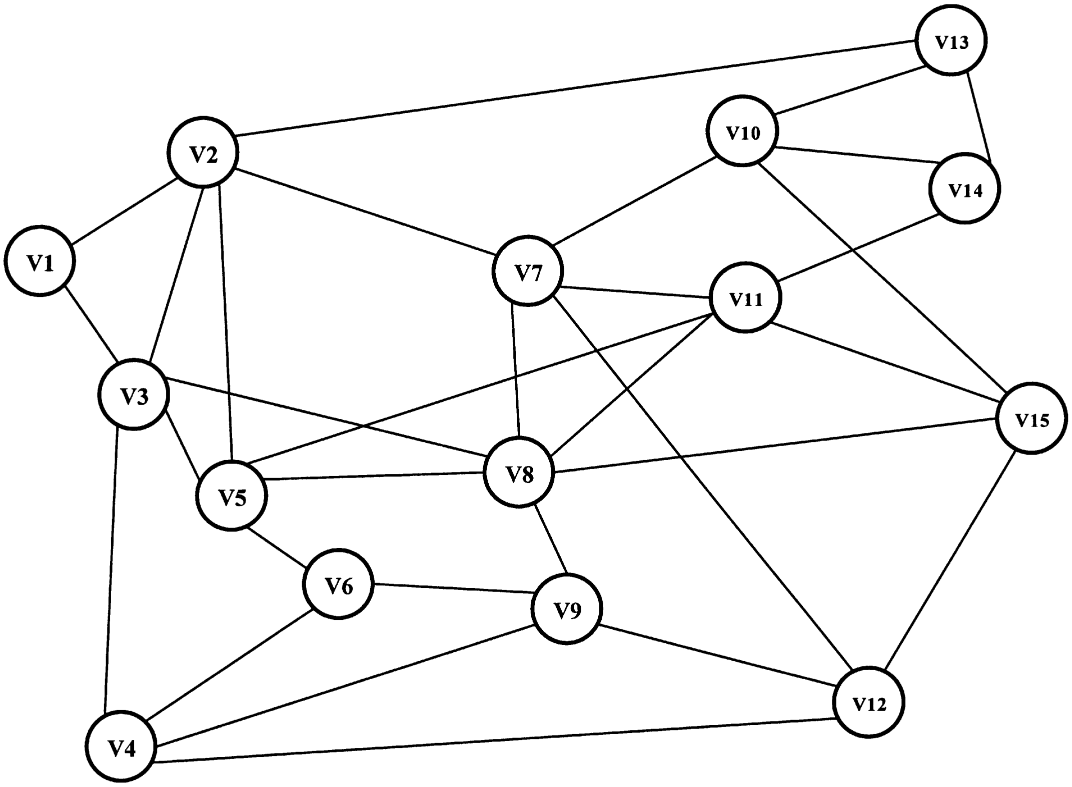

Fig. 3. Randomly generated topology with 15 nodes and 29 links.

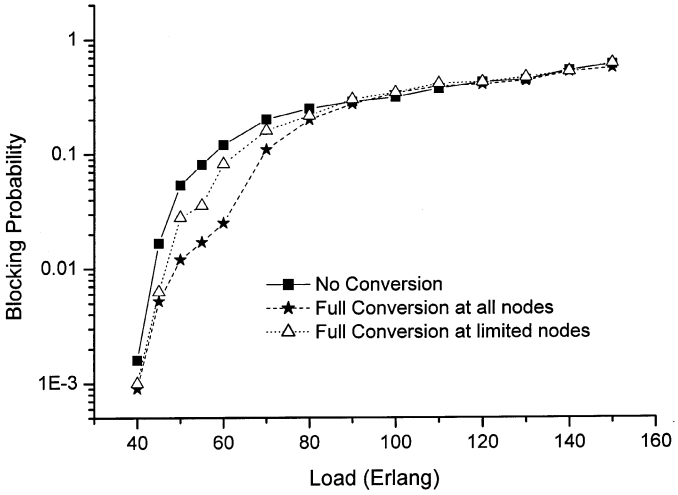

Fig. 4. Blocking probabilities for NSFNet with and without converters.

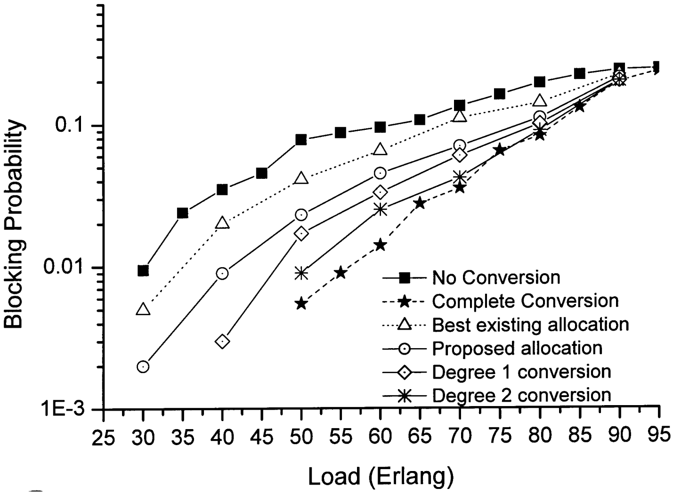

Fig. 5. Overall blocking probability in the randomly generated topology with eight wavelengths per link. The plot shows simulation values for no conversion, complete conversion, limited number of WCs, and limited conversion with degree one and two. In the limited number of WCs, we compare the performance of our algorithm with the best existing allocation.

without WCs has low blocking probability (crossover effect). This phenomenon is due to the suboptimal routing algorithm. Since the networks are usually designed to only have below 1% or 2% blocking probability, this need not be considered.

The randomly generated topology (Fig. 3) has been explored in Fig. 5. We assume the number of wavelengths is eight, and this is a single-fiber all-optical network. The allocation and RWA algorithm we use to do the comparison is proposed in [11]. To our knowledge, it is the best existing allocation scheme without being restricted to any particular network model or assumption. Because this allocation scheme is proposed in a different node configuration called "share per node," we need to modify the algorithm so that it can be applied in this scenario. Based on the node statistics and bottleneck links, we place WCs at corresponding nodes. In this example, we place FWCs at four nodes. They are $V2$, $V4$, $V8$, and $V10$ in our algorithm and $V4$, $V5$, $V12$, and $V13$ in the best existing allocation scheme. In Fig. 5, we can see the blocking probability of both algorithms is much better than that of no wavelength conversion. And our method can give significantly better performance. For example, when the load is 50 Erlang,

TABLE I
SIMULATION RESULT. N IS THE NUMBER OF NODES IN THE NETWORK. L IS THE NUMBER OF LINKS IN THE NETWORK. [6] IS THE RESULT OF OUR IMPLEMENTATION OF THE ALGORITHM IN [6].

Network	N	L	Lower Bound	[5]	*BI_RWA*
ARPANet	20	31	33	34	34
UKNet	21	39	19	22	20
EON	20	39	18	18	18
NSFNet	14	21	13	13	13

the blocking probability of our method and the best existing allocation scheme is 2.3% and 4.1%, respectively.

Fig. 5 also shows a better performance can be obtained by putting a limited range of WCs at every node. The blocking probability is reduced significantly when the degree of conversion is one. When the degree of conversion is two, the performance is very close to that of the complete wavelength conversion. However, putting a limited range of WCs at every node is still very expensive. As a compromise, we may consider putting a limited range of WCs at limited nodes.

Our algorithm, which is based on the BI paradigm, allows us to get an intuitive and clear picture about the load distribution with "noncritical" nodes hidden in an abstract node. In addition, our algorithm is general enough to be applied to any arbitrarily connected networks. With the help of the BI-RWA algorithm, we can easily manage and balance network resources (especially bandwidth and converters).

B. Static BI-RWA

For the static traffic, one of the most important goals is to minimize the number of wavelengths needed to accommodate the given requests. The proposed BI-RWA algorithm with backtracking is applied to several existing or planned network topologies to verify its efficiency. The networks considered are the ARPANet [7], NSFNet [8], the European Optical Network (EON) proposed in [9], and a hypothetical UK topology reflecting the current BT networks [10]. Note, those topologies are also evaluated in [6] and the results in [6] are near optimal. In our simulation, we assume the incoming connection requests are uniform. There is a single-fiber WDM optical network without WCs. The simulation results are shown in Table I. As can be seen, the application of the BI paradigm can lead to optimal or near-optimal results, and sometimes compare favorably with those in [6].

The lower bound is calculated in [6]. It means the minimum number of distinct wavelengths required to support the uniform traffic. The lower bound may not always be achieved using heuristic algorithms. However, it is a very useful measure for any lightpath allocation algorithm to see how far it is from the optimum. In brief, we need to find a cut in the topology. Say, there are five nodes on one side of the cut and nine nodes on the other side, and three links go through the cut. So we can calculate that in order to satisfy a uniform traffic, at least $9 * 5 = 45$ routes have to be set up between the left-hand side nodes and right-hand side nodes. Then, at least $45/3 = 15$ wavelengths for each of the three links are needed to support those routes.

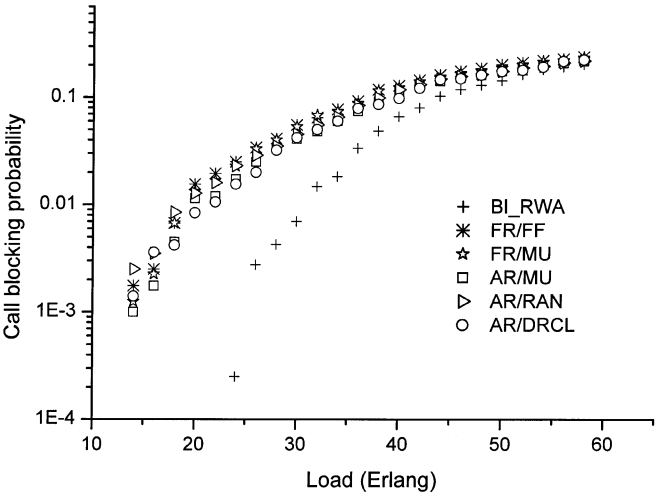

Fig. 6. Blocking probabilities for the NSFNet with eight wavelengths.

C. Dynamic BI-RWA

As for the dynamic traffic, according to a recent survey on RWA approaches [18], we compare our algorithm with well-reputed algorithms such as fixed routing with first-fit wavelength assignment (FR/FF); fixed routing with most used/pack wavelength assignment (FR/MU); alternate routing with most used/pack wavelength assignment (AR/MU); alternate routing with random wavelength assignment (AR/RAN); and alternate routing with distributed relative capacity loss (AR/DRCL). DRCL is based on relative capacity loss (RCL) and is more efficient in a distributed environment. In [18], AR/DRCL offers the best performance.

In Fig. 6, we assume there is a single-fiber WDM optical network without WCs. The number of wavelengths on each link is eight. Results show that the BI-RWA has the best performance, followed by AR/DRCL, AR/MU, AR/RAN, FR/MU, and FR/FF. For example, in the case $|W| = 8$, when the total load is around 35 Erlang, the blocking probability using the BI-RWA is only 4.85 $\times 10^{-2}$, compared to 1.18 $\times 10^{-1}$ using FR/FF, 1.15 $\times 10^{-1}$ using FR/MU, 9.175 $\times 10^{-2}$ using AR/MU, 9.9 $\times 10^{-2}$ using AR/RAN, and 7.9 $\times 10^{-2}$ using AR/DRCL. The simulation results favorably compare our proposed algorithm with the related algorithms.

In Figs. 7 and 8, the dynamic multifiber RWA algorithm is evaluated. The network is treated with even links and unit basic cost. The network with even links means the same number of fibers for every link. The unit basic cost means each fiber for every link has a unit cost. We assume eight wavelengths per fiber.

Fig. 7 shows the call blocking probability of NSFNet with two fibers per link, and Fig. 8 shows call blocking probability of NSFNet with five fibers per link. As expected, the blocking performance improves dramatically with the use of multiple fibers. For example, at a blocking probability of 0.04, in the single-fiber case ($|F| = 1$, $|W| = 8$), the load is about 38, while in the two-fibers case ($|F| = 2$, $|W| = 8$), the load is about 90, and

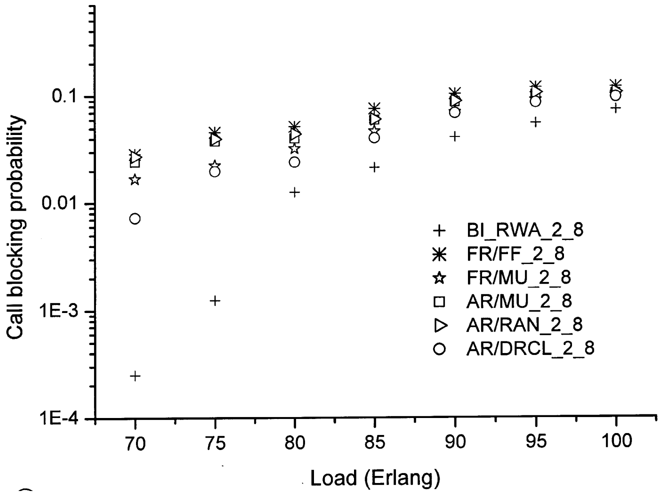

Fig. 7. Blocking probabilities for the NSFNet with two fibers per link, e.g., FR/FF_2_8 means FR/FF algorithm with $|F| = 2$ and $|W| = 8$.

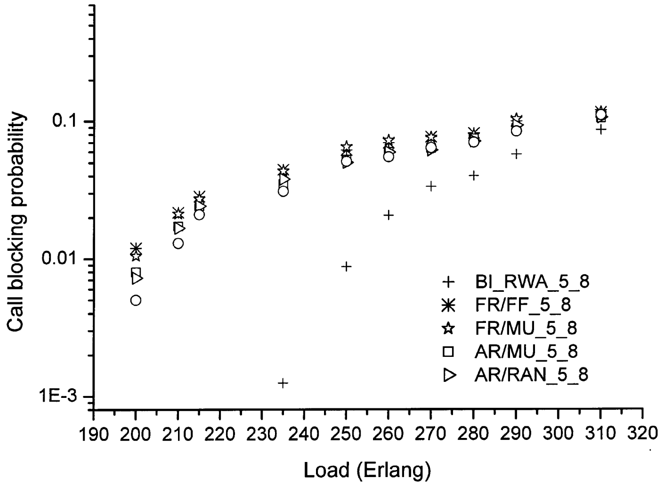

Fig. 8. Blocking probabilities for the NSFNet with five fibers per link, e.g., FR/FF_5_8 means FR/FF algorithm with $|F| = 5$ and $|W| = 8$.

the throughput increases nearly 137%. And in both figures, our algorithm consistently obtains a much lower blocking probability than the other algorithms.

All these results show that when the load is relatively low in each case (that means the RWA algorithm plays a more important role in handling resources, since there are more free resources, and with a better management and allocation, a request is more likely to be accepted), the BI-RWA performs much better than the other algorithms.

VII. CONCLUSIONS

In this paper, inspired by some artificial intelligence abstraction concepts, we have designed a framework to solve the problems of placement of WCs as well as RWA in all optical networks. A simple heuristic for the placement of WCs in an arbitrary mesh network and a general RWA algorithm have been proposed. Simulation results have demonstrated that our algorithms performed very well under various networking scenarios. We believe that this framework would provide a general approach to solve various additional problems in the ONRAM, including traffic grooming, optical traffic engineering, and network failure allocation/recovery.

VIII. ACKNOWLEDGEMENT

The authors would like to thank Dr. G. Xiao for his program and helpful discussions.

REFERENCES

[1] B. Mukherjee, “WDM optical communication networks: Progress and challenges,” *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1810–1824, Oct. 2000.
[2] C. Frei, “Abstraction techniques for resource allocation in communication networks,” Ph.D. dissertation, Swiss Federal Inst. Technol. (EPFL), Lausanne, Switzerland, 2000.
[3] K. C. Lee and V. O. Li, “A wavelength-convertible optical network,” *IEEE J. Lightwave Technol.*, vol. 11, pp. 962–970, May/June 1993.
[4] K. R. Venugopal, M. Shiyakumar, and P. S. Kumar, “A heuristic for placement of limited range wavelength converters in all-optical networks,” in *Proc. INFOCOM’99*, vol. 2, 1999, pp. 908–915.
[5] A. Mokhtar and M. Azizoglu, “Adaptive wavelength routing in all-optical networks,” *IEEE/ACM Trans. Networking*, vol. 6, pp. 197–206, Apr. 1998.
[6] S. Baroni and P. Bayvel, “Wavelength requirements in arbitrarily connected wavelength-routed optical networks,” *IEEE/OSA J. Lightwave Technol.*, vol. 15, pp. 242–251, Feb. 1997.
[7] R. Ramaswami and K. N. Sivarajan, “Routing and wavelength assignment in all-optical networks,” *IEEE/ACM Trans. Networking*, vol. 3, pp. 489–500, Oct. 1995.
[8] ——, “Design of logical topologies for wavelength-routed all-optical networks,” in *Proc. IEEE INFOCOM’95*, 1995, pp. 1316–1325.
[9] M. J. O’Mahony, D. Simeonidou, A. Yu, and J. Zhou, “The design of a European optical network,” *IEEE/OSA J. Lightwave Technol.*, vol. 13, pp. 817–828, May 1995.
[10] S. Appleby and S. Steward, “Mobile software agents for control in telecommunications networks,” *BT Technol. J.*, vol. 12, pp. 104–116, Apr. 1994.
[11] G. Xiao and Y. W. Leung, “Algorithms for allocating wavelength converters in all-optical networks,” *IEEE/ACM Trans. Networking*, vol. 7, pp. 545–557, Aug. 1999.
[12] J. Yates, J. Lacey, D. Everitt, and M. Summerfield, “Limited-range wavelength translation in all-optical networks,” in *Proc. IEEE INFOCOM’96*, vol. 3, 1996, pp. 954–961.
[13] V. Sharma and E. A. Varvarigos, “Limited wavelength translation in all-optical WDM mesh networks,” in *Proc. IEEE INFOCOM’98*, vol. 2, 1998, pp. 893–901.
[14] T. Tripathi and K. Sivarajan, “Computing approximate blocking probabilities in wavelength routed all-optical networks with limited-range wavelength conversion,” in *Proc. IEEE INFOCOM’99*, vol. 1, 1999, pp. 329–336.
[15] H. Harai, M. Murata, and H. Miyahara, “Allocation of wavelength convertible nodes and routing in all-optical networks,” in *Proc. SPIE, All-Optical Communication Syst.: Architecture, Control, Network Issues III*, vol. 30, Oct. 1997, pp. 277–287.
[16] S. Subramaniam, M. Azizoglu, and A. K. Somani, “On the optimal placement of wavelength converters in wavelength-routed networks,” in *Proc. IEEE INFOCOM’98*, vol. 2, 1998, pp. 902–909.
[17] P. J. Wan, L. Liu, and O. Frieder, “Optimal placement of wavelength converters in trees and trees of rings,” *Comput. Commun. Networks*, pp. 392–397, 1999.
[18] H. Zang, J. P. Jue, and B. Mukherjee, “A review of routing and wavelength assignment approaches for wavelength-routed optical wdm networks,” *Opt. Networks Mag.*, vol. 1, no. 1, pp. 47–60, Jan. 2000.

Ding Zhemin received the B.S. degree from Nanjing University, Nanjing, China, in 1999. Currently, he working toward the Ph.D. degree in the Computer Science Department, Hong Kong University of Science and Technology (HKUST), Kowloon.

He is currently a Research Assistant in the networks research lab. His research interest is in optical networks, focusing on optical network resource allocation and management.

Mounir Hamdi (S'89–M'90) received the B.S. degree in computer engineering (with distinction) from the University of Louisiana, Lafayette, in 1985 and the M.S. and Ph.D. degrees in electrical engineering from the University of Pittsburgh, Pittsburgh, PA, in 1987 and 1991, respectively.

He has been a Faculty Member in the Department of Computer Science, Hong Kong University of Science and Technology, since 1991, where he is now Associate Professor of computer science and the Director of the Computer Engineering Program that has some 350 undergraduate students. From 1999 to 2000, he held Visiting Professor positions at Stanford University, Stanford, CA, and the Swiss Federal Institute of Technology, Zürich, Switzerland. His general areas of research are in high-speed packet switches/routers and all-optical networks, in which he has published more than 180 research publications, received numerous research grants, supervised some 20 postgraduate students, and for which he has served as Consultant to various international companies. Currently, he is working on high-speed networks including the design, analysis, scheduling, and management of high-speed switches/routers, wavelength division multiplexing (WDM) networks/switches, and wireless networks. He is currently leading a team that is designing one of the highest capacity chip sets for Terabit switches/routers. This chip set is targeted toward a 256 × 256 OC-192 switch and includes a crossbar fabric chip, a scheduler/arbiter chip, and traffic management chip.

Dr. Hamdi was on the Editorial Boards of the IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE C*ommunication Magazine*, *Computer Networks*, *Wireless Communications and Mobile Computing*, and *Parallel Computing* and has been on the program committees of more than 50 international conferences and workshops. He was a Guest Editor of IEEE *Communications Magazine*, the IEEE JOURNAL ON SELECTED AREAS OF COMMUNICATIONS, and *Optical Networks Magazine* and has Chaired more than five international conferences and workshops, including the IEEE GLOBECOM/ICC Optical Networking Workshop, the IEEE ICC High-Speed Access Workshop, and the IEEE IPPS HiNets Workshop. He is the Chair of IEEE Communications Society Technical Committee on transmissions, access, and optical systems and Vice Chair of the Optical Networking Technical Committee, as well as ComSoc Technical Activities Council. He received the Best Paper Award out of 152 papers at the International Conference on Information and Networking in 1998. In addition to his commitment to research and professional service, he is also a dedicated teacher. He received the Best 10 Lecturers Award (through university-wide students voting for all university faculty, held once a year) and the Distinguished Teaching Award from the Hong Kong University of Science and Technology. He is a Member of ACM.