Energy-efficient Multicast Algorithm for Survivable WDM Networks

Xiaojuan Pu[1] and Young-Chon Kim[2]*

[1]*Department of Computer Engineering, Chonbuk National University, Jeonju 54896, Korea*
[2]*Department of Information Technology, Chonbuk National University, Jeonju 54896, Korea*

In recent years, multicast services such as high-definition television (HDTV), video conferencing, interactive distance learning, and distributed games have increased exponentially, and wavelength-division multiplexing (WDM) networks are considered to be a promising technology due to their support for multicast applications. Multicast survivability in WDM networks has been the focus of extensive attention since a single-link failure in an optical network may result in a massive loss of data. But the improvement of network survivability increases energy consumption due to more resource allocation for protection.

In this paper, an energy-efficient multicast algorithm (EEMA) is proposed to reduce energy consumption in WDM networks. Two cost functions are defined based on the link state to determine both working and protection paths for a multicast request in WDM networks. To increase the number of sleeping links, the link cost function of the working path aims to integrate new working path into the links with more working paths. Sleeping links indicate the links in sleep mode, which do not have any working path. To increase bandwidth utilization by sharing spare capacity, the cost function of the protection path is defined to use sleeping fibers for establishing new protection paths. Finally, the performance of the proposed algorithm is evaluated in terms of energy consumption, and also the blocking probability is evaluated under various traffic environments through OPNET. Simulation results show that our algorithm reduces energy consumption while maintaining the quality of service.

Keywords : Multicast algorithm, Routing, Energy-efficiency, Survivability, Protection, WDM network
OCIS codes : (060.0060) Fiber optics and optical communications; (060.4510) Optical communication

I. INTRODUCTION

Over the last few years, the increase in bandwidth-intensive multicast applications such as HDTV, video conferencing, interactive distance learning, live auctions, and distributed games has driven an enormous increase in the volume of Internet traffic [1, 2]. Such explosive growth in traffic has led to an increase in the energy consumed by networks. Baliga *et al*. have estimated such consumption to be about 4% of the total energy consumed in broadband-enabled countries [3]. As a result, energy efficiency is now an important factor in network design and operation.

Wavelength-division multiplexing (WDM) has developed as a technology to support bandwidth-intensive multicast applications by establishing a lighttree to transmit data from one source node to multiple destination nodes. However, failure of a single fiber on the light-tree will disrupt the transmission of information to several destination nodes. Thus, multicast survivability is important in WDM networks. Several studies have proposed protection schemes to reduce network energy consumption [4-7]. In the proposed schemes, the devices used in the protection paths switch into sleep mode and promptly wake up when a failure occurs. For example, [4] found that a significant reduction in energy consumption could be achieved by using a provisioning solution that packed the working and protection paths into different fibers. However, the schemes proposed in such studies did not consider multicast traffic.

Various schemes have been proposed to protect against link failure in multicast networks, including tree-based [8],

---

*Corresponding author: yckim@jbnu.ac.kr
Color versions of one or more of the figures in this paper are available online.

link-based, and segment-based protection [9]. Tree-based protection finds two link-disjoint light trees that start from the same source and end at the same destination nodes. Link-based and segment-based protection schemes respectively provide protection paths for each link and segment on the working tree. In [10], the authors proposed a multicast protection scheme through spanning paths (MPSP) to reduce the total bandwidth allocation. However, network energy consumption was not taken into account in the design of such multicast algorithms. Reference [11] proposed a green multicast grooming algorithm (GMG) based on dedicated spanning path protection. However, the authors did not consider the use of spare capacity that could be shared between different multicast connections.

The objective of this paper is to design a multicast algorithm for reducing energy consumption in survivable WDM networks while sharing spare capacity. In order to do this, we define two link cost functions, one for the working path and another for the protection path. The two link cost functions used in the routing procedure allow the proposed algorithm to pack the working and protection paths into different fibers to increase both the number of sleeping fibers and the amount of spare capacity that is shared. The proposed algorithm is then evaluated in terms of the number of sleeping fibers, energy consumption, and blocking probability.

The rest of this paper is organized as follows: Section 2 introduces conventional multicast algorithms. Section 3 details the proposed energy-efficient multicast algorithm (EEMA) for survivable WDM networks. In Section 4, the proposed algorithm is evaluated through a simulation, and the conclusions are given in Section 5.

II. RELATED WORK

2.1. Multicasting

Data can be transmitted in various ways on a network: by unicasting, broadcasting, anycasting, or multicasting. In multicasting, the same information is sent to a group of nodes, and upon the arrival of a multicast request, a light-tree is established to transmit the traffic. The light-tree is an extension of the concept of a light path, incorporating multicasting capabilities [12]. A light-tree for a multicast session can be derived from the Steiner tree problem, in which a minimum cost light-tree should be found, and since the Steiner tree problem is *NP*-complete [13], a number of heuristic algorithms have been proposed. These algorithms can be categorized into three groups: minimum path heuristics (MPH), spanning tree-based heuristics (STH), and metaheuristics. The MPH algorithm repeatedly includes the closest new destination node in the partially built light-tree until all destination nodes have joined the tree [14, 15]. The STH algorithm is a fast algorithm that first constructs a closure graph of the multicast nodes from the original graph by using the cost of the shortest path between each

source or destination pair [14]. The closure graph is then used to obtain a minimum spanning tree. Then, the shortest paths from the original graph are used to replace the edges of this minimum spanning tree. Finally, the algorithm removes any cycles to obtain the multicast tree. Metaheuristics, such as simulated annealing [15], genetic algorithms [16], and Tabu search [17], have also been investigated to solve the Steiner tree problem. In this paper, an MPH heuristic algorithm is used to search for the working tree, because such an algorithm guarantees minimum cost paths.

2.2. Multicast Protection

To improve network reliability, a protection tree is established to transmit traffic when a link failure occurs. Several approaches have been proposed in the literature [8-11] to improve network reliability. In [8], the authors propose a multicast tree-based protection scheme that protects a working tree by deriving a link-disjoint protection multicast tree. Two trees are said to be link-disjoint if they do not have any links in common. When a link failure occurs, the destination nodes that are affected reconfigure their switches so that data can be received from the protection tree. The drawback of using this scheme is that in many cases, it is difficult to find two link-disjoint light-trees. In [9], the authors investigated the use of link-based and segment-based protection schemes. A segment is defined as the sequence of edges from the source or from any splitting node, to a leaf node or a splitting node on the working tree. In [10], the authors proposed a spanning path-based protection scheme. A spanning path is a path from a leaf node to any other leaf node in the working tree; this scheme can reduce resource allocation, compared to link-based or segment-based protection. The protection schemes are divided into dedicated protection and shared protection schemes. In a dedicated protection scheme, a spare resource is specifically allocated for a particular working path. In a shared protection scheme, network resources along a protection path are reserved for multiple working paths when considering a single link failure. This implies that a shared protection scheme has higher capacity efficiency than a dedicated protection scheme.

Capacity-sharing schemes for multicast networks can be classified into three categories: self-sharing intrarequest sharing, and interrequest sharing [10]. Self-sharing schemes involve sharing capacity between the working and protection paths of the same multicast. In Fig. 1, the multicast is (0, {5, 3}), where node 0 is the source node. The working-tree is {0-1-2-3, 0-5} and the protection paths are {0-1-5, 0-5-4-3}. The links (0-1) and (0-5) can be shared. An intrarequest sharing scheme shares link capacity among the protection paths of the same multicast connection. In Fig. 2, the multicast request is {0, {5, 3}}. The protection paths for working-paths (0-1-5) and (0-1-2-3) are (0-5) and (0-5-4-3), respectively. The capacity on link 0-5 can be shared by the two protection paths. Interrequest sharing is capacity sharing among protection paths of different
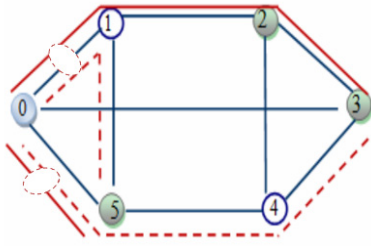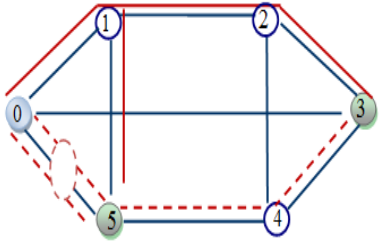
FIG. 1. Self-sharing.
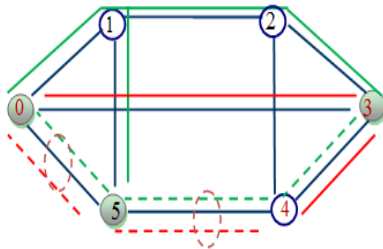

FIG. 2. Intrarequest sharing.

FIG. 3. Interrequest sharing.

multicast connections. In Fig. 3, there are two multicast connections: {0, {5, 3}} and {3, {0, 4}}. The capacity of links 0-5 and 5-4 can be shared by two multicast connections.

2.3. Conventional Multicast Algorithm

In [10], the authors proposed a multicast protection scheme through spanning paths (MPSP). The key idea of MPSP is focused on minimizing the total bandwidth allocation by identifying a protection path for each spanning path to appropriately select the parts of these protection paths that form a protection tree. In this case, the scheme proposed by the authors considered self-sharing, intrarequest sharing, and interrequest sharing. However, this algorithm did not consider the energy consumed by the network.

In [11], the authors proposed a green multicast grooming algorithm based on spanning path dedicated protection (GMG) that considered the energy consumption of the network. This scheme dynamically adjusts the link cost to search for working and protection paths to decrease energy consumption. However, this method does not share spare capacity among the different multicast connections.

III. ENERGY-EFFICIENT MULTICAST ALGORITHM

3.1. Energy Consumption

In this paper, a multicast request is accepted if both a working and a protection tree are available. The energy-efficient routing and resource allocation problem for a multicast in a survivable WDM network is formulated as follows. The given network is denoted as G (N, L), where N is the node set and L is the link set. $R_i(s_i, D_i, b_i, \Delta t_i)$ denotes the ith multicast request with required bandwidth b_i from the source node s_i to the destination node set D_i with a request holding time of Δt_i. The energy consumption of a router port, a transceiver, and an amplifier are denoted as E_r, E_t and E_a respectively. The remaining notations are defined as follows.

$l(\in L)$: a link of network G

h_l^∂ : holding time of ∂^{th} wavelength on link l

h_l : holding time of the fiber on link l

x_l^∂ : $x_l^\partial = 0$ if the state of ∂^{th} wavelength on link l is P or U; otherwise, $x_l^\partial = 1$.

x_l : $x_l = 0$ if the state of the fiber on link l is P or U; otherwise, $x_l = 1$.

A_l : The number of amplifiers on link l

WT_i : Working tree for multicast request R_i

PT_i : Protection tree for multicast request R_i

SP_i^j : The jth spanning path on WT_i

PP_i^j : Protection path for SP_i^j, where PP_i^j and SP_i^j are link-disjoint

C_l^r : Residual capacity on link l

C_l^s : Sharable spare capacity on link l

S_l : $S_l = 1$ if link l has sharable spare capacity; otherwise, $S_l = 0$.

α : Weight factor

RC_l^W : Relative link cost of l to search for the working tree

RC_l^P : Relative link cost of l to search for the protection path

AC_l : Absolute link cost of l, where AC_l is the physical length of link l

ε : Small value to encourage spare capacity sharing

w_l, p_l, m_l, u_l: Binary variables {0, 1}, $u_l = 0$ if the state of link l is U; otherwise, $u_l = 1$.

This paper assumes that a link is in one of four states: dedicated working path (w), dedicated protection path (p), mixed working and protection path (m), or unused (u). The energy consumed by devices in a p or u state can be neglected, because devices used exclusively in protection paths are switched into sleep mode in survivable optical WDM networks. The total energy required by the network is consumed by three of its parts: the transceiver, router

port, and amplifier. The number of amplifiers on a specific link l is $A_l = [d_l/d_0] + 2$, where d_l is the length of fiber link l, d_0 is the distance between two adjacent amplifiers, and the number 2 indicates there are two amplifiers allocated at the ends of a fiber link. Therefore, the energy consumed by the network (E) can be defined as Eq. (1).

$$E = 2(E_r + E_t) \cdot \sum_{l \in L} \sum_{\partial} x_l^\partial \cdot h_l^\partial + E_a \cdot \sum_{l \in L} x_l \cdot A_l \cdot h_l \tag{1}$$

3.2. Link Cost Function

When a multicast request arrives, a working tree WT_i and its corresponding protection tree PT_i, which is a set of protection paths, must be computed. To decrease energy consumption, the working and protection paths should be allocated to different fibers to increase the number of fibers in sleep mode. The working paths should be packed into fibers with more working paths, and the protection paths should be packed into fibers with more protection paths. To achieve this, two link-cost functions are defined to calculate the dynamic link-costs RC_l^W and RC_l^P to find the working and the protection paths.

Before the working tree is computed, the link-costs in the network are adjusted according to Eq. (2). If the residual capacity of a link is less than the required bandwidth, the link-cost will be set to infinity. If the link has enough residual capacity, we define the link-cost to be adaptively adjusted according to the link-state. In Eq. (2), the cost of the link with a state p increases α times, while the cost of the link with a state of w decreases α times. Therefore, we define the link cost function for a new working path so as to increase the cost of candidate links under state p and decrease the cost of the candidate links under state w. The purpose of this function is to pack the working paths into the links with more working paths to save energy.

$$\mathrm{RC}_l^W = \begin{cases} \mathrm{AC}_l[(u_l + p_l) \cdot \alpha + m_l + w_l/\alpha], & \mathrm{if}(C_l^r \geq b) \\ +\infty & \mathrm{otherwise} \end{cases} \tag{2}$$

Before the protection path PP_i^j is computed for a specific spanning path SP_i^j, the cost of the links on the spanning path is set to infinity, to satisfy the link-disjoint constraint. Then the link-costs are adjusted according to Eq. (3), and if the summation of sharable and residual capacities on a link is less than the required bandwidth, the link-cost is set to infinity to avoid selection. To increase the spare capacity that is shared, Eq. (3) sets a very small value of ϵ ($\epsilon = 0.01$) for links with sharable capacity. If the link does not have sharable capacity, the link-cost is dynamically adjusted according to its link state. In Eq. (3), the cost of a link in state w increases α times, while the cost of a link in state p decreases α times. The purpose of this function is to pack the protection paths into the links with more protection paths to use resource efficiently and reduce energy consumption.

$$\mathrm{RC}_l^P = \\ \begin{cases} \mathrm{AC}_l[(u_l + m_l + p_l)/\alpha + w_l \cdot \alpha] \cdot (1 - S_l) + \varepsilon \cdot S_l, & \mathrm{if}(C_l^r + C_l^s \geq b) \\ +\infty & \mathrm{otherwise} \end{cases} \tag{3}$$

3.3. Algorithm Description

This section uses the flowchart shown in Fig. 4 to introduce the details of the EEMA algorithm. The EEMA (Energy-Efficient Multicasting Algorithm) generates an auxiliary graph to search for the working and protection trees. If a new request arrives, the EEMA uses the FINDROUTE function to compute the working and protection trees. If a route for the multicast request is successfully found, the EEMA allocates resources to each link on the working and protection trees and updates the auxiliary graph. Here resource allocation includes both wavelength and bandwidth allocation, and the request is blocked if the EEMA cannot find a working tree or a protection tree for the multicast request.

3.3.1. Multicast Routing

This subsection explains how to select working and protection trees for a dynamic multicast request, with the objectives of reducing energy consumption and improving sharing of spare capacity. The routing procedure for the EEMA shows "FINDROUTE" in Fig. 5. When the multicast request arrives, the EEMA adjusts the link-cost according to Eq. (2) and computes a working tree WT_i, using an MPH algorithm [10]. Then, the EEMA divides the working tree into several spanning-paths, adjusts the link-cost according to Eq. (3), and uses Dijkstra's algorithm to compute a protection path for each spanning-path. Then, a minimum-cost protection tree is constructed using the Minimum-Cost Spanning Tree algorithm.
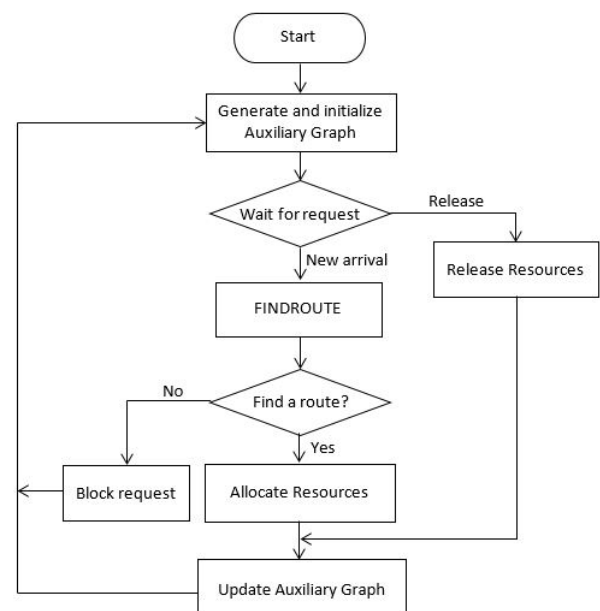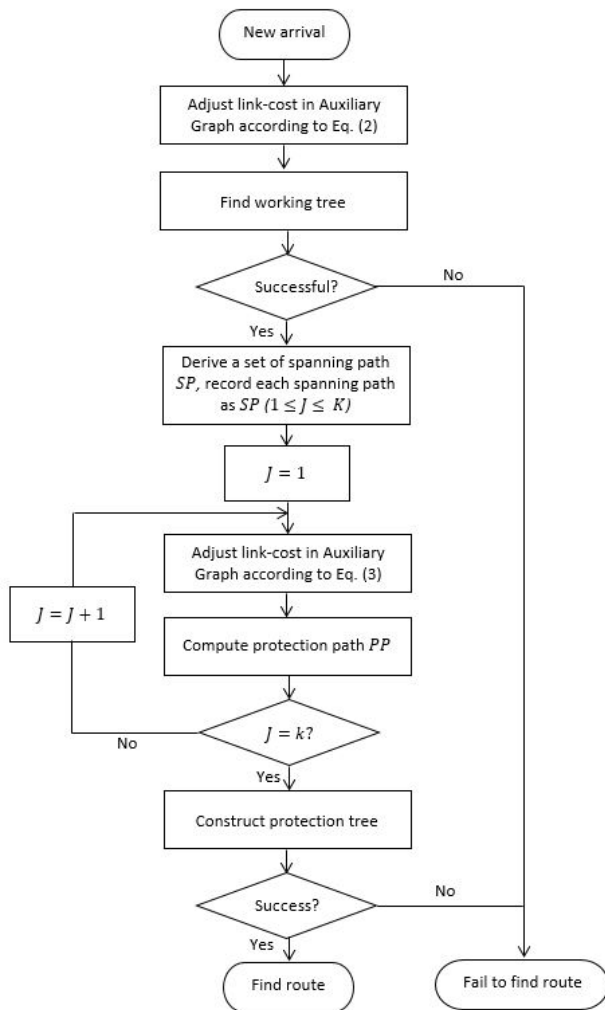

FIG. 4. Flowchart of the EEMA.

Figure 6 shows an example illustrating the routing procedure for the EEMA. Figure 6(a) shows the physical topology in which a multicast $R_1(0, \{2, 3\})$ has been established. The auxiliary graph (AG) has nodes and links that correspond to the nodes and links in the physical graph G, and AG is generated to search for the working and protection trees. Each link in AG has a link-state and a link-cost, as shown in Fig. 6(b). Assuming that the multicast request $R_2(0, \{3, 4\})$ arrives and needs to be established, the first step is to find a working-tree. The link-costs are adjusted according to Eq. (2). The result is shown in Fig. 6(c), where the weighting factor α is 8 and the working tree is $\{(0\text{-}3), (3\text{-}4)\}$, as shown in Fig. 6(d). The second step is to derive a set of spanning paths where θ is the node degree and $N_\theta = 1$ is the set of nodes on the working tree with node degree 1. Each node pair in $N_\theta = 1$ is considered as a spanning path. In this example, $N_\theta = 1 = \{0, 4\}$ and the spanning path is 0-3-4. The third step is to compute a protection-tree, formed by a set of protection paths. Here, the basic idea is to compute a protection path for each spanning path, deleting the ones that overlap, to form a minimum cost protection tree. Before searching for the protection path, the link costs in AG are adjusted and the cost of the link in the corresponding spanning path is set to infinity to satisfy the link-disjoint constraint, such as 0-3 and 3-4. Figure 6 shows the procedure for computing link cost and deciding the route of the second source and destination pair, after establishing the route of the first source and destination pair. The cost of a link in state u is initialized to 1 and the cost of a link in state p is changed to 0.125, according to Eq. (3). Finally, the EEMA decides on the least-cost path as a protection path, as shown in Fig. 6(f).
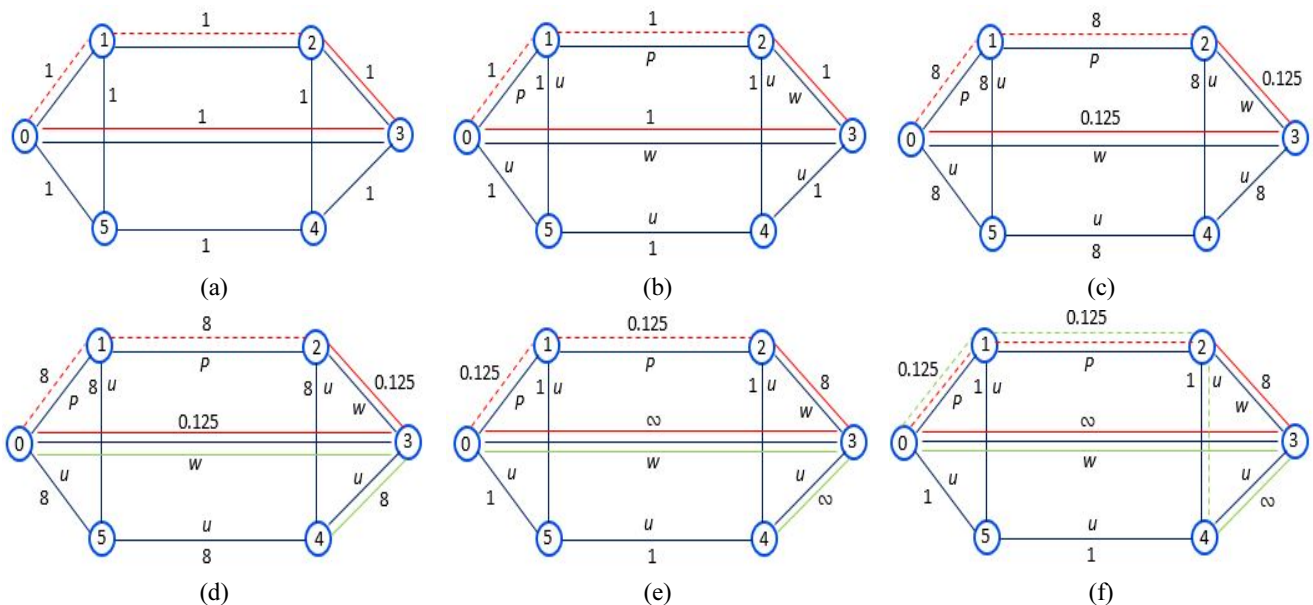

FIG. 5. Flowchart of FINDROUTE.

FIG. 6. Example illustrating the routing procedure.

3.3.2. Resource Allocation and Release

After the working and protection trees have been computed for the multicast request, the wavelength and bandwidth for both trees need to be allocated. To allocate resources for the working tree, the EEMA assigns a wavelength by using a first-fit scheme [17, 18] for each link of the tree, and then assigns the required b units of bandwidth as the working capacity. To allocate resources for the protection tree, EEMA assigns b_l^j units of bandwidth for each link of the tree. The pseudo-code to calculate b_l^j is shown in Fig. 7. The EEMA then checks whether the link has a sharable spare capacity. If the sharable spare capacity is larger than the required bandwidth, this link does not need any additional bandwidth. But the EEMA allocates $b - C_l^s$ units of bandwidth for protection capacity on the chosen wavelength if the sharable capacity is less than b.

Whenever a multicast connection drops from the network, the wavelength and bandwidth allocated to the working and protection trees need to be released. The procedure to release these resources is comprised of two parts: releasing the resources on the working tree, and releasing those on the protection tree. For the working tree, the EEMA simply releases the allocated bandwidth for each link. For the protection tree, the procedure is more complex: To release the resources on each link of the protection tree, a minimum needed-protection capacity MNC_l is calculated. Assume that l is the link of the protection tree PT that needs to be released, and that the protection capacity of PT is b units, carried by wavelength λ_k. MNC_l is calculated according to Eq. (4), where $V_l^{t'}$ is the working capacity on link l' protected by link l using wavelength λ_k. Let B_l^P be the protection bandwidth reserved on link l using wavelength λ_k. If $B_l^P - b < \mathrm{MNC}_l$, the protection capacity on link l can maintain network survivability after releasing b units of capacity, and the EEMA releases b units of protection capacity as residual capacity. If $B_l^P - b < \mathrm{MNC}_l$, the protection capacity on link l cannot be released.

$$\mathrm{MNC}_l = Max\left\{ V_l^{t'}, l \in L \right\} \tag{4}$$

3.3.3. Failure Restoration Procedure

If a link failure occurs in the working tree, the multicast request is restored using the protection paths that were derived. This subsection uses an example to describe how to restore a failed multicast request. When a link failure occurs, the working tree is divided into two parts, M1 and M2. M1 stores the nodes that are not affected by the link failure, while M2 stores the nodes that are affected. A protection path is selected to connect M1 to M2.

Figure 8(a) shows an example where the working tree is (0, {2, 5}) and the protection tree is {0-5, 5-4-2}. If link 0-1 fails, the working tree is divided into M1 = {0} and M2 = {1, 5, 2}, and the working tree is reconfigured as shown in Fig. 8(b). Figures 8(c) and 8(d) illustrate how to restore the multicast request when a failure occurs on links 1-2 and 1-5 respectively.

```
for each protection path PP_j do
  for each link l in PP_j do
    if link l is on the working tree T
        b_l^j = 0;
    else if link l is on the allocated protection path
        b_l^j = 0;
    else
        b_l^j = b - C_l^s
```

FIG. 7. Pseudocode for bandwidth allocation.
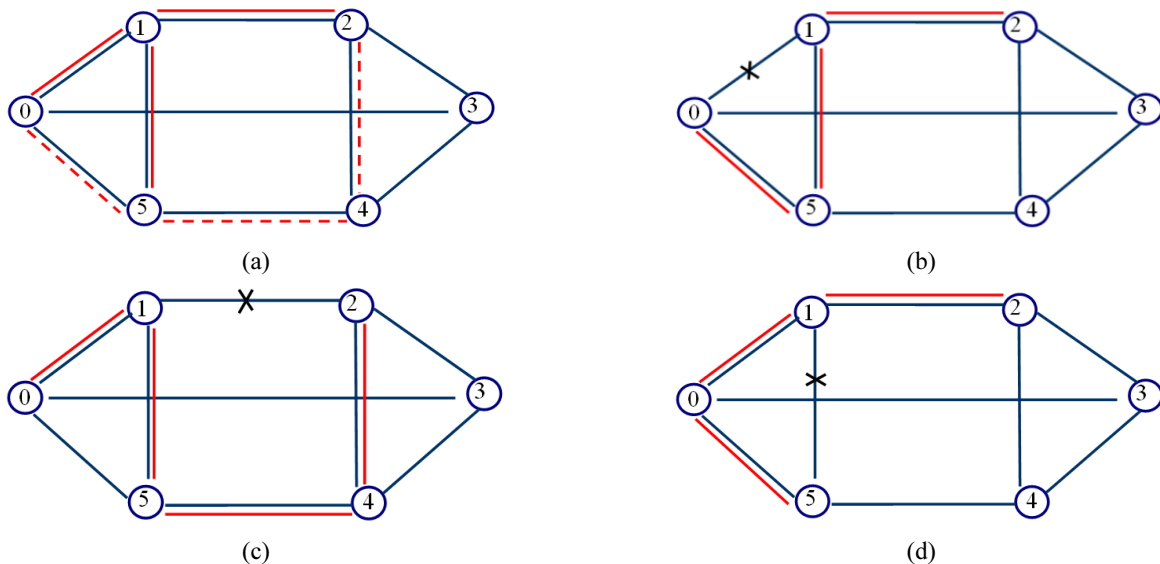

(a)

(b)

(c)

(d)

FIG. 8. Failure restoration procedure.

IV. PERFORMANCE EVALUATION

4.1. Simulation Assumptions

The proposed algorithm and conventional algorithms were evaluated over an NSF network with 14 nodes and 21 links. This paper assumes that each node has full wavelength conversion and splitting capability, each link is comprised of only one fiber, and each fiber carries two bidirectional wavelength channels. The bandwidth for each wavelength is OC-48. The idle or the protection devices in network are switched into sleep mode, and their energy consumption is negligible. The energy consumption of the active devices is set according to [11], as shown in Table 1.

10,000 multicast demands are generated according to a Poisson distribution with average arrival rate λ. The holding time of each multicast follows a negative exponential distribution with mean $1/\mu$. For simplicity, μ is set to 1. Thus, the network load becomes λ Erlang. The required bandwidth for each multicast is fixed to OC-1.

4.2. Performance Metrics

The performance of the proposed algorithm is evaluated in terms of its blocking probability, average energy consumption (AEC), and protection bandwidth ratio (PBR), as defined below.

Blocking probability: the ratio of the number of multicast requests that are blocked to the number of all multicast requests that have arrived.

AEC: the ratio of the total network energy consumption (E) to the number of serving demands (N_s).

$$AEC = E/N_s \tag{5}$$

PBR: the ratio of the bandwidth reserved for the protection path P_b to the total allocated bandwidth, as shown in (6), where W_b is the bandwidth allocated for the working path. A smaller value of PBR can be seen to indicate a higher degree of resource sharing,

$$PBR = \frac{P_b}{W_b + P_b} \tag{6}$$

4.3. Simulation Using OPNET

The simulation model is implemented using OPNET Modeler 14.5 running on Windows 7 Professional with a 2.8 GHz CPU and 4 GB of RAM. OPNET has a three-tiered hierarchy that consists of a network domain, node domain, and process domain. The node model specifies an

TABLE 1. Energy consumption of components

Component	Energy consumption (W)
Router port	1000
Transceiver	73
Amplifier	8

object in the network domain, and the process model specifies an object in the node domain. The OPNET simulation is event-driven; the simulation time advances when an event occurs. Events can be generated by BEGSIM/ENDSIM interrupts, self-interrupts, or packet arrivals. In OPNET, discrete events and continuous models are created using an object-oriented paradigm in Visual Basic, C, or Visual C++.

The EEMA is executed on a mesh network with 14 nodes and 21 links, as shown in Fig. 9. Two kinds of nodes can be seen in this figure: a control node (node 15), and 14 mesh nodes (nodes 1-14). The control node generates multicast requests and runs the EEMA algorithm. The process model for the control node in Fig. 10 includes five states: *Init, Generate_req, Routing, Allocate_resource*, and *Release_resource*. The function of each state is described as follows.

Init obtains the topology information, which includes the number of links, the link length, and the number of wavelengths per fiber.

Generate_req generates a multicast request, including its source, destination, and holding time.

Routing searches the working and protection trees for multicast requests, using the EEMA algorithm.

Allocate_resource allocates wavelength and bandwidth for each link on the working and protection trees.

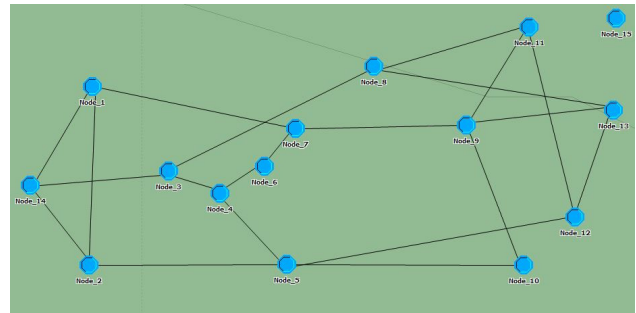*Release_resource* releases the wavelength and bandwidth of each link on the working and protection trees.
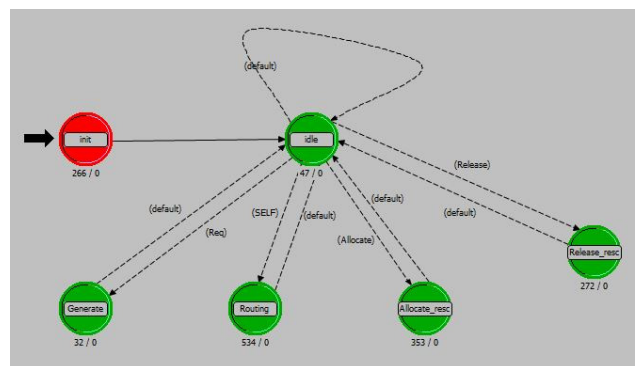

FIG. 9. Network model of the NSF network.

FIG. 10. Process model of the nodes.

4.4. Simulation Results and Analysis

Figure 11 compares the average number of sleeping fibers for MPSP (Multicast Protection Scheme through Spanning paths), GMG (Green Multicast Grooming), and the EEMA (Energy-Efficient Multicasting Algorithm). The average number of sleeping fibers for the EEMA and GMG are higher than that for MPSP, because the EEMA and GMG gather more protection paths into sleeping fibers. The number of sleeping fibers for the EEMA is higher than that for GMG because the EEMA shares the spare capacity of the links among different multicast connections.

Figures 12-14 show the average energy consumption of

the core router, transceiver, and amplifier, respectively. The AEC of the core router is 90% of the total energy consumption which implies that the core router is the main consumer of energy. The AEC of the core router, transceiver, and amplifier for EEMA are lower than those for MPSP and GMG. This is because the EEMA increases the number of sleeping fibers and switches the router ports, transceivers, and amplifiers on those fibers into sleep mode to reduce energy consumption.

Figure 15 presents the average energy consumption for MPSP, EEMA, and GMG. The AEC for the EEMA can be seen to be lower than those for MPSP and GMG at all times. The EEMA gathers the working and protection paths into different fibers to increase the number of sleeping fibers and the spare capacity that is shared. The EEMA can save up to 26% and 12% of energy, compared to MPSP and GMG, respectively, and the energy savings is larger under higher traffic load. This implies that the number of shared links for the EEMA increases under a heavy traffic load.

Figure 16 shows the average number of hops for MPSP, EEMA, and GMG. The average number of hops for the EEMA and GMG are higher than that for MPSP because
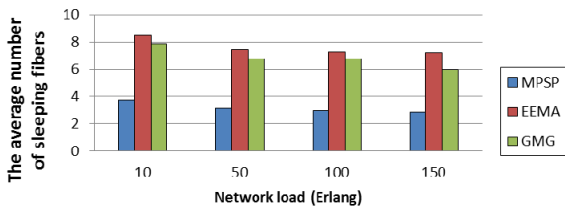

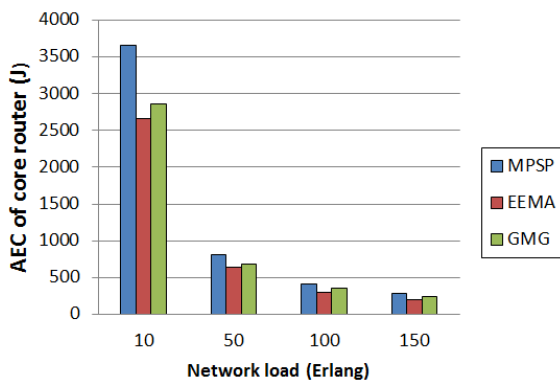FIG. 11. Comparison of the average number of sleeping fibers.

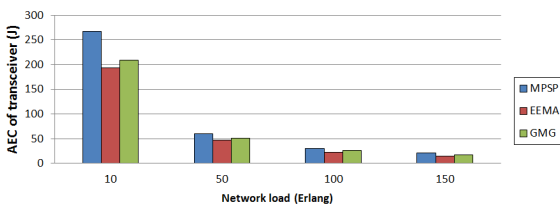FIG. 12. Average energy consumption of the core router.

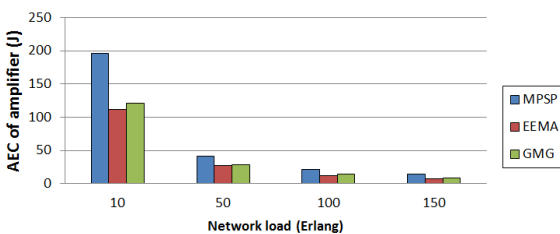FIG. 13. Average energy consumption of the transceiver.

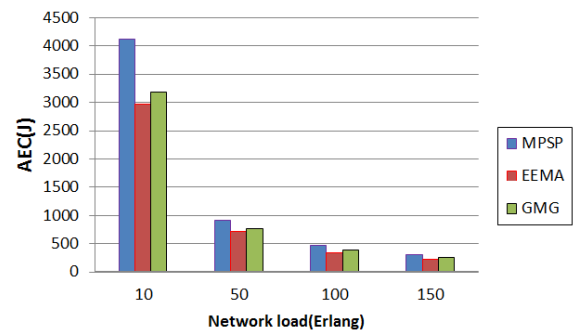FIG. 14. Average energy consumption of the amplifier.

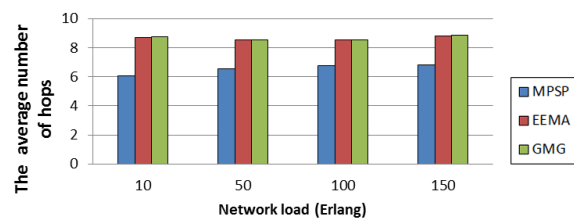FIG. 15. Comparison of average energy consumption.

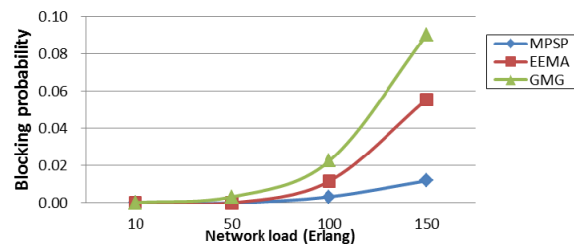FIG. 16. Comparison of average number of hops.

FIG. 17. Comparison of blocking probability.

the EEMA and GMG choose a route with more active fibers as the working path, and choose a route with more sleeping fibers as the protection path, while MPSP always chooses the shortest paths. Figure 17 clearly shows that the blocking probability for the EEMA is higher than that for MPSP, because the protection paths of EEMA are allocated to longer routes that use sleeping fibers, as shown in Fig. 16. Figure 17 shows that the blocking probability for the EEMA is lower than that for GMG. The EEMA considers interrequest sharing to increase resource utilization. As shown in Fig. 18, the PBR for the EEMA is lower than that for GMG, which indicates that the EEMA has a higher degree of capacity sharing. Figures 19 and 20 compare the AEC and blocking probability of the EEMA based on the weight factor. As the figures suggest, with an increase in weight factor, the AEC decreases, while the blocking probability increases. Since more protection paths are collected into sleeping fibers, more devices enter the sleep state.
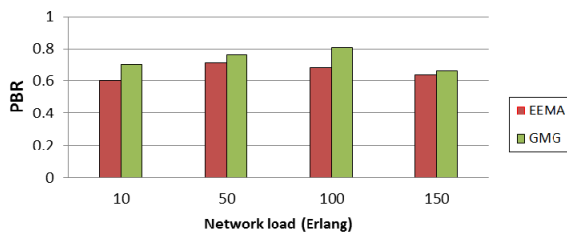
V. CONCLUSION

In this paper, we have proposed an energy-efficient multicast algorithm (EEMA) to achieve efficient routing and bandwidth allocation in survivable WDM networks. The working and protection paths of a multicast request are selected using two link-cost functions, defined by taking into account energy consumption. The proposed link-cost functions allow EEMA to pack working and protection paths into different fibers and switches, so that more components enter sleep mode. The proposed algorithm was compared to the conventional multicast algorithms MPSP and GMG. The results showed that the EEMA increased both the number of sleeping fibers and the amount of spare capacity. These resulted in decreasing energy consumption. Compared to conventional MPSP and GMG, the EEMA saved up to 26% and 12% of the energy consumed, respectively.

FIG. 18. Comparison of protection-bandwidth ratio.
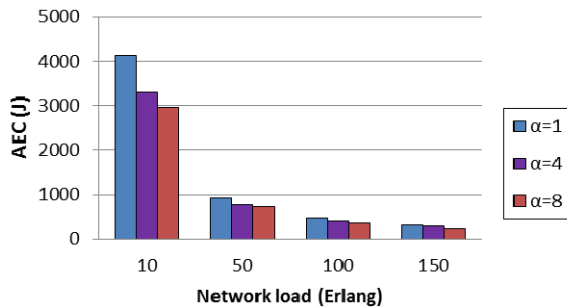
FIG. 19. AEC according to α.
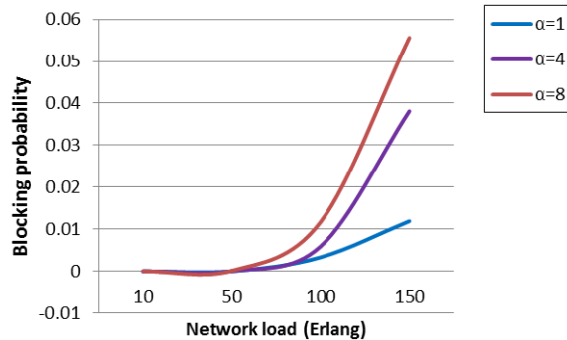
FIG. 20. Blocking probability according to α.

REFERENCES

1. M. Ali and J. S. Deogun, "Power-efficient design of multicast wavelength-routed networks," IEEE J. Sel. Areas Commun. **18**, 1852-1862 (2000).
2. B. Li and Y. C. Kim, "Efficient logical topology design considering multi-period traffic in IP-over-WDM networks," J. Opt. Soc. Korea **19**(1), 13-21 (2014).
3. J. Baliga, K. Hinton, and R. Tucker, "Energy consumption of the Internet," Conference on the Optical Internet (COIN), Australian Conference on Optical Fibre Technology (ACOFT), Melbourne, Australia (2007).
4. A. Jirattigalachote, C. Cavdar, P. Monti, L. Wosinska, and A. Tzanakaki, "Dynamic provisioning strategies for energy efficient WDM networks with dedicated path protection," Opt. Switching Networking **8**(3), 201-213 (2011).
5. N, H. Bao, L. M. Li, H. Yu, Z. Zhang, and H. Luo, "Power-aware provisioning strategy with shared path protection in optical WDM networks," Opt. Fiber Technol. **18**(2), 81-87 (2012).
6. F. Musumeci, M. Tornatore, J. L. Vizcaíno, Y. Ye, and A. Pattavina, "Energy-efficiency of protected IP-over-WDM networks with sleep-mode devices," J. High Speed Networks **19**, 19-32 (2013).
7. R. He and B. Lin, "Dynamic power-aware shared path protection algorithms in WDM mesh networks," J. Commun. **8**(1), 55-65 (2013).
8. C. Boworntummarat, L. Wuttisittikulkij, and S. Segkhoonthood, "Light-tree based protection strategies for multicast traffic in transport WDM mesh networks with multi-fiber systems,"

IEEE ICC 04 **3**, 1791-1795 (2004).

9. N. K. Singhal, L. H. Sahasrabuddhe, and B. Mukherjee, "Provisioning of survivable multicast sessions against single link failures in optical WDM mesh networks," J. Lightwave Technol. **21**, 2587-2594 (2003).

10. H. Luo, L. Li, H. Yu, and S. Wang, "Achieving shared protection for dynamic multicast sessions in survivable mesh WDM networks," IEEE J. Sel. Areas Commun. **25**(9), 83-95 (2007).

11. C. Yu, Y. Liu, W. Hou, Y. Yu, J. Wu, and P. Jiang, "A new green multicast grooming protection algorithm in WDM optical networks," Optik - International Journal for Light and Electron Optics **125**(2), 657-662 (2014).

12. L. H. Sahasrabuddhe and B. Mukherjee, "Light-trees: optical multicasting for improved performance in wavelength-routed networks," IEEE Commun. Mag., February (1999).

13. M. R. Garey and D. S. Johnson, "The NP-completeness column: An ongoing guide," J. Algorithms in Cognition, Informatics and Logic **64**, 125-192 (2009).

14. L. Kou, G. Markowsky, and L. Berman, "A fast algorithm for Steiner trees," Acta Inf. **15**, 141-145 (1981).

15. K. A. Dowsland, "Hill-climbing simulated annealing and the Steiner problem in graphs," Eng. Optim. **17**, 91-107 (2007).

16. C. C. Ribeiro and M. C. De Souza, "Tabu search for the Steiner problem in graphs," Networks **36**, 138-146 (2000).

17. J. Zhou and X. Yuan, "A study of dynamic routing and wavelength assignment with imprecise network state information," International Conference on Parallel Processing Workshops, 207-213 (2002).

18. S. Kaur and R. Kaler, "Ultrahigh speed reconfigurable logic operations based on single semiconductor optical amplifier," J. Opt. Soc. Korea **16**(4), 432-442 (2012).