INSTITUTE OF POLICY AND PLANNING SCIENCES

Discussion Paper Series

No. 1073

Optimal Multicast Routing Using Genetic Algorithm
for WDM Optical Networks

by
Johannes Hamonangan Siregar, Yongbing Zhang,
and Hideaki Takagi

January 2004

UNIVERSITY OF TSUKUBA
Tsukuba, Ibaraki 305-8573
JAPAN

Optimal Multicast Routing Using Genetic Algorithm for WDM Optical Networks

Johannes Hamonangan Siregar
Doctoral Program in Policy and Planning Sciences
University of Tsukuba
Tsukuba-shi, Ibaraki 305-8573, Japan
E-mail: siregar@sk.tsukuba.ac.jp

Yongbing Zhang
Institute of Policy and Planning Sciences
University of Tsukuba
Tsukuba-shi, Ibaraki 305-8573, Japan
E-mail: ybzhang@sk.tsukuba.ac.jp
and
Hideaki Takagi
Vice President, University of Tsukuba
Tsukuba-shi, Ibaraki 305-8577, Japan
E-mail: takagi@sk.tsukuba.ac.jp

Abstract

We consider the multicast routing problem for large-scale wavelength division multiplexing (WDM) optical networks where transmission requests are established by point-to-multipoint connections. To realize multicast routing in WDM optical networks, some nodes need to have light (optical) splitting capability. A node with splitting capability can forward an incoming message to more than one output link. We consider the problem of minimizing the number of split-capable nodes in the network for a given set of multicast requests. The maximum number of wavelengths that can be used is specified a priori. A genetic algorithm is proposed that exploits the combination of alternative shortest paths for the given multicast requests. This algorithm is examined for two realistic networks constructed based on the locations of major cities in Ibaraki Prefecture and those in Kanto District in Japan. Our experimental results show that the proposed algorithm can reduce more than 10% of split-capable nodes compared with the case where the split-capable node placement optimization is not performed while the specified number of wavelengths is not exceeded.

1 Introduction

In the foreseeable future, the network traffic may grow dramatically due to multimedia applications such as tele-conferencing, distributed database access, distance learning and Internet TV. These applications may need wide bandwidth and high quality of services. A wavelength division multiplexing (WDM) optical network offers a great potential for future high speed applications in large-scale networks because of its wide bandwidth and high-speed data transmission. Data transmission at multiple carrier wavelengths over a single fiber is available using WDM technology [1, Sect. 1.3.1]. One wavelength is dedicated to each channel between two adjacent nodes on a single fiber link. The path that traverses a series of links from a source to a destination is established by using an all-optical route over which a set of available wavelengths is used.

An optical communication path between a pair of a source and a destination is called the *lightpath*, and it may span multiple fiber links. Without wavelength conversion capability, a lightpath must use the same wavelength on all the fiber links through which it traverses; this property is known as the *wavelength-continuity constraint* [2]. In order to establish a lightpath between a source and a destination, one needs to determine the path from the source to the destination and then assign a wavelength to the path. By adding the wavelength conversion capability to a node, an incoming wavelength to the node can be converted to a different outgoing wavelength and therefore the wavelength-continuity constraint is relaxed.

A lightpath is viewed as a *point-to-point* light connection from a source to its destination. The concept of a lightpath can be extended to a *light-tree* whereby a point-to-multipoint connection is set up using a single or multiple wavelengths [3]. The data transmission from one source to a single destination is known as *unicast*, while the transmission from one source to all nodes is known as *broadcast*. Between these two extremes lies *multicast*, which refers to the data transmission targetting to a selected set of destinations [4, Sect. 8.1]. Let us consider a given set of multicast requests using the unicast lightpaths. A large number of lightpaths should be used because the number of lightpaths required for each multicast request equals to the number of its destinations. By introducing some nodes with wavelength splitting capability and constructing a multicast light-tree for each multicast request, the network performance can be improved. A *split-capable node* (or simply *split node*) is a node with splitting capability that can forward an incoming message to more than one output link. However, the implementation of a split node may be expensive due to the large amount of amplification and fabrication [5, Sect. 2.2].

In this paper, we focus on the multicast routing problem and attempt to place the split nodes in an efficient way so as to minimize the number of split nodes for a given set of multicast requests. The problem of the split node placement is a combinatorial optimization and the approaches for this problem can be categorized as *exact* or *heuristic*. Exact algorithms use exhaustive search by checking all feasible solutions and therefore they are impractical. On the other hand, heuristic algorithms are based on probablistic performance with computational results such as random placement of split nodes. Heuristic algorithms provide practical solutions even though they may not be strictly optimal.

In this paper, we first employ a minimum spanning tree algorithm to construct a light-tree for each multicast request. We then use a genetic algorithm to minimize the number of split nodes by examining the alternative shortest paths for each destination. Genetic algorithms (GAs) have been applied to various optimization problems. In general, a GA uses a mechanism of natural selection from biology concept [6, 7, 8]. This is an evolution of individuals from one generation to the next, based on the elimination of weak individuals and the reproduction of strong individuals. The individuals are analogous to the possible solutions of the problem. Strong individuals are related to the nearly optimal solutions that we search in optimization problems. These individuals will survive over a number of generations until the strongest one remains, that is, an optimal solution of the problem is obtained. For reproducing individuals in the population, genetic operators such as selection, crossover, and mutation are used. These operators will explore more combinations of individuals which may lead to an optimal solution of the problem.

The rest of the paper is organized as follows. In section 2 we review some previous approaches to the optical multicast routing problem. Section 3 contains a detailed formulation of the network model and the split node placement problem. The implementation of GA for the split node placement problem is given in section 4. Results of experiments with GA are presented in section 5. In section 6 we make some concluding remarks.

2 Related Work

Recently, various approaches have been developed to address the optical multicast problems. Most work is finding the minimum number of wavelengths used for the given set of multicast requests. A survey of the optical multicast is given in [9, 10]. Malli et al. [11] present the efficiency of multicasting over unicasting in all-optical WDM networks based on the number of wavelengths.

They assume a subset of nodes supporting the splitting capability and construct a multicast tree algorithm for a given set of sources and destinations. Then the nodes on each tree are assigned the same wavelength. Sahasrabuddhe and Mukherjee [3] formulate the multicast problem as an optimization problem with one of two possible objective functions: for a given traffic matrix, (i) minimize the network-wide average message hop distance, or (ii) minimize the total number of transceivers in the network. They solve the problem as a mixed-integer linear programming problem. Li et al. [12] consider the problem of constructing a routing tree with minimal number of wavelengths on the tree and they propose an approximation algorithm. Zhang et al. [13] propose four WDM multicast routing algorithms for the problem where the splitting capability of the nodes is given and the performance criterion is the number of wavelengths used.

In designing a WDM optical network that can support multicast, the issue to provide enchanced performance becomes important. Ideally the design must balance several conflicting performance criteria: minimize the number of hops traversed; minimize the number of splitters at each node in the network; minimize the blocking probability; maximize the power splitter utilization; minimize the optical power budget; minimize the number of wavelengths by solving the routing and wavelength assignment problem, and so on. The optical splitter is a key component to realize multicast in optical networks [14]. Therefore, the use of nodes with such splitting capability affects the performance of approaches developed for the optical multicast problems.

There have been several studies of the optimization of splitter placement in optical multicast networks. Splitters are placed in distinct nodes because the number of splitter is limited due to amplification and economic reasons.

Ali [5, 15] considers the splitter placement problems for static and stochastic traffic in wavelength-routed networks. In [5, Sect. 3.2], the traffic between nodes is static and the objective is to maximize the number of multicasts in the network by (i) finding the best combination of (tree, wavelength) pair for each multicast, and (ii) allocating a fixed number of split nodes in the network. A genetic algorithm is employed to provide the fast near optimal solution. In [15] the traffic between nodes is stochastic and the objective is to minimize the blocking probability. The blocking performance is used to guide an iterative algorithm for the placement of split nodes, where the total number of split nodes to be placed in the network is fixed.

3 Network Model and Formulation

We introduce a network model for the split node placement problem using graph theoretic terminology. Consider an undirected bipartite graph $G = (V, L)$, where V is the set of vertices representing the network nodes and L is the set of undirected edges representing the bi-directional fiber links in the network. Let $|L|$ denote the number of links and $N = |V|$ denote the number of nodes in the network. The maximum number of neighboring nodes of a node in the network is denoted by $B < N$. The bi-directional link between nodes u and v is denoted by l_{uv}. It is assumed that each link in the network has the same number of wavelengths denoted by W. A multicast request i is denoted by m_i and the set of destination nodes of m_i is denoted by D_i. The source, the number of destinations, and the selected destination of a multicast request m_i are denoted by $s_i, |D_i|$, and d_{ij} $(1 \leq j \leq |D_i|)$, respectively. A given set of multicast requests is denoted by $M = \{m_1, m_2, \ldots, m_{|M|}\}$, where $|M|$ denotes the number of multicast requests. It is assumed that the set of

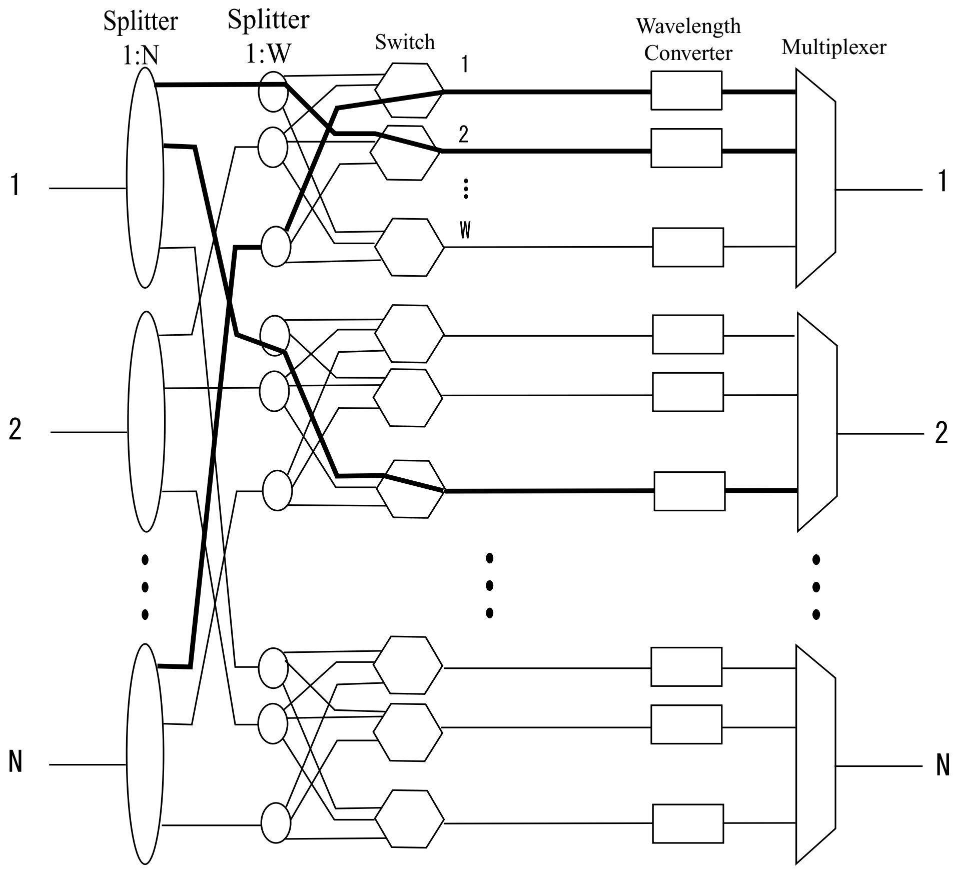


Figure 1: The split node architecture.

multicast requests in the network are given in advance.

The nodes in the network are classified into two categories, *split-capable* or *split-incapable*. The former can forward the incoming message to only one outgoing link. On the other hand, the latter can split the incoming signal into multiple outgoing links. The architecture of a split node is shown in Fig. 1, where an input signal can be split into BW signals. Wavelength conversion may be performed in order to avoid the wavelength conflict. When a node works as a split node, an input signal is split at the splitter of the incoming link and then forwarded to appropriate output links.

Let us consider a lightpath for multicast request m_i from source s_i to destination d_{ij}. Suppose that a lightpath from s_i to d_{ij} consists of successive links $l_{s_i v_1}, l_{v_1 v_2}, \ldots, l_{v_n d_{ij}}$, where $v_1, v_2, \ldots, v_n$ are the intermediate nodes between s_i and d_{ij} along the path. If the lightpath does not include a split node, the same wavelength must be used through those links. Let w_k $(1 \leq k \leq W)$ denote a wavelength assigned on the lightpath of multicast request m_i from s_i to d_{ij}. The lightpaths from s_i to the rest of destination nodes can be established using the same wavelength w_k only if they have no common links with other existing lightpaths. Otherwise, another wavelength w_{k+1} should be used. If a lightpath includes a split node then an incoming wavelength w_k at the split node may convert to a different wavelength w_p $(1 \leq p < k)$ which is not used by other existing lightpaths. Another wavelength w_{k+1} should be used if all the diferent wavelengths are already used by other existing lightpaths. By using split nodes, the number of required wavelengths can be decreased.

For each multicast request, we first apply a minimum spanning tree algorithm [16] based on the least number of links from the source to the destinations. We then mark the nodes that require the splitting capability. Every branching node in this spanning tree is a split node. In order to minimize the number of split nodes, a straightforward way is to examine repeatedly all the alternative shortest paths to each destination from each source to see whether the number of split nodes can be reduced. However, this method is impractical because there may be too many alternative shortest paths for each source-destination pair, especially in a large-scale network.

An example of the multicast routing is shown in Fig. 2, where the multicast request m_1 is from source s_1 to three destination nodes d_{11}, d_{12}, and d_{13}. A light-tree can be found using a minimum spanning tree algorithm as shown in Fig. 2 (a). It needs three lightpaths each of which uses a distinct wavelength

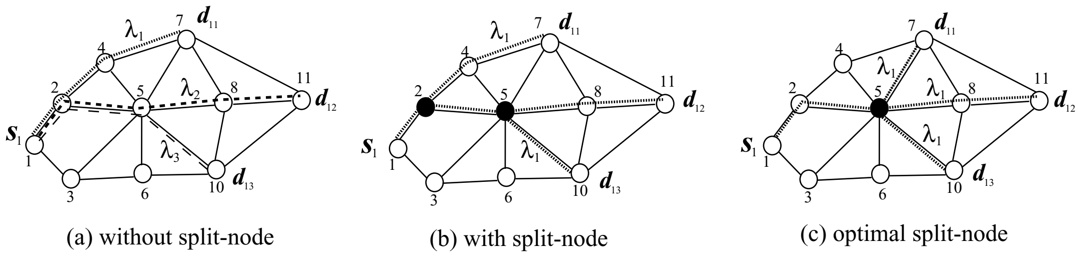


Figure 2: An example of the multicast routing without and with split nodes.

to send messages from s_1 to d_{11}, d_{12}, and d_{13} if there is no split node. On the other hand, if split nodes are used at nodes 2 and 5 denoted by black circles in Fig. 2 (b), the signal from s_1 can be split at these nodes. In this case, it needs only one wavelength. Furthermore, if we perform the placement optimization for the split nodes, only one split node (node 5) is needed as shown in Fig. 2 (c).

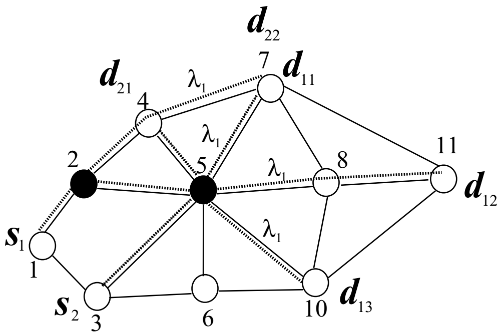
When the set of multicast requests is established, we can use the wavelength assignment algorithm to minimize the number of wavelengths in the light-tree. Using the minimal number of split nodes may cause the number of wavelengths to increase. However, we choose to construct multicast routing trees with minimal number of split-nodes rather than with minimal number of wavelengths. This is because a large number of wavelengths can be available nowadays. The maximum number of wavelengths used for our genetic algorithm is set to twice the number of wavelengths required in the initial spanning trees.

4 Genetic Algorithm

In this section, we present a genetic algorithm (GA) for the split node placement problem. In our GA, each individual in the population represents all the possible paths for all of the multicast requests. A path denotes a series of links from a source to its each destination and should be one of the shortest paths between the source and the destination. We create the initial population randomly. Feasible solutions are generated by repeating appropriate genetic operators such as crossover and mutation. With constraint handling, our GA selects only feasible solutions. When the stopping condition is satisfied, we find the solution to the problem even though it may not be strictly optimal.

4.1 Representation of an Individual

An individual in the population consists of all the paths for the given set of multicast requests. A path is represented by a series of node identification numbers along the path from a source to its destination.



(a) Network

multicast request	*(s,d)* pair	shortest path
m_1	(1,7) (1,11) (1,10)	1-2-4-7 1-2-5-8-11 1-2-5-10
m_2	(3,4) (3,7)	3-5-4 3-5-7

(b) Multicast requests

1	2	4	7	1	2	5	8	11	1	2	5	10	3	5	4	3	5	7

(c) Individual string

Figure 3: Example of an individual string.

Figure 3 shows an example of how to represent an individual. In this example, there are two multicast requests, one with three destinations and the other with two destinations.

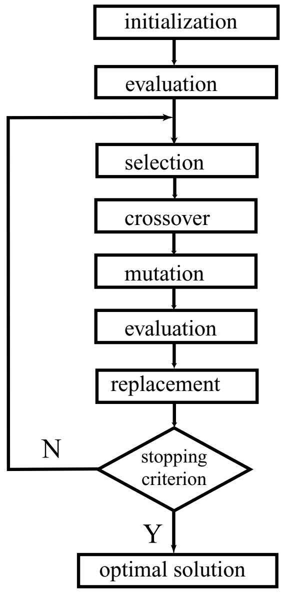


Figure 4: Flowchart of the GA procedures.

4.2 Procedures of the Proposed GA

Figure 4 shows the flowchart of the GA for searching the optimal combination of the paths for a given set of multicast requests with the minimal number of split nodes. In the following, we describe each procedure of the GA in detail

Initialization: Each individual of the population is a possible solution to the problem. The GA starts with an initial population which is generated from a random seed. It selects an arbitrary combination of shortest paths from the table of shortest paths. The table is computed in advance.

Evaluation of fitness value: The objective function to optimize is the number of split nodes. This evaluation is based on the number of common nodes on the paths of multicast requests. To maintain the uniformity over the problem domain, we use a fitness value for the objective function normalized to a convenient range. The normalized objective function indicates the fitness of an individual that the selection uses to evaluate. Individuals with good fitness value will be selected for the next generation.

Selection: The tournament selection method is used for selecting two parents to produce new individuals for the next generation [6, p. 121]. This selection leaves only those individuals with the highest fitness values in the population.

Crossover: Crossover explores the diversity in the individuals of the population. We use the path crossover operator for exchanging the subroutes between two individuals [17, p. 355]. The paths of requests in the individuals must have the same source and destination nodes to apply the crossover. Crossing sites for the path crossover operator are limited to the nodes contained in both individuals. The probability of crossover is given in the parameter setting. Using the random procedure, the crossing site is selected. For example, two individuals, parent 1 and parent 2, for two paths of requests are selected to apply the path crossover operator as shown in Fig. 5. There are two potential crossing sites, 7 and 16, and one of them, 7, is selected to be the crossing site. Two children, child 1 and child 2, are generated by exchanging the subroutes of their parents after node 7.

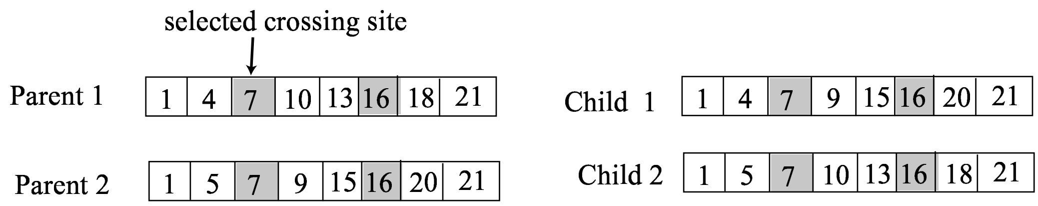


Figure 5: Example of path crossover.

Mutation: The mutation procedure is applied after the crossover on each child independently. Every child gets an opportunity for changing each node according to the probability of mutation. This probability is also given in the parameter setting. The mutation plays a role to restore lost genetic values when the population converges too fast [6, p. 14]. The fitness value is evaluated for the individual after mutation.

Replacement: The new individual replaces the old individual or their parent in order to maintain a fixed population size if the fitness value of the new individual is higher than that of their parent. If the fitness value of the new individual is lower than that of their parent, the new individual is discarded and the next generation employs the old individual.

4.3 Handling Constraints

In our GA, the initialization, crossover, and mutation procedures breed new individuals with new fitness values. If the solutions do not satisfy the constraint conditions, they are infeasible. It is a path that cannot be realized through a physical link in the network. We avoid producing infeasible solutions in the initialization, crossover, and mutation procedures. By using the path representation for the set of multicast requests in the initialization procedure, only feasible solution is produced. In the crossover procedure only feasible solution is produced too. By the path crossover operator, a new path may be realized in the network. In the mutation procedure, it may produce an infeasible solution by changing a node to a different one. We then discard such solution and do not perform that mutation procedure.

4.4 Stopping Criteria

A disadvantage in the optimization with GA is the difficulty in deciding when to stop. Although the statistical variables such as average and best fitness values are available in each generation, their values change almost unexpectedly as the generations evolve. Stopping after a certain number of iterations with no improvement or when the change in the average fitness is small may cause the algorithm to stop too early or too late.

Another stopping criterion may be that if the average fitness attains the value we expect then the iteration stops. In this case, the number of iterations may become large and the long computation time may be needed. In this paper, we have used a stopping criterion based on the number of generations. Our algorithm stops when the generation counter exceeds the preset maximum number of generations.

5 Experimental Results

In this section, we show some experimental results obtained by applying the GA to two practical networks. The network models are constructed by referring to the Nippon Telegraph and Telephone Corporation (NTT) service network. Some major cities in an area in Japan, which are extracted from NTT's public web site (http://www.ntt-east.co.jp/tariff/ryoukin/), are selected as the network nodes. Then, by the Delaunay triangulation [18, p. 175] for the set of node locations, we get a fictitious set of links in the network. *Note that the resulting networks are by no means related to the real networks of NTT or any other companies.* Given the network topology, we have applied Dijkstra algorithm [18, p. 273] to determine the shortest path for every pair of the source and destination nodes.

5.1 Ibaraki Network

We first consider a network consisting of 14 nodes that represent major cities in Ibaraki Prefecture, which we call Ibaraki Network for the sake of convenience, as shown in Fig. 6. In our experiment, the sets of connection requests are randomly generated.

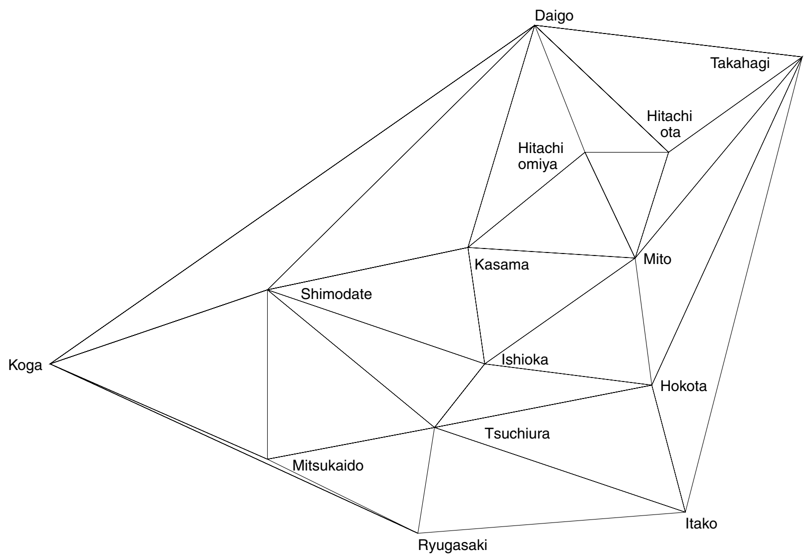


Figure 6: Configuration of the Ibaraki Network.

The GA parameters which we use for the Ibaraki Network are as follows:

- Population size = 20
- Mutation probability = 0.00333
- Crossover probability = 0.6
- Maximum number of generations = 50

The results are presented in Fig. 7, which shows the performance of the worst, best and average solutions in the population. The x-axis represents the

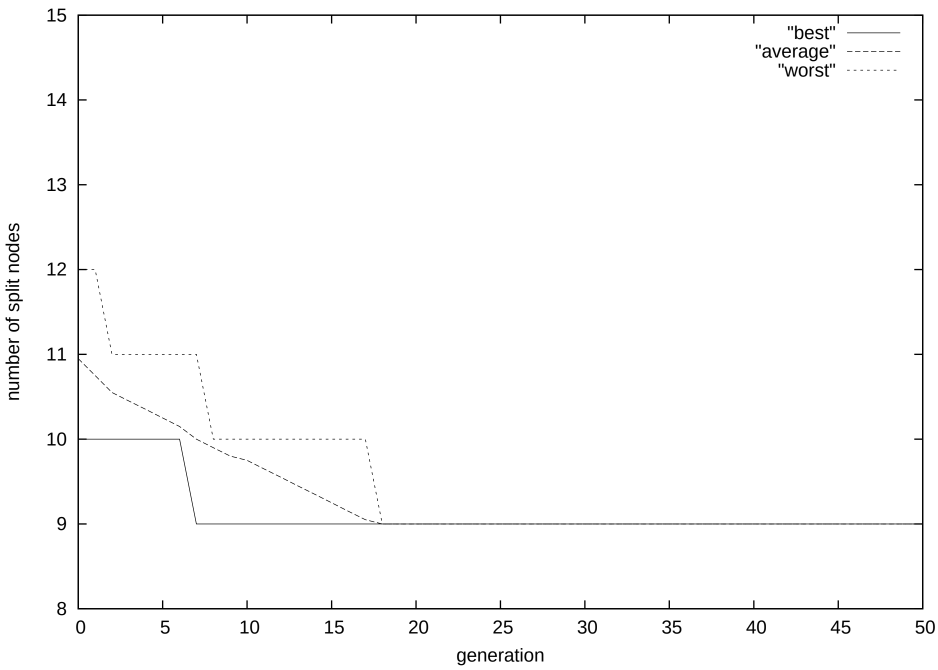


Figure 7: GA performance for the Ibaraki Network.

number of generations and the y-axis is the value of the objective function, i.e., the number of split nodes. It can be seen that the initial generation starts with the average number of split nodes 11, the worst 12, and the best 10. The minimum number of wavelengths in the intial generation is 4, so that the maximum number of wavelengths in GA is set to 8. The GA improves the individuals' values (the number of required split nodes) by exploiting the search space with mutation. Then in generation 19, all individuals exhibit the same value, and we judge that the optimal solution is reached with the value 9. The maximal number of wavelengths used in the last generation is 5, which does not exceed twice the initial number of wavelengths. It can be seen that the GA makes only small improvement to find the number of split nodes, because alternate paths in the Ibaraki Network are limited.

5.2 Kanto Network

We next examine a network that consists of 82 nodes representing major cities in the Kanto District, which we call Kanto Network, as shown in Fig. 8. A total of 20 multicast requests, each of which has a selected group of destinations with the number of destinations between 4 and 40. The source and destinatons are selected randomly for each multicast request. The simulation run has been repeated 1000 times.

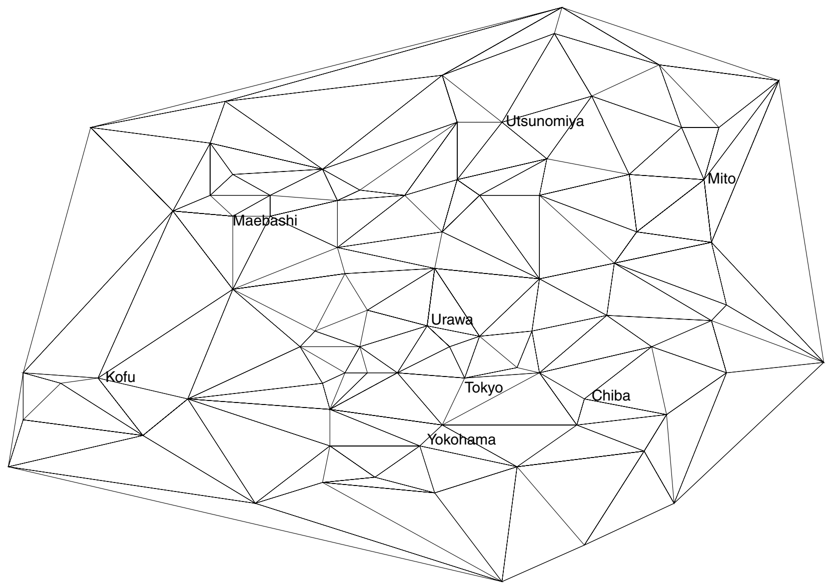


Figure 8: Configuration of the Kanto Network.

The GA parameters which we use for the Kanto Network are as follows:

- Population size = 50
- Mutation probability = 0.00333
- Crossover probability = 0.6
- Maximum number of generations = 200

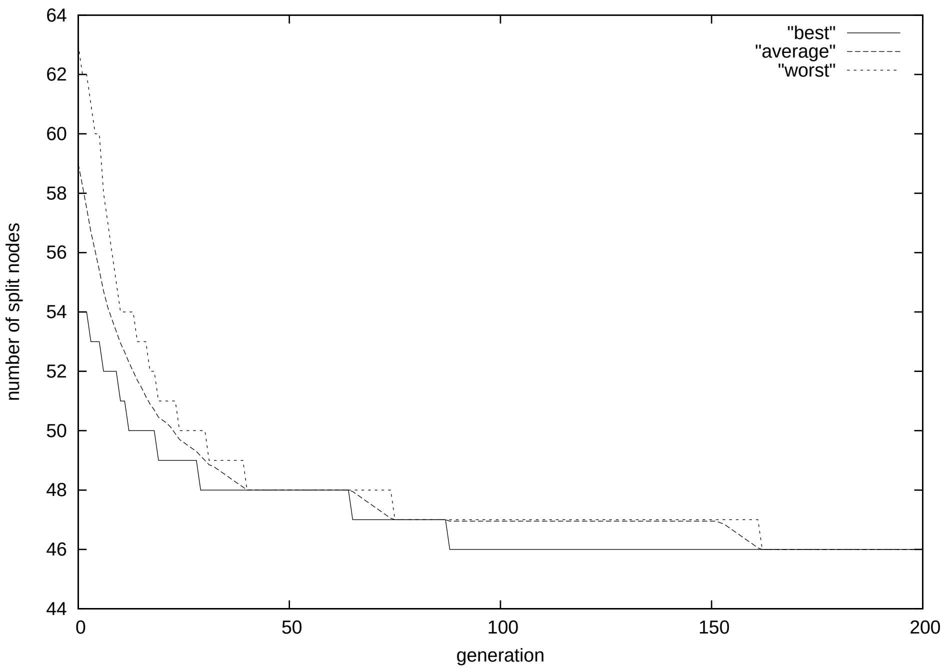


Figure 9: GA performance for the Kanto Network.

In Fig. 9 we show the number of split nodes required for a given set of multicast requests in a single simulation run, where the generation starts with the average value 59, the worst value 63, and the best value 54. The minimum number of wavelengths in the intial generation is 12, so that the maximum number of wavelengths in GA is set to 24. During generations from 40 to 64, the GA comes to the premature convergence at the value 48. However, after generation 64, the GA improves the individuals' values by further exploitating the search space of solution with mutation procedure. Again during generations from 75 to 87, the GA comes to the premature convergence at the value 47. In generation 161, the GA finds the solution at the value 46, even though it may not be optimal. The maximal number of wavelengths used in the last generation is 16, which does not exceed twice the initial number of wavelengths.

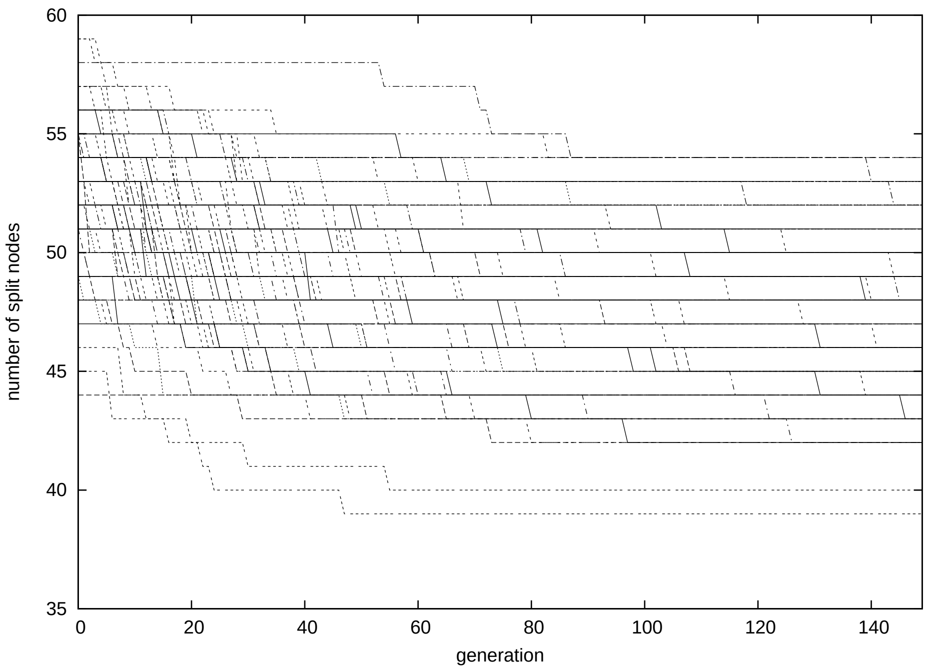


Figure 10: Best performance of 100 simulation runs for the Kanto Network.

In Fig. 10 we show the best value of the number of split nodes obtained from 100 simulation runs. They have been selected from 1000 simulation runs as those that can clearly depict the results. Each simulation generates a non-increasing curve (dashed line). Similar results are obtained for other runs, and they together appear as a solid line. It can be seen that our proposed GA works very efficiently and yields better placement of split nodes. In most cases our GA converges within 100 generations. It can also be seen that the reduction in the number of required split nodes ranges from 10% to 30% of the initial value.

The average number of the required split nodes (solid line) with 95% confidence interval (dotted lines) is shown in Fig. 11 by using 1000 simulation runs. It can be remarked that our GA estimates the average number of split nodes and that the 95% confidence interval band is quite stable.

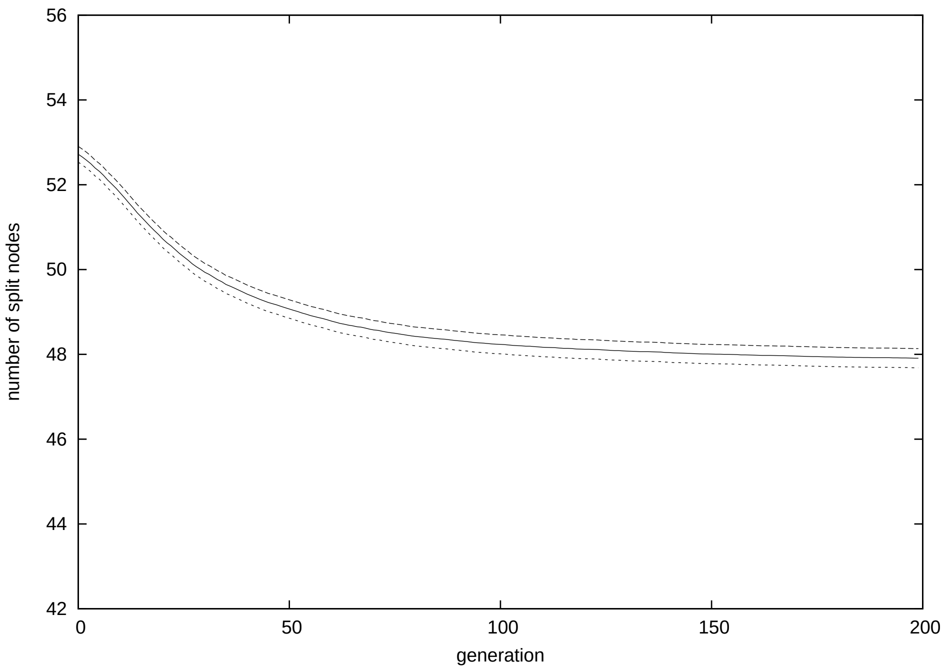


Figure 11: Average number of split nodes with 95% confidence interval for the Kanto Network.

6 Conclusion

The purpose of our GA method is to produce a possibly optimal solution for a certain population of individuals in a limited number of generations. This method has been tested to obtain the optimal split node placement for the Ibaraki and Kanto Networks constructed from the NTT office locations.

A simple solution to this problem is employing the exhaustive search. There is a major difference between the GA and the exhaustive search algorithm in the search space for the optimal solution. The exhaustive search must compare all combinations of the shortest paths. By using the minimum spanning tree algorithm, the multicast tree can be constructed to find a possible split node placement. However, for large networks where the number of combinations grows, it needs explosive computational time for searching the optimal solution.

In this paper, we have proposed a GA that restricts trials to feasible solutions. By using appropriate probabilities of crossover and mutation, the diversity in the population is obtained and the solutions converge mostly in a limited number of generations. Our experimental results have shown that the reduction in the number of split nodes required for a given set of multicast requests can be more than 30% in the best case while the number of required wavelengths does not grow as much.

Acknowledgement

The authors wish to thank Professor Kounosuke Kawashima of Tokyo University of Agriculture and Technology for providing them with the web site URL of the NTT office locations. This work is supported in part by the University of Tsukuba under the University Research Projects, Research Grant (A).

References

[1] R. Ramaswami and K. N. Sivarajan, *Optical Networks: A Practical Perspective*, Second Edition, San Francisco: Morgan Kaufmann, 2002.

[2] H. Zang, J. P. Jue, and B. Mukherjee, A Review of Routing and Wavelength Assignment Approaches for Wavelength-Routed Optical WDM Networks, *Optical Networks Magazine*, vol. 1, no.1, pp. 47–60, January 2000.

[3] L. H. Sahasrabuddhe and B. Mukherjee, Light-Trees: Optical Multicasting for Improved Performance in Wavelength-Routed Networks, *IEEE Communications Magazine*, vol. 37, no. 2, pp. 67–73, February 1999.

[4] C. S. R. Murthy and M. Gurusamy, *WDM Optical Networks: Concepts, Design, and Algorithms*, Upper Saddle River, New Jersey: Prentice-Hall, 2002.

[5] M. Ali, *Transmission-Efficient Design and Management of Wavelength-Routed Networks*, Boston: Kluwer Academic Publishers, 2001.

[6] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning,* Reading, MA: Addison-Wesley, 1989.

[7] L. Davis, *Handbook of Genetic Algorithms,* New York: Van Nostrand Reinhold, 1991.

[8] M. Mitchell, *An Introduction to Genetic Algorithms,* Cambridge, MA: MIT Press, 1996.

[9] G. N. Rouskas, Optical Layer Multicast: Rationale, Building Blocks and Challenges, *IEEE Network*, vol. 17, no. 1, pp. 60–65, January/February 2003.

[10] A. Ding and G.-S. Poo, A Survey of Optical Multicast over WDM Networks, *Computer Communications*, vol. 26, no. 2, pp. 193–200, February 2003.

[11] R. Malli, X. Zhang, and C. Qiao, Benefit of Multicasting in All-Optical Networks, *Proc. of SPIE, All Optical Networking*, pp. 209–220, November 1998.

[12] D. Li, X. Du, X. Hu, L. Ruan, and X. Jia, Minimizing Number of Wavelengths in Multicast Routing Trees in WDM Networks, *Networks*, vol. 35, no. 4, pp. 260–265, July 2000.

[13] X. Zhang, J. Y. Wei, and C. Qiao, Constrained Multicast Routing in WDM Networks with Sparse Light Splitting, *Journal of Lightwave Technology*, vol. 18, no. 12, pp. 1917–1927, December 2000.

[14] D. Papadimitriou, D. Ooms, and J. Jones, Optical Multicast: A Framework, *Internet Draft*, February 2001.

[15] M. Ali, Optimization of Splitting Node Placement in Wavelength-Routed Optical Networks, *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 8, pp. 1571–1579, October 2002.

[16] H. Takahashi and A. Matsuyama, An Approximate Solution for the Steiner Problem in Graphs, *Mathematica Japonica*, vol. 24, no. 6, pp. 573–577, 1980.

[17] M. Gen and R. Cheng *Genetic Algorithms and Engineering Optimization,* New York: John Wiley & Sons, 2000.

[18] J. O'Rourke, *Computational Geometry in C,* New York: Cambridge University Press, 1994.