International Journal of Management Science
Vol 9, No 1, May 2003

Wavelength Assignment Optimization in SDH over WDM Rings*

Jibok Chung**
Electronics and Telecommunications Research Institute
161 Gajeong-dong, Yuseong-gu, Daejeon, 305-350, Korea

Heesang Lee***
Department of Industrial Engineering, Hankuk University of Foreign Studies
Yongin, Mohyun, Wangsan 89, Kyungki 449-791, Korea

Chimoon Han****
Department of Electronic Engineering, Hankuk University of Foreign Studies
Yongin, Mohyun, Wangsan 89, Kyungki 449-791, Korea

(Received Jun. 2002 ; Revised Feb. 2003 ; Accepted Mar. 2003)

ABSTRACT

In this study, we propose a mathematical model based on the graph theory for the wavelength assignment problem arising in the design of SDH (Synchronous Digital Hierarchy) over WDM (Wavelength Division Multiplexing) ring networks. We propose a branch-and-price algorithm to solve the suggested models effectively within reasonable time in realistic SDH over WDM ring networks. By exploiting the structure of ring networks, we developed a polynomial time algorithm for efficient column generation and a branching rule that conserves the structure of column generation. In a computer simulation study, the suggested approach can find the optimal solutions within reasonable time and show better performance than the existing heuristics.

---

* The work reported here was partially supported by KOSEF, No.R01-2001-000-00364-0

** E-mail: jbchung@etri.re.kr
*** E-mail: leehee@san.hufs.ac.kr
**** E-mail: cmhan@san.hufs.ac.kr

1. INTRODUCTION

The fast growth of the Internet has brought about rapid exhaustion of core transport capacity. WDM (Wavelength Division Multiplexing) technologies have emerged as a viable technology to deal with the increasing telecommunication traffic in a cost-effective way and will continue to be deployed on a large scale [13].

Currently, fiber optic ring networks in the form of TDM (Time Division Multiplexing) that is standardized as SDH (Synchronous Digital Hierarchy) technology have been deployed widely by the telecommunication service providers for the past decades due to their advanced protection and network management capabilities. Considering the progress of telecommunication evolution, current SDH rings will not completely replaced with the all-optical WDM rings since the traffic grooming is more efficient when SDH layer technology over WDM layer is used. Therefore, WDM rings will co-exist with the current SDH rings at least in near future. This hybrid transmission architecture is called as *SDH over WDM rings* [13, 5].

When there is a traffic requirement, an optical channel, called *lightpath*, with dedicated wavelength between the source node and the destination node of the optical channel must be established. Two lightpaths that traverse the same physical link must be assigned the different wavelength. This requirement is called *wavelength-continuity constraint.*

Figure 1 shows an example of lightpath connection and the wavelength assignment in SDH over WDM rings. As we can see in Figure 1, the terminating equipments in SDH over WDM rings are OADMs (*Optical Add-Drop Multiplexer*) and SDH ADMs. Since OADMs can selectively add and drop wavelengths at a node, it can optically bypass the wavelengths that do not carry any traffic from or to the specific node. This means that SDH ADMs are required at a node if and only if it carries traffic terminating at this node. Therefore, in SDH over WDM ring architecture, a good design algorithm can reduce the number of SDH ADMs by assigning wavelength efficiently.

It is well known that the total system cost of SDH over WDM rings is dominated by the number of SDH ADMs rather than the number of wavelengths. The number of SDH ADMs and the number of wavelengths may not be minimized simultaneously [4-5]. Therefore, it is necessary to use the number of SDH ADMs as small as possible in order to design the cost-effective SDH over WDM ring networks.

The wavelength assignment problem in WDM networks, which is to minimize the number of wavelengths while satisfying the wavelength-continuity constraint, corresponds to the path-coloring problem in graph theory. See [8] and [11] for the related path-coloring problem. However the traditional graph-coloring algorithms can not be directly applied to the wavelength assignment problem in SDH over WDM rings because it has a different objective function. Therefore, new models and algorithms must be developed to minimize the number of the SDH ADMs.

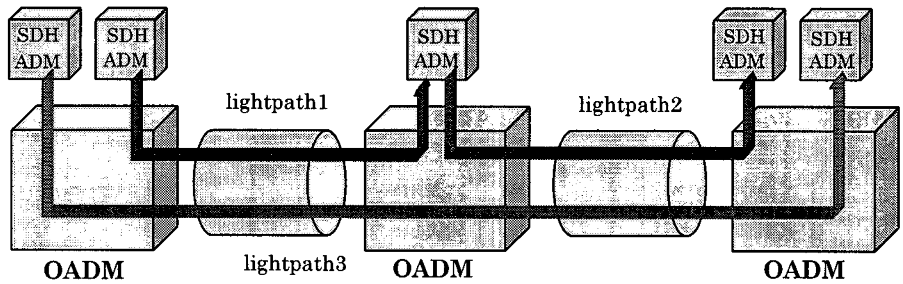

Figure 1. Structure of SDH over WDM Ring

In this paper, we assume that the routes of lightpaths are pre-fixed. Hence we focus on the wavelength assignment problem and do not consider the routing problem. We propose a new graph-theoretic integer programming formulation that has reasonably many constraints but has huge number of variables. We suggest a branch-and-price method to get an optimal solution. We develop a polynomial time algorithm for the effective column generation to handle many variables.

The remainder of this paper is organized as follows. In section 2, we describe the wavelength assignment problem for SDH over WDM rings and present some related backgrounds. In section 3, we propose an Integer Programming formulation. In section 4, we suggest an exact algorithm to solve the formulation proposed in section 3. In section 5, we show, by computational experiments, that the suggested algorithm can find optimal solutions for real-sized SDH over WDM ring networks within reasonable time. We also do the performance comparisons with some generic heuristics. In section 6, we give some concluding remarks and discuss further research topics.

2. BACKGROUNDS

1.1 State of the arts

Many works has been conducted for the problems related with WDM ring networks as we can see in Table 1. [4] showed that the number of wavelength and the number of SDH ADMs can not be minimized simultaneously. They proposed two kinds of heuristics for the wavelength assignment and some kinds of post-assignment transformations such as merging, combining and splitting. They also analyzed the lower bound and the worst-case example for each heuristic. [10] showed that the wavelength assignment problem to minimize the number of SDH ADMs is NP-Hard. They also showed that minimum ADMs problem equals to the maximum ADMs sharing problem. [12] considered the traffic grooming and wavelength assignment problem simultaneously. They analyzed the lower bound and proposed two heuristics, each for minimizing wavelength and number of SONET ADMs. They analyzed the performance of suggested algorithms for realistic different traffic patterns such as uniform, non-uniform and distance-dependent traffics. Recently, [16] proposed a flow-based integer programming formulation that has exponentially many constraints with respect to the number of lightpaths. They also suggested some heuristic algorithms.

Table 1. Review of previous works

Ref. Number	Ring architecture	Addressed problems	Main result
[15]	all-optical WDM ring	Wavelength Assignment	Upper bound for wavelength required
[3]	all-optical WDM ring (4-fiber bi-directional)	Wavelength Assignment	Analysis of wavelength required for uniform traffic
[9]	all-optical WDM ring	Routing and Wavelength Assignment	IP formulation Exact solution algorithms Greedy heuristics
[4]	SDH over WDM ring	Wavelength Assignment	SDH ADMs and wavelength may not be minimized simultaneously. Greedy heuristics (Assign First and Cut First)
[12]	SDH over WDM ring (Bi-directional)	Traffic Grooming Wavelength Assignment	Lower bound for wavelength and SDH ADMs
[10]	SDH over WDM ring	Wavelength Assignment	Proof of NP-completeness (3+e)/(1+e) approximation algorithms Greedy heuristics
[16]	SDH over WDM ring	Wavelength Assignment	Circle Segment heuristics Flow based IP formulation
This paper	SDH over WDM ring	Wavelength Assignment	Graph theoretic IP Formulation Branch-and-price algorithm

1.2 Wavelength Assignment in SDH over WDM ring Problem (WA_SWR)

In this section, we will describe the WA_SWR problem by using the following small example. Figure 2 shows two kinds of wavelength assignment examples for the same traffic set of T on 6-node ring, where { 0, 1, 2, 3, 4, 5 } is a set of nodes and T = { (0, 2), (3, 5), (2, 4) } is the set of light paths. In Figure 2, 6-node ring is represented linearly for convenience. We call lightpath 1, lightpath 2, lightpath 3 for each traffic requirement between (0, 2), (3, 5), (2, 4), respectively.

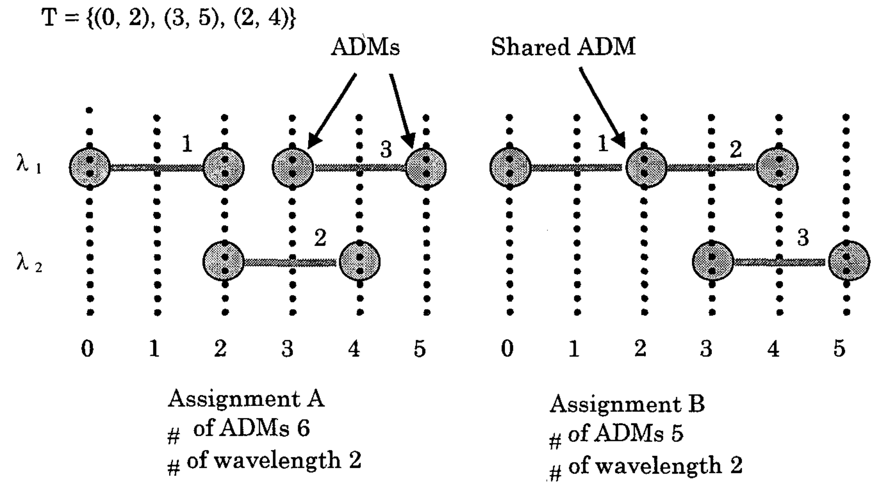

Figure 2. Example of wavelength assignment

As we can see in Figure 2, lightpath 2 and lightpath 3 can not be assigned the same wavelength because they traverse the same physical link (3, 4). Therefore, it requires at least two wavelengths for each assignment. However, the assignment A requires 6 SDH ADMs while the assignment B requires 5 SDH ADMs since there exists one ADM that is shared at node 2.

As we see in this example, it is necessary to maximize the number of ADM sharing as much as possible. The conditions for ADM sharing are a) two lightpaths are assigned same wavelength and b) two lightpaths have common end points [4].

There are four kinds of relations between a pair of lightpaths: (a) they traverse the same physical link, (b) they do not traverse the same physical link but do not have a common end point, (c) they do not traverse the same physical link and they have one common end point and (d) they do not traverse the same physical link and they have two common end point. In case of (a), they can not be assigned the same wavelength due to the wavelength continuity constraint and consequently the ADM sharing can not happen. In case of (b), they can be assigned the same wavelength but the ADM sharing can not happen since they violate the sharing condition. In case of (c) and (d), they can be assigned the same wavelength and the ADM sharing can happen. In case of (d), there exists an optimal wavelength assignment such that the pair of lightpaths is assigned with the same wavelength [16]. Therefore, we can first find the pair of lightpaths in case (d) and assign the same wavelength for them, and then find the optimal wavelength assignment for the rest of the lightpaths. In case of (c), we may think that assigning the wavelength first for the pair of lightpaths may bring about much ADM sharing. However, as we can see in Figure 3, this greedy heuristic method may require 11 SDH ADMs but the minimum number of SDH ADMs is 8. Therefore, it is necessary to develop an optimization model and an exact algorithm to design the cost-effective SDH over WDM ring networks.

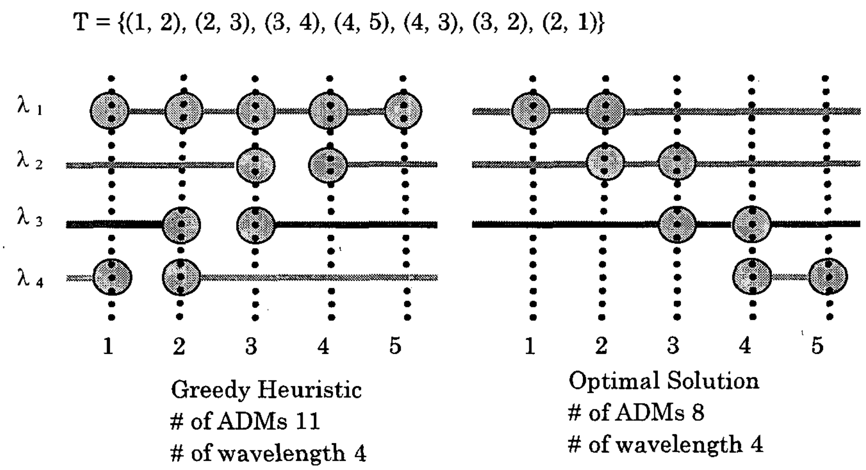

Figure 3. Bad case example of greedy heuristic

3. MATHEMATICAL MODELS

In this section, we will describe a mathematical model for the WA_SWA problem that is the problem of assigning a wavelength to each lightpath (i.e coloring a corresponding path on the graph) in order to maximize the ADMs sharing while satisfying the wavelength-continuity constraint.

The path-coloring problem can be transformed as the well-known vertex-coloring problem by using a *conflict graph*, where we define a vertex in the conflict graph for each lightpath and define an edge between two vertices of the conflict graph if the routes of two lightpaths traverse the same physical link [6], [8]. The set of lightpaths on ADM ring topology constitutes a special conflict graph so called the *Circular Arc Graph* (CAG). The set of vertices with the same color of the conflict graph can be found by finding an *Independent Set* (IS) on the corresponding CAG [8], where an independent set is the set of vertices such that there is no edge connecting any pair of vertices of the independent set [6]. Now we introduce the WA_SWR problem as the following Integer Programming (IP) problem.

Let $G = (V, E)$ be an undirected conflict graph, where V is the set of vertices and E is the set of edges. Let n be the cardinality of V and S be the set of all ISs of G. We define a binary variable $x_s = 1$ if s will be given an unique color (wavelength), while $x_s = 0$, otherwise. Let W_s be the weight of s. In this paper, we means by W_s the number of SDH ADMs required to assign the wavelength for all the lightpaths in s. By using the above notation, we have:

(WA_SWR)

$$\text{Minimize} \quad \sum_s W_s x_s \tag{1}$$

$$\text{Subject to} \quad \sum_{\{s:i\in S\}} x_s = 1, \quad \forall i \in V, \tag{2}$$

$$x_s \in \{0,1\}, \quad \forall s \in S. \tag{3}$$

The objective function (1) means to minimize the total number of SDH ADMs required to assign the wavelength for each lightpaths. The constraints (2) and (3) imply that one vertex of the conflict graph must be included in one IS, (i.e, A lightpath must have the unique wavelength).

As we can see in Figure 2 in section 2, it is easy to find that W_s equals the two times the number of lightpaths minus the number of shared ADMs. For the example in Figure 2, W_1 equals $2\times2-0=4$ and W_2 equals $2\times1-0=2$ in assignment A, while W_1 equals $2\times2-1=3$ and W_2 equals $2\times1-0=2$ in assignment B.

It can be notated mathematically as $2\sum_{i\in s} x_i - \sum_{i,j\in E'} x_i x_j$, where E' is the subset of E such that the ADMs sharing between the lightpaths is possible. Note that W_s is not an explicit function with respect to the lightpaths. This makes it difficult to apply the existing graph algorithms directly to generate the favorable columns. In section 4, we will explain in detail how we resolve this difficulty.

4. A BRANCH-AND-PRICE APPROACH

Our IP formulation has only one constraint for each vertex of the conflict graph (lightpath). However, the number of decision variables is huge since the number of all ISs of a graph G can be exponetially large. Therefore generating all ISs of a graph to get the explicit formulation is intractable (except for very small size SDH over WDM ring networks). Hence solving even the *Linear Programming* (LP) relaxation may be computationally difficult if we try to solve the explicit formulation. We resolve this difficulty by using only subset of the decision variables and generating more variables when they are needed. This technique, called *column generation technique*, is successfully used to solve the LP problems with many variables [2]. An optimal solution to LP problem with many columns can be obtained without explicitly including all columns since only a very small subset of all columns will be in an optimal solution.

Branch-and-price approach that is a column generation techniques for IP within a branch-and-bound method has an additional computational advantage: The column generation formulation of an IP may have a stronger LP relaxation than a canonical compact IP formulation, which is a very important advantage for avoiding long computation time of a branch-and-bound algorithm. See [2] and [11] for general expositions of the column generation approach to IP.

In our problem, the column generation procedure for the LP relaxation is described as follows. Begin with S', a subset of S, the set of all ISs. Solve the LP relaxation of restricted to all $s \in S'$. This gives a feasible solution for the LP relaxation and a dual value π_i for each constraint i of the original (primal) LP relaxation. Now, determine if we need more columns (i.e., if we need more ISs) by solving the following subproblem that is denoted by SP.

(SP)

$$\text{Max } \sum_{i \in s} \pi_i - W_s, \ \forall s \in S \tag{4}$$

$$\rightarrow \text{Max } \sum_{i \in s} \pi_i x_i + \sum_{i,j \in E'} x_i x_j - 2\sum_{i \in s} x_i, \ \forall s \in S \tag{5}$$

Note that SP is to find the set of ISs that maximizes the function (4). The first term of the objective function (4) is the sum of dual variables associated with each IS and the second term is the weight of the IS. As noted in [7], when the weight of IS can be defined explicitly, *Maximum Weighted Independent Set* (MWIS) in the CAG can be solved by computing the MWIS in the *Interval Graph*

(IG) iteratively, where IG is a special case of CAG without forming a cycle [6]. However, since the weight in our problem is not explicitly defined (i.e., the objective is not a simple summation of the each lightpath), it requires a modified algorithm to deal with SP.

4.1 Algorithm for SP in IG

Before describing the algorithms for SP in CAG, let us consider the following SP in IG.

Input : IG G with weight w_i for each lightpath i.
Output : A set of lightpaths.
Objective : Maximizes the function (5)

We suggest that SP in IG is transformed to the longest path problem in a modified network. Note that the longest path problem is NP-Hard in general network but the problem is solvable in polynomial time for acyclic network. Since our modified network for SP has no directed cycle, we can find the longest path in polynomial time using the topological ordering algorithm [1]. Therefore, SP in IG can be computed within polynomial time bound. The detailed procedures are as follows;

Algorithm for SP in IG :

Step 0 : **Initialization.** Sort the lightpaths from the most left node
Step 1 : **Network Construction.** Construct the associated network as follows:
For all pairs of lightpath i, j $(i < j)$, if two lightpaths do not overlap, add edge (i, j).
If two lightpaths have a common end point, set the edge weight as $w_j + 1$.
Else, set the edge weight as w_j
Step 2 : **Network Transformation.** Add artificial node s, t as follows:
For all lightpaths j, add edge (s, j) with edge weight w_j and (j, t) with edge weight 0.
Step 3 : **Solve.** Find longest path from node s to node t.
Step 4 : **Get Solution for SP.** The set of arcs included in the longest path is the solution for SP in IG.

In Figure 4 below, (c) is the transformed network from the IG in (b), and (d) is the resulting solution after performing step 3 for the transformed network in (c).

4.2 Algorithm for SP in CAG

Now we will show that SP in CAG can be solvable in polynomial time by using the above algorithm for SP in IG iteratively. By using the fact that when the G is the CAG, the remaining graph after deleting all intersecting nodes with arbitrary node reduces to the IG. SP in CAG can be solved in polynomial time bound since the number of node in G is finite.

Algorithm for SP in CAG:

Step 0 : **Initialization.** Label all nodes as unmarked. Let wt_max be negative infinite and $wt(v)$ be the weight of node v. $N(v)$ is the set of adjacent node with node v

Step 1 : **Network Construction.** Select any unmarked node v and construct IG G' by deleting the node v and $N(v)$. Increase the weight of nodes which has a common end point with v by 1.

Step 2 : **Solve.** Solve the SP in corresponding G' and find the solution wt^*.

Step 3 : **Solution update.** If $wt(v) + wt^* > wt\_max$, then $wt\_max = wt(v) + wt^*$.

Step 4 : **Terminate.** If all nodes are marked, terminate. Else, let node v as marked and go to step 1.

In Figure 4 below, (a) is the original CAG and (b) is the resulting IG after performing step 1 for the lightpaths 7 selected.

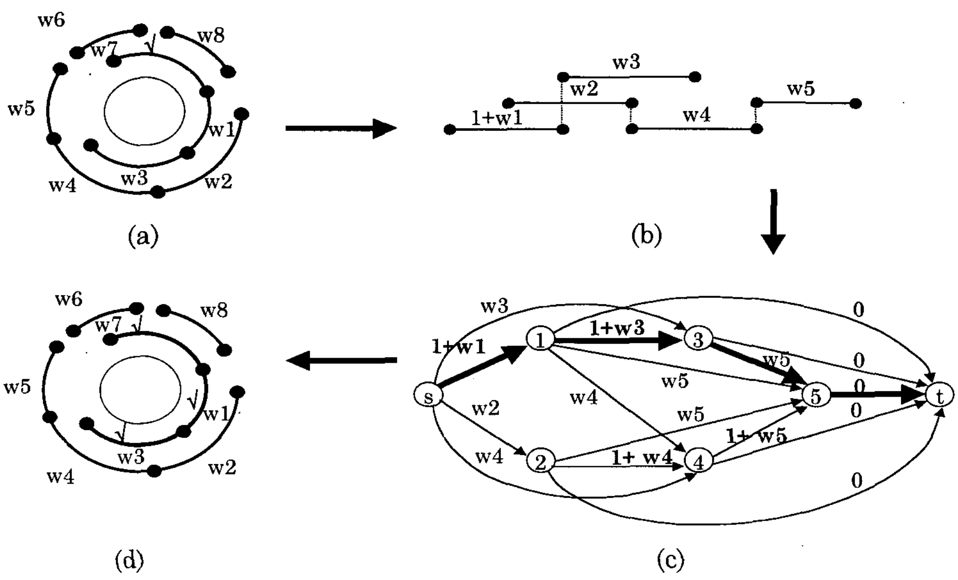

Figure 4. Step 1 for algorithm for SP in CAG

4.3 Branching Rule

When the optimal solution of LP relaxations obtained through the above column generation procedure is not integral, we need to branch some fractional variables. For effective implementation of branch-and-price approach, it is necessary to have a branching rule that conserves the structure of column generation structure.

First, moving clockwise once around the ring from arbitrary starting point, we can index the vertices according to the order in which the counterclockwise endpoints of the corresponding lightpaths occur (tie-break arbitrarily). Then we will select the pair of vertices that are independent and most nearest with respect to the index. We will call it as *minimal distance branching rule*.

The suggested branching rule is composed of two operations, called SAME(i, j) and DIFFER(i, j). SAME(i, j) is an operation to collapse two vertices i and j (i < j) into single vertex. It means to merge two lightpaths corresponding to the vertex i and j into one lightpath by extending the counterclockwise endpoint of j to the clockwise endpoint of i. DIFFER(i, j) is an operation to add edge between the vertex i and j. It means to extend the clockwise endpoint of i such that it is larger than the counterclockwise endpoint of j but smaller than the counterclockwise endpoint of j+1. (See the example in Figure 5 for more explanation)

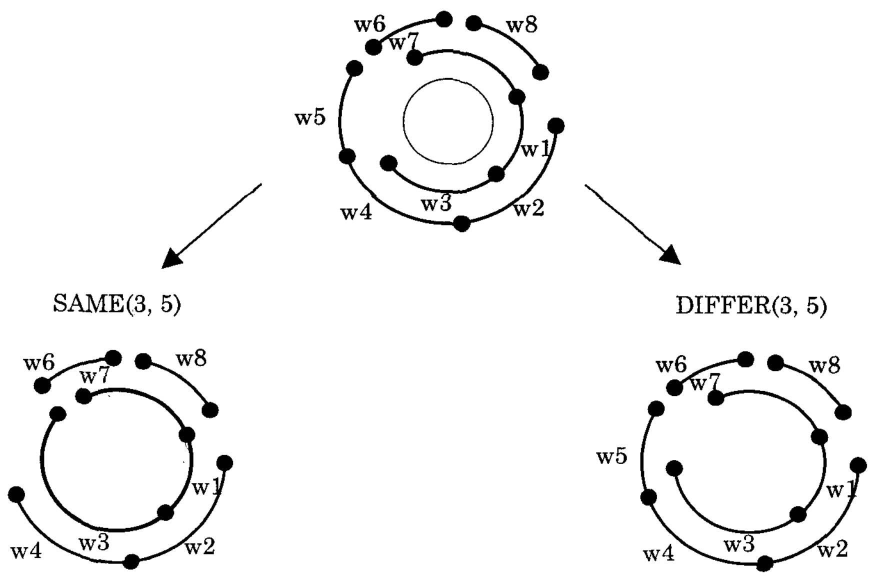

Figure 5. Example of minimal distance branching

In this section, we will show that the suggested branching rule conserves the structure of the column generation procedure by using the *quasi–circular 1's property*. Let $M(G) = (m_{ij})_{nn}$ be the augmented adjacent matrix for CAG G and U_i (V_i) be the set of consecutive 1's below (rightward) from the i[th] diagonal element of $M(G)$; regard the matrix as wrapped around a cylinder. If the union of U_i and V_i covers all 1's in $M(G)$, then $M(G)$ satisfy the quasi-circular 1's property. Refer to [14] for the details of quasi-circular 1's property.

Proposition 1. An undirected graph G is a CAG *if and only if* $M(G)$ has quasi-circular 1's property [14].

Theorem 1. The suggested branching rule conserves the structure of column generation procedure.

Proof. We will show that the graph $\overline{G}$ after performing the branching operation for CAG G is also CAG. Let $M(\overline{G})=(m'_{ij})_{nn}$ be the augmented adjacent matrix for $\overline{G}$ and $U'_i\,(V'_i)$ be the set of consecutive 1's below (rightward) from the i[th] diagonal element of $M(\overline{G})$. Let A_i be the set of elements in $\{1,\cdots,i\}$ at the i[th] column and B_i be the set of elements in $\{i+1,\cdots,n\}$ at the i[th] column. Assume that the index of vertex i precedes that of vertex j and SAME(i, j) operation collapses two vertices i and j into i'. By definition, SAME(i, j) operation takes the value 1 if at least one of the row i and j has value 1. For the new 1's in B_i at the i[th] column of $M(\overline{G})$, it is easy to know that $U'_{i'}$ includes the all 1's which was in U_i and U_j. Furthermore, $U'_{i'}$ is consecutive because we select the vertex j which is the first 0's when we are moving below (rightward) from the i[th] diagonal element of $M(G)$. For the new 1's in A_i at the i[th] column of $M(\overline{G})$, they must be included U_j of $M(G)$. If it is not true, it contradicts the assumption that the G has the quasi-circular 1's property. Therefore, SAME(i, j) operation conserves the quasi-circular 1's property. By definition, DIFFER(i, j) operation changes the value of $m'_{ij}\,(m'_{ji})$ from 0 to 1. Because we select the vertex j which is first 0's when we are moving below (rightward) from the i[th] column of $M(\overline{G})$, it is easy to know that $U'_i\,(U'_j)$ includes the 1's of $m'_{ij}\,(m'_{ji})$ and is consecutive. Therefore, DIFFER(i, j) operation conserves the quasi-circular 1's property.

5. EXPERIMENTS RESULTS

The proposed algorithm has been coded in C and experimented on an engineering workstation (167Mhz CPU) using an IP optimization callable library, CPLEX 6.0. By the experiment, we want to show that the suggested algorithm is computationally feasible to implement in real-sized SDH over WDM rings. We also want to compare the suggested algorithms with the generic heuristics developed by [4].

Table 2. Problem instance (average of 5 instances)

	$G(n,d)$	VERT	EDGE	DEGREE
1	$G(5, 0.3)$	6	11.2	4
2	$G(5, 0.5)$	10	34.8	6.8
3	$G(5, 0.7)$	14	65.2	7.8
4	$G(5, 0.9)$	18	113.6	9.8
5	$G(10, 0.3)$	27	282.6	16.6
6	$G(10, 0.5)$	45	799.6	26
7	$G(10, 0.7)$	63	1564.8	34.4
8	$G(10, 0.9)$	81	2615	43.2
9	$G(15, 0.3)$	63	1594.8	36.4
10	$G(15, 0.5)$	105	4325.6	56.8
11	$G(15, 0.7)$	147	8736.2	77.8
12	$G(15, 0.9)$	188	14226.4	96.8
13	$G(20, 0.3)$	114	5307.2	62.8
14	$G(20, 0.5)$	190	14677.8	102.8
15	$G(20, 0.7)$	266	28681.4	139.4
16	$G(20, 0.9)$	342	47771.2	175.6

We experiment four classes of problem instances that have 5, 10, 15, and 20 ring nodes. To know the effect of demand on the ring to the performance, we divide each class of node sizes into four demand sets by setting the demand density of 0.3, 0.5, 0.7, and 0.9, which is defined as the probability that requires one lightpath for a pair of ring node. Input parameters of twenty sets are summarized in Table 2. In Table 2, $G(n,d)$ represents that n is the number of ring nodes from 5 to 20, and the d is the demand density from 0.3 to 0.9. "VERT", "EDGE" and "DEGREE" in Table 2 denotes respectively the number of vertices, the number of edges and the average degree in the conflict graph. Note that "VERT" is the same in a given set but the "EDGE" and "Degree" can be different according to the overlapping conditions for each instance. The largest size of the graphs has 342 nodes and 47,886 edges.

Average performances of five instances for each set are displayed in Table 3. In Table 3, "Heuri" denotes the number of SDH ADMs obtained by the heuristic of [4]. This heuristic is also used as an initial solution of the suggested algorithms. "COLs" denotes the average number of columns generated and "BnBs" denotes the average number of branch and bound tree nodes to get the final integer solution. "LP" denotes the average of the optimal objective values of the LP relaxations and "OPT" denotes the average of the minimum numbers of SDH ADMs obtained by the suggested algorithms.

Table 3. Average performance of branch-and-price approach

	Heuri	COLs	LP	BnBs	OPT.	Time
1	10	2.2	9.6	0	9.6	0.01
2	15.4	3.6	14.8	0	14.8	0.016
3	19.8	9	17	0	17	0.04
4	25.4	17.2	19.8	0	19.8	0.088
5	43.2	21.2	39.2	0	39.2	0.15
6	70.6	52.2	59.4	0	59.4	0.598
7	94.6	101.4	77	0	77	1.748
8	118.6	177.6	88.2	0	88.2	4.48
9	100.6	74.8	91.2	0.8	91.2	1.214
10	158.6	216.8	138.6	0	138.6	7.166
11	224	431.4	176.2	0.4	176.2	27.018
12	272.8	769.6	201.8	0.2	201.8	75.136
13	178	197.8	161.8	0.2	161.8	8.066
14	290.6	475	247.2	0.2	247.2	41.636
15	398.8	1206	316.6	2.6	316.6	219.29
16	498.2	1763	367	0	367	522.64

As you can see in Table 3, our branch-and-price algorithm can solve any instance of 100 instances in less than 523 seconds. Many instances can be solved within a few seconds. The branch-and-price algorithm does not require generating huge number of columns to get the optimal solution for the LP relaxations. Moreover, after getting an LP optimal solution on the root node of branch and bound tree, the algorithms can be terminated without traversing many branch and bound tree nodes. This can be possible since our graph theoretic formulation for WA_SWR has a very strong LP relaxation as we can see in Table 3. Note that the value of LP is near the value of OPT.

As we can see in Figure 6, our branch-and-price algorithm outperforms the existing heuristics AFH (Assign First Heuristic) and CFH (Cut First Heuristic) about 15%, 24% respectively in terms of the number of SDH ADMs. Furthermore, the performance gap increases as the traffic load becomes heavy. Between two heuristics, CFH uses more SDH ADMs than AFH since it generates additional lightpaths by splitting an original lightpath into two lightpaths if it is necessary.

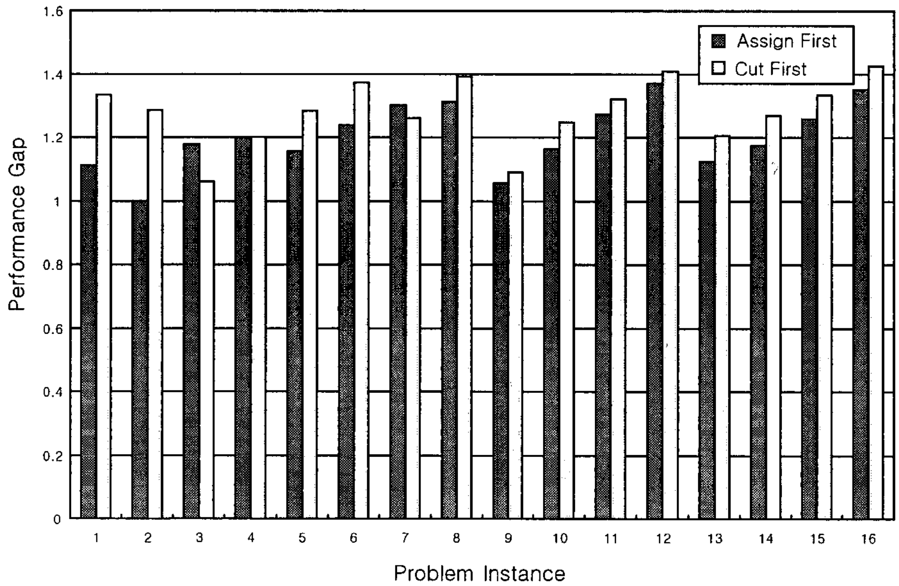

Figure 6. Performance comparison with heuristics with respect to SDH ADMs

In terms of wavelengths, our algorithm requires 10% less than AFH but 3% more than CFH as we see in Figure 7. Note that minimization of SDH ADMs does not imply wavelength minimization. The objective of our model is not minimizing the number of wavelengths but minimizing the number of SHD ADMs.

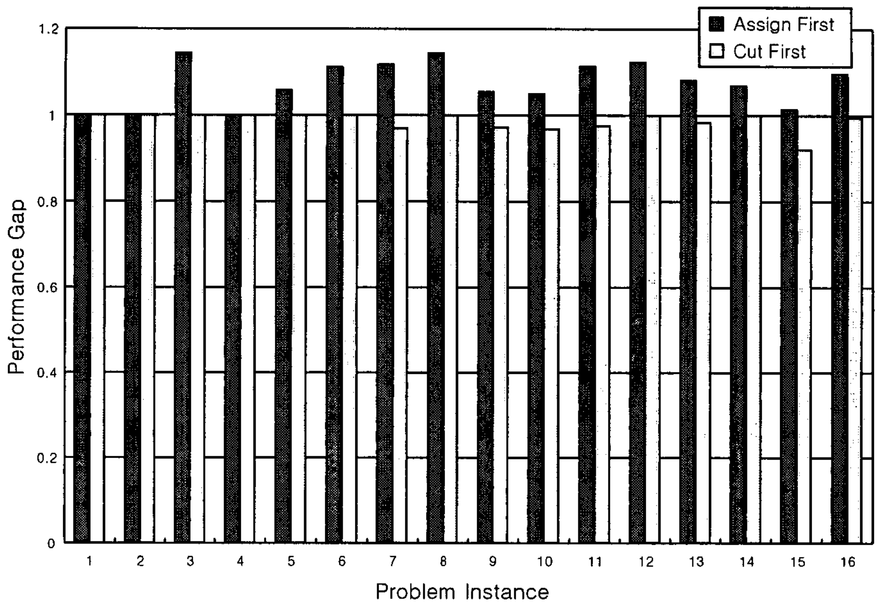

Figure 7. Performance comparison with heuristics with respect to wavelengths

6. CONCLUSIONS

In this paper, we have shown that the wavelength assignment problem to minimize the number of SDH ADMs can be formulated as an IP problem, and the problem can be exactly solvable up to 20 ring nodes that are enough large to meet explosive future demands. To accomplish this, we exploited properties of CAG and suggested a branch-and-price algorithm to get the exact optimal solutions. The computer simulation shows that the suggested algorithm could find the optimal solution within reasonable time and better performance than the known heuristic methods.

In this paper, we do not consider the traffic grooming of SDH traffic and survivability problem. Extending the current research to the traffic grooming and/or restoration optimization is one of the further research topics for SDH over WDM ring networks.

RERFERENCE

[1] R. K. Ahuja, T. L. Magnanti and J. B. Orlin, *Network flows: theory, algorithms and applications*, Prentice Hall Inc, 1992.

[2] J. Birge and K. Murty eds., Mathematical Programming: State of the Art 1994, University of Michigan, 1994.

[3] G. Ellinas et al., *A Novel Wavelength Assignment Algorithm for 4-fiber WDM Self-Healing Rings*, IEEE ICC '98, 1998.

[4] O. Gerstel et al., *Wavelength Assignment in a WDM Ring to Minimize Cost of Embedded SONET Rings*, IEEE Infocom '98, 1998.

[5] O. Gerstel et al., *Combined WDM and SONET Network Design*, IEEE Infocom '99, 1999.

[6] M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, 1980.

[7] J. Y. Hsiao et al., "An Efficient Algorithms for Finding a Maximum Weighted 2-independent Set on Interval Graph," *Information Processing Letters*, No.43, 1992, 229-235.

[8] H. Lee and J. Chung, "Wavelength Assignment Optimization in Uni-Directional WDM ring," *IE interface*, Vol.13, No.4, 2000.

[9] T. Lee et al., "Optimal Routing and Wavelength Assignment in WDM Ring Networks," *IEEE JSAC*, Vol.18, No.10, 2000, 2146-2154.

[10] Liu et al., *Wavelength Assignment in WDM Rings to Minimize SONET ADMs*, IEEE Infocom '00, 2000.

[11] A. Mehrotra and M. A. Trick, "A Column Generation Approach for Graph Coloring," *INFORMS Journal on Computing*, 8, 1996, 344-354.

[12] H. Shiraz et al., "Effective Wavelength Assignment Algorithms for Optimizing Design Cost in SONET/WDM Rings," *J. of Lightwave Technology*, Vol.19, No.10, 2001, 1427-1439.

[13] T. E. Stern and K. Bala, *Multiwavelength Optical Networks: A layered approach*, Addison-Wesley, 1999.

[14] A. Tucker, "Matrix Characterization of Circular-Arc Graph," *Pacific J. of Mathematics*, Vol.39, No.2, 1971, 535-545.

[15] A. Tucker, "Coloring a Family of Circular Arcs," *SIAM J. Applied Math*, 29(3), 1975, 493-502.

[16] X. Yuan and A. Fulay, *Wavelength Assignment to Minimize the Number of SONET ADMs in WDM Rings*, IEEE ICC'02, 2002.