On RWA Algorithms for Scheduled Lightpath Demands in Wavelength Division Multiplexing Networks

Sooyeon Park†, **Jong S. Yang**††, **and Young-Cheol Bang**†

† Department of Computer Engineering, Korea Polytechnic University, Korea

†† Korea Institute of Industrial Technology Evaluation and Planning, Seoul, Korea

Summary

We have proposed and evaluated a novel heuristic algorithm of routing and wavelength assignment (RWA) for scheduled lightpath demands (SLD). Our algorithm works on directed and undirected optical networks, and is shown here to have a performance gain in terms of the number of wavelengths upto 65.8 % compared with the most recently introduced method, TDP_RWA. Also, we show the execution time of our proposed method is feasible.

Key words:

WDM networks, RWA, Scheduled Lightpath Demeands

1. Introduction

Emerging real-time multimedia communication applications and increasing traffic demands in networks cannot be supported by the current Internet architecture. To support such services that require the electronic processing speeds, the wavelength division multiplexing (WDM) technology employed in optical networks is developed and will replace Internet backbone in near future. To send a message from a source to a destination, WDM optical network should setup a connection between optical layer of each end node such as the one in circuit-switched networks. This connection process can be done by determining a path between the two end nodes, and such a path is referred as a lightpath. Once a connection is setup the entire bandwidth on the lightpath is reserved until it is terminated.

As explained, a message is supposed to be sent though a lightpath that requires a route and a wavelength for each connection so that it should be effective to select lightpaths for request demands to utilize bandwidths of optical networks efficiently. This is known as the problem of routing and wavelength assignment (RWA). Espe-cially in the case of no wavelength converter, given a physical optical network and the requested demands, the RWA problem is to select a suitable route and wavelength among the many possible choices for each connection so that no two paths sharing a link are assigned the same wavelength [2].

In the most common way, RWA at optical networks deals separately with route and wavelength assignment such that a route of the light path is selected at first and then a wavelength is assigned on the selected route in efficient way. RWA problem on the optical networks is known to be NP-complete [12], and many heuristic approaches have been proposed in [2][4][5][6]. Also more mechanisms are intensively surveyed in [3].

The optical transport network (OTN) provides the optical virtual private network (OVPN) as the key service network, where there exist three types of lightpath demands; static, scheduled, and dynamic. Static lightpath demands are given by OTVN clients to minimize connectivity and bandwidth requirements, and dynamic lightpath demands are to establish and release connection in time. At releasing time, reserved resources are also released. In the case of the scheduled lightpath demand (SLD), request demands are based on specific time. This type of demand is to increase the capacity of network at specific times on certain links. For example, periodical data backup performed at specific time can use SLD.

In real world OTN, most of demands will be static and scheduled as explained in [7][10]. This is because traffic load in a transport network is fairly predictable for its periodic nature. Fig. 1 shows the traffic on the New York-Washington link of the Abilene backbone network during a typical week. A similar periodic pattern was observed on all the other links of the network in the same period. Fig.1 is a clear evidence of the link between the intensity of communication among humans using the network, and the network traffic load [7].

Manuscript received February 5, 2007
Manuscript revised February 25, 2007

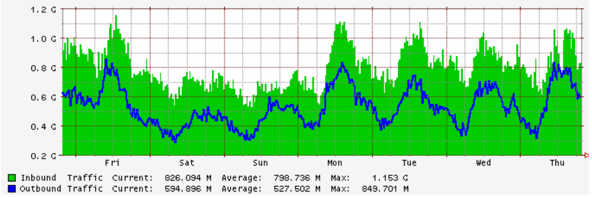


Fig. 1 Traffic on the New York-Washington link of the Abilene backbone network form Oc-tober 13, 2005 to October 20, 2005

One of characteristics of RWA is to reuse wavelength [8] [9] such that if there are two paths between arbitrary nodes without sharing links, two paths can use same wavelength at the same time [8][9]. For example in Fig. 2, consider two paths from 1 to 3 (path A) and from 3 to 5 (path B). Since these two paths do not share any link, there will be no problem for A and B to use the same wavelength at any time. Thus, it is clear that if we find more link-disjoint paths in a given network then the number of wavelength may be decreased. In the case that two paths are sharing any link, if ser-vice times of two paths are not overlapped then it is also clear only single wavelength is enough for both paths.

Using these simple notions of an link-disjoint path and usage-time, we consider an optical network, G = (V, E), where V and E are the set of vertices and the set of links, respectively. is a set of SLDs, and each SLD can be represented by 4 tuples (s, t, u, w), , where s and t are source and destination nodes of a demand, u and w are setup and teardown times of a demand. λ is a wavelength that is assigned at lightpath. Let be a set of groups such that each , , consists of which does not share usage-time. Then our objective is to pro-pose a new algorithm to reduce the number of wavelengths and give the feasible computation time for SLD.

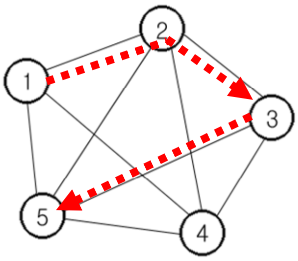


Fig. 2 Wavelength Routing Network

The paper is organized as follows. We discuss related works on RWA in Section 2. We propose a new RWA algorithm regarding to SLD in Section 3. Performance evaluation of the proposed algorithm is presented in Section 4. Finally, we conclude in Section 5.

2. Related Works

Due to its importance for the efficiency of optical networks, RWA algorithms have been intensively studied in the literature [2][3][4][5][6]. Among those algorithms, BGAforEDP [3][4] is known to be one of the simplest and most efficient algorithms introduced so far.

BGAforEDP is to find link-disjoint paths based on the shortest path algorithm to assign a route and a wavelength to a lightpath. It is shown that BGAforEDP is suitable for static demands but not appropriate for SLD because of characteristics such that starting time and teardown time are not considered for each static request [6]. For SLD, the Combinatorial Optimal Solution (COS) has been introduced [10], but it is complex and impractical for the computational time [5]. Using COS, time-complexity increases exponentially as the number of demands increase. For this reason, Steven [8] and Stern [9] considered the bound and branch (B&B) search algorithm to reduce the computational time, but B&B still does not give feasible computational time [8][9]. So Chlamtac[1] introduced the meta-heuristic TABU search algorithm. In the case of TABU, the performance is somewhat low with high complexity. The method called sRWA, based on FF [3], was introduced [10], but usage-time of each connection was not considered.

Recently new method called TDP-RWA [6] was proposed to take both the computational time and the number of wavelength into consideration. TDP-RWA separates SLDs into several groups such that usage-time of connection for each SLD cannot be overlapped in the same group but can be overlapped between different groups. That is, SLDs in same group can have same wavelength. Also, SLDs in other groups may have a same wavelength if lightpaths are link-disjoint each other with paths that a wavelength has already been assigned. Due to simplicity of algorithm, computational time decreases dramatically but more number of wavelength are required than other algorithms [6]. In this paper we propose a new RWA algorithm for SLDs that is feasible in terms of the computational time and outperforms any other heuristic algorithms in terms of wavelengths.

3. Proposed Algorithm

As introduced, RWA problems on optical network focus on minimizing the number of wavelengths as well as selecting appropriate routes to minimize the end-to-end delay to support request demands. In this section, we describe our proposed algorithm named SP_TDP_RWA that takes following two observations into account for SLD.

(1) Link-disjoint lightpaths can use the same wavelength
(2) If the usage-times of lightpaths with sharing links are different each other, the same wavelength can be assigned to those lightpaths

Based on observation 1 and 2, it is clear that if there exist enough link-disjoint lightpaths we may reduce the number of wavelengths as many as possible. So, a mechanism to select routes can decide the number of wavelengths. To derive an efficient algorithm, we first divide request demands into several groups such that request demands in the same group have the different service time each other, and sort groups in non-decreasing order based on number of request demands included. Then SP_TDP_RWA has following five steps.

(1) Select routes that use the minimum number of links for group to be considered, and then delete links selected from a given network to compute link-disjoint paths for other groups.
(2) Repeat step 1 until all groups are considered.
(3) Assign the same wavelength to all routes computed in step 1 and 2.
(4) Restore a given graph to consider request demands of which wavelength are not assigned.
(5) Repeat step 1 through 5 until all request demands are considered.

Especially, step 1 should be carefully treated. To minimize the number of links to constitute a lightpath, a new path *P'* can be selected by concatenating minimum number of links to any previously selected lightpath *P* if *P'* satisfies the delay bound, $\max(diam(G), \sqrt{|E|})$. Otherwise, the shortest path is used as a lightpath *P'*. In this manner, we minimize the number of links used for a group to be considered so that probability of finding link-disjoint paths for other group can be increased. Also, our mechanism is working well with both undirected and directed networks.

Consider
$\Delta_T = \{\Delta_{T1} = \{\delta_{11}, \delta_{12}, \ldots\ldots, \delta_{1|\Delta_{T1}|}\}, \Delta_{T2} = \{\delta_{21}, \delta_{22}, \ldots\ldots, \delta_{2|\Delta_{T2}|}\}, \ldots\ldots, \Delta_{Tk} = \{\delta_{k1}, \delta_{k2}, \ldots\ldots, \delta_{2|\Delta_{T2}|}\}\}$
and NSFNET in Fig. 3, where Δ_T is sorted in non-decreasing order based on number of δ included. Then Fig.4 shows how to compute routes and assign wavelength for arbitrary Δ_{Ti} with $1 \le i \le k$ in Table.1, where $\max(diam(G), \sqrt{|E|})$ is 4.

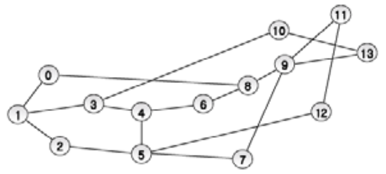


Fig. 3 NFSNET Topology

Table 1: Set of SLDs for ΔTi

SLD	SLD = (s, t, u, w)				Number of Links
	Source	Destination	Setup time	Teardown time	
δ_{i1}	3	8	2:00	3:00	3
δ_{i2}	4	6	3:00	5:00	1
δ_{i3}	6	10	13:00	14:00	3
δ_{i4}	1	9	19:00	21:00	3

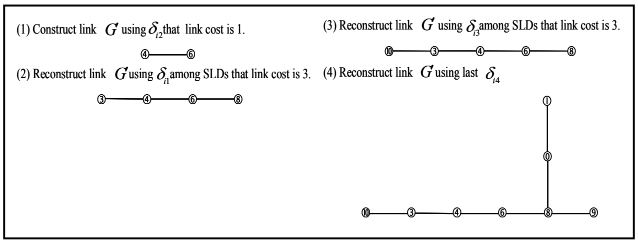
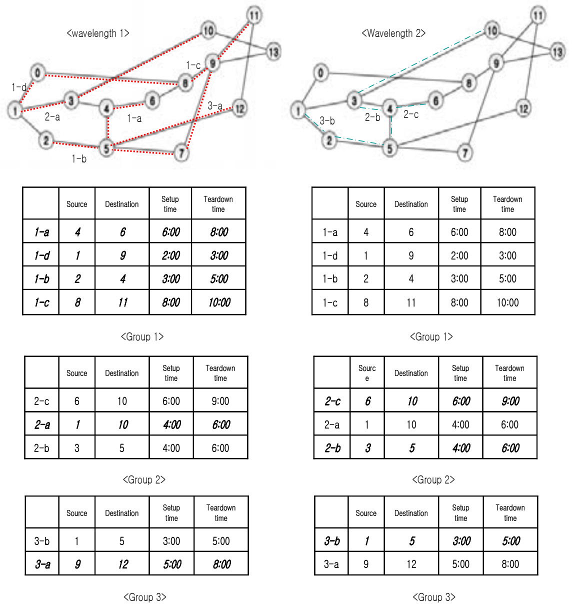


Fig. 4. Path Selection and G' construction using ΔTi

In step 1, since the number of links for the shortest path $P_{\delta_{i2}}$ for δ_{i2} is one that is minimum among the computed shortest paths for all $\delta \in \Delta_{Ti}$, we choose $P_{\delta_{i2}}$ and construct G' by adding links on $P_{\delta_{i2}}$, where G' is initially empty. Next (step 2), since the number of links on the computed shortest path for δ_{i1} is minimum among the computed shortest paths for $\delta \in \Delta_{Ti} = \Delta_{Ti} - \{\delta_{12}\}$, $P_{\delta_{i1}}$ is considered. The computed shortest path of *G* is 3-1-0-8 and a path reusing $P_{\delta_{i2}}$ (4-6) is 3-④-⑥-8, where both paths are satisfying the delay bound, $\max(diam(G), \sqrt{|E|}) = 4$. Thus, we select path (3-④-⑥-8) for δ_{i1}, since only two more links are used by sharing links on $P_{\delta_{i2}}$ instead of using three links for the shortest path 3-1-0-8. After then links (3, 4) and (6, 8) are added to G'. In this manner, ⑥-④-③-10 and 1-0-⑧-9 are selected for $P_{\delta_{i3}}$ and $P_{\delta_{i4}}$ in step 3-4, respectively. Suppose G'' is G - G' . Then repeat to find a lightpath for any $\delta \in \Delta_T - \{\Delta_{Ti}\}$ if a path is available. After check every SLDs in Δ_T, the same λ_i is assigned to lightpaths selected.

Fig. 5 shows the entire process to assign routes and wavelengths.



	Source	Destination	Setup time	Teardown time
1-a	***4***	***6***	***6:00***	***8:00***
1-d	***1***	***9***	***2:00***	***3:00***
1-b	***2***	***4***	***3:00***	***5:00***
1-c	***8***	***11***	***8:00***	***10:00***

<Group 1>

	Source	Destination	Setup time	Teardown time
2-c	6	10	6:00	9:00
2-a	***1***	***10***	***4:00***	***6:00***
2-b	3	5	4:00	6:00

<Group 2>

	Source	Destination	Setup time	Teardown time
3-b	1	5	3:00	5:00
3-a	***9***	***12***	***5:00***	***8:00***

<Group 3>

	Source	Destination	Setup time	Teardown time
1-a	4	6	6:00	8:00
1-d	1	9	2:00	3:00
1-b	2	4	3:00	5:00
1-c	8	11	8:00	10:00

<Group 1>

	Source	Destination	Setup time	Teardown time
2-c	***6***	***10***	***6:00***	***9:00***
2-a	1	10	4:00	6:00
2-b	***3***	***5***	***4:00***	***6:00***

<Group 2>

	Source	Destination	Setup time	Teardown time
3-b	***1***	***5***	***3:00***	***5:00***
3-a	9	12	5:00	8:00

<Group 3>

Fig. 5 Example of wavelength assignment: Bold type font is SLD that is selected for a certain wavelength, where a, b, c and d are selected order. For example, selected order in Group 1 is 1-a, 1-b, 1-c and 1-d.

```
SP_TDP_RWA (G, Δ)
01 $\Delta_T = TDP - Selector(G, \Delta)$
02 $\lambda = 1, k = 0$
03 $while( |\Delta_T| \neq k)$
04     $G' = ADD\_SLD\ (\Delta_{Tk}, \lambda, G)$
05     $G'' = G - G'$
06     $G' = Other\_Grou\ p( \Delta_T - \{\Delta_{Tk}\}, G'') \cup G'$
07     $\lambda = \lambda + 1$
08     $k = k + 1$
```

Fig. 6. Proposed new SP_TDP_RWA Algorithm

The function, SP_TDP_RWA() in Fig. 6, consists of three functions. TDP-Selector( ), ADD_SLD( ), and Other_Group( ). TDP-Selector() groups service-time disjoint SLDs, $\Delta_T = \{\Delta_{T1}, \Delta_{T2}, ..., \Delta_{Tk}\}$. ADD_SLD() assigns wavelength to set of SLDs in the checked Δ_{Tk}. Other_Group() is to find link-disjoint paths of SLDs in $\Delta_T - \{\Delta_{Ti}\}$ and assigns the wavelength to them. At step 2, λ is wavelength to be assigned to SLDs that are selected at each process, and k is number of elements in set Δ_T with $\Delta_T = \{\Delta_{T1}, \Delta_{T2}, ..., \Delta_{Tk}\}$. The steps 3-9 are repeated until all groups of Δ_T are checked. At step 05, links of G' are deleted from G to find link-disjoint paths for SLDs in $\Delta_T - \{\Delta_{Ti}\}$. Once all SLDs for a certain wavelength are checked and there are SLDs of which a wavelength is not assigned, the other wavelength is considered for them with *G*.

```
ADD_SLD ($\Delta_{Tk}, \lambda$, G)
01 for j = 1 to $|\Delta_{Tk}|$
02   Find $\delta_{kn} = (s_n, t_n, u_n, w_n)$ with the smallest cost at $\Delta_{Tk}$
03   Find shortest path P($s_n, t_n$) in G
04   new_path_distance = Find_New_Path($\delta_n$, G)
05   if new_path_distance $\leq d$
06     if new_path_distance $\leq$ (distance of ($s_n, t_n$) in G)
07       Assign $\lambda$ to new path of ($s_n, t_n$)
08       $G'(V', E') = G'(V', E') \cup$ link of new paht of ($s_n, t_n$)
10     else
11       Assign $\lambda$ to path of ($s_n, t_n$) in G
12       $G'(V', E') = G'(V', E') \cup$ path of ($s_n, t_n$) in G
13   else
14     Assign $\lambda$ to path of ($s_n, t_n$) in G.
15     $G'(V', E') = G'(V', E') \cup$ path of ($s_n, t_n$) in G
```

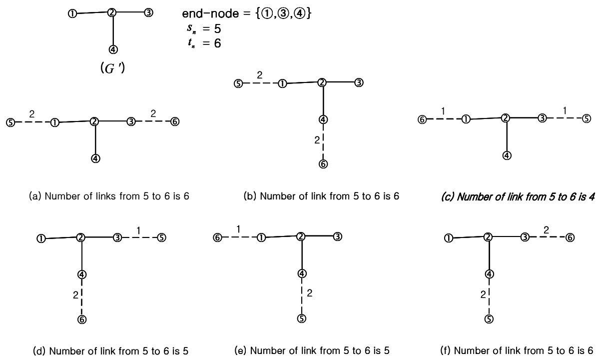
Fig. 7 ADD_SLD Algorithm

ADD_SLD( ) in Fig. 7 is the function to assign wavelength for $\Delta_{kj} = \{\delta_{k1}, \delta_{k2}, ..., \delta_{k|\Delta_{kj}|}\}$. The steps 1-15 are repeated until all $\delta_k \in \Delta_{kj}$ are checked. At step 02, we find $\delta_{kn} = (s_n, t_n, u_n, w_n)$ with the smallest number of links, and shortest path of (s_n, t_n) is computed at step 03 from *G*. Find_New_Path( ) at step 4 is the function to find new path reusing links of previously selected paths. Steps 5-12 are executed if the cost of new path is less than or equal to $\mathbf{max}(\mathbf{dim}(G), \sqrt{|E|})$. Steps 6-8 are executed when cost of new path is less than that of the shortest path in network *G*. At step 7, we assign λ to a new path, and add new links to G'. If the cost of new path is longer than that of shortest path in network *G*, steps 10-11 assign a wavelength to the shortest path, and add links of shortest path G'.

```
$\delta_n = \{s_n, t_n, u_n, w_n\}$
$end\_node = \{E_1, E_2, E_3, ......, E_l\}$
Find_New_Path($\delta_n$, G)
01  k = 1
02  while( k $\neq$ |end_node|)
03      m = 1
04      while(m $\neq$ |end_node|)
05          Find shortest path $P_1(s_n, E_k)$ in G
06          Find shortest path $P_2(E_k, E_m)$ in G'
07          Find shortest path $P_3(E_m, t_n)$ in G
08          if new_path_distance > distance of $P_1$ + distance of $P_2$ + distance of $P_3$
09              new_path_distance = $P_1 + P_2 + P_3$
10              new_path = $s_n \rightarrow E_k \rightarrow E_m \rightarrow t_n$
```

Fig. 8. Find_New_Path algorithm that find new path

The function, Find_New_Path( ), in Fig. 8 is to find new path reusing links of G' which means new path shares the links of paths previously selected. To concatenate new source and destination to previously selected paths, steps 2-10 are repeated until checking all end nodes of which degree of node is 1 in link G'. Steps 5-7 are to find a path that is from s_n to t_n via any end-node. Fig. 9 shows an example to find a new path of (5, 6). Path of (5, 6) in (c) is shortest, so this can be used as a new path.



(a) Number of links from 5 to 6 is 6 (b) Number of link from 5 to 6 is 6 *(c) Number of link from 5 to 6 is 4*

(d) Number of link from 5 to 6 is 5 (e) Number of link from 5 to 6 is 5 (f) Number of link from 5 to 6 is 6

Fig. 9 Example to find new path as share links of G'. Number on link represents the number of links between two nodes. If not represented, it means 1.

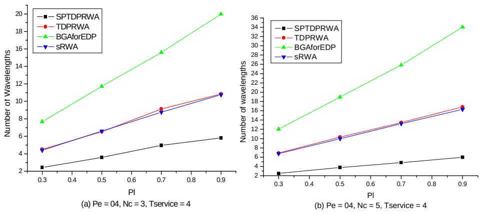
```
Other_Group($\Delta_T - \{\Delta_{Tk}\}, G''$)
01  i = 1
02  while (i $\neq$ |$\Delta_T - \{\Delta_{Tk}\}$|)
03      Find shortest path of $P(s_i, t_i)$ in G''
04      if $P(s_i, t_i)$ exists in G''
05          $G' = G' \cup P(s_i, t_i)$
06          $G'' = G'' - P(s_i, t_i)$
```

Fig. 10 Algorithm to find path to assign wavelength to SLD in other groups

Other_Group( ), in Fig. 10 is the function to find link-disjoint paths that do not share link with Δ_{Tk}, and assign the same wavelength to SLDs in $\Delta_T - \{\Delta_{Tk}\}$. We repeat the steps 2-6 until all SLDs in $\Delta_T - \{\Delta_{Tk}\}$ are checked. At step 3, we check to know whether a path from *s* to *t* is in G''. If disjoint path exists, add links of path to G', and then delete G' from G''. And then, we repeat SP_TDP_RWA() to assign new wavelength to SLDs that are not assigned starting from Δ_{Tk+1} again. In this manner, we can assign wavelengths to all SLDs.

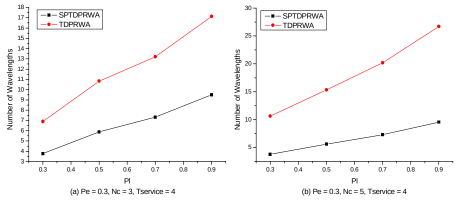
4. Performance Evaluation

We compare the performance of SP_TDP_RWA with those of BGAforEDP[4], TDP_RWA[6] and sRWA[10] in terms of the number of wavelengths. Network topologies used for performance evaluation are randomly generated directed networks [11]. In this evaluation, *Pe* is the probability of links in a network, therefore if *Pe* is 1, this means complete graph. *Pl* is the probability of SLDs in a network. If *Pl* is 1, this means that every request occurs between every two node pairs. If *Pl* is 0.3, then requests are used only 30% among every request that occurs between every two node pairs. Because several requests between two nodes are possible to occur, we evaluated performance when several requests occurred. If *Nc* is 3, this means that request occurs between two node pairs. *Tservice* means average service-time of requests. If *Tservice* is 4, this means that each service-time is 4 hours. In this paper, we supposed that network has 20 nodes. We implemented evaluations 1000 times for each evaluation condition and obtain the average number of wavelengths assigned.



(a) Pe = 04, Nc = 3, Tservice = 4 (b) Pe = 04, Nc = 5, Tservice = 4

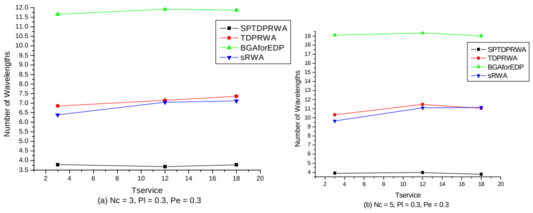
Fig. 11. Number of wavelengths as *Pl* increases (*Pe* = 0.4, *Nc* = 3 or 5, *Tservice* = 4)



(a) Pe = 0.3, Nc = 3, Tservice = 4 (b) Pe = 0.3, Nc = 5, Tservice = 4

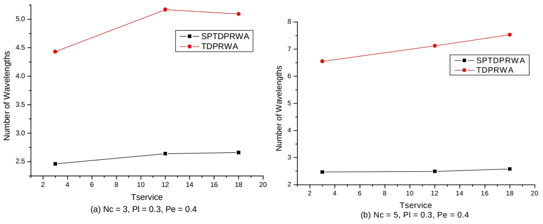
Fig. 12 Number of wavelengths as Pl increases (Pe = 0.3, Nc = 3 or 5, Tservice 4)

Fig. 11 shows number of wavelength as *Pl* increases in random networks with 20 nodes when *Nc* is 3 or 5, *Pe* is 0.4 and *Tservice* is 4 hours. And Fig. 12 shows number of wavelength as *Pl* increases when *Nc* is 3 or 5, *Pe* is 0.3 and *Tservice* is 4. As shown in Fig. 11 and 12, SP_TDP_RWA, our proposed algorithm, outperforms any other algorithm in terms of wavelength at any condition with fixed *Tservice*.



(a) Nc = 3, Pl = 0.3, Pe = 0.3 (b) Nc = 5, Pl = 0.3, Pe = 0.3

Fig. 13 Number of wavelengths as Tservice increases (Nc = 3 or 5, Pl = 0.3, Pe = 0.3)



(a) Nc = 3, Pl = 0.3, Pe = 0.4 (b) Nc = 5, Pl = 0.3, Pe = 0.4

Fig. 14 Number of wavelengths as Tservice increases (Nc = 3 or 5, Pl = 0.3, Pe = 0.4)

Fig. 13 shows number of wavelength as *Tservice* increases in random networks with 20 nodes when *Nc* is 3 or 5, *Pl* is 0.3 and *Pe* is 0.3. And Fig. 14 shows number of wavelength as *Tservice* increases when *Nc* is 3 or 5, *Pl* is 0.3 and Pe is 0.4. From Fig. 13 and 14, we may also conclude that SP_TDP_RWA outperforms TDP_RWA as *Tservice* is increased. Also, the number of wavelength using SP_TDP_RWA is very steady.

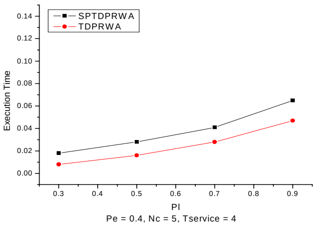


Fig. 15 Average execution time as Pl increases (Pe = 0.4, Nc = 5, Tservice = 4)

In Fig. 15, we illustrate average execution time as *Pl* increases in random directed networks when *Nc* is 5, *Pe* is 0.4, and *Tservice* is 4 hours. SP_TDP_RWA requires some more execution time than TDP_RWA, but this is small enough to ignore for whole performance.

SP_TDP_RWA is to minimize the number of links for selecting paths using single wavelength for a certain group so that probability of finding link-disjoint paths using the same wavelength is increased for SLDs in other groups. For this reason, SP_TDP_RWA outperforms TDP_RWA[6] from 44.4% to 65.8% in terms of wavelength with feasible execution time.

5 CONCLUSIONS

In this paper, we proposed new wavelength assignment algorithm, SP_TDP_RWA, for Scheduled Lightpath Demand (SLD) in OVPN. Using characteristics of the RWA of SLD, we made time-disjoint SLDs in the same group, and then we found paths sharing links of previously selected paths for SLDs in the same group to increase number of link-disjoint paths available for SLDs in other groups. As a result, SP_TDP_RWA dramatically decreased the number of wavelength. Also execution time is feasible when comparing TDA_RWA of which execution time is best among mechanisms introduced so far.

References

[1] I. Chlamtac, A. Ganz, and G. Karmi.: Lightpath Communications: An Approach to High-Bandwidth Optical WAN's," IEEE Transactions on Communications, vol. 40, no. 7, pp. 1171-1182, July 1992.

[2] R. Ramaswami and K. Sivarajan,: Routing and wavelength assignment in all-optical networks, IEEE/ACM Trans. Networking, vol. 3, pp. 489-500, October 1995.

[3] Zang, H., Jue, J.P., Mukherjee, B.: A Review of Routing and Wavelength Assignment Approaches for Wavelength-routed Optical WDM Networks, Optical Networks Magazine, Vol. 1. No. 1. (2000) 47–60

[4] Manohar, P., Manjunath, D., Shevgaonkar, R.K.: Routing and Wavelength Assignment in Optical Networks from Edge Disjoint Path Algorithms, IEEE Communications Letter, Vol. 5. (2002) 211–213

[5] Kirovski, D., Potkonjak, M.: Efficient Coloring of a Large Spectrum of Graphs, Proc. 35th Conf. Design Automation, (1998) 427–432

[6] Hyun Gi Ahn, Tae-Jin Lee, Min Young Chung, and Hyunseung Choo,: RWA on Scheduled Lightpath Demands in WDM Optical Transport Networks with Time Disjoint Paths, LNCS 3391, pp. 342-351, ICOIN 2005.

[7] Advanced Networking for Research and Education [Online]. Available: http://abilene.internet2.edu/

[8] Steven G. Finn, Richard A. Barry, Optical Services in Future Broadband Networks, IEEE Network Magazine Nov/Dec. 1996.

[9] T.E. Stern, K. Bala, S. Jiang and J. Sharony,: Linear Lightwave Networks: Performance Issues, IEEE/OSA Jr. of Lightwave Technology, May/June 1993.

[10] Kuri, J., Puech, N., Gagnaire, M., Dotaro, E., Douville, R.: Routing and Wavelength Assignment of Scheduled Lightpath Demands, IEEE Journal on Selected Areas in Communications, Vol. 21. No. 8. (2003) 1231-1240
[11] A.S. Rodionov and H. Choo,: On Generating Random Network Structures: Connected Graphs, Springer-Verlag Lecture Notes in Computer Science, vol. 3090, pp. 483-491, September 2004.
[12] J. S. Choi, N. Golmie, F. Lapeyrere, F. Mouveaux, and D. Su,: A functional classification of routing and wavelength assignment schemes in DWDM networks: Static case, in Proc. VII Int. Conf. on Optical Communication and Networks, Jan. 2000.

Soo-Yeon Park received the B.S. degree in Department of Computer Engineering from Korea Polytechnic University, Korea in 2004, the M.S. degree in Department of Computer Engineering from Korea Polytechnic University, Korea in 2007. She is the Ph.D. degree candidate in Department of Computer Engineering from Korea Polytechnic University, Korea. Her research interests include routing algorithms, optical networks, mobile computing, network mobility, and network security.

Young-Cheol Bang received the B.S. degree in computer science from University of Oklahoma, Norman, OK in 1994, the M.S. degree in computer science from the University of Oklahoma, Norman, OK in 1997, and the Ph.D. degree in computer science from the University of University, Norman, OK in 20000. From 2000 to 2002, he was a Senior Researcher at Electronics and Telecommunications Research Institute, Daejeon, Korea, Dr. Bang is currently an Assist-Engineering at Korea Polytechnic University. His research interests include routing protocols, optical networks, mobile computing, and parallel and distributed computing.

JongS Yang received the B.S.(Korea Advanced Institute Science & Technology, in 1990), M.S (Univ. of Seoul, in 1998) and Ph.D (Kyung Hee Univ. in 2003) respectively. After working as a researcher (from 1990) in the semiconductor Division, Electronics and Telecommunications Research Institute, he has been an principal researcher at Korea Institute of industrial Technology Evaluation & Planning, since 1992. His research interest includes digital signal processing, medical device, orient medical science, mobile computing, routing protocols, optical networks, home networking, and multimedia systems.