Technology Management, Economics, and Policy Discussion Paper Series

SEOUL NATIONAL UNIVERSITY
College of Engineering
Technology Management, Economics, and Policy Program

Disjoint Paths Pair Computation Procedure for SDH/SONET Networks

A. Mohammad Hassan, Jörn Altmann

TEMEP Discussion Paper No. 2013:107

서 울 대 학 교
공 과 대 학, 기술경영경제정책 대학원과정
151-742 서울시 관악구 관악로 599

Technology Management, Economics, and Policy Program
College of Engineering, Seoul National University
599 Gwanak-Ro, Gwanak-Gu, Seoul 151-742, South-Korea
Phone: ++82-2-880-9140, Fax: ++82-2-880-8389

The *Technology Management, Economics, and Policy Program (TEMEP)* is a graduate program at Seoul National University. It includes both M.Sc. and Ph.D. programs, which are integrated into the graduate programs at the College of Engineering.

The *TEMEP Discussion Paper Series* is intended to serve as an outlet for publishing research about theoretical, methodological, and applied aspects of industrial economics, especially those related to the institute's areas of specialization, namely management of technology, information and communication technology, telecommunication, services, health industries, energy industries, as well as infrastructures for development. In particular, paper submissions are welcome, which analyze current technology and industry related issues, discuss their implications, and suggest possible alternative policies. The objective is to gain insights into important policy issues and to acquire a balanced viewpoint of policy making, technology management, and economics. It will enable us to identify the problems in industries accurately and to come up with optimal and effective guidelines. In addition, another important aim of this series is to facilitate communication with external research institutes, individual researchers, and policy makers worldwide.

Research results disseminated by TEMEP may include views on policies for information and communication technologies, technology management, telecommunication, energy, health, and development economics, but the institute itself takes no institutional policy position. If appropriate, the policy views of the institute are disseminated in separate policy briefs. Thus, any opinion expressed in the TEMEP Discussion Paper Series is those of the author(s) and not necessarily the opinion of the institute.

Finally, the current editor of this series would like to thank Prof. Dr. Almas Heshmati for establishing this working paper series in January 2009 and for guiding its development during the course of the first 55 issues. In particular, Prof. Heshmati provided invaluable contributions for setting up the local IT infrastructure, registering the series with RePEc, disseminating the working papers, and setting the quality requirements.

Jörn Altmann, Editor

Office: 37-305
Technology Management, Economics, and Policy Program
College of Engineering
Seoul National University
599 Gwanak-Ro, Gwanak-Gu
Seoul 151-742
South-Korea

Phone: +82-70-7678-6676
Fax: +1-501-641-5384
E-mail: jorn.altmann@acm.org

Disjoint Paths Pair Computation Procedure for SDH/SONET Networks

A.M. Hassan[1], Jörn Altmann[2]

[1] College of Engineering
Seoul National University
Seoul, South Korea
eng.hassan2009@yahoo.com

[2] Department of Industrial Engineering
College of Engineering
Seoul National University
Seoul, South-Korea
jorn.altmann@acm.org

October 2013

Abstract: The increasing demand for bandwidth and the error-prone, costly, and long-lasting service provisioning process force carriers to find new ways of automatic service provisioning. Despite the introduction of numerous path computation procedures, suitable for WDM-technology-based networks, very few studies exist that discusses the SDH/SONET specific multiplexing requirements. Furthermore, the existing disjoint path computation procedures do not always find the best possible path between two nodes. In this paper, we propose a path computation procedure capable of addressing the SDH/SONET multiplexing requirements and the shortcomings of the existing disjoint path computation procedures. Our disjoint paths pair computation procedure is applied to the topology of the NSF.net network. The simulation results, obtained with Matlab, suggest that the proposed procedure outperforms existing path computation procedures. The proposed procedure covers the demand for bandwidth with fewer resources. Furthermore, it finds paths considering the capacity units of SDH/SONET. It is also observed that the time complexity is tolerable.

Keywords: Resource allocation, disjoint paths, path computation procedure, SDH/SONET technology, optimal path selection, transport networks, service provisioning.

JEL Classification Numbers: C61, L86, L96, M15.

1. Introduction

In the current environment, in which data communication has surpassed voice communication, and the use of bandwidth-intensive applications (e.g., IP-TV and P2P) became part of daily live, network providers are flooded with requests for network capacity.

SDH/SONET is the main transport network technology, which is used for providing capacity and connectivity to the upper layer (i.e., IP layer). However, due to the static nature of SDH/SONET, the operation of IP networks suffers from high operational costs at the transport layer. Several scholars pointed out that the service provisioning process cost covers significant part of the operational expenses of transport service providers [1] [2] [3] [4]. In order to provision a network service, network providers have to adjust their network management system and execute manual and time-consuming activities [2]. This cost issue can get even more serious, considering the fact that provisioning of differentiated services to enterprise customers develops to be one of the main sources of revenue for network service providers [5].

Therefore, it is necessary to provision network services with the shortest possible time. By implementing an automated service provisioning process [20], network operators may save substantial operational expenditure [1],[21]. As part of this, there is a need for automating the computation of paths between nodes.

To address the need for a multi-domain, multi-layer, and fast path computation procedure, the Internet Engineering Task Force (IETF) introduced the Path Computation Element Protocol (PCEP) and the architecture for Path Computation Element (PCE) [6].

The path computations problem is mostly studied as routing and wavelength assignments by many scholars. They have proposed numerous procedures, facilitating automatic computation and allocation of network resources between the IP/MPLS routers. Despite the existence of many solutions addressing routing and wavelength assignment problems in WDM network, very little research has been done on how to deal with the SDH/SONET technology specific requirements (i.e., with its none-linear multiplexing structure). Furthermore, the existing disjoint path computation procedures do not always find the best possible path between two nodes.

In this paper, the objective is to propose a path computation procedure that is capable of addressing the SDH/SONET multiplexing requirements and the shortcomings of the existing disjoint path computation procedures.

The contribution of this paper is a new disjoint paths pair computation procedure, which works for SDH/SONET networks and considers the joined cost of disjoint paths.

The remainder of this paper is organized as follows. Section 2 describes related works on path computation and discusses the SDH/SONET multiplexing requirements. Section 3, we describe the advantages of our disjoint paths pair computation procedure over existing path computation procedures. Section 4 introduces our disjoint paths pair computation procedure. Section 5 presents the workings of our procedure within the NSFnet topology and shows its performance advantages over existing path computation procedures. Finally, section 6 concludes the paper with a brief summary and discussion.

2. Path Computation and SDH/SONET

The path computation problem, also known as the routing and wavelength assignment (RWA) problem, has been studied by scholars for a long time. A brief overview is given with the following 12 literature references. The problem of searching for diverse routes with shared links was studied in [7]. A graph transformation technique addressing the adaption constraints has been proposed in [8]. An integrated, topology-independent solution to the RWA problem has been introduced in [9]. The multi-constraint shortest path problem (MCSP), which finds a feasible path subject to multiple additive constraints, has been studied in [10]. The K-shortest path problem has been studied in [11]. A path computation algorithm that assumes independent arrivals of requests for bandwidth has been analyzed in [12]. A QoS routing algorithm (SAMCRA) that has been designed for multi-constraint path search has been introduced by [13]. Path computation and traffic grooming has been described in [14] [15] [16]. An open source PCE, called Network Aware Resource Broker (NARB) has been designed and developed within the DRAGON project [17].

The majority of studies that investigated the path computation problem have one thing in common. For their studies, they assume a network model that allows splitting the link capacity into arbitrary portions [18]. However, this assumption does not hold for SDH/SONET-based transport networks due to their specific multiplexing structure [19]. This structure only allows splitting the link capacity into specific portions, which must be considered when searching for paths in SDH/SONET networks. Only few studies considered this. A path computation procedure for dynamic service provisioning with protection and inverse multiplexing for SDH/SONET networks discribed in [18]. The authors introduced two variables, α to give priority to links having available trails and β to consider the trail's capacity. Although their findings suggest an approximation of values for α and β, a generalization does not work as networks can differ significantly. Networks have different protection mechanism, node degree, priority levels, and policies. Therefore, the suggested values for α and β may not work for all types of network. In contrast, our disjoint paths pair computation procedure considers the exact number and types of available VCs that can fulfill the required bandwidth.

The SDH/SONET multiplexing structure is defined through the transmission unit of SDH-technology, which is a synchronous transport module (STM-1) consisting of multiple virtual containers (VC) of different capacities. The bandwidth assignment needs to consider this. For example, in order to provision a service with a capacity of 2 Mbps (i.e., equivalent of 1 VC12), a higher level container needs to be used. By assigning 1 VC12 to a STM-1, which would be capable of carrying a VC4 (150 Mbps), the STM-1 link cannot carry a VC4 anymore. Therefore, the container needs to be sub-divided into 2 VC3 and a third VC3 that is further subdivided into multiples of VC12 (For more details on the SDH standard hierarchy, please refer to [19]). Due to this specific multiplexing structure, the capacity of a SDH/SONET link cannot be split into arbitrary capacity units and, therefore, cannot fulfill all demand for bandwidth [18]. Consequently, there is a need for a procedure that addresses this specific SDH/SONET technology requirement when computing paths in SDH/SONET networks.

3. Shortcomings of Existing Disjoint Path Computation Procedures

Existing procedures for finding a pair of disjoint paths between a source node and a sink node do not always find the optimal pair of disjoint paths. For instance, if we consider the network shown in Figure 1, existing path computation procedures do not find the lowest-cost pair of disjoint paths between source node 1 and sink node 6.

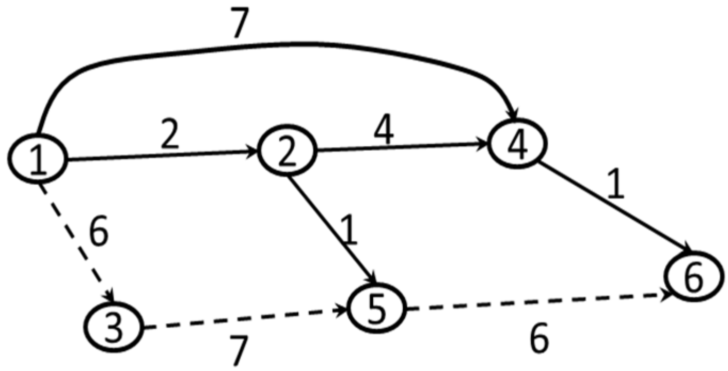

Figure 1. Example network.

Using Dijkstra's algorithm to compute the disjoint paths between the two specified nodes, the first shortest path between the source node 1 and sink node 6 is the path that originates at node 1, passes node 2 and node 4, and ends at sink node 6. The total cost of the path is 2 + 4 + 1 = 7. Then, following the current practice, we remove this path (i.e., the nodes and links between the source node and the sink node) from the network topology and execute Dijkstra's algorithm again on the remaining network for finding the second path, which is disjoint to the first path. With respect to the example of Figure 1, it means that Dijkstra's algorithm finds path 1-3-5-6 (dashed lines) after removing the first path (i.e., 1-2-4-6 but not the source node 1 and the sink node 6) from the topology graph. As path 1-3-5-6 has a cost of 19, the total cost for both disjoint paths is 7 + 19 = 26.

The problem with the currently proposed path computation procedures is that they do not find all possible pairs of disjoint paths. Actually, there are four paths between node 1 and node 6 in our example network. Those paths are 1-2-4-6, 1-4-6, 1-2-5-6, and 1-3-5-6. Each costs 7, 8, 9, and 19 respectively. Among these paths, there are three pairs of disjoint paths that connect the specified nodes (1-2-4-6, 1-3-5-6), (1-4-6, 1-2-5-6), and (1-4-6, 1-3-5-6). Since we search for the lowest cost pair of disjoint paths between node 1 and node 6, the pair of paths (1-4-6, 1-2-5-6) is chosen. Its total cost is only 8 + 9 = 17, compared to the cost of the other pairs of disjoint paths, 26 and 27. Therefore, we can state that the existing disjoint path calculation procedures do not always find the lowest cost solution. Their result is 26, 9 units more costly.

4. Proposed Path Computation Procedure

To address the SDH/SONET-specific requirements, we propose to consider the SDH/SONET standard units of capacity for our disjoint path computation. For instance, if a path with a capacity of 2 Mbps between two nodes has to be found in a SDH-technology-based network, the path calculation procedure has to search for a path capable of supporting a VC12 signal, which runs at the rate of 2.304 Mbps [19]. Therefore, we suggest searching for paths of a specific VC type instead of a specific bit rate (e.g., 1 Mbps). For the path search, only the sub-graph of the original SDH-based network, which can fulfill the specific VC type, is considered. This way, it becomes possible to compute paths in SDH-based networks correctly.

In addition to this, our procedure avoids the shortcomings of existing disjoint path computation procedures. It always finds the lowest-cost pair of disjoint paths in a transport network by searching for all possible paths first and, then, checking pairs of paths for being disjoint.

```
Generate graph G(V, E) from the topology of the network;
Remove all links with a capacity less than demanded by the provisioning request
from the network graph;
Find all possible paths between source node and sink node, using a BFS algorithm
and store the paths in the list of paths P;
FOR í = 1 TO number-of-paths-in-P {
    Calculate path cost Y(i) for path í;
} END FOR
FOR í =1 TO number-of-paths-in-P {
  FOR j = í TO number-of-paths-in-P {
      IF (path í and path j are disjoint) THEN {
          Calculate Z by summing up path cost Y(i) of path i and path cost Y(j)
          of path j;
          Add the disjoint paths pair (i,j) and its cost Z to the list of disjoint
          paths DP;
      } END IF
  } END FOR
} END FOR
lowest-cost = max-integer;
FOR k = 1 TO number-of-paths-in-DP {
  IF (lowest-cost > DP(k).Z) THEN {
      lowest-cost = DP(k).Z;
      lowest-cost-path = DP(k).pair;
  } END IF
} END FOR
RETURN(lowest-cost, lowest-cost-path);
```

Figure 2. Proposed disjoint path computation procedure.

In detail, our path computation procedure builds first a network graph G = (V, E), containing n nodes $v \in V$ and m edges $(v, w) \in E$, and represents the graph in an adjacency matrix. In the second step, a slightly modified version of the breadth first search (BFS) algorithm is used to find all possible paths P between a source node and a sink node in the graph G. In the third step, the cost Yi(s, d) of all paths is calculated. It is assumed that Ci is the cost of an edge (v, w) from node v to node w. Then, the cost of a path with k edges from the source node s to the sink node d is as defined in the following equation:

$$Yi\ (s,\ d) = \sum_{1}^{k} Ci \qquad (1)$$

In step 4, all possible pairs of disjoint paths are generated from the list of all possible paths and the sum of their costs is calculated. In the final step, from all the pairs of disjoint paths, the pair of paths is selected that has the lowest cost. The detailed algorithm of our disjoint paths pair computation procedure is given in Figure 2.

Furthermore, a pair of disjoint paths between the source node and the sink node is assumed to be appropriate, if the quality of service QoS of the two disjoint paths is less than a threshold QoS user set by the user.

5. Analysis

To describe how the algorithm works, we consider an example network, simulate a set of path computations, and calculate the savings achieved with our procedure.

5.1.Simulation Assumptions

For the simulation, we consider the NSFnet network topology (Figure 3). Furthermore, we assume that the bandwidth demanded by the customer is equivalent of 7 VC12 and that the customer prefers the lowest cost pair of disjoint paths between source node 1 and sink node 13. The links of the network have the following characteristics. The initial link capacity (IC) of each link is set to STM-16. The allocated capacity (AC) for each link is set randomly using a uniform distribution in the range between 1VC12 and 1008VC12 (Table 1). The link cost is represented by the link length, which is set according to a setting of the NSFnet network.

Table 1. Simulation variables, variable units, and their descriptions.

Variable	Unit	Description
IC	SDH/SONET	Each link has a fixed capacity of a STM-16 (i.e., 1008 VC12 ≈ 2Gbps)
AC	VC12	Traffic is assigned to each link in the range between 1 VC12 and 1008 VC12 according to a uniform distribution.
Cost	km	The link cost (i.e., link length) is fixed and depends on the length of the links within the NSFnet network. The links vary between 1 km and 100 km

5.2. NSFnet Network Example

After generating the graph (i.e., a 14x14 adjacency matrix) from the network (Figure 3), the algorithm removes the links with a capacity less than 7VC12 (as this is capacity demanded by the user) from the graph. In our example, as shown in Figure 3, none of the links have been removed though. In the next step, the path computation algorithm searches for all possible paths between the source node (node 1) and the sink node (node 13) by means of a BFS algorithm. The result of the BFS is a list of 9 different paths between the source node and the sink node. The paths are (1-4-12-13), (1-2-8-10-13), (1-4-5-6-11-13), (1-4-5-6-9-10-13), (1-4-5-7-8-10-13), (1-3-6-11-13), (1-2-3-6-11-13), (1-3-6-9-10-13), and (1-2-3-6-9-10-13). Using these paths, 36 pairs of paths can be generated. However, only nine of these 36 pairs have disjoint paths (Table 2).

Table 2. All pairs of disjoint paths found between node 1 and node 13.

No.	Two disjoint paths	Sum of the path cost (Path length)
1	1-2-8-10-13 & 1-4-12-13	235
2	1-2-8-10-13 & 1-4-5-6-11-13	**215**
3	1-2-8-10-13 & 1-3-6-11-13	225
4	1-2-3-6-11-13 & 1-4-5-7-8-10-13	275
5	1-3-6-11-13 & 1-4-12-13	250
6	1-3-6-11-13 & 1-4-5-7-8-10-13	**220**
7	1-4-12-13 & 1-2-3-6-11-13	305
8	1-4-12-13 & 1-2-3-6-9-10-13	310
9	1-4-12-13 & 1-3-6-9-10-13	255

Among all 9 combinations of disjoint paths between the specified source node and sink node, the paths (1-2-8-10-13) and (1-4-5-6-11-13) returned the lowest cost pair of disjoint paths with a total cost of 215 (Table 2). Figure 3 illustrates the two disjoint paths found (dotted lines).

To compare the proposed path computation procedure with existing disjoint path computation procedures on the NSFnet network (Figure 3), we execute Dijkstra's shortest path algorithm, on which the disjoint path computation procedures rely. The result of the first execution of Dijkstra's algorithm is the shortest path (i.e., lowest cost) path (1-4-5-7-8-10-13) with a total cost of 100. Then, in order to follow existing disjoint path computation procedures, we have to remove the first path (i.e., remove nodes and links of the first path, except for node 1 and node 13) from the graph. Based on the resulting graph, a disjoint path to the first path can be found by executing Dijkstra's algorithm again. The path found is (1-3-6-11-13) with a path cost of 120.

The total lengths (i.e., total cost) of the two paths found using the existing path computation procedure is equal to 220 km (dashed lines in Figure 3). The length of the two disjoint paths returned with our proposed disjoint paths pair computation procedure is 5 km shorter. This suggests that our disjoint path computation procedure performs better than the existing disjoint path computation procedures.

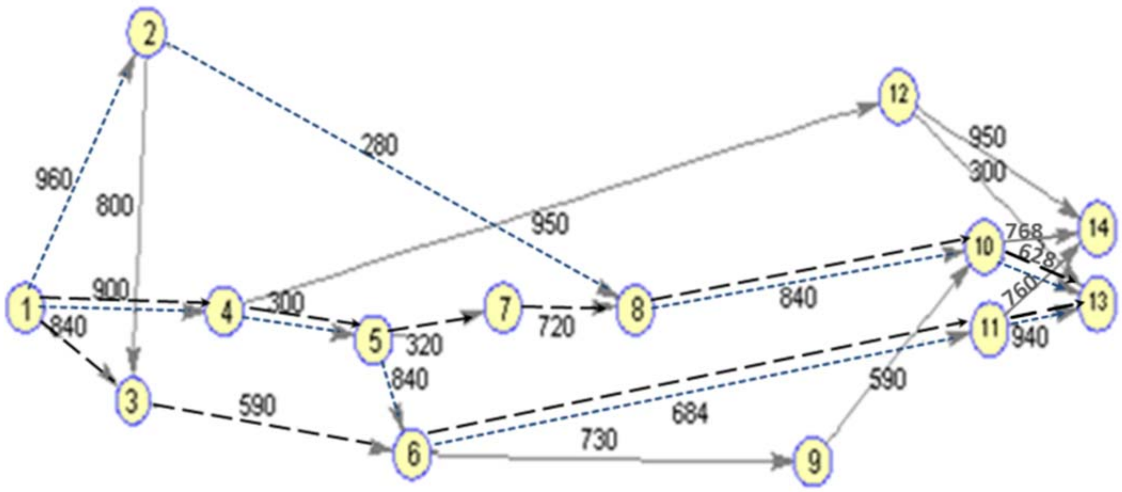

Figure 3. Disjoint paths found with the proposed disjoint paths pair computation procedure (dotted lines) and disjoint paths found with existing disjoint path computation procedures (dashed lines).

5.3. Evaluation of the Proposed Procedure

To evaluate the proposed disjoint paths pair computation procedure in more detail, we estimate the savings in capacity and the reduction of path length and, then, compare these values with those of the existing disjoint path computation procedures. For the evaluation, we run both disjoint path computation procedures on the NSFnet network with different source nodes and sink nodes for each simulation run. The nodes are chosen randomly using a uniform distribution. The values of the allocated capacity are also set randomly for each link and each simulation run (see Section V.A). The capacity of the disjoint paths that is requested is 7 VC12.

In order to be able to calculate the path length differences to existing path computation procedures, we calculate the lowest cost disjoint paths pair for each simulation run. The reduction of path lengths averaged across all simulation runs is shown in Figure 4. Figure 4 also shows the number of disjoint paths pairs averaged across all simulation runs, depending on the number of paths between the source node and the sink node. One to ten different paths have been found between the source node and the sink node in 30 simulation runs. The vertical axis on the left hand side describes the average frequency of pairs of disjoint paths. The vertical axis on the right hand side shows the average path length reduction.

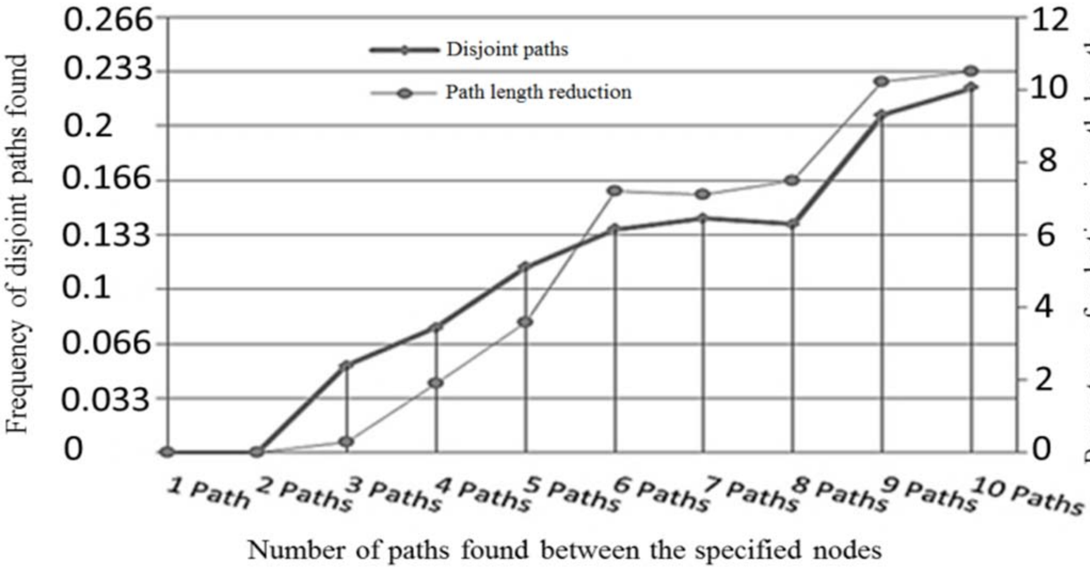

Figure 4. Average percentage of path length reduction and average frequency of disjoint paths found with the proposed disjoint path pair computation procedure over existing path computation procedures.

Figure 4 suggests that, if the number of paths found between the source node and the sink node increases, the average reduction in path length increases. It is also observed that, as the average number of pairs of disjoint paths found between the nodes increases, the average path length reduction also increases. The average reduction is at its highest, when 10 different paths are found between the two nodes. In contrast, the average reduction is zero, if only one or two paths are found between the source node and the sink node.

Another way of evaluating the proposed path computation procedure is the estimation of capacity savings. Figure 5 illustrates how much capacity will be saved, if the demand for path selections (e.g., through service provisioning) increases.

The capacity saving is calculated as a percentage of the total annual network capacity. In order to correctly estimate the capacity savings, we convert the initial representation of the link capacity (IC) from a single STM-16 signal to 144 signals of capacity 7 VC12. Specifically, as a STM-16 signal carries 1008 VC12 signals, a 1 km link with a capacity of STM-16 is equivalent to a 144 km link having a capacity of 7 VC12. Having the total length of the NSFnet network to be 5500 km and assuming a STM-16 signal on each link, the equivalent length of the network is 792000 km with a capacity of 7 VC12 per link.

The results shown in Figure 5 suggests that the capacity savings of the proposed disjoint paths pair calculation procedure increases as the demand for network services (e.g., service provisioning) increases. More clearly, for our example network, the average yearly capacity savings is found to be 0.288 percent of the total network capacity, if demand for only a single service provisioning per week exists. The average yearly capacity savings reaches 2.16 percent of the total network capacity, if the service provisioning demand increases to 7 times per week.

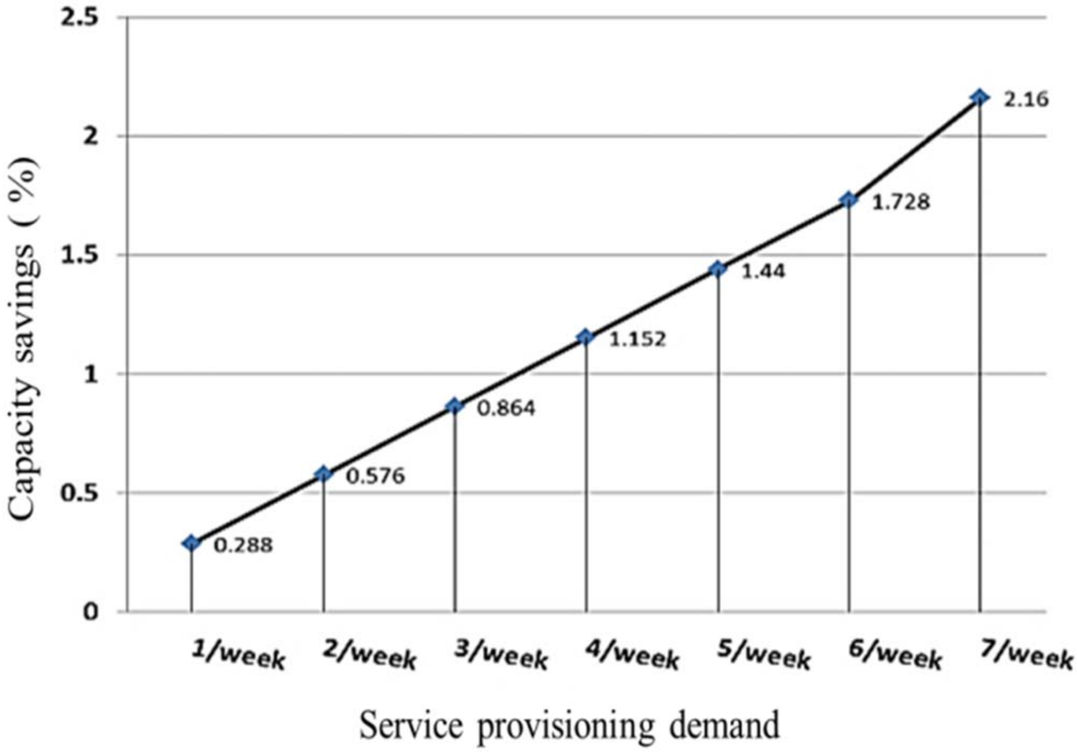

Figure 5. Capacity savings of the proposed disjoint paths pair computation procedure compared to the existing disjoint path computation procedures.

The time complexity of our procedure is $O[|P|^2 + |V| + |E|]$, where $|P|$ is the number of paths, $|V|$ the number of nodes, and $|E|$ the number of links. Its complexity is based on the BFS algorithm ($O[|V| + |E|]$). It is more complex than Eppstein's implementation of the K-shortest path algorithm ($O[|E| + |V|\log|V| + K]$), if $|P| \approx |V|$.

While the proposed algorithm can be more time consuming than the K-shortest path algorithm, it returns the lowest cost pair of disjoint paths. The existing disjoint path computation procedures, which return first the shortest path and, then, the shortest path of the remaining network, do not achieve that. Considering these results, we can state that our disjoint paths pair computation procedure achieves significant resource allocation improvement compared to existing procedures at a tolerable time complexity.

6. Conclusion

The continuously growing demand for bandwidth brings forward the need for an automatic provisioning process. Scholars mostly studied this problem as a routing and wavelength assignment problem, considering the WDM networks requirements but overlooking the SDH/SONET specific requirements. Furthermore, the existing disjoint path computation procedure does not always find the lowest-cost pair of disjoint paths between two nodes.

This paper introduced a disjoint paths pair computation procedure that considers the requirements of SDH/SONET and the finds the lowest-cost pair of disjoint paths.

Our simulations findings suggest that our disjoint paths pair computation procedure on the NSFnet topology saves network resources and, therefore, reduces cost for network providers. It achieves that at tolerable time complexity.

Acknowledgement

This research was performed within the European ONE project (contract no. 258300) and funded by the International Research and Development Program of the National Research Foundation of Korea. It is part of the Ministry of Education, Science and Technology of Korea (Grant number: K21001001625-10B1300-03310).

References

[1] R. Chahine, A. Kristadter, A. Iselt, and S. Pasqualini, “Operation cost reduction using ASON/ASTN,” Optical Society of America, 2005.

[2] S. Verbrugge, D. Colle, M. Pickavet, P. Demeester, S. Pasqualini, A. Iselt, A. Kirstädter, R. Hülsermann, F. Westphal, and M. Jäger, “Methodology and input availability parameters for calculating OpEx and CapEx costs for realistic network scenarios,” Journal of Optical Networking, vol. 5, no. 6, June 2006.

[3] S. Pasqualini, A. Kirstädter, A. Iselt, R. Chahine, S. Verbrugge D. Colle, and M. Pickavet, “Influence of GMPLS on network providers' operational expenditures,” IEEE Communication Magazine, July 2005.

[4] A. Kirstädter, A. Iselt, C. Winkler, and S. Pasqualini,"A quantitative study on the infuence of ASON/GMPLS on OPEX," International Journal of Electronics and Communication (AEÜ), Elsevier, vol. 60, no. 1, pp. 30-44, January 2006.

[5] W. Fawaz, B. Daheb, O. Audouin, B. Berde, M. Vigoureux, M. Du-Pond, and G. Pujolle, “Service level agreement and provisioning in optical networks,” IEEE Communication Magazine., vol. 42, no. 1, pp. 36-43, January 2004.

[6] J. P. Vasseur, and J. L. Le Roux, “Path computation element communication protocol,” RFC-5440, March 2009. [Accessed at http://datatracker.ietf.org/doc/rfc5440/?include_text=1 on August 2013]

[7] J. Q. Hu, “Diverse routing in optical mesh networks,” IEEE Transaction on Communication, vol. 51, no. 3, pp. 489-494, March 2003.

[8] B. Jabbari, S. Gong, and E. Oki, “On constraints for path computation in multi-layer switched networks.” IEICE Transactions, vol. 90-B, no. 8, pp. 1922-1927, August 2007.

[9] S. Janardhanan, A. Mahanti, D. Saha, and S. K. Sadhukhan, “A routing and wavelength assignment (RWA) technique to minimize the number of SONET ADMs in WDM rings,” 39th HICSS, pp.1-10, January 2006.

[10] F. Kuipers, T. Korkmaz, M. Krunz, and P. V. Mieghem, “An overview of constraint-based path selection algorithms for QoS routing,” IEEE Communications Magazine, December 2002.

[11] Z. Ziang and J. Zhao, "Multi- constraint-pruning: an algorithm for finding K shortest paths subject to multiple constraints," IEICE 08 SB 0083, APCC 2008, 2008.

[12] G. Banerjee and D. Sidhu, "Path computation for traffic engineering in MPLS networks," Springer, 2001.

[13] P. V. Mieghem, and F. A. Kuipers, "Concepts of exact quality of service algorithms," IEEE/ACM Transactions on Networking, vol. 12, no. 5, pp. 851-864, October 2004.

[14] E. Oki, K. Shiomoto, and S. Daisaku, "Dynamic multilayer routing schemes in GMPLS-based IP+optical networks," IEEE Communications Magazine, vol. 43, no. 1, pp. 108-114, January 2005.

[15] S. Zhang and B. Ramamurthy, "Dynamic traffic grooming algorithms for reconfigurable SONET over WDM networks," IEEE Journal on Selected Areas of Communications, vol. 21, no. 7, pp. 1165-1172, 2003.

[16] K. Zhu, H. Zhu, and B. Mukherjee, "Traffic engineering in multigranularity heterogeneous optical WDM mesh networks through dynamic traffic grooming," IEEE Network, vol. 17, no. 2, pp. 8-15, 2003.

[17] T. Lehman, J. Sobieski, and B. Jabbari, "DRAGON: A framework for service provisioning in heterogeneous grid networks," IEEE Communications Magazine, vol. 44, No. 3, pp. 84-90, March 2006.

[18] M. Ramachandran, N. Usha Rani, and T. A. Gonsalves, "Path computation algorithms for dynamic service provisioning with protection and invers multiplexing in SDH/SONET networks," IEEE/ACM Transaction on Networking, vol. 18, no. 5, October 2010.

[19] SDH "Telecommunication standard primer," Tektronix, [Accessed at www.tektronix.com/optical on March 2013].

[20] M. Yannuzi, A. Jukan, X. Masip-Bruin, M. Chamania, R. Serral-Gracia, V. Lopez, O. Gonzalez de Dios, A. Azañon, M. Maciejewski, C. Brunn, M. Roth, J. Altmann, "The Internet and transport network management ecosystems: a roadmap toward convergence," Optical Networking Design and Modeling (ONDM 2012), April 2012.

[21] A. M. Hassan, J. Altmann, V. López, "Control Plane Framework Emergence and its Deployment Cost Estimation," Conference of Telecommunication, Media and Internet Techno-Economics (CTTE 2012), June 2012.

TEMEP Discussion Papers

2010-57: Kibae Kim, Jörn Altmann and Junseok Hwang, “Measuring and Analyzing the Openness of the Web2.0 Service Network for Improving the Innovation Capacity of the Web2.0 System through Collective Intelligence”

2010-58: Alireza Abbasi and Jörn Altmann, “A Social Network System for Analyzing Publication Activities of Researchers”

2010-59: Jörn Altmann, Costas Courcoubetis and Marcel Risch, “A Marketplace and its Market Mechanism for Trading Commoditized Computing Resources”

2010-60: Fatmawati Zifa and Jörn Altmann, “A Empirically Validated Framework for Limiting Free-Riding in P2P Networks Through the Use of Social Network Information”

2010-61: Ashraf Bany Mohammed, Jörn Altmann and Junseok Hwang, “Cloud Computing Value Chains-Understanding Business and Value Creation in the Cloud”

2010-62: Ashraf Bany Mohammed and Jörn Altmann, “A Funding and Governing Model for Achieving Sustainable Growth of Computing e-Infrastructures”

2010-63: Junseok Hwang, Jihyoun Park and Jörn Altmann, “Two Risk-Aware Resource Brokering Strategies in Grid Computing: Broker-Driven vs. User-Driven Methods”

2010-64: Juthasit Rohitratana and Jörn Altmann, “Agent-Based Simulations of the Software Market under Different Pricing Schemes for Software-as-a-Service and Perpetual Software”

2010-65: Radoslaw R. Okulski and Almas Heshmati, “Time Series Analysis of Global Airline Passengers Transportation Industry”

2010-66: Alireza Abbasi and Jörn Altmann, “On the Correlation between Research Performance and Social Network Analysis Measures Applied to Research Collaboration Networks”

2010-67: Kibae Kim, Jörn Altmann and Junseok Hwang, “An Analysis of the Openness of the Web2.0 Service Network Based on Two Sets of Indices for Measuring the Impact of Service Ownership”

2010-68: Bory Seng, “The Driving Forces Underlying the Growth of Total Factor Productivity in Cambodia, Quantitative and Qualitative Studies”

2010-69: Michael Maurer, Vincent C. Emeakaroha, Ivona Brandic and Jörn Altmann, “Cost and Benefit of the SLA Mapping Approach for Defining Standardized Goods in Cloud Computing Markets”

2010-70: Khin Swe Latt and Jörn Altmann, “A Cost-Benefit-Based Analytical Model for Finding the Optimal Offering of Software Services”

2010-71: Jung Eun Lee, Younghoon Kim, Yeonbae Kim and Donghyuk Choi, “The Impact of Technology Licensing Payment Mechanisms on Firms’ Innovative Performance”

2010-72: Dongook Choi and Yeonbae Kim, “Effects of Piracy and Digital Rights Management on the Online Music Market in Korea”

2011-73: Tai-Yoo Kim, Jihyoun Park, Eungdo Kim and Junseok Hwang, “The Faster-Accelerating Digital Economy”

2011-74: Ivan Breskovic, Michael Maurer, Vincent C. Emeakaroha, Ivona Brandic and Jörn Altmann, “Towards Autonomic Market Management in Cloud Computing Infrastructures”

2011-75: Kiran Rupakhetee and Almas Heshmati, “Rhetorics vs. Realities in Implementation of e-Government Master Plan in Nepal”

2011-76: Alireza Abbasi, Jörn Altmann and Liaquat Hossain, “Identifying the Effects of Co-Authorship Networks on the Performance of Scholars: A Correlation and Regression Analysis of Performance Measures and Social Network Analysis Measures”

2011-77: Ivona Brandic, Michael Maurer, Vincent C. Emeakaroha and Jörn Altmann, “Cost-Benefit Analysis of the SLA Mapping Approach for Defining Standardized Cloud Computing Goods”

2011-78: Kibae Kim and Jörn Altmann, “A Complex Network Analysis of the Weighted Graph of the Web2.0 Service Network”

2011-79: Jörn Altmann, Matthias Hovestadt and Odej Kao, “Business Support Service Platform for Providers in Open Cloud Computing Markets”

2011-80: Ivan Breskovic, Michael Maurer, Vincent C. Emeakaroha, Ivona Brandic and Jörn Altmann, “Achieving Market Liquidity through Autonomic Cloud Market Management”

2011-81: Tai-Yoo Kim, Mi-Ae Jung, Eungdo Kim and Eunnyeong Heo, “The Faster-Accelerating Growth of the Knowledge-Based Society”

2011-82: Mohammad Mahdi Kashef and Jörn Altmann, “A Cost Model for Hybrid Clouds”

2011-83: Daeho Lee, Jungwoo Shin and Junseok Hwang, “Application-Based Quality Assessment of Internet Access Service”

2011-84: Daeho Lee and Junseok Hwang, “The Effect of Network Neutrality on the Incentive to Discriminate, Invest and Innovate: A Literature Review”

2011-85: Romain Lestage and David Flacher, “Access Regulation and Welfare”

2012-86: Juthasit Rohitratana and Jörn Altmann, “Impact of Pricing Schemes on a Market for Software-as-a-Service and Perpetual Software”

2012-87: Bory Seng, “The Introduction of ICT for Sustainable Development of the Tourism Industry in Cambodia”

2012-88: Jörn Altmann, Martina Meschke and Ashraf Bany Mohammed, “A Classification Scheme for Characterizing Service Networks”

2012-89: Nabaz T. Khayyat and Almas Heshmati, “Determinants of Mobile Telecommunication Adoption in the Kurdistan Region of Iraq”

2012-90: Nabaz T. Khayyat and Almas Heshmati, “Determinants of Mobile Phone Customer Satisfaction in the Kurdistan Region of Iraq”

2012-91: Nabaz T. Khayyat and Jeong-Dong Lee, “A New Index Measure of Technological Capabilities for Developing Countries”

2012-92: Juseuk Kim, Jörn Altmann and Lynn Ilon, “Using Smartphone Apps for Learning in a Major Korean University”

2012-93: Lynn Ilon and Jörn Altmann, “Using Collective Adaptive Networks to Solve Education Problems in Poor Countries”

2012-94: Ahmad Mohammad Hassan, Jörn Altmann and Victor Lopez, “Control Plane Framework Emergence and its Deployment Cost Estimation”

2012-95: Selam Abrham Gebregiorgis and Jörn Altmann, “IT Service Platforms: Their Value Creation Model and the Impact of their Level of Openness on their Adoption”

2012-96: Ivan Breskovic, Jörn Altmann and Ivona Brandic, “Creating Standardized Products for Electronic Markets”

2012-97: Netsanet Haile and Jörn Altmann, “Value Creation in IT Service Platforms through Two-Sided Network Effects”

2012-98: Soyoung Kim, Junseok Hwang and Jörn Altmann, “Dynamic Scenarios of Trust Establishment in the Public Cloud Service Market”

2012-99: Lynn Ilon, “How Collective Intelligence Redefines Education”

2013-100: Ivan Breskovic, Ivona Brandic and Jörn Altmann, “Maximizing Liquidity in Cloud Markets through Standardization of Computational Resources”

2013-101: Juseuk Kim, Lynn Ilon and Jörn Altmann, “Adapting Smartphones as Learning Technology in a Korean University”

2013-102: Baseem Al-Athwari, Jörn Altmann and Almas Heshmati, “A Conceptual Model and Methodology for Evaluating E-Infrastructure Deployment and Its Application to OECD and MENA Countries”

2013-103: Mohammad Mahdi Kashef, Azamat Uzbekov, Jörn Altmann and Matthias Hovestadt, “Comparison of Two Yield Management Strategies for Cloud Service Providers”

2013-104: Kibae Kim, Wool-Rim Lee and Jörn Altmann, “Patterns of Innovation in SaaS Networks: Trend Analysis of Node Centralities”

2013-105: Netsanet Haile and Jörn Altmann, “Estimating the Value Obtained from Using a Software Service Platform”

2013-106: A. Mohammad Hassan and Jörn Altmann, “Business Impact Analysis of a Mediator between the Network Management Systems of the IP/MPLS Network and the Transport Network”

2013-107: A. Mohammad Hassan and Jörn Altmann, “Disjoint Paths Pair Computation Procedure for SDH/SONET Networks”