Survivable Node-Disjoint Routing in Multi-Domain Networks

Maria Samonaki, Cristian Bermudez Serna and Carmen Mas-Machuca
Chair of Communication Networks (LKN), Technical University of Munich (TUM), Munich, Germany
maria.samonaki@tum.de, cristian.bermudez-serna@tum.de, cmas@tum.de

***Abstract*—This paper aims at finding node-disjoint paths in multi-domain networks while avoiding to disclose each domain topology and minimizing routing cost. In order to maintain the privacy of the domains, the proposed solutions exploit a full mesh Topology Aggregation scheme that limits the exchanged information. Each domain provides information only on the existence and total cost of the two shortest node-disjoint paths for every pair of aggregated links. This information is then utilized on the inter-domain aggregated topology for the computation of two node-disjoint paths with minimum cost for every demand in the network. Four approaches are proposed and evaluated in terms of average cost per demand and blocking probability. Their performance is also compared to the respective approaches for link-disjoint routing. Two of the proposed node-disjoint routing schemes keep low blocking probability (with a median of 0-3 %), while incurring 5 % higher cost when compared to link-disjoint routing.**

***Index Terms*—Multi-domain networks; Survivability; Topology Aggregation; Intra-domain disjointness information**

I. Introduction

With the ongoing advances in networking technologies, communication networks are characterized by high transmission rates, which means that failures can lead to the loss of significant amount of data. Many research studies address higher connection availability and network survivability while aiming to reduce the costs. Providing survivability becomes a greater challenge for multi-domain networks [1]–[4]. A multi-domain network consists of several single-domain networks connected by inter-domain links. Each domain is considered as an independent network that has its own rules of operation and management [5]. An *internal node* can view only local network information, thus it supports only intra-domain routing. A *border node* (e.g., Internet Exchange points (IXPs)) can view both the local and the global network information; it can perform both intra-domain and inter-domain routing in the multi-domain network.

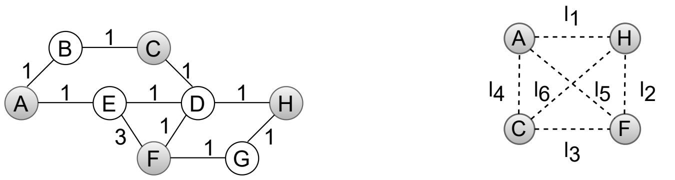
Due to scalability constraints and confidentiality issues, domain-specific information, such as the exact topology, resources, service availability, etc., is usually not shared externally. In order to maintain privacy, domain operators utilize Topology Aggregation (TA), to represent the physical topology as an aggregated logical topology, where aggregated links connect the border nodes of the domain. The work in [6] proposes a TA model in order to provide survivability in multi-domain optical networks using p-Cycles. In [7], the Shared Risk Link Group (SRLG) distribution among the aggregated links is advertised, while in [8], domains provide the shortest path between each pair of their border nodes for domain-disjoint routing. Works in [9], [10] focus on link-disjoint inter-domain routing based on TA. In [11], TA is utilized and an ILP formulation is proposed for link- and domain-disjoint lightpath provisioning. However, none of these works take into account node-disjoint routing. In this case, working and backup paths of the demands cannot share any common components, which offers stronger protection than link-disjoint or SRLG-disjoint based solutions. On the other hand, it is not as restrictive as domain-disjoint routing, where a second disjoint path for a certain demand can be very costly or may not even exist.

This paper proposes four different approaches for node-disjoint routing with minimum total cost, based on TA and intra-domain disjointness information in multi-domain networks. Following a similar approach with [10], Full Mesh Aggregation (FMA) is applied to abstract the physical topology of each domain (e.g., the topology of Fig. 1(a), where the border nodes are shown in grey) into a full mesh scheme (shown in Fig. 1(b)). This way, each domain only needs to provide information on: i) the cost of the shortest path for every aggregated link, and ii) the existence and total cost of the two shortest node-disjoint paths for every pair of aggregated links. Based on this information, the inter-domain node-disjoint routing is then presented, along with how it can be performed in a centralized or a distributed multi-domain architecture. The average total cost of the path pair per demand and the blocking probabilities of the four approaches are offered, along with a comparison to the corresponding link-disjoint based approaches in [10]. To the best of our knowledge, this is the first paper that provides a complete formulation and evaluation of node-disjoint routing in multi-domain networks towards minimum total cost.

The rest of the paper is organized as follows. Section II presents the applied node-disjoint routing schemes. Section III describes the construction of the intra-domain disjointness information. Section IV develops the inter-domain routing. Section V describes the inter-domain communication and orchestration. Section VI presents the evaluation setup, experiments and results. Finally, Section VII concludes this work.

II. Node-Disjoint Routing Schemes

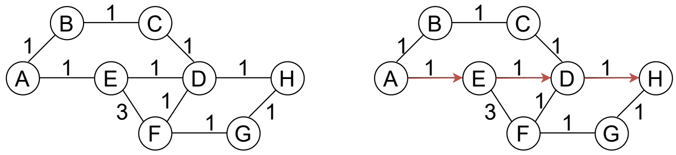
Network survivability is addressed in the context of finding a disjoint path pair between a source and a destination with minimum total cost. This paper focuses on *node-disjoint* routing and follows two different approaches. The first one,



(a) Physical Topology.

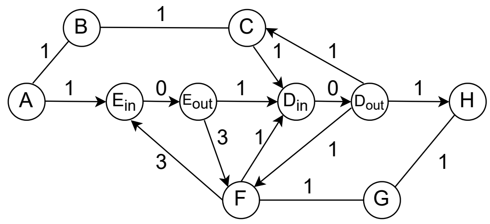
(b) Full Mesh Aggregation.

Fig. 1. Physical vs. Aggregated intra-domain topology.

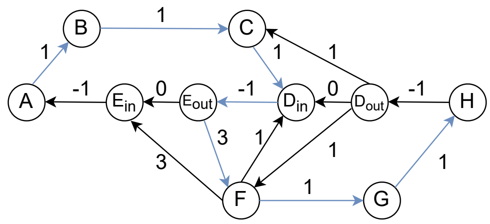


(a)

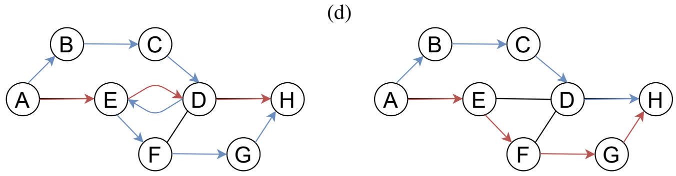
(b)



(c)



(d)



(e)

(f)

Fig. 2. Suurballe's algorithm for two node-disjoint paths with minimum total cost.

called the Two-Step approach, finds the shortest path between a pair of nodes by applying Dijkstra's algorithm, removes the traversed nodes, and then finds a second shortest path based on the remaining topology. However, this method will fail if there are traps in the network, e.g., in Fig. 2(b), a node-disjoint path pair between A and H cannot be found after the removal of nodes E and D.

The second approach tackles the problem in a joint way, aiming to find the optimal solution while avoiding such trap topologies. Namely, the original version of Suurballe's algorithm [12] starts with finding the shortest path (shown with the red arrows in Fig. 2(b)) using Dijkstra's algorithm. It then applies a node-splitting technique, where each node of the shortest path, apart from the source and the destination, is replaced by a pair of adjacent nodes; one with all of the incoming adjacencies of the original node, and one with all of the outgoing adjacencies, as shown in Fig. 2(c). The pair of adjacent nodes resulting from the split are connected by a zero cost uni-directional edge from the incoming node to the outgoing node. Next, in Fig. 2(d), it reverses the directions of the traversed edges and negates their costs, in order to find the second shortest path (noted by the blue arrows) by applying the Bellman-Ford algorithm, which allows negative weights on the graph. If a reversed edge is traversed by the second path, as in Fig. 2(e), it will be removed from the final result and the two paths will be reconstructed by combining the edges of the first and the second shortest path, as shown in Fig. 2(f).

The Two-Step approach and Suurballe's algorithm are applied for every pair of aggregated links in a domain to provide the intra-domain disjointness information. On top of that, they are also applied for inter-domain node-disjoint routing between every pair of border nodes.

III. Intra-domain Disjointess Information

This section describes the procedure to gather intra-domain disjointness information from a particular domain. This information is given as a $L_a \times L_a$ matrix T, where L_a is the number of aggregated links, which depends on the number of border nodes, N_b, in each domain as $L_a = N_b(N_b - 1)/2$.

A. Construction of Two-Step Matrix T_{TS}

The calculation of the T_{TS} matrix for the Two-Step approach is as follows. Each element of T_{TS}, t_{ij}, stores the cost of the first shortest path (FSP) for link l_i plus the cost of the second shortest path (SSP) for link l_j, whereas t_{ji} stores the total cost of the FSP for link l_j and the SSP for link l_i. The exact algorithm can be seen in the flowchart of Fig. 3. Let us consider a particular example on the aggregated domain topology of Fig. 1(b), and focus on the aggregated links l_1 and l_3, with $src(l_1) = A$, $dst(l_1) = H$, $src(l_3) = C$, $dst(l_3) = F$. To find the FSP for l_1, the nodes C and F have to be excluded from the physical topology, to prevent their selection in the FSP. Once the FSP is found, the nodes C, F are added back to the physical topology, and the nodes of the FSP, including the nodes A and B, are removed. If a SSP for l_3 can be found, t_{13} is equal to the total cost of the two paths. Otherwise, t_{13} is set to 0, which is the case in this example. Note that the removal of the nodes C, F may prevent finding even the FSP for l_1. In that case, nodes C, F are added back to the topology, and an attempt to find the FSP for link l_1 is made again. If now a path is found, t_{13} is set to 1, as obviously a second node-disjoint path for link l_3 cannot be found, and as an indicator that the FSP of l_1 occupies C or/and F.

B. Construction of Suurballe Matrix T_{Suur}

The computation of the T_{Suur} matrix for Suurballe's algorithm is explained below. Similarly with T_{TS}, each element t_{ij} is associated to l_i and l_j, with priority given to l_i. The procedure can be found in the flowchart of Fig. 3. Focusing on the same example as in the previous paragraph, the first step, again, is to find the FSP for l_1 after removing nodes C, F. Next, the known steps of Suurballe's algorithm are followed and T_{Suur} is constructed in the same way as T_{TS}. However, if node C and/or node F were part of the FSP for l_1 in the original network, i.e., before removing them from the physical topology, then their addition back to the network after the reverse of the edges and the negation of the costs, will lead to negative cycles. Thus, Bellman-Ford algorithm would fail to find a SSP for l_3. Therefore, after the reverse of the edges and the negation of the costs, the $max_{(u,v)\in E}(|c(u,v)|)$ of the topology is added to the costs of all links, where E is the link set of the graph and $c(u,v)$ is the cost of the link (u,v). This modification will prevent the formation of negative cycles and the SSP for link l_3 can now be obtained. Finally, to get the real cost values of the shortest paths, the $max_{(u,v)\in E}(|c(u,v)|)$ is subtracted from every one of their links.

IV. Routing On The Inter-Domain Level

Once the aggregation matrix is computed for each domain, two node-disjoint paths can be obtained for any pair of border nodes (referred in this section as source src and destination dst) on the inter-domain aggregated topology. This section explains how the Two-Step approach and Suurballe's algorithm, can be applied to both types of matrices, T_{TS} and T_{Suur}.

A. Two-Step over Two-Step Approach

The Two-Step over Two-Step (ToT) approach uses the T_{TS} matrices along with the Two-Step approach on the inter-domain level to obtain two node-disjoint paths between any pair of border nodes, aiming at minimum total cost. The entire approach can be found in the flowchart of Fig. 5. Each domain will provide its T_{TS} matrix along with the single shortest path (SP) costs for every of its aggregated links. Then the Two-Step approach will be performed on the inter-domain aggregated topology from src to dst. The FSP is obtained based on the single SP costs of all aggregated links. Next, all the traversed nodes, except for src and dst, are removed from the network. Before finding the SSP, the remaining intra-domain aggregated links of the traversed domains have to change their costs according to the T_{TS} matrices; if two intra-domain aggregated links belong to the inter-domain path pair, their total cost should correspond the total cost of their two node-disjoint intra-domain paths. Let us consider as example the network depicted in Fig. 4, where the FSP from $src = A$ to $dst = N$ is found as $A-C-E-G-L-N$. The intermediate nodes of the FSP, i.e., C, E, G, L, are removed, so the only aggregated links that need to adjust their costs are (A,B), (D,F), (K,N) and (M,N). Note that in the first and the last traversed domain, the remaining aggregated links that need to modify their costs are the ones that have the same src and dst with the FSP, respectively. In Domain 4, (K,M) should not change its cost, since the same domain is not allowed to be traversed with more than one aggregated links by the SSP; intra-domain node-disjointness cannot be

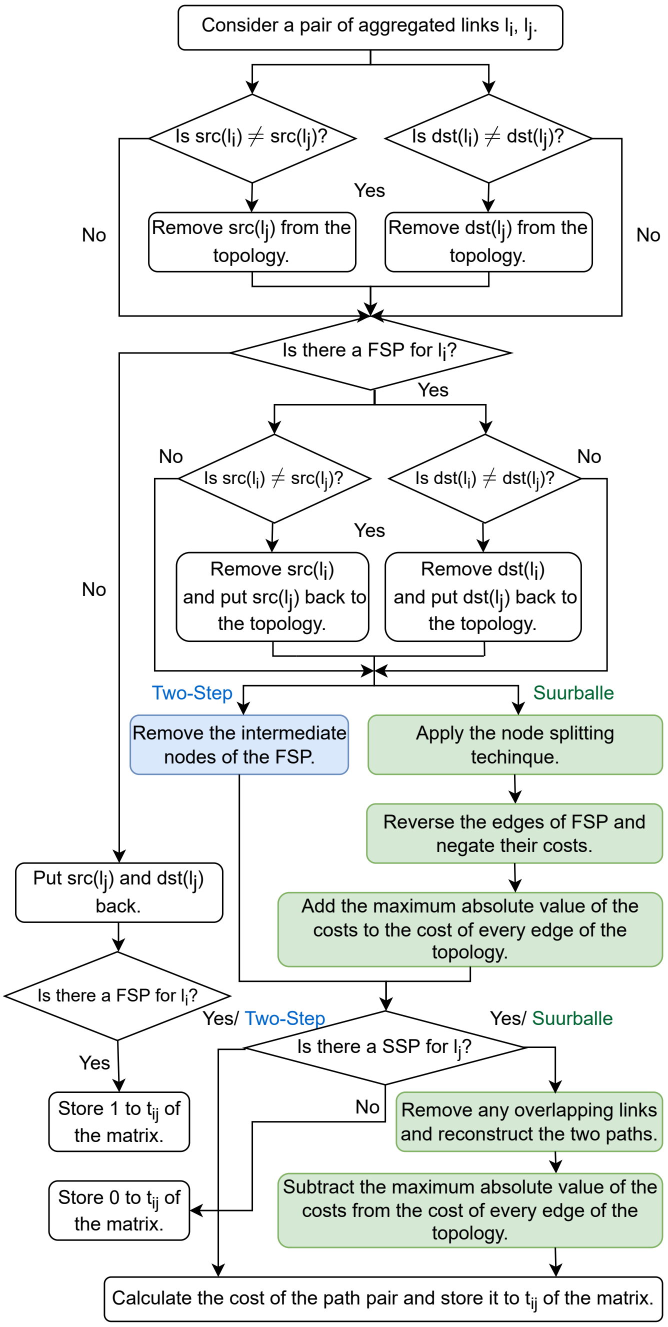


Fig. 3. Construction of intra-domain node-disjointness information for a pair of aggregated links.

guaranteed for more than two aggregated links. In every case, if l_i is an intra-domain link of the FSP, the cost of the link l_j, $j \neq i$, will be set to $t_{ij} - SP(l_i)$ as long as $t_{ij} \neq 0$ and $t_{ij} \neq 1$, where t_{ij} is the element of the T_{TS} with the total cost of the node-disjoint path pair for l_i, l_j, and $SP(l_i)$ is the cost of the single shortest path of l_i. If $t_{ij} = 0$, l_j is removed from the topology, whereas if $t_{ij} = 1$, $src(l_j)$, $dst(l_j)$ are also removed. Now, the SSP is obtained from the remaining aggregated topology. Finally, there is a case that the FSP traverses a domain through an aggregated link l_i, and the SSP

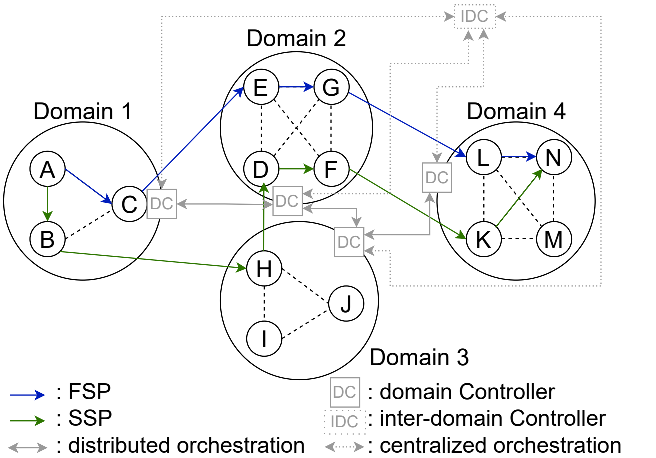


Fig. 4. Inter-domain routing example of two node-disjoint paths from Node A to Node N.

traverses the same domain through a single node, nd, (and not through an aggregated link l_j). In this scenario, l_i will change its cost to $min\{t_{ij}\} - SP(l_i)$, where $t_{ij} \neq 0$ and l_j (and thus t_{ij}) is selected among the links with $src(l_j) = nd$ and $dst(l_j) \in \{src(l_i), dst(l_i)\}$. Since the actual cost of l_i being disjoint with nd cannot be acquired, this modification aims to "penalize" its cost because of this disjointness requirement.

B. Suurballe over Two-Step Approach

The second approach, Suurballe over Two-Step (SoT) approach, utilizes the T_{TS} matrix, and applies Suurballe's algorithm for inter-domain node-disjoint routing. Following the procedure of the flowchart in Fig. 5, the first step is to construct T_{TS} for each domain, and find the FSP for a pair of border nodes based on the single SP costs for all aggregated links. Next, according to Suurballe's algorithm, the node-splitting technique is applied, the traversed links of the FSP are reversed and their costs are negated. The link costs are changed in the exact same way and in the same cases as in the ToT approach, i.e, aggregated links in the first and the last traversed domain with the same src and dst as the FSP, respectively, and aggregated links in intermediate traversed domains with different source and destination from the link occupied by the FSP (which basically translates to domains with at least 4 border nodes). After the cost adjustments, the SSP is found on the modified topology and the final result is obtained by reconstructing the two paths after removing the overlapping links. The scenario described above as the final step for ToT approach, needs to be checked here as well.

C. Two-Step over Suurballe Approach & Suurballe over Suurballe Approach

The Two-Step over Suurballe (ToS) and the Suurballe over Suurballe (SoS) are the two approaches that utilize the T_{Suur} matrices for intra-domain disjointness information, and then apply the two approaches (Two-Step and Suurballe) on the inter-domain aggregated topology for two node-disjoint paths

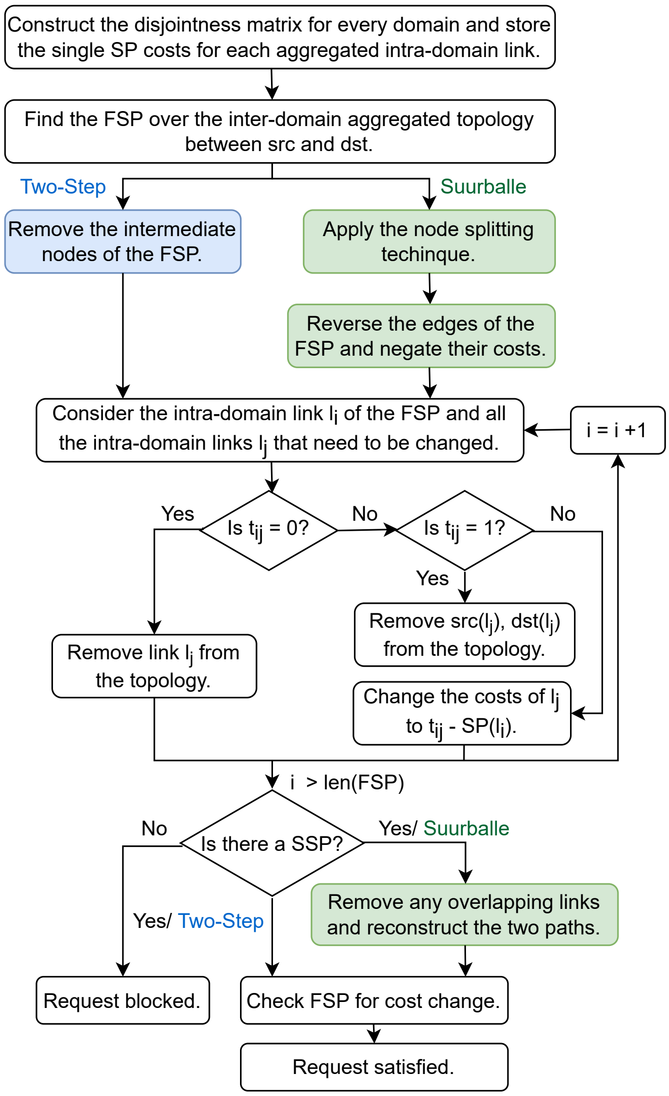


Fig. 5. Inter-domain node-disjoint routing over the aggregated topology, where len(FSP) is the length of the FSP.

pursuing minimum total cost for every demand. ToS and SoS are entirely symmetrical to ToT and SoT, respectively; they can be implemented based on the flowchart of Fig. 5.

V. Inter-domain Communication and Orchestration

Leveraging the capabilities of Software-Defined Networking (SDN), a *centralized* or *distributed* approach can enable inter-domain communication and orchestration. As depicted in Fig. 4 with doted gray lines, in the centralized approach every Domain Controller (DC) exchanges information exclusively with the Inter-Domain Controller (IDC). Each DC presents a censored version of its domain, which is used by the IDC to create an abstracted view of the inter-domain network. The IDC is responsible to orchestrate multi-domain services by splitting the configurations in the involved domains and requesting the provisioning of resources to associated DCs.

The distributed approach is depicted in Fig. 4 with a solid gray line. In this approach, every DC exchanges information with its neighboring DCs. The information presented by the DCs represent a censored version of its domain and what has been learnt from other domains. This way, each DC can build an abstract view of the multi-domain topology. Communication services are provisioned by each DC, which splits the configuration in the involved domains and triggers the provisioning of resources to the respective DCs. Depending on the use case, the most appropriate mechanism can be selected. For example, in the centralized approach the IDC acts as a client contracting services from multiple domain operators, while the distributed approach can be used when different domain operators cooperate directly.

VI. Numerical Results

The problem instance of the Cost266 European network [13] is considered as the aggregated inter-domain topology to evaluate the performance of the four proposed approaches. Each country is considered as a domain, so there are 21 domains in total. Six of them have more than 1 border node and need to provide their disjointness matrices. The intra-domain topologies for Germany, France and Poland are taken from the instances of germany50, france and polska from [13]. For Italy, Spain and UK, GARR 1999_01, RedIris and Janet Backbone networks are utilized, respectively, from [14]. Note that some network topologies may contain duplicate links, which are removed from our simulation setup. In the original topology of the Cost266 network, some of the border nodes within a country-domain are not connected with each other; the necessary aggregated links are also added to create the full mesh aggregation scheme in each domain. Regarding the costs of the links, two cases are considered: i) the link lengths (in kilometers) are assigned as costs both for the intra- and the inter-domain topology, and ii) the costs are randomly generated cost units (cu) in $[100, 10000]$ for all the intra-domain topologies, and in $[3000, 15000]$ for the inter-domain topology. The four approaches for node-disjoint routing are compared with the corresponding ones in [10] for link-disjoint routing. Requests are generated between every pair of border nodes over the inter-domain aggregated topology.

Fig. 6 shows the average cost of the path pair per demand for each of the four approaches for link- and node-disjointness, when the link costs are the link lengths on the physical network. For both problem cases, the bars show the average cost of the disjoint path pair over the number of satisfied requests for each approach, whereas the beige line shows the average total cost over the same number of requests among the 4 approaches (that is, considering only the demands that are satisfied by all 4 approaches). Since it is more restrictive, the node-disjoint routing problem demonstrates overall a higher average cost than the link-disjoint one, but limited to a $\sim 5\,\%$ increase per method. Regarding link-disjoint routing, it can be seen that the bars and the line coincide, since all the requests on the inter-domain aggregated network are satisfied by every one of the 4 approaches. In the node-disjoint problem though,

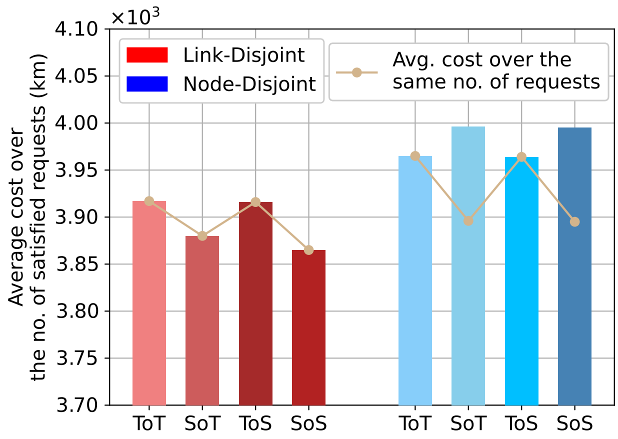


Fig. 6. Average cost over the inter-domain disjoint path pairs in Cost266 network with link lengths as costs.

ToT and ToS cannot satisfy almost $8.5\,\%$ of the requests, whereas SoT and SoS manage to find a node-disjoint path pair for all the requests. This causes that the former two have a lower average cost than SoT and SoS respectively. However, in order to perform a fair comparison, the cost associated to the $91.5\,\%$ of the requests coped by all 4 approaches is shown with the beige line. It can be observed that the cost per demand is lower for the SoX methods. So, it seems that the greater impact on the average cost comes from the approach applied on the inter-domain aggregated topology. Specifically, in Fig. 6 the intra-domain routing method does not seem to affect the results for the node-disjoint case. This is mainly because only 6 out of the 21 domains have the ability to affect the final result.

Fig. 7 shows the similar analysis when the costs are generated randomly, as described in the first paragraph of this section. It can be seen that the differences among the approaches are consistent in terms of average cost performance. In this case, $\sim 0.5\,\%$ of the requests cannot be satisfied by the ToX approaches when aiming at link-disjoint paths, whereas in the node-disjoint case, the same two approaches block almost $10\,\%$ of the requests. On the other hand, the SoX approaches are entirely successful both for the link and the node-disjoint routing. Here, a slight difference can be observed between the two ToX (ToT and ToS) and the two SoX (SoT and SoS) approaches. Although the ToX approaches demonstrate higher average cost and blocking ratio, they are simpler and easier to implement compared to the SoX ones, which involve more and complex steps, as it can also be clearly seen from the flowcharts.

As the number of blocked requests depends on the link costs, the blocking probability of each approach is analyzed over 1000 independent simulations with random generated costs for the links of the entire physical network. As expected, the blocking probability for link-disjoint paths is tremendously lower than for node-disjoint paths for all methods. Focusing on the node-disjoint routing problem, Fig. 8 shows that the two

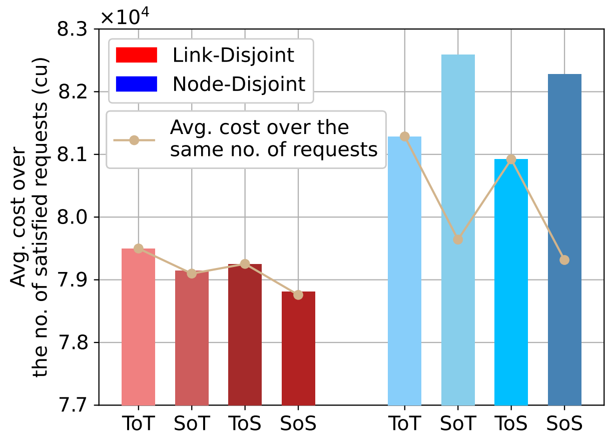


Fig. 7. Average cost over the inter-domain disjoint path pairs in Cost266 network with randomized costs.

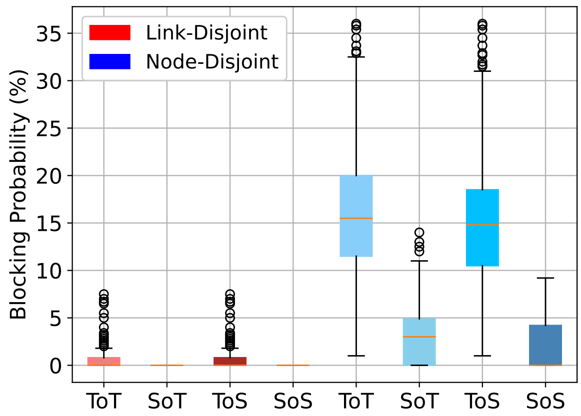


Fig. 8. Blocking probability over the inter-domain aggregated topology of Cost266 network.

ToX approaches present an intensely higher blocking probability than the SoX ones. Specifically, the ToX approaches have a median of $\sim 15\,\%$, with a slight improvement from ToT to ToS. Moreover, they never satisfy all the demands on the inter-domain network, presenting a minimum blocking probability of $1\,\%$. On the other hand, SoT performs quite effectively compared to ToT with a median of $3\,\%$ blocked requests. It also achieves a maximum of $11\,\%$ blocking ratio on the network, lower than the median of the ToX methods. As expected, the most optimal approach is the one applying Suurballe's algorithm both on the intra- and the inter-domain levels. The SoS approach manages to achieve a $0\,\%$ median of blocking probability, while it demonstrates the lowest maximum blocking ratio as well.

VII. Conclusions

This work proposes four different approaches for survivable node-disjoint inter-domain routing in multi-domain networks. Using TA to abstract the physical topology of the domains maintains their privacy across the inter-domain network. The only information each domain exposes is related to the existence and the total cost of two node-disjoint paths for every pair of aggregated links. The proposed methods have been evaluated in terms of average cost per demand and blocking probability, and have been compared to the respective ones for link-disjoint routing. Results show that even though the node-disjoint routing is more expensive than link-disjoint solutions, the cost difference is limited to $5\,\%$. As expected, the most optimal approach in terms of cost is when Suurballe's algorithm is applied to both intra- and inter-domain levels. Moreover, it is the only one that achieves a median of $0\,\%$ blocked requests on the inter-domain aggregated topology. Hence, SoS has been shown to be the best method independently on the cost assignment on the physical network.

VIII. Acknowledgements

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany (BMBF) in the programme of ”Souverän. Digital. Vernetzt.” joint project 6G-life, project identification number 16KISK002.

References

[1] Z. Ji-hong, M. Hong-bao, and Q. Hua, “Research on survivability in multi-service-based multi-domain optical networks,” in *2009 11th International Conference on Advanced Communication Technology*, vol. 01, 2009, pp. 161–165.

[2] F. Xu, F. Gu, H. Alazemi, M. Peng, and N. Ghani, “Novel path protection scheme for multi-domain networks,” in *2011 IEEE GLOBECOM Workshops (GC Wkshps)*, 2011, pp. 130–135.

[3] S. Zhang, S. Du, Y. Peng, and K. Long, “Segment-shared protection for multi-domain optical mesh networks with export policy limitation,” in *2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)*, 2012, pp. 1290–1293.

[4] S. Hong, J. P. Jue, P. Park, H. Yoon, H. Ryu, and S. Hong, “Survivable virtual topology design in multi-domain optical networks,” *Journal of Optical Communications and Networking*, vol. 8, no. 6, pp. 408–416, 2016.

[5] L. Guo, “LSSP: A novel local segment-shared protection for multi-domain optical mesh networks,” *Computer Communications*, vol. 30, pp. 1794–1801, 06 2007.

[6] H. Drid, S. Lahoud, B. Cousin, and M. Molnar, “A topology aggregation model for survivability in multi-domain optical networks using p-cycles,” in *2009 Sixth IFIP International Conference on Network and Parallel Computing*, 2009, pp. 211–218.

[7] C. Gao, Y. Zhu, and J. P. Jue, “SRLG-aware topology aggregation for survivable multi-domain optical networks,” in *2011 IEEE International Conference on Communications (ICC)*, 2011, pp. 1–5.

[8] C. Gao, M. M. Hasan, and J. P. Jue, “Domain-disjoint routing based on topology aggregation for survivable multi-domain optical networks,” in *2011 IEEE Global Telecommunications Conference - GLOBECOM*, 2011, pp. 1–5.

[9] Q. Zhang, M. M. Hasan, X. Wang, P. Palacharla, and M. Sekiya, “Survivable path computation in PCE-based multi-domain networks,” *Journal of Optical Communications and Networking*, vol. 4, no. 6, pp. 457–467, 2012.

[10] C. Gao, H. C. Cankaya, and J. P. Jue, “Survivable inter-domain routing based on topology aggregation with intra-domain disjointness information in multi-domain optical networks,” *Journal of Communications and Networking*, vol. 6, no. 7, pp. 619–628, 2014.

[11] P. Pavarangkoon and E. Oki, “An optimization approach for survivable lightpath provisioning in multi-domain optical networks,” *Optical Switching and Networking*, vol. 22, pp. 26–41, 2016.

[12] J. W. Suurballe, “Disjoint paths in a network,” *Networks*, vol. 4, pp. 125–145, 1974.

[13] SNDlib. Networks and Problem instances. Accessed: Oct. 26, 2022. [Online]. Available: https://bit.ly/3TCHflH

[14] The Internet Topology Zoo. Datasets. Accessed: Oct. 26, 2022. [Online]. Available: https://bit.ly/3WhiwWb