Improving the Connectivity Resilience of a Telecommunications Network to Multiple Link Failures Through a Third-Party Network

Amaro de Sousa
Instituto de Telecomunicações
DETI, Universidade de Aveiro, Portugal
asou@ua.pt

***Abstract*—Currently, telecommunication networks are fully resilient, in terms of connectivity, to single link failures. On the other hand, multiple simultaneous link failures are becoming a concern to network operators, mainly due to malicious human activities. Full connectivity resilience to multiple link failures is too costly and other solutions must be envisaged. For a given maximum number of simultaneous link failures, the connectivity resilience metric adopted here is the minimum number of network node pairs that can still communicate for any set of failing links. In this work, the connectivity resilience to multiple link failures is improved by resorting to a third-party network for temporary additional connectivity (i.e., while the failing links are not reestablished). In such a solution, some nodes must be selected to act as gateway nodes between the two networks. For a given network topology and a given number of gateway nodes, the aim is to select the most appropriate gateway nodes so that the connectivity resilience is improved as much as possible. To address this problem, a Gateway Node Selection (GNS) algorithm is proposed where the most damaging sets of failing links are identified and, then, a set cover problem type is defined and solved to select the gateway nodes. The computational results demonstrate the effectiveness of the proposed GNS algorithm over two well-known network topologies.**

Keywords—connectivity resilience, multiple link failures, integer linear programming, telecommunication networks

I. INTRODUCTION

The resilience to failures of a given telecommunication network is generally defined as the capacity of the network to maintain its services after the failures. The network resilience to failures can be evaluated at different levels. Low impact (single) failures cause the rerouting of some service flows, potentially degrading service latency. Medium impact failures can also cause link congestion degrading not only service latency but also the available throughput. Large-scale failures have the potential of disconnecting the network in different components causing connectivity disruption between many pairs of network nodes, i.e., services between nodes in different components can no longer be supported.

In general, current telecommunication networks are fully resilient, in terms of connectivity, to single link failures. On the other hand, large-scale failures are becoming a concern to operators of telecommunication networks due to different reasons, as natural disasters [1] or malicious human activities [2]. In the latter case, current telecommunication networks are deployed over optical infrastructures which are vulnerable to many physical-layer attacks [3]. In particular, link cuts are a relatively straightforward method of a physical-layer attack and, depending on the selected links to be cut, it can severely degrade the services supported by the network.

Multiple link failures have been considered in different contexts in the last two decades. The impact of multiple link failures on optical networks based on wavelength division multiplexing (WDM) has been studied long time ago [4–5]. Multiple link failures have been modelled as shared risk link groups (SRLGs), i.e., groups of links with high probability of simultaneous failure [6–7]. The typical example is when multiple links share a single duct and, thus, the unattended cut of the duct makes all links to be simultaneously cut. More recently, multiple link failures have been modelled as regional failures in the context of natural disasters [1, 8–9]. In all these works, there is correlation of some kind between the failing links. Concerning uncorrelated failures, the particular case of dual link failures has also been addressed in different works [10–13]. Nevertheless, the case of more than two simultaneous uncorrelated failures (which can happen in malicious human activities when the attacker knows the network topology), has only been considered by very few works [14–16].

Full connectivity resilience to multiple link failures is too costly for a network operator as it requires too many physical links. Instead, operators need to enhance the connectivity resilience to such failures with solutions that do not represent a prohibitive investment. A recent example is the common emergency packet transport network proposed in [17] for disaster recovery. In that proposal, a third-party entity builds an emergency network with the surviving resources of multiple network operators (affected by a regional disaster) that can be jointly used by them. The emergency network is built in several steps to avoid confidential information leakage between network operators.

Here, we consider a simpler approach which does not require the third-party entity to exist (and also avoids confidential information leakage between operators). Consider an operator of a network deployed on an area (a region, a country or a continent) where other networks (of other operators) also exist. In the case of a multiple link failure, a cost effective solution is to resort to a third-party network to provide temporary connectivity between some network nodes while the failing links are not reestablished. The business relation between the two network operators can be one-way (the third-party operator charges the temporary connectivity provided when needed) or two-way (each operator provides the temporary connectivity when needed by the other), which might even eliminate the service costs charged between operators depending on the Service Level

Agreement (SLA) defined between them. In any case, some nodes of one network must be selected to act as gateway nodes to the other network.

The connectivity resilience of the network is assumed to be the minimum number of network node pairs that can still communicate after any set of failing links (for a given maximum number L of simultaneous failures). Consider the example in Fig. 1 where two nodes of the network are set as gateway nodes, each one connecting to a node of the third-party network (in general, the number of gateway nodes can be two or more). Then, consider the simultaneous failure of the 3 links highlighted in dashed red in Fig. 2. Without the third-party network, this failure causes the network to be split in two components (one with 2 nodes and one with 5 nodes) and, before the recovery of at least one failing link, only nodes belonging to the same component can communicate between them. With the provision of a temporary "virtual link" by the third-party network between the gateway nodes (with a throughput defined in accordance to the SLA between the two operators), all network nodes can communicate between them. In fact, the two selected gateway nodes are optimal for any simultaneous failure of 3 links: these gateway nodes guarantee that any set of 3 failing links can disconnect at most one node from all others.

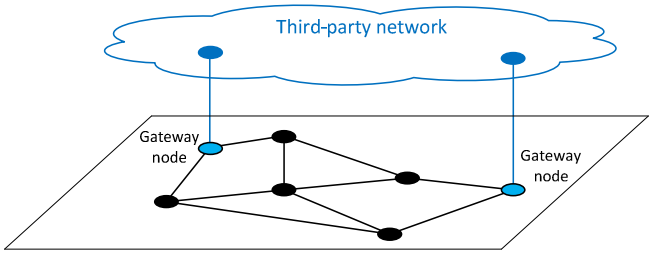

Fig. 1. Use of 2 gateway nodes to a third-party network.

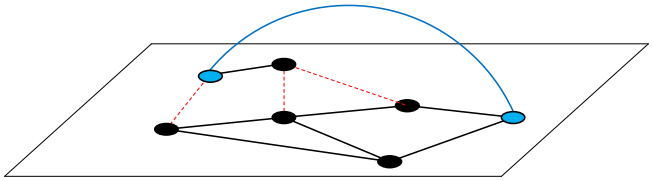

Fig. 2. Provision of a temporary virtual link between the gateway nodes for the simultaneous failure of 3 links (highlighted in dashed red).

For a given network topology and a given number of gateway nodes, the aim is to select the gateway nodes to the third-party operator so that the connectivity resilience is improved as much as possible. To address this problem, a Gateway Node Selection (GNS) algorithm is proposed where the most damaging sets of failing links are identified and, then, a set cover problem type is defined and solved to select the most appropriate gateway nodes.

The paper is organized as follows. Section II describes how the connectivity resilience is evaluated for a given network either without gateway nodes or with a given set of gateway nodes. Section III describes the GNS algorithm. In section IV, a set of computational results is presented and the effectiveness of the GNS algorithm is discussed. Finally, Section V ends with the main conclusions of the work.

II. Connectivity Resilience Evaluation

First, consider the connectivity resilience evaluation of a given telecommunications network without gateway nodes. The network is defined by an undirected graph $G = (N, E)$. Each node of set N is labelled as $1, 2, 3, \ldots, n$, where $n = |N|$. Each link of set E is defined by (i, j), with $i, j \in N$ and $i < j$.

For a given a positive integer L, the connectivity resilience of graph G for L simultaneous link failures is defined as the minimum number of node pairs that can communication whatever set of L failing links. The connectivity resilience is given by the optimal solution of the Critical Link Detection (CLD) optimization problem defined as follows.

Consider a binary variable v_{ij} for each link $(i, j) \in E$ which is equal to 1 if link (i, j) is a critical (failing) link; or equal to 0, otherwise. Consider also a binary variable u_{ij} for each pair of nodes $i, j \in N$, with $i < j$, which is equal to 1 if nodes i and j can communicate in the surviving network (i.e., in the network without the critical links). The CLD problem is defined by mixed integer linear programming as:

$$\text{Minimize } \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} u_{ij} \quad (1)$$

Subject to:

$$\sum_{(i,j)\in E} v_{ij} \leq L \quad (2)$$

$$u_{ij} + v_{ij} \geq 1 \qquad ,(i,j) \in E \quad (3)$$

$$u_{ij} \geq u_{\{ik\}} + u_{\{kj\}} - 1 \qquad ,i,j \in N: i < j, k \in N(i,j) \quad (4)$$

$$v_{ij} \in \{0,1\} \qquad ,(i,j) \in E \quad (5)$$

$$u_{ij} > 0 \qquad ,i,j \in N: i < j \quad (6)$$

The objective function (1) is the minimization of the connectivity resilience, i.e., of the number of node pairs that can communicate in the surviving network. Constraint (2) guarantees that the number of critical links is not higher than L. Constraints (3) guarantee that the end nodes of a link $(i, j) \in E$ can communicate if it is not a critical (failing) link, i.e., if $v_{ij} = 0$.

Constraints (4) guarantee that each pair of nodes $i, j \in N$, with $i < j$, can communicate if there is a third node k such that k can communicate with both i and j. In constraints (4), $u_{\{ik\}}$ represents u_{ik} if $i < k$, or u_{ki} otherwise (the same meaning for $u_{\{kj\}}$). In order to minimize the number of constraints (4), it is enough to consider k as the neighbor nodes of either i or j. In the notation of constraints (4), set $N(i, j)$ is the set of neighbor nodes of the node (among i and j) with the lowest degree and, if nodes i and j are neighbors (i.e., if $(i, j) \in E$), one is excluded as neighbor of the other.

Finally, constraints (5–6) are the variable domain constraints. Note that only variables v_{ij} are declared as binary. Variables u_{ij} do not need to be declared as binary since they will be set to binary values in any optimal solution.

Consider now the connectivity resilience evaluation of a given telecommunications network with a given set of gateway nodes. We can model this case by:

- considering an augmented graph G' by adding to graph G one extra link per pair of gateway nodes, and
- assuming that the extra links never fail (they represent the virtual links provided by the third-party operator only when needed).

So, the connectivity resilience of a network with a set of gateway nodes is the optimal solution of the CLD model

defined over the augmented graph G' and setting the variables v_{ij} equal to 0 for all extra links (i,j). This optimization model will be referred henceforward as CLD-S model.

III. Gateway Node Selection (GNS) Algorithm

Consider the notation introduced in the previous section. The GNS algorithm aims to select a set of nodes on a given graph G to be configured as gateway nodes. The algorithm requires the determination not only of the most damaging set of failing links (i.e., the set of Critical Links) but also the second, third (and so on) most damaging sets of failing links.

The solution of the CLD model (presented in the previous section II) gives the most damaging set of failing links. To compute the next (i.e., the second) set of most damaging failing links, we need to add to the CLD model a constraint excluding the first solution from its feasible set. Consider the binary values of the variables v_{ij} of the optimal solution of the CLD represented by $\tilde{v}_{ij}^1$ and consider the value L_1 given by the sum of all values $\tilde{v}_{ij}^1$, i.e., $L_1 = \sum_{(i,j)\in E}(\tilde{v}_{ij}^1)$. The CLD-1 model is then defined as:

Minimize (1)

Subject to:

(2–6)

$$\sum_{(i,j)\in E}(\tilde{v}_{ij}^1 \times v_{ij}) \leq L_1 \qquad (7.1)$$

The solution of CLD-1 model gives the second most damaging set of failing links. Again, considering the values of v_{ij} of the optimal solution of CLD-1 represented $\tilde{v}_{ij}^2$ and $L_2 = \sum_{(i,j)\in E}(\tilde{v}_{ij}^2)$, the CLD-2 model is defined as:

Minimize (1)

Subject to:

(2–6), (7.1)

$$\sum_{(i,j)\in E}(\tilde{v}_{ij}^2 \times v_{ij}) \leq L_2 \qquad (7.2)$$

The solution of CLD-2 model gives the third most damaging set of failing links. All next sets of most damaging failing links are computed by iteratively repeating this procedure. In general, the CLD-*k* model has one constraint for each of the *k* most damaging sets of failing links and its optimal solution provides the (*k*+1)th most damaging set of failing links.

Then, the selection of the gateway nodes is based on the previously computed most damaging sets of failing links. Note that the k^{th} most damaging set, defined by the binary parameters $\tilde{v}_{ij}^k$, splits the network in two or more components. By computing the surviving graph $G_k = (N, (i,j) \in E: \tilde{v}_{ij}^k = 0)$, we can determine the 2 largest components (in number of nodes). Consider N_1^k and N_2^k as the set of nodes of the largest and second largest component of the surviving graph G_k. In order to guarantee that the nodes of the 2 largest components can communicate in the k^{th} most damaging set of failing links, one needs to ensure that one gateway node belongs to N_1^k and another gateway node belongs to N_2^k.

The selection of the gateway nodes is based on a set cover problem (SCP). Consider the number of gateway nodes (to be selected) given by B and the number of most damaging sets (to be considered) given by K. Consider a binary variable x_i for each node $i \in N$ which is equal to 1 if node i is selected as a gateway node, or equal to 0 otherwise. Consider also the parameter c_i which is given by the closeness centrality value of node $i \in N$ in graph G (i.e., $c_i = 1/\sum_{j\neq i} p_{ij}$ and p_{ij} is the length of the shortest path from node i to node j in G). The selection of the gateway nodes is based on the optimal solution of the following SCP-*K* model:

Maximize $\sum_{i\in N} c_i x_i$ (8)

Subject to:

$$\sum_{i\in N} x_i \leq B \qquad (9)$$

$$\sum_{i\in N_1^k} x_i \geq 1 \qquad , k = 1 \dots K \qquad (10)$$

$$\sum_{i\in N_2^k} x_i \geq 1 \qquad , k = 1 \dots K \qquad (11)$$

$$x_i \in \{0,1\} \qquad , i \in N \qquad (12)$$

The objective function (8) is the maximization of the closeness centrality of the nodes selected as gateway nodes. Constraint (9) guarantee that the number of gateway nodes is not higher than B. Constraints (10–11) guarantee that each of the 2 largest components of each set of failing links contains (is covered by) one gateway node. Finally, constraints (12) are the variable domain constraints.

Note that, when the set of constraints (9–12) admits multiple solutions, the objective function (8) gives preference to the selection of more central nodes, minimizing in this way the average shortest path length from each node to the closest gateway node.

Algorithm 1. Gateway Node Selection Algorithm

1. Determine the most damaging set E_1 of failing links by solving CLD model (defined in Section II)
2. Define SCP-1 model based on the nodes N_1^1 and N_2^1 of the largest and second largest component defined by E_1
3. Determine a set of gateway nodes ρ_{opt} by solving SCP-1 model
4. Determine the connectivity resilience r_{opt} of the set of gateway nodes ρ_{opt} by solving the CLD-S model (defined in Section II)
5. $k \leftarrow 2$, *continue* $\leftarrow$ TRUE
6. **While** *continue* is TRUE **Do**
   a. Determine the k^{th} most damaging set E_k of failing links by solving CLD-(*k*–1) model
   b. Define SCP-*k* based on the nodes N_1^k and N_2^k of the largest and second largest component defined by E_k
   c. Determine a set of gateway nodes ρ_{aux} by solving SCP-*k* model
   d. **If** SCP-*k* model is feasible **Then**
      - Determine the connectivity resilience r_{aux} of the set of gateway nodes ρ_{aux} by solving the CLD-S model
      - **If** $r_{aux} > r_{opt}$ **Then**
        $r_{opt} \leftarrow r_{aux}$, $\rho_{opt} \leftarrow \rho_{aux}$
        EndIf
      - $k \leftarrow k + 1$

      Else
      - *continue* $\leftarrow$ FALSE

      EndIf

   EndWhile

One difficulty in using the SCP-*K* model (8–12) is that one cannot know in advance the proper value of *K* (i.e., how many most damaging sets of failing links to include in the model). The value of *K* should be as large as possible without turning the model infeasible. The Gateway Node Selection (GNS) algorithm proposed in this work is an iterative method that, at each iteration, the next most damaging set of failing links is computed and the resulting set cover problem is solved. The method ends when the set cover problem becomes infeasible. During all iterations, the connectivity resilience of each set cover solution is computed and the method selects the solution with the highest connectivity resilience value. The proposed GNS algorithm is presented in Algorithm 1. At the end of the algorithm, ρ_{opt} is the set of selected gateway nodes and r_{opt} is its resilience value (as provided by the CLD-S model).

IV. Computational Results

Consider the Janos-us (with 26 nodes, 42 links and 325 node pairs) and Germany50 (with 50 nodes, 88 links and 1225 node pairs) well-known network topologies [18] presented in Fig. 3. The figure also highlights in red the most damaging set of $L = 6$ failing links when no gateway nodes are used. In Germany50, the failing link which is not visible in the figure is link (10,17).

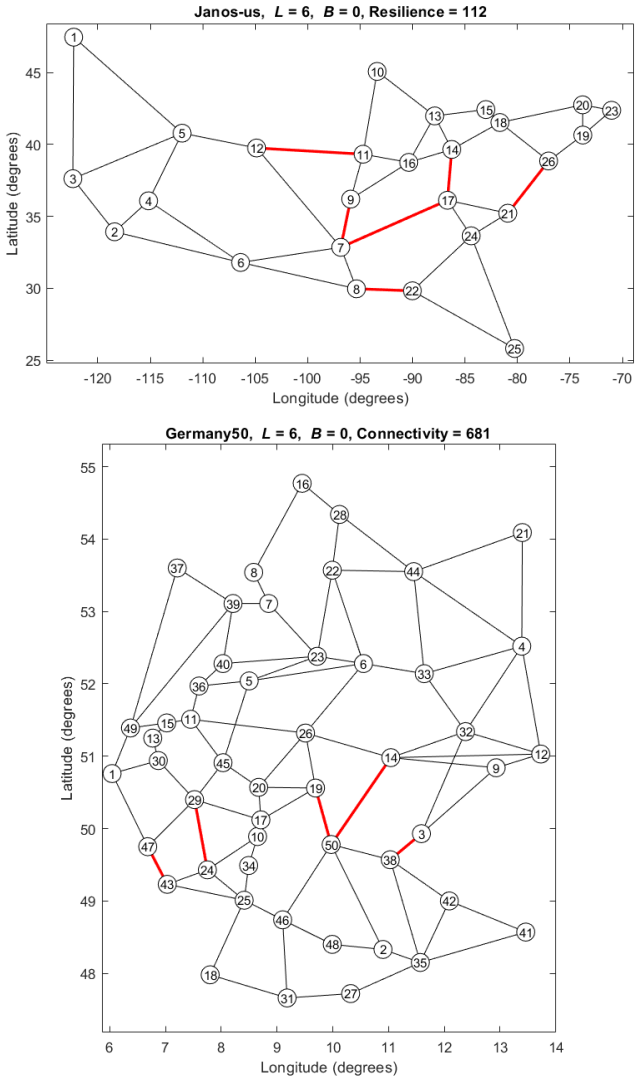

Fig. 3. Geographical networks highlighting in red the most damaging set of $L = 6$ failing links.

In the Janos-us case, the failing links split the topology in one component of 12 nodes, one component of 9 nodes and one component of 5 nodes. So, the connectivity resilience of Janos-us is $\binom{12}{2} + \binom{9}{2} + \binom{5}{2} = 112$ node pairs which is 34.5% of the total number of node pairs. In the Germany50 case, the failing links split the topology in one component of 34 nodes and one component of 16 nodes. So, the connectivity resilience of Germany50 is $\binom{34}{2} + \binom{16}{2} = 681$ node pairs which, in this case, is 55.6% of the total number of node pairs.

The GNS algorithm was run for these two network topologies considering the number of failing links $L = 3$, 4, 5 and 6 and the number of gateway nodes $B = 2$, 3 and 4. The GNS algorithm was implemented in MATLAB and all optimization models were solved using CPLEX 12.8 software package. All results were obtained on a PC platform with an Intel Core i7 8th generation processor and 32 GBytes of RAM.

Table I shows the connectivity resilience (in number of node pairs that can communicate) of each network for each value of *L*. Not surprisingly, a higher number of failing links results in a lower connectivity resilience value (more failing links disconnect more node pairs) and the absolute resilience values are higher for Germany50 since its total number of node pairs is higher than the total number of node pairs of Janos-us.

TABLE I. Connectivity Resilience (in No. of Node Pairs) without Gateway Nodes

Network	*L = 3*	*L = 4*	*L = 5*	*L = 6*
Janos-us	205	157	154	112
Germany50	1084	1000	796	681

Table II presents the computational results of the GNS algorithm for Janos-us network. Column 'Res' presents the connectivity resilience value of each solution. Then, column 'NIter' indicates how many iterations were run by the GNS algorithm and column 'Time' presents the running time (in seconds) of the GNS algorithm. The performance (in number of iterations and running time) observed in Table II shows that the GNS algorithm is efficient for Janos-us (largest running time below 3 minutes) implying that larger cases (in terms of *L* and *B* values) can also be solved.

TABLE II. GNS Computational Results for Janos-us

L	*B*	*Res*	*NIter*	*Time (s)*
3	2	256	18	4.4
	3	300	25	7.3
	4	300	28	9.2
4	2	220	20	3.5
	3	256	42	9.5
	4	256	48	12.3
5	2	172	30	4.0
	3	205	113	32.6
	4	205	125	38.0
6	2	130	9	1.1
	3	152	13	1.8
	4	180	351	168.9

In terms of connectivity resilience, Fig. 4 presents the resilience gains (in percentage) obtained by the selected gateway nodes, i.e., the resilience gains from the values without gateway nodes (presented in Table I for Janos-us) and

the values with gateway nodes (presented in Table II). The results in Fig. 4 show that the gains are very much parameter dependent. For example, with $B = 2$ gateway nodes, a high resilience gain of 40.1% is obtained for $L = 4$ failing links while an only moderate resilience gain of 11.7% is obtained for $L = 5$ failing links. Moreover, more gateway nodes do not always provide better resilience: increasing from 2 to 3 gateway nodes has improved the connectivity resilience for all cases but increasing from 3 to 4 gateway nodes has only improved the connectivity resilience for $L = 6$ failing links.

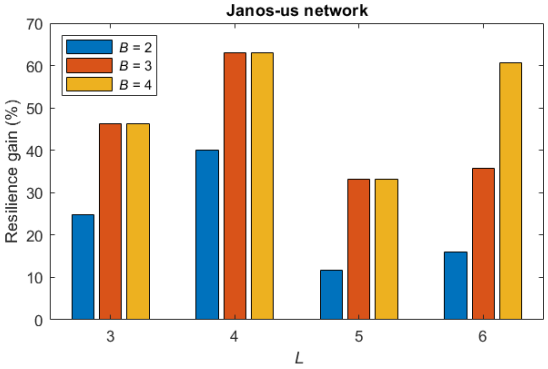

Fig. 4. Resilience gains for Janos-us network.

Table III presents the computational results of the GNS algorithm for Germany50 network (the meaning of each column is the same as in Table II).

TABLE III. GNS COMPUTATIONAL RESULTS FOR GERMANY50

C	*B*	*Res.*	*No. Iter.*	*Time (s)*
3	2	1129	16	84.9
	3	1129	18	112.9
	4	1129	24	186.7
4	2	1082	10	119.4
	3	1084	15	192.4
	4	1084	24	321.1
5	2	961	6	28.9
	3	1000	29	163.8
	4	1000	41	306.1
6	2	856	24	85.8
	3	856	26	117.4
	4	952	101	731.2

In the Germany50 case, the running times of GNS algorithm are in the order of some minutes for the problems instances with the largest L and B values (a little over 12 minutes in the worst case), indicating that the algorithm cannot solve much larger cases (in terms of L and B values).

As in the Janos-us case, Fig. 5 presents the resilience gains (in percentage) from the resilience values without gateway nodes (Table I for Germany50) to the resilience values with gateway nodes (Table III). In this case, it is clear that the resilience gains increase for larger number of failing links L, a behavior that is very different to the one observed on the Janos-us problem instances, highlighting also that the resilience gains are also dependent on the topology of the network. Nevertheless, an observation that is common to both networks is that more gateway nodes do not always provide better resilience. In the Germany50 case, increasing from 2 to 3 gateway nodes has only improved the connectivity resilience for $L = 4$ (very slightly) and for $L = 5$ (significantly) failing links and did not improve the connectivity resilience for $L = 2$ and 6 failing links. Like in Janos-us, increasing from 3 to 4 gateway nodes has only improved the connectivity resilience for $L = 6$ failing links.

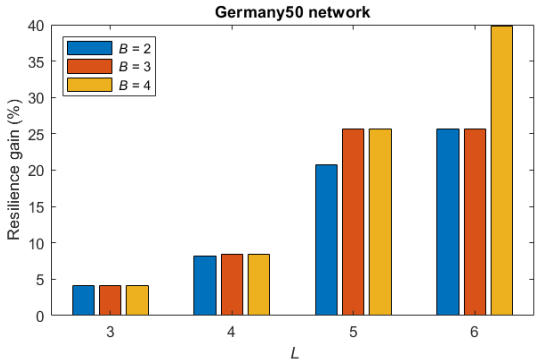

Fig. 5. Resilience gains for Germany50 network.

For illustration purposes, Fig. 6 presents for both networks the gateway nodes (highlighted in blue) provided by the GNS algorithm for $B = 4$ gateway nodes and considering $L = 6$ failing links. The figure also shows the most damaging set of failing links (highlighted in red) of these solutions.

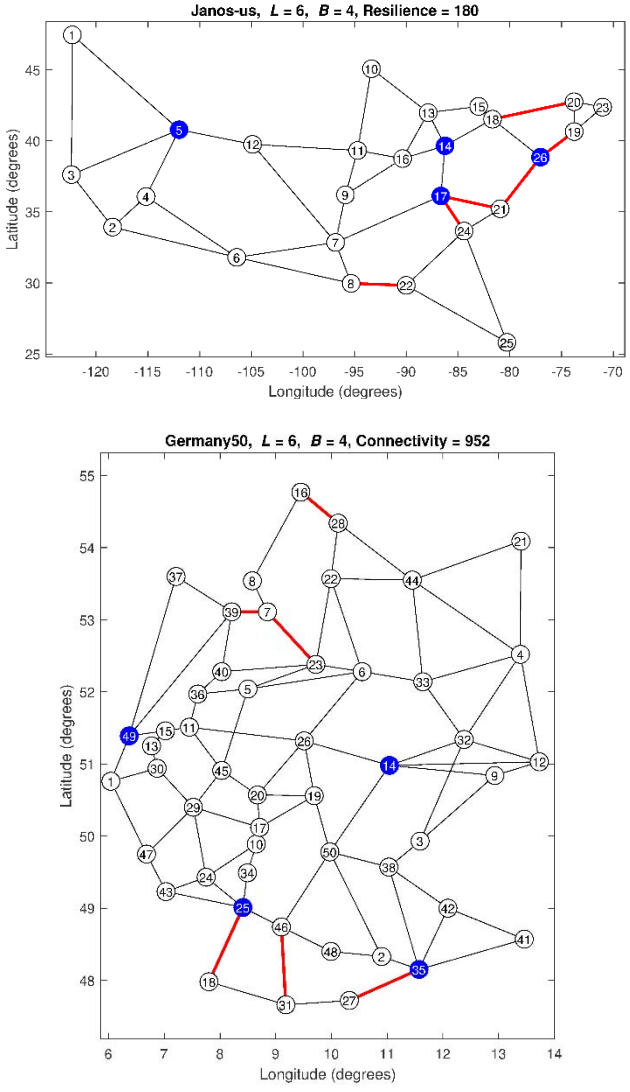

Fig. 6. The solutions with $B = 4$ gateway nodes (in blue) for $L = 6$ failing links (the most damaging set of the solution highlighted in red).

In the Janos-us case, the failing links split the topology in one component of 19 nodes, one component of 4 nodes and one component of 3 nodes. So, the connectivity resilience of Janos-us with the four selected gateway nodes is now $\binom{19}{2} + \binom{4}{2} + \binom{4}{2} = 180$ node pairs, which represents an increase of 60.7% from the solution without gateway nodes (presented in Fig. 3). In the Germany50 case, the failing links split the topology in one component of 44 nodes and two components of 3 nodes each. So, the connectivity resilience of Germany50 with the four selected gateway nodes is now $\binom{44}{2} + 2 \times \binom{3}{2} = 952$ node pairs, which represents an increase of 39.8% from the solution without gateway nodes (presented in Fig. 3).

V. Conclusions

Multiple simultaneous link failures are becoming a concern to network operators, mainly due to malicious human activities. Current telecommunication networks are deployed over optical infrastructures which are particularly vulnerable to link-cut attacks, a relatively straightforward method of a physical-layer attack. Full connectivity resilience to multiple link failures is too costly and, instead, operators need to enhance the connectivity resilience to multiple link failures with solutions that do not represent a prohibitive investment. One cost effective solution is to use a third-party network to provide temporary connectivity between some network nodes while the failing links are not reestablished. In such a solution, some nodes must be selected to act as gateway nodes between the two networks.

This work has addressed the gateway node selection problem aiming to improve the connectivity resilience of a given network as much as possible to multiple link failures. To address this problem, a Gateway Node Selection (GNS) algorithm was proposed where the most damaging sets of failing links are identified and, then, a set cover problem type is defined and solved to select the gateway nodes.

Two well-known network topologies were used in the computational results and the GNS algorithm was run for the simultaneous failure of up to a maximum of $L = 6$ links and to the selection of up to $B = 4$ gateway nodes. The computational results have shown that the proposed algorithm can solve all problem instances. Concerning the connectivity resilience gains obtained by using a third-party network, the results have shown that the gains are very much case dependent, i.e., they depend on the particular network topology, and on the particular values of L and B.

Acknowledgments

This paper is based upon work from COST Action CA15127 ("Resilient communication services protecting end user applications from disaster-based failures – RECODIS") supported by COST Association. The work was financially supported by FCT, Portugal, under the project CENTRO-01-0145-FEDER-029312. This work was also funded by FCT/MCTES through national funds and when applicable co-funded EU funds under the project UIDB/EEA/50008/2020.

References

[1] T. Gomes, et al., "A survey of strategies for communication networks to protect against large-scale natural disasters", RNDM, pp. 11–22, Sept 2016.

[2] M. Furdek, L. Wosinska, R. Goścień, K. Manousakis, M. Aibin, K. Walkowiak, S. Ristov, M., and J. L. Marzo, "An overview of security challenges in communication networks", RNDM, pp. 43–50, Sept 2016.

[3] N. Skorin-Kapov, M. Furdek, S. Zsigmond, and L. Wosinska, "Physical-layer security in evolving optical networks," IEEE Commun. Mag., vol. 54, no. 8, pp. 110–117, Aug 2016.

[4] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee, "Survivable WDM mesh networks", IEEE J. of Lightwave Technology, vol. 21, no. 4, pp. 870–883, April 2003.

[5] J. Zhang, K. Zhu, and B. Mukherjee, "Backup reprovisioning to remedy the effect of multiple link failures in WDM mesh networks", IEEE J. on Selected Areas in Communications, vol. 24, no. 8, pp. 57–67, Aug 2006.

[6] S. Yuan and B. Wang, "Highly Available Path Routing in Mesh Networks Under Multiple Link Failures", IEEE Trans. on Reliability, vol. 60, no. 4, pp. 823–832, Dec 2011.

[7] B. Jaumard and H. A. Hoang, "Design and dimensioning of logical survivable topologies against multiple failures", IEEE/OSA J. of Optical Communications and Networking, vol. 5, no. 1, pp. 23–36, Jan 2013.

[8] J. Tapolcai, L. Rónyai, B. Vass, and László Gyimóthi, "List of Shared Risk Link Groups Representing Regional Failures with Limited Size", IEEE INFOCOM, pp. 1–9, Atlanta, USA, May 2017.

[9] B. Nedic, M. Gunkel, T. Gomes, and R. Girão-Silva, "SRLG-disjointness and geodiverse routing – a practical network study and operational conclusions", RNDM, pp. 1–8, Longyearbyen, Norway, Aug 2018.

[10] S. Ramasubramanian and A. Chandak, "Dual-link failure resiliency through backup link mutual exclusion", IEEE/ACM Trans. On Networking, vol.16, no.1, pp.157–169, Feb 2008.

[11] N.-H- Bao et al., "On Exploiting Sharable Resources With Resource Contention Resolution for Surviving Double-Link Failures in Optical Mesh Networks", J. of Lightwave Technology, vol. 30, no. 17, pp. 2788–2795, Sept 2012.

[12] V. Liu and D. Tipper, "Spare capacity allocation using shared backup path protection for dual link failures", Computer Communications, vol.36, no.6, pp. 666–677, Mar 2013.

[13] H. Guo, G. Shen, and S.K. Bose, "Routing and Spectrum Assignment for Dual Failure Path Protected Elastic Optical Networks", IEEE Access, vol. 4, pp. 5143–5160, Aug 2016.

[14] T. N. Dinh, Y. Xuan, M. T. Thai, P. M. Pardalos, and T. Znati, "On new approaches of assessing network vulnerability: hardness and approximation", IEEE/ACM Tran. on Networking, vol. 20, no. 2, pp. 609–619, 2012.

[15] S. Yin et al., "Shared-Protection Survivable Multipath Scheme in Flexible-Grid Optical Networks Against Multiple Failures", J. of Lightwave Technology, vol. 35, no. 2, pp. 201–211, Jan 2017.

[16] C. Natalino, A. de Sousa, L. Wosinska, and M. Furdek, "On the Trade-offs between User-to-Replica Distance and CDN Robustness to Link Cut Attacks", RNDM, pp. 1–7, Longyearbyen, Norway, Aug 2018.

[17] S. Xu et al., "Multi-Carrier Interconnection-based Emergency Packet Transport Network Planning in Disaster Recovery", DRCN, pp. 109–116, Munich, Germany, Mar 2017.

[18] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski, "SNDlib 1.0 – survivable network design library," Networks, vol. 55, no. 3, pp. 276–286, May 2010.