Network Robustness Improvement based on Alternative Paths Consideration

Jean-Claude Lebègue, Daniel Delahaye, Jacco Hoekstra

Network Robustness Improvement based on Alternative Paths Consideration

Jean-Claude Lebègue
Aeroline
Sopra Steria
Toulouse, France
jean-claude.lebegue@soprasteria.com

Daniel Delahaye
OPTIM, ANITI
ENAC LAB
Toulouse, France
daniel@recherche.enac.fr

Jacco Hoekstra
Faculty of Aerospace Engineering
TU Delft
Delft, Netherlands
J.M.Hoekstra@tudelft.nl

***Abstract*—Many services have a complex network infrastructure (transport, Internet, etc.). Disruptions caused by accidents or weather conditions can paralyze traffic for a long time. But how to improve the robustness of networks? This robustness improvement would reduce the time of exposure to disruptions and their impact on overall traffic. Lots of studies have been done on identifying critical links and nodes but not so many analyze the paths. In this paper, we propose a new method to measure network robustness based on alternative paths. Beyond improving the French low-cost flight network robustness by 15%, the method attempts to show the relevance of analyzing network vulnerability from a path-based approach.**

***Keywords*—Network robustness; Link adding; Floyd-Warshall algorithm; Simulated annealing; Operations Research.**

I. Introduction

Every year, the world's population grows, and with it the size of our networks (Internet, transportation, power grid, etc.) to meet the increasing demand. Telecommunication networks keep expanding at a fast pace. At its beginning, in the seventies, the Internet was a military network counting less than hundreds of users, now there are billions. The transportation network is not only growing according to a vertical axis but also to a horizontal one. Vertical growth is related to the increase in the size of the network associated with a transportation mode whereas horizontal growth refers to the number of transportation means. For instance, the power grid is mainly concerned with vertical growth caused by the city's size increasing. The bigger the network, the more complex it is.

This complexity brings a lot of new challenges to deal with. The quality of service should remain at least the same, no matter the size of the network. People expect to spend the same amount of time doing what they used to do before the network gets bigger. Maybe the most important challenge is the load, especially during peak hours. The network is supposed to offer an acceptable service even when capacity is reached. In a society where competition between services is a norm, it can be a reason to shift from one service to another. Finally, the network must be robust to disruptions. At a time like ours, information is a precious resource. Therefore, a telecommunication network is expected to be robust against all kinds of cyberattacks.

Robustness is so important that a branch of the research has been dedicated to the topic. The analysis of network vulnerability has developed several models to study networks. These models try to quantify to which extent infrastructure elements are vulnerable to breakdowns, congestion, attacks, etc. The methods to tackle the problem are numerous but can be gathered into two groups: topological vulnerability analysis and system-based vulnerability analysis. The metrics based on the former come from complex network theory and usually use graph properties without considering the dynamics of the transportation networks. The models from the latter method overcome this aspect by integrating notions from transportation theory. Vulnerability is usually quantified by the difference in cost between a nominal state and a disrupted state.

Robustness does not focus only on identifying critical elements but also defines methods to improve the network's robustness. The strategies to improve the robustness are numerous. The rewiring strategy gets a lot of attention lately thanks to its network properties conservation aspect, in particular, the node degree conservation. Sometimes the choice of a rewiring strategy seems to be justified by economic benefits however this kind of argument is not as evident as one may think. Firstly, it depends on the characterization of the network. On transportation networks, rewiring between non-directly connected elements involves adding a new connection. Secondly, establishing new connections at the strategic level may be a possible and interesting option.

The following paper presents a new topological method to improve network robustness based on alternative paths. The strategy chosen here is the addition of new elements. It is well known that robustness is correlated to the number of links in a network and by extension the number of paths. However, path enumeration is a time-consuming task. Instead of using classical k shortest path algorithms, the authors propose a modified Floyd-Warshall algorithm to quantify this property and use a simulated annealing algorithm to find the set of links whose addition will improve the network robustness.

The paper is structured as follows. The next section provides state-of-the-art robustness improvement methods. Section III three describes the model and an algorithm for link adding which maximizes the robustness. Section IV describes the

methodology. The section V presents the results obtained with such a method and discusses the benefits and shortcomings of our approach and the last section highlights future works.

II. Related Work

A lot of measures are defined in the literature to model network robustness. Some models characterize robustness by the size of the largest connected component after the removal of the graph's elements (nodes or links) [1] [2] [3] [4] [5]. These approaches are purely topological and therefore do not capture transportation features. Moreover, there are other robustness models based on the shortest path [6], and among them are the centrality indices [7][8][9]. These models consider only one path connecting Origin-Destination (OD) pairs whereas the traffic assignment methods use several. Another attempt uses the network spectrum especially the algebraic connectivity [10][11]. This measure is a well-known qualitative index to compare OD pair's connectivity but lacks precision.

Improving network robustness is a complex task because of the size of the state space. Different approaches were developed to address the problem often based on a random approach. Several papers have shown that a topological change of a network structure can significantly improve its robustness [1][5][12]. The most acknowledged strategies consist in adding and/or removing elements from a network [13][14][15] or rewiring existing connections to form new connections [1][4][16].

Floyd-Warshall's algorithm received a lot of attention from the research community. Several works were done to improve the computation time by different means such as GPUs and parallelism [17], cache optimization [18], matrix multiplication optimization [19]. The algorithm was also generalized to compute the k shortest paths [20][21]. However, it seems that nothing has been done on attempting to assess the number of paths connecting all node pairs.

The method proposed in this paper is a path-based robustness method that considers OD pairs. The passenger-centric robustness is defined by the number of alternative paths connecting the OD pairs. The more there are alternatives to a path the more robust it is. The network robustness improvement is also done by a topological strategy. We choose the adding strategy because instead of the other methods it always increases the number of paths making it the most consistent with the robustness model.

III. Model

Path-based models are exceedingly rare. Those that do follow this approach only focus on the shortest path. From the passenger's perspective, the alternative at each stage of a trip is as important as the chosen path to the destination.

A. Problem formulation

Let us consider a transportation network $T = (G, C, D)$ [22][23] where:

- $G = (N, L)$ is a strongly connected digraph with N being the set of nodes and L the set of links.
- C is a set of unit costs (cost per unit of flow) of transport on the different links of the network.
- D represents the demand which is the number of people traveling from an origin node o to a destination node d in the network. Moreover, $O_N \subseteq N$ and $D_N \subseteq N$ are respectively the sets of origin nodes and destination nodes.

In this paper, the transportation network robustness is related to the richness of alternatives. The more paths connect OD pairs, the more there are alternatives for rerouting the passengers when a disruption happens. The diversity of paths helps to maintain a network connection, which is one of the most fundamental criteria when it comes to analyzing robustness because it ensures the existence of a path between all pairs of nodes.

We are interested in a robust transportation network model based on alternative paths between an OD pair. Let us consider a path $p = (o, a, b, d)$ connecting an OD pair od and $r_o^{od}, r_a^{od}, r_b^{od}$ being the respective robustness of the nodes o, a and b. These robustness values are characterized by the number of alternative paths between a node $i \in (o, a, b)$ and the destination node d. We are looking for a path model that combines these r_i^{od} values. The goal behind this definition is to capture the alternatives at each node i along the path p. A robust path always possesses an alternative to the destination in case of a disruption on the nominal path p.

The model is hardly applicable to large transportation networks because it involves computing the shortest paths for all paths connecting all the OD pairs, and this operation is very time-consuming and requires a lot of memory resources. Instead of the model previously described, we focus on a less restrictive model in the following observation. In general, the shortest paths connecting two nodes are remarkably similar. They slightly differ from the shortest path. It is based on this principle that the Yen algorithm [24] computes the k shortest paths. As illustrated by Figure 1, the shortest path (green-plained path) between O and D is (O, C, D) with a cost of 4. To compute the two other paths (green-dashed paths), the algorithm is going to iteratively remove (red-dashed links) the links (C, D) and (O, C) from the network to respectively find the paths (O, C, A, D) and (O, A, D). Both paths have a cost of 7.

To make the running time acceptable, we limit the method to the computation of the paths connecting an OD pair. The main benefit of this approach is to combine the computation of the distinct paths connecting the OD pairs and the computation of the alternatives along them. The robustness of- an OD pair is quantified by the paths connecting its origin to its destination.

The model proposed by the authors can be defined by the following optimization problem:

$$\begin{cases} \max\limits_{p \in \mathcal{P}(G \setminus L)} f(p) = \sum\limits_{o \in O_N} \sum\limits_{d \in D_N} \Omega_{od}(L \cup p) & (1) \\ card(p) < K & (2) \end{cases}$$

The strategy adopted here consists of adding elements. The

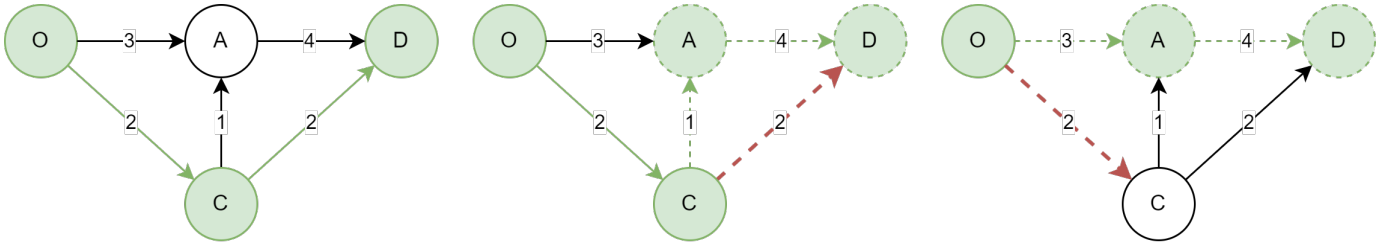

Figure 1. Yen 3 shortest paths (green paths) computation process between OD pair od based on the disconnection (red-dashed links) of each of the links forming the shortest path (green-plained path)

model looks for the set of links p that is going to increase the number of paths from all origin nodes to all destination nodes. The Ω_{od} function gives the number of paths connecting nodes o and d in graph G where the set of links p has been added to the set L. The only constraint (Inequality 2) stands for limiting the size of the solution and therefore the combination of possibilities.

B. Modified Floyd-Warshall algorithm

The heuristic chosen to solve the problem is the simulated annealing [25]. This metaheuristic is a well-known algorithm in the operations research community to deal with optimization problems. This approach can reach the quasi-global optimum solution in a reasonable computation time.

Data: adjacency matrix of a strongly directed digraph
Result: set p of links that best improve the robustness of the network
$p \leftarrow$ random set of non-existing links
$f_p \leftarrow$ objective function evaluation for p
$T \leftarrow$ initial temperature
repeat
  for $k <$ *number of transitions* **do**
    generate a neighbor n of p
    evaluate the objective function f_n for n
    $r \leftarrow$ random number in [0,1]
    if $f_n > f_p$ *or* $r < e^{\frac{f_n - f_p}{T}}$ **then**
      $f_p \leftarrow f_n$
      $p \leftarrow n$
    end
  end
  decrease the temperature T
until *temperature is sufficiently low*;

Figure 2. Simulated annealing pseudo-code to solve the optimization problem (Equation 1).

The main principle of the simulated algorithm (Figure 2) consists in generating a neighbor n and comparing its objective evaluation f_n with the current decision evaluation f. If f_n is better, f and p are updated. However, these variables have a non-null probability to be updated even if f_n is worst so that the algorithm does not stay in a local maximum. In our implementation, we used the well-known metropolis rule [25] to accept from time to time a bad solution.

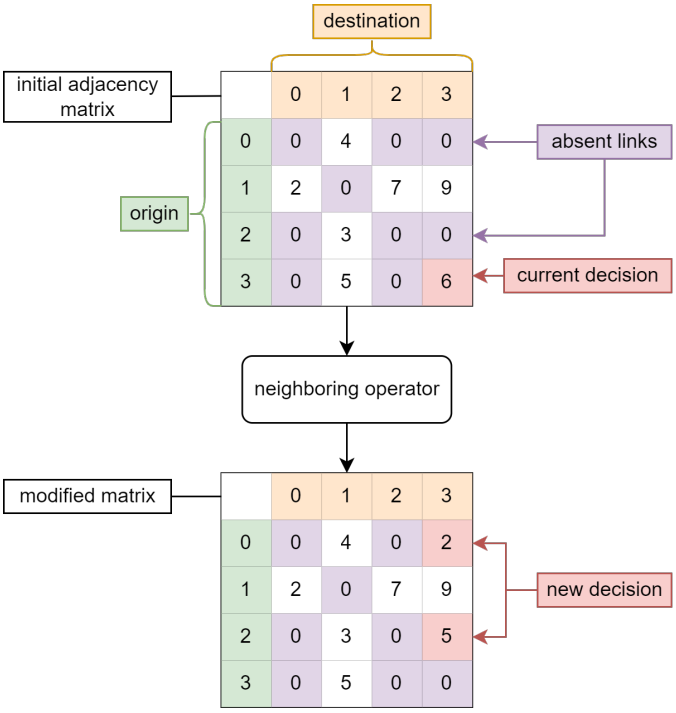

Figure 3. Simulated annealing neighboring operator functioning. The operator modifies the current decision to a new decision according to three operations: adding, removing, and swapping.

The core of the simulated annealing algorithm is the neighboring operator because it is the process that is going to modify the taken decision (Figure 3). The neighboring operator implemented is threefold: ten percent of the time, the size of the decision is increased as illustrated in Figure 3. The decision goes from one link to two. Another ten percent is used to reduce the size of the decision by one element. The rest of the time is spent swapping the elements in the decision with the same number of other random elements from the absent links set.

Now that the method to solve the optimization problem has been presented, the last thing to do is to find a way to evaluate the objective function (Equation 1). It is a tough task to do it fast due to the number of possibilities. Here, the authors propose a modified Floyd-Warshall [26] algorithm (Figure 4) to tackle the problem. The approach proposed here does not focus at all on computing the shortest paths between all pairs of nodes instead it estimates the number of alternative paths.

Data: strongly directed digraph $G(N, L)$
Result: alternative paths matrix P
$P \leftarrow$ paths set matrix
for $k \in N$ **do**
  for $i \in N \setminus \{k\}$ **do**
    for $j \in N \setminus \{k, i\}$ **do**
      $P_{ij} \leftarrow P[i][k] \oplus P[k][j]$
      filter the paths in P_{ij}
      $P[i][j] \leftarrow P[i][j] \cup P_{ij}$
    end
  end
end

Figure 4. Modified Floyd-Warshall algorithm to assess the number of alternative paths between all pairs of nodes without looking for the shortest paths.

	France	Europe
nodes	47	391
links	363	10786

TABLE I. NUMBER OF NODES AND LINKS OF THE FRENCH AND EUROPEAN LOW-COST FLIGHT NETWORKS

Like the classical Floyd-Warshall algorithm, the version proposed in this paper builds the paths connecting OD pairs (i, j) by passing through a transit node k. The paths P_{ij} are the result of the concatenation of the paths $P[i][k]$ (connecting OD pair (i, k)) and the paths $P[k][j]$ (connecting OD pair (k, j)). Then, the set P_{ij} is filtered to remove cycles and expensive paths. Finally, the set P_{ij} of paths going through k is added to $P[i][j]$ which contains the other paths connecting (i, j).

IV. METHODOLOGY

The robustness improvement method presented in the previous section was tested on the French and European low-cost flight networks on 30/06/2018. The network data come from the Eurocontrol database. The major features of both networks are summarized in table I. Each node is an airport. Two nodes are connected by a link if there is at least one flight between the two airports in the data. The travel time of the leg is the difference between the off-block time of the departure airport and the in-block time of the arrival airport.

No information was available on the actual demand, so both sets O_N and D_N were considered to be equal to N because it is the worst-case scenario. We assume that the newly added flight was performed by an A320 aircraft at the speed of 863 km/h (average time speed). The new leg travel time is approximated by the en-route travel time between the two connected airports which is lesser than the real value. The maximum number of possible added links was set to three because our goal was to improve the robustness through minimal topological changes. Besides, the number of transitions was fixed to 50 and the decreasing cooling coefficient to 0.97 to allow a suitable state space exploration. Finally, the maximum travel time was respectively fixed to 15000 which is approximately four hours (resp. 13000 about 3.6 hours), and the number of legs to 4 for the French network (resp. European network).

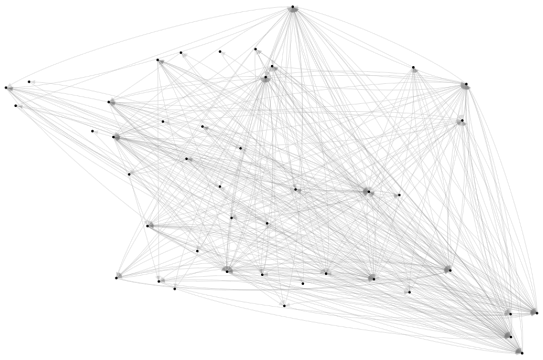

Figure 5. France strongly connected low-cost flights network of June 30th, 2018 composed of 47 nodes and 363 links

The raw data set contained internal and external flight data. In this study, we focus only on internal flights. All the external data were removed from the data set.

V. RESULTS

A. French low-cost network

The first study case was done on the French low-cost flight network. The network is shown in Figure 5. The best solution is ((LFBO, LFCK), (LFKB, LFKJ), (LFKB, LFMN)). When this set is added to the network, the number of alternatives is increased by 14.9%, and these alternatives concern 34.3% of the OD pairs.

Figure 6 shows the distribution of the newly added paths for only the improved OD pairs after integrating the links ((LFBO, LFCK), (LFKB, LFKJ), (LFKB, LFMN)) to the transportation network. Most of the OD pairs improved have their number of paths increased by less than two: 45.9% received one new path and 35.9% received two. The OD pair (LFKB, LFLL) gets the maximum number of alternatives with 29 new paths. However, having a lot of alternatives is not that relevant because all these paths will not be used for reassignment in case of a disruption. In this study case, 88% of the OD pairs improved get less than five paths.

Considering O_N and D_N represent the worst-case scenario. We release these constraints to study the influence of the demand on the result. The results are plotted on Figure 7. For varied sizes of the solution (from one in blue to three in grey in Figure 7), we compute the improvement for several demands. The X-axis is related to the percentage of $N \times N$ (worst case scenario for O_N and D_N) used as the demand and the Y-axis is the improvement brought by the found solution.

For all three curves, the improvement is often slightly better for 10% and 40% of the demand. They also have almost the same shape: the improvement decreases from 10 to 30%, then increases at 40% and decreases again except for the three-size solution where it increases from 50% to 70% and finally decreases. The improvement's variations are small, roughly 2% for all the curves, so they can be considered

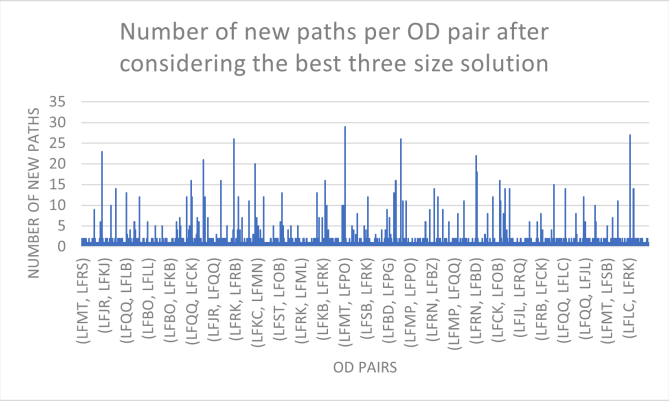

Figure 6. Distribution of the newly added paths per OD pair after adding the links (LFBO, LFCK), (LFKB, LFKJ), and (LFKB, LFMN) on the French low-cost flight network

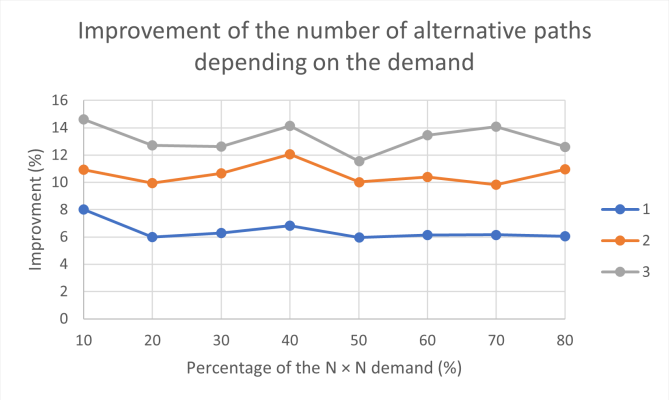

Figure 7. Improvement in the percentage of the number of alternative paths depending on the percentage of the number of OD pairs considered for different solution sizes

constant. Therefore, the demand has very little influence on the improvement. Moreover, for each OD pair, the three-size solution has always an improvement greater than the two-size solution whose improvement is also always greater than the one-size solution. Indeed, for the same demand, a larger solution should at least contain the same links found by a smaller solution.

B. European low-cost network

The last study concerns the European low-cost flight network (Figure 8). We looked for a solution with a size capacity still lesser than three. The best-found solution was ((EDDL, LSZB), (LFOK, LTCN), (EDDK, EHAM)) which connects Düsseldorf to Berne, Paris-Vatry to Kahramanmaraş, and Konrad-Adenauer, to Schiphol. The improvement is about 2.76%.

This time, the solution affected 3% of the OD pairs. We remind the OD pairs considered for this case is 152 490. We also computed the distribution of the added paths for the improved OD pairs (Figure 9). Like the previous case, the path distribution is not uniform. The main OD pairs have been granted from one to three new path even though others got

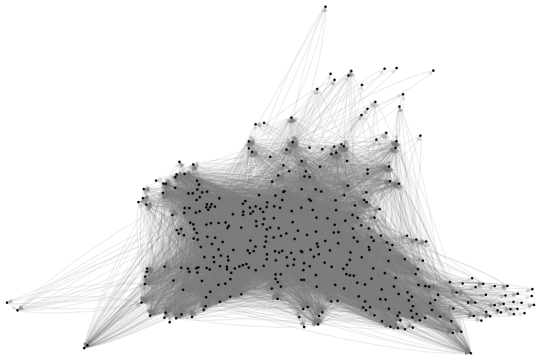

Figure 8. Europe strongly connected low-cost flights network of June 30th, 2018 composed of 391 nodes and 10786 links

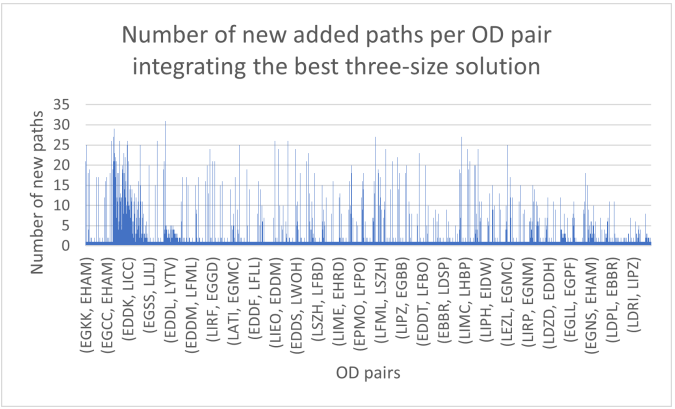

Figure 9. Distribution of the newly added paths per OD pair after adding the links (EDDL, LSZB), (LFOK, LTCN), and (EDDK, EHAM) on the European low-cost flight network

more than twenty paths. Two OD pairs ((EDDK, LEBL) and (EDDL, EHAM)) got almost thirty new paths.

VI. Conclusion

The paper has presented a new model of transportation network robustness based on alternative paths between OD pairs. We also developed a method to improve it. The method is the combination of a simulated annealing metaheuristic and a modified Floyd-Warshall algorithm. The latter is used to assess the number of paths connecting the OD pairs. The method has been tested on two static study cases: the French and European low-cost flight networks. The number of alternatives in the French (resp. European) network has been improved by 15% (resp. 3%), and these alternatives concern 34% (resp. 3%) of the OD pairs. The difference in results between these two studies is due to the size of the decision variable and the network's size. Adding three links to the 44-node French network has more impact than adding three links to the 391-node European network. By slightly changing the network topology, the studies have shown it is possible to greatly improve the number of alternatives. Having several alternatives is interesting during a disruption because it reduces the costs generated and the resources to mobilize to reassign the passengers.

The main drawback of robustness path models is the running time and memory resource consumption, especially on large networks. It can be helpful to look for the parallel version of the Floyd-Warshall algorithm presented here. Besides, the decision is slightly changed between two simulated annealing iterations, so it is worthwhile to search for a way to update the paths based on these changes.

The model in its current version only maximizes the global number of alternative paths in the network. However, above a certain threshold, it is no more relevant to have alternatives for one OD pair. An upgrade can maximize the number of enhanced OD pairs as well as the alternative paths.

The model proposed is static, so it can be interesting to develop a dynamic version to fully capture the key features of the transportation networks and make the model more realistic. Considering schedules, filling rates, and operational constraints on planes can help to analyze the quality of the solution, and on the other hand reduce the state space for the decision and the number of alternatives to compute.

REFERENCES

[1] A. A. M. C. M. Schneider, S. H. J. S. Andrade, and H. J. Herrmann, "Mitigation of malicious attacks on networks," *Proceedings of the National Academy of Sciences*, vol. 108, pp. 3838 – 3841, 2011.

[2] L. H. L. Bai, Y. Xiao and S. Lao, "Smart rewiring: Improving network robustness faster," *Chinese Physics Letters*, vol. 32, p. 078901, 2015.

[3] M. P. Q. Cai and S. Alam, "Interdependency and vulnerability of multipartite networks under target node attacks," *Complex.*, vol. 2019, pp. 2 680 972:1–2 680 972:16, 2019.

[4] H. J. H. V. H. P. Louzada, F. Daolio and M. Tomassini, "Smart rewiring for network robustness," *IO: Theory eJournal*, 2013.

[5] A. Zeng and W. Liu, "Enhancing network robustness for malicious attacks," *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 85 6 Pt 2, p. 066130, 2012.

[6] M. Y. A. Sydney, C. M. Scoglio and P. Schumm, "Characterizing the robustness of complex networks," *arXiv: Networking and Internet Architecture*, 2008.

[7] T. Bröhl and K. Lehnertz, "Centrality-based identification of important edges in complex networks," *Chaos*, vol. 29 3, p. 033115, 2019.

[8] D. C. E. Yu and J. Zhao, "Identifying critical edges in complex networks," *Scientific Reports*, vol. 8, 2018.

[9] J. S. I. Kivimäki, B. Lebichot and M. Saerens, "Two betweenness centrality measures based on randomized shortest paths," *Scientific Reports*, vol. 6, 2016.

[10] C. M. S. A. Sydney and D. M. Gruenbacher, "Optimizing algebraic connectivity by edge rewiring," *Appl. Math. Comput.*, vol. 219, pp. 5465–5479, 2013.

[11] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, pp. 298–305, 1973.

[12] J. W. Wang, "Robustness of complex networks with the local protection strategy against cascading failures," *Safety Science*, vol. 53, pp. 219–225, 2013.

[13] R. L. A. Beygelzimer, G. Grinstein and I. Rish, "Improving network robustness by edge modification," *Physica A-statistical Mechanics and Its Applications*, vol. 357, pp. 593–612, 2005.

[14] W. Abbas, M. Shabbir, H. Jaleel, and X. Koutsoukos, "Improving network robustness through edge augmentation while preserving strong structural controllability," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 2544–2549.

[15] D. S. P. V. Mieghem, C. L. F. A. Kuipers, D. L. R. van de Bovenkamp, and H. Wang, "Decreasing the spectral radius of a graph by link removals." *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 84 1 Pt 2, p. 016101, 2011.

[16] H. Chan and L. Akoglu, "Optimizing network robustness by edge rewiring: a general framework," *Data Mining and Knowledge Discovery*, vol. 30, pp. 1395–1425, 2015.

[17] H. N. Djidjev, S. Thulasidasan, G. Chapuis, R. Andonov, and D. Lavenier, "Efficient multi-gpu computation of all-pairs shortest paths," *2014 IEEE 28th International Parallel and Distributed Processing Symposium*, pp. 360–369, 2014.

[18] J. S. Park, M. Penner, and V. K. Prasanna, "Optimizing graph algorithms for improved cache performance," *IEEE Trans. Parallel Distributed Syst.*, vol. 15, pp. 769–782, 2004.

[19] S. C. Han, F. Franchetti, and M. Püschel, "Program generation for the all-pairs shortest path problem," *2006 International Conference on Parallel Architectures and Compilation Techniques (PACT)*, pp. 222–232, 2006.

[20] S. K. Lam and T. Srikanthan, "Accelerating the k-shortest paths computation in multimodal transportation networks," *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems*, pp. 491–495, 2002.

[21] E. Minieka, "On computing sets of shortest paths in a graph," *Commun. ACM*, vol. 17, pp. 351–353, 1974.

[22] D. Delahaye, "Optimisation de la sectorisation de l'espace aérien par algorithmes génétiques," Ph.D. dissertation, ENAC, Toulouse, France, 1995.

[23] M. Patriksson, *The Traffic Assignment Problem: Models and Methods*, ser. Dover Books on Mathematics. Dover Publications, 2015.

[24] J. Y. Yen, "Finding the k shortest loopless paths in a network," *Management Science*, vol. 17, no. 11, pp. 712–716, 1971. [Online]. Available: https://doi.org/10.1287/mnsc.17.11.712

[25] S. C. D. Delahaye and M. Mongeau, "Simulated annealing: From basics to applications," *Handbook of Metaheuristics*, 2018.

[26] R. W. Floyd, "Algorithm 97: Shortest path," *Communications of the ACM*, vol. 5, p. 345, 1962.