Models and Algorithms for Network Interdiction Problems

Luis Salazar Zendeja[1], Diego Cattaruzza[1], Martine Labbé[2], Frédéric Semet[1]

[1] Ecole Centrale de Lille, Lab INOCS @Inria Lille-Nord Europe, Lille, France
{luis.salazar-zendeja,diego.cattaruzza,frederic.semet}@inria.fr
[2] Université Libre de Bruxelles, Brussels, Belgium
mlabbe@ulb.ac.be

Keywords : *minimum spanning tree, network problems, interdiction problems.*

1 Introduction

Network Interdiction problems consist of two opposing forces interacting with each other. The operator of the network, referred as the defender, wants to find an optimal solution of network related problems, such as the calculation of the minimum shortest path, minimum spanning tree or the maximum flow network problem. The second individual, called interdictor, desires to worsen the objective of the defender by changing the topology of the network, i.e., removing or modifying the dimension of the edges. However, the interdiction is limited by a budget.

This research focuses on models, algorithms and results for the Minimum Spanning Tree Interdiction Problem (MSTIP). To the best of our knowledge only in [1] are stated mathematical formulations of the MSTIP. The authors propose two enumeration algorithms and several mixed integer formulations. Our goal is to contribute to the resolution of the MSTIP through new mathematical formulations.

The MSTIP can be defined as follows. Let $G = (V, E)$ be an undirected graph where V is the vertex set such that $|V| = n$. E represents the edge set. Each edge $e = \{i, j\}$ is associated with a weight c_e, a budget requirement b_e to interdict the edge and a interdiction increment d_e that is added to c_e if the edge is interdicted. When interdiction represents total edge deletion, d_e is set to a large enough value for all $e \in E$. The objective of the MSTIP is to determine an interdiction plan such that the interdiction budget B is respected and the weight of the minimum spanning tree in the resulting graph is as large as possible.

A special case of the MSTIP considers the edge interdiction budget $b_e = 1$ and the interdiction increment $d_e = +\infty$ for all $e \in E$. This particular case of the MSTIP is usually referred as the B most vital edges interdiction problem. In this case, the interdiction constraint is called cardinality constraint since exactly B edges are to be removed.

The authors in [2] propose an algorithm to tackle the B most vital edges of the minimum spanning tree. In the latter work, a preprocessing on the network is made by only considering a subset of the network's edges which is called the *sparse B-edge connected certificate*. Moreover, it is proved that all B edges removed from the network belong to this certificate. Then, [1] uses these results and develop their own algorithms along with mixed integer formulations for the B most vital edges of the problem. Note that the MSTIP generalizes the B most vital edges by considering interdiction budget associated with edges and increment on the edge cost rather than complete deletion.

2 Formulations for the MSTIP

Throughout our research, four mathematical formulations of the MSTIP are defined. The MSTIP can be formulated as a $max - min$ bilevel problem where the attacker selects the edges to interdict in order to maximize the value of the minimum spanning tree selected by the defender. The model that is obtained, $[MSTIP]$, has a non-linear objective function which is linearized using classical linearization techniques. The lower level uses the formulation of the minimum spanning tree presented in [3]. That allowed to develop optimality constraints to obtain a single-level problem.

The second and third models are called $[D - MSTIP]$ and $[SD - MSTIP]$, respectively. $[D - MSTIP]$ is obtained by computing the dual of the inner problem of $[MSTIP]$.

$[SD - MSTIP]$ is obtained by first reducing the number of variables and constraints in [MSTIP] before computing the dual of the inner problem. The last formulation modifies the objective function of $[MSTIP]$ and includes one additional constraint that impose to either interdict or select an edge. This model only considers complete deletion of edges, thus its name $[C - MSTIP]$.

3 Computational Experimentation

Preliminary computational results are reported here. Table 1 shows the resolution time in seconds that the different models require to obtain the optimal solution on three test instances. All of the instances are of 20 nodes. The interdiction budget is set to 1 for all edges in E. Values of B equal to 3, 5 and 7 are considered. Parameters c_e are generated randomly, uniformly distributed in $[1, 100]$ for all $e \in E$. If an edge is interdicted, the penalization for that edge is defined as $d_e = \max_{e \in E} c_e + 1 - c_e$ for all $e \in E$. Our results are compared to $[B - MSTIP]$ which is the formulation defined in [1]. The models were solved using CPLEX on a Ubuntu 18.04.3 64-bit computer with an Intel Xeon(R) CPU E5-2630 v4 (2.20GHz) and 62.8 GB of RAM.

n	B	$B - MSTIP$	$MSTIP$	$D - MSTIP$	$SD - MSTIP$	$C - MSTIP$
	3	11.79	201.67	7.73	9.06	70.22
20	5	203.57	4455.50	92.89	78.42	384.50
	7	1576.83	30822.23	968.87	755.09	1670.83

TAB. 1: Resolution time of test instances

These preliminary results show that models $[D - MSTIP]$ and $[SD - MSTIP]$ outperform the model proposed in [1].

4 Conclusions

Our formulations take advantage of the formulation proposed in [3] to model the Minimum Spanning Tree Problem. We develop optimality conditions or use duality theory to reduce the bi-level formulation into a single level formulation for the MSTIP. We use classical linearization techniques to get rid on non-linear terms when these appear. Encouraging preliminary results are obtained on random generated instances.

References

[1] Bazgan, C., Toubaline, S., and Vanderpooten, D. Efficient determination of the k most vital edges for the minimum spanning tree problem. *Computers & Operations Research 39*, 11 (Nov 2012), 2888–2898.

[2] Liang, W., and Shen, X. Finding the k most vital edges in the minimum spanning tree problem. *Parallel Computing 23*, 13 (Dec 1997), 1889–1907.

[3] Martin, R. Using separation algorithms to generate mixed integer model reformulations. *Operations Research Letters 10*, 3 (Apr 1991), 119–128.