A tri-level Network Protection Problem

Margarida Carvalho[1], Pierre Hosteins[2], Rosario Scatamacchia[3]

[1] Polytechnique Montréal, Montréal, Canada
carvalho@iro.umontreal.ca

[2] University Lille Nord de France, IFSTTAR, COSYS, ESTAS, Villeneuve d'Ascq, France
pierre.hosteins@ifsttar.fr

[3] Politecnico di Torino, Dipartimento di Ingegneria Gestionale e della Produzione, Torino, Italy
rosario.scatamacchia@polito.fr

Mots-clés : *tri-level optimisation, network interdiction, cutting plane algorithm, network protection.*

1 Introduction

Network Interdiction Problems (NIPs) are problems where an attacker tries to disable some network elements (usually edges or vertices) submitted to an attack budget, while a defender reacts to such an attack in order to optimise some kind of connectivity on the network. Classic examples are, among many others, the Maximum Flow Interdiction or Shortest Path Interdiction problems, where an attacker disables edges to minimise the maximum flow or maximise the length of the shortest path between two nodes. Such problems have many possible applications, such as assessing the robustness of a transportation or energy distribution network, studying the structure of social networks or biological networks and of course military and security applications. Given their attacker-defender structure, they are often modeled as bi-level optimisation problems. We refer the reader to the recent study of [1] for an overview of NIPs and the Mathematical Programming techniques used to solve them.

Though the focus is often on finding the most critical set of elements to disable from an attacker's point of view, a few works exist in the literature that instead focus on how to protect the network from the malign intervention of an attacker. These problems very often take the form of a *fortification* of a subset of graph elements, i.e. rendering some graph elements impervious to attacks, see e.g. [2]. To our knowledge, no approach has been studied to act instead on the relative deletion costs of the different graph elements. However, we feel that in many situations it should be possible to allocate more resources to protect a specific part of the network with respect to other parts, therefore requiring a larger effort on the part of the attacker to disable the respective graph elements. We refer to such a problem as the Network Protection Problem (NPP) in the following.

2 Tri-level formulation and cutting plane reformulation

Next we model the NPP as a tri-level problem. We consider that the defender has a protection budget W, the attacker an attack budget K and that both edges and nodes can be interdicted. Define the following variables :

— w_i, $w_{ij} \geq 0$: deletion costs for nodes $i \in V$ and edges $\{i, j\} \in E$;
— v_i, v_{ij} : binary variable equal to 1 if $i \in V$ or $\{i, j\} \in E$ is deleted from G, 0 otherwise ;
— x : general variables of the follower to optimise the connectivity measure on the network.

We call $X(v)$ the domain of definition of the lower level (defender) variables. The NPP is :

$$\max_{w} C_d \tag{1}$$

$$\sum_{i \in V} w_i + \sum_{\{i,j\} \in E} w_{ij} \ \leq W \tag{2}$$

$$C_d = \min_{v} C_a(w) \tag{3}$$

$$\sum_{i \in V} w_i v_i + \sum_{\{i,j\} \in E} w_{ij} v_{ij} \ \leq K \tag{4}$$

$$C_a = \max_{x} C'_d(x) \tag{5}$$

$$x \in X(v), \tag{6}$$

where C_d is the objective value of the defender at the higher level, C_a is the objective of the attacker at the second level and finally C'_d is the objective of the defender at the lower level. We will concentrate on problems where the lower level is a polynomial problem and the two lower levels can be transformed into a single level (by, e.g., inner dualisation), making the problem bi-level, where the lower level is the traditional NIP and the upper problem tries to allocate the deletion costs optimally. We have chosen continuous values for the deletion costs w for the sake of simplicity but one could define them as integers.

We write the model in a simpler form which enumerates all the second level solutions with their respective objective value. Define $\mathcal{S}$ as the set of second level (attacker) solutions : a $s \in \mathcal{S}$ is defined by parameters $\bar{v}_i^s$ and $\bar{v}_{ij}^s$ equal to 1 if the graph elements are deleted in solution s, along with the connectivity C_s associated to s. We can reformulate the above MINLP as :

$$\max_{w} C_d \tag{7}$$

$$\sum_{i \in V} w_i + \sum_{\{i,j\} \in E} w_{ij} \ \leq W \tag{8}$$

$$C_d \leq C_s + (1 - \pi_s)(C_{max} - C_s) \qquad s \in \mathcal{S} \tag{9}$$

$$(K + \varepsilon)\pi_s \geq K + \varepsilon - \sum_{i \in V} w_i \bar{v}_i - \sum_{\{i,j\} \in E} w_{ij} \bar{v}_{ij} \qquad s \in \mathcal{S} \tag{10}$$

with π_s binary variables which activate the constraint for a solution $s \in \mathcal{S}$ if its attack budget is less than the maximum budget K and where ε is small enough to find the optimal solution. Therefore, each solution has an associated variable and constraint. We can adopt a Cutting Plane (CP) algorithmic approach, i.e. start with an empty (or very small) set $\mathcal{S}$ and solve the model iteratively by adding violated cuts. At each iteration, we add a binary variable π_s and we can branch on it : the branching node where $\pi_s = 1$ is closed immediatly since we cannot improve on the solution found while the node where $\pi_s = 0$ forces the model to make the newly found attacker's solution infeasible in the next iteration. The procedure converges when the model become infeasible.

3 Conclusions and perspectives

We will assess the above algorithmic framework for solving the protection version of several classic NIPs on a set of benchmark instances. The model has mixed lower level variables but only continuous upper level variables : since it is known that such models in general suffer from the non-existence of an optimal solution, we will prove its existence. Depending on the time and results available, we will also try to derive algorithms to compute relevant upper bounds, for example by using a restricted sample of lower level defender's solutions [2]. If possible, we will also try to derive the Σ_p^2-completeness of the application of our model to some classic NIPs in order to justify a CP algorithm to solve our bi-level approach.

Références

[1] J. C. Smith and Y. Song. A survey of network interdiction models and algorithms. *European Journal of Operational Research*, 1–15, 2019.

[2] L. Lozano and J. C. Smith. A Backward Sampling Framework for Interdiction Problems with Fortification. *INFORMS Journal on Computing*, 29(1) :123–139, 2017.