A Branch-and-Cut algorithm for the Proactive Countermeasures Selection Problem

A. Ridha Mahjoub[1], M.Yassine Naghmouchi[1,2], Nancy Perrot[2]

[1] Université Paris-Dauphine, PSL Research University, CNRS, LAMSADE, 75016, Paris, France
{mahjoub,mohamed-yassine.naghmouchi}@lamsade.dauphine.fr

[2] Orange Labs, France
nancy.perrot@orange.com

Key words : *Polyhedra, Branch-and-Cut, countermeasure selection.*

1 Introduction

We consider the Proactive Countermeasures Selection Problem, defined in [1]. An instance of the PCSP is given by a triplet (G, K, D). Here, G = (V, A) is a directed graph called the Risk Assessment Graph [2], where $V = S \cup T$, $S \cap T = \emptyset$. Each arc (i, j) of A has a weights $w_{ij} \in \mathbb{R}_+$. $K = \{(t, k) : k \in K_t, t \in T\}$ is a set of available countermeasures such that K_t is the set of countermeasures associated with t. The placement of k on t has a positive cost $c^t_k \in \mathbb{R}+$, and it increases the weight of t-ongoing arcs by a factor $\alpha^k_t \in \mathbb{R}_+$. $D = (d^s_t)_{s\in S,t\in T} \in \mathbb{R}_+$ is a positive security threshold vector. The PCSP consists in selecting a set of countermeasures, at minimal cost, such that for each $(s, t) \in S \times T$ the length of the $s - t$ shortest path is at least d^t_s. A bilevel model, as well as two compact and extended formulations, were introduced in [1] to solve the PCSP.

In this abstract, we give polyhedral results and a Branch-and-Cut algorithm developed for solving the extended formulation as well as some numerical results showing the efficiency of the algorithm.

2 Polyhedral investigation and Branch-and-Cut algorithm

We study the polytope associated to the path formulation. We characterize the dimension of the polytope by considering *the essential countermeasures set*, i.e., the countermeasures such that if we remove at least one of them, the PCSP does not have a solution. We then introduce several classes of valid inequalities, namely *the path covering inequalities*, *the countermeasures path inequalities* and *the essential-by subsets removing-countermeasure inequalities*. We discuss when these inequalities define facets. We also study the optimality conditions for the PCSP.

The polyhedral results are used within a Branch-and-Cut algorithm. We develop a preprocessing phase considering the essential countermeasures equations and the optimality condition inequalities. We devise separation routines for the basic and valid inequalities. In particular, we propose exact separation algorithms for both the basic inequalities and the countermeasures path inequalities.We also prove that the separation problems of path covering inequalities and essential-by subsets removing-countermeasure inequalities are NP-Complete requiring then the use of heuristics to separate them. In addition, we have provided a primal heuristic in order to reduce the size of the Branch-and-Cut tree and accelerate the resolution of the problem.

3 Numerical Results

We conduct extensive experimentations on random and realistic instances of the PCSP problem. In this section, we present some numerical tests of the compact formulation and the path formulation. The computational study shows the efficiency of the polyhedral results. The optimality condition inequalities and valid inequalities play an important role in the resolution of the problem as we can see in Table 1. The entries of the table are the following :

$\vert S\vert$	:	number of access points ;
$\vert T\vert$	:	number of asset-vulnerability nodes ;
$I\_\vert S\vert,\vert T\vert$	:	name of the instance ;
$\vert A\vert$	:	number of arcs ;
$\vert K\vert$	:	number of countermeasures ;
N	:	number of nodes in the Branch & Cut tree ;
$NOpt$	:	the number of instances solved to optimality / total number of instances ;
Gap	:	the relative error between the best upper bound (the optimal solution if the problem has been solved to optimality) and the lower bound obtained at the root,
CPU	:	total CPU time (in the format hh :mm :ss).

			Compact formulation				Branch and Cut			
Name	$\vert A\vert$	$\vert K\vert$	N	Gap	CPU	$NOpt$	N	Gap	CPU	$NOpt$
I_10,100	139.2	201	34	0.07	0 :1 : 5	5/5	43.6	0.06	0 :01 :2	5/5
I_20,200	500.2	667.2	70.8	0.09	1 :38 :2	5/5	153	0.08	0 :12 :2	5/5
I_30,300	1103.6	1100.8	116	0.16	4 :10 :5	1/5	177	0.08	0 :49 :3	5/5
I_40,400	1864.8	981.8	120.2	0.39	-	0/5	63.8	0.09	3 :49 :3	3/5
I_50,500	2676.4	1674.4	138	0.23	3 :41 :1	1/5	157.2	0.09	3 :22 :3	4/5
I_60,600	3276.2	1613	88.2	0.41	-	0/5	79.8	0.10	1 :52 :3	3/5
I_70,700	3988.4	1876.2	57.8	0.49	-	0/5	104	0.09	2 :57 :6	3/5
I_80,800	4694	2854.4	68.2	0.51	-	0/5	92.6	0.13	3 :52 :7	2/5
I_90,900	6204.6	2765.4	54	0.45	-	0/5	79	0.12	3 :19 :5	3/5
I_100,1000	10866.4	3596	-	-	-	0/5	107	0.19	4 :42 :3	1/5
I_110,1100	14444.6	4987.2	-	-	-	0/5	103	0.23	4 :49 :5	1/5
I_120,1200	19546.6	5503.6	-	-	-	0/5	69	0.37	-	0/5

TAB. 1 – Comparison with the compact formulation for a family of random instances

Références

[1] A. R. MAHJOUB, M. Y. NAGHMOUCHI, and N. PERROT. *A Bilevel Programming Model for Proactive Countermeasure Selection in Complex ICT Systems.* Electronic Notes in Discrete Mathematics, 2018, vol. 64, p. 295-304.

[2] M. Y. NAGHMOUCHI, N. PERROT, N. Kheir,A. R. MAHJOUB, and J. P. Wary, *A new risk assessment framework using graph theory for complex ICT systems.* Managing Insider Security Threats (pp. 97-100). CCS, ACM.