

Table of Contents

Introduction	1
1. Coordinated defender strategies for border patrols	4
1.1. Introduction	4
1.2. Problem Formulation and Notation	6
1.2.1. Stackelberg Security games	6
1.2.2. Resource Combination in a SSG	9
1.3. Decomposition Model	12
1.3.1. (COMB) is an equivalent SSG formulation	13
1.3.2. Cutting Odd-Set Constraints	16
1.3.3. Recovering an implementable strategy	18
1.4. Case Study: Carabineros de Chile	22
1.4.1. Payoff estimation	24
1.4.2. Building software for Carabineros	26
1.4.3. Robustness of our approach	27
1.5. Computational Experiments	28
1.5.1. Performance of (COMB)	28
1.5.2. Performance of the alternative sampling method	30
1.6. Conclusions and future work	31
2. Abstractions to handle large scale models.	32
2.1. Layer Generating Algorithm	33
2.2. Multi-Layer Generating Algorithm	36
2.3. Conclusions	37
3. Solving Stackelberg equilibrium in stochastic games	39
3.1. Introduction	39
3.2. Literature Review	41
3.2.1. General results	42
3.2.2. Math programming to solve Stochastic Games	43
3.3. Motivational Example	46
3.4. Numerical Example 1	49
3.5. Special case: Myopic Follower Strategies (MFS)	53
3.5.1. Stackelberg operator and Value function iteration	54
3.5.2. Policy Iteration	58
3.6. General Case	60

3.6.1.	Definition of the Stackelberg Operator in the general case	61
3.6.2.	Numerical Example 2	62
3.6.3.	Value Function and Policy Function for the general case	65
3.6.4.	What does it fail?	68
3.7.	Mathematical Programming approach	70
3.8.	Computational Experiments	73
3.8.1.	Performance of algorithms: (MFS) case.	73
3.8.2.	Stackelberg Security Games	75
3.8.3.	General Case	76
3.8.4.	Sensitivity analysis in β	78
3.9.	Conclusions and Future work	80
4.	Stackelberg games of water extraction	81
4.1.	Introduction	81
4.2.	A first study	82
4.3.	Preliminary results	83
4.4.	Robust approach	84
4.5.	Conclusions and future (current) work	86
	Conclusion	86
	Bibliography	89

List of Tables

1.1. Tabular representation for the feasible schedule in Figure 1.8.	23
3.1. Transition matrix and payoffs for each player.	49
3.2. Bi-matrix games in $\tau = 2$	50
3.3. Bi- matrix games at stage 1.	51
3.4. Transition matrix and payoffs for each player in the numerical example 2. . .	62
3.5. Bi- matrix games at stage 14 for numerical example 2.	62
3.6. Bi- matrix games at stage 15 for numerical example 2.	62
3.7. Iterations 14 and 15 for numerical example 2.	63
3.8. Transformation of the matrix in numerical example 2 to a Leader-controller case.	65
3.9. Solution times and iterations performed by algorithms in myopic instances. .	74
3.10. Resolution time and iterations performed by algorithms in Leader Controller Instances.	75
3.11. Resolution time and iterations performed by algorithms in General instances, for instances where T is not detected to be not contractive.	77
3.12. Transition matrix and payoffs for each player in the Numerical Example 3. .	79
4.1. Parameters in our preliminar study.	83

List of Figures

1.1.	Variables $\mathbf{z} \in [0, 1]^{ E }$ that satisfy (1.26) and (1.27) but violate (1.28) for $m = 2$.	12
1.2.	Variables $\mathbf{z} \in [0, 1]^{ E }$ and $\mathbf{c} \in [0, 1]^{ J }$ that do not satisfy (1.29) and (1.30) with $m = 2$.	13
1.3.	Construction to identify implementable mixed strategy \mathbf{x} given vectors $\mathbf{z}, \mathbf{c}, \mathbf{g}$ feasible for (COMB).	16
1.4.	Representation of the algorithm which detects the Odd min cut set. The dashed edges shows the min cut of this graph.	18
1.5.	Example of the warm start algorithm.	21
1.6.	A Carabinero conducts surveillance.	23
1.7.	Harsh border landscape.	23
1.8.	Feasible schedule for a week. Elaborated by Carlos Casorrán.	23
1.9.	Three crime flow networks, one per type of crime. Elaborated by Carlos Casorrán.	25
1.10.	Robustness of the solution method to variations in the parameters λ and h .	28
1.11.	Performance for $m = 2$, $m = 3$ and $m = 10$: Solving time (s.) vs. number of nodes in graph.	29
1.12.	Kullback-Leibler distance between z^* and \hat{z} over instances of different size.	30
2.1.	Layers of targets generated by the Greedy Multi-Layer Algorithm algorithm.	37
3.1.	Expected reward for each player and SSE in the game played in state s_1 at stage $\tau = 2$.	51
3.2.	Expected reward for each player and SSE in the game played in state s_2 at stage $\tau = 2$.	51
3.3.	Expected reward for each player and SSE in the game played in state s_1 at stage 1.	52
3.4.	Expected reward for each player and SSE in the game played in state s_2 at stage 1.	52
3.5.	Value Function vs time in numerical example 1.	53
3.6.	Value Function vs time in numerical example 2.	64
3.7.	Value Function vs time in numerical example 2 in the myopic case ($\beta_B = 0$).	64
3.8.	Value Function vs time in numerical example 2 in Leader-controller case.	65
3.9.	Convergence of v_A starting from different starting values (v_A^0, v_B^0) .	69
3.10.	Convergence of v_B starting from different starting values (v_A^0, v_B^0) .	69
3.11.	Performance of VI and PI in myopic instances.	74
3.12.	Performance of VI and PI in Leader Controller instances.	75

3.13. Performance of value function iteration and policy iteration in security games instances.	76
3.14. Percentage of instances where the Algorithm 12 returns undefined.	77
3.15. Performance of value function iteration and policy iteration in General random instances generated, for instances where it is not detected that the operator T is not contractive.	78
3.16. Sensitivity analysis in solution times for VI algorithm when β_A changes.	78
3.17. Sensitivity analysis in solution times for PI algorithm when β_A changes.	79
3.18. Impact of β_A and β_B on the convergence in VI in Numerical Example 3.	79
4.1. Groundwater level and consumption for $\beta_A \in \{0.2, 0.7, 0.9\}$	83
4.2. Policies for $\beta_A \in \{0,2, 0,7, 0,9\}$	84