Regstar: Efficient Strategy Synthesis for Adversarial Patrolling Games

David Klaška, Antonín Kučera, Vít Musil, and Vojtěch Řehák

Faculty of Informatics, Masaryk University, Brno, Czech Republic

Abstract

- An efficient strategy synthesis method applicable to adversarial patrolling problems with **long distances** among targets and possibly **imperfect intrusion detection**.
- Experimentally evaluated that it is applicable to **real-world patrolling** graphs of reasonable sizes.
- Proof that **regular** (finite-memory) Defender's strategies are **arbitrarily close** to the power of general strategies.

Patrolling Games

Attacker

- aims to proceed attack
- attack takes time
- knows patroller(s) position and strategy

Targets

parameters

- level of importance
- attack length
- visibility of attack

Patroller

- visits targets
- aims to disturb attacks
- knows nothing about attacker

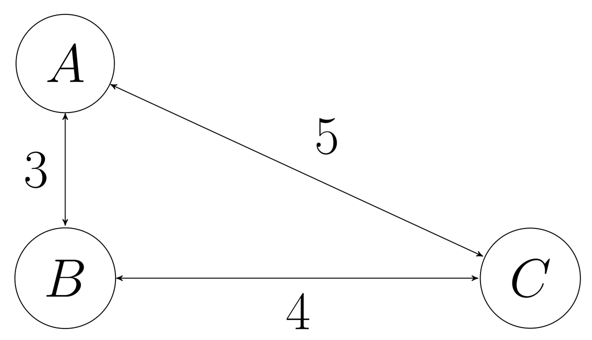
The goal is to find the best (randomized) strategy for the patroller.

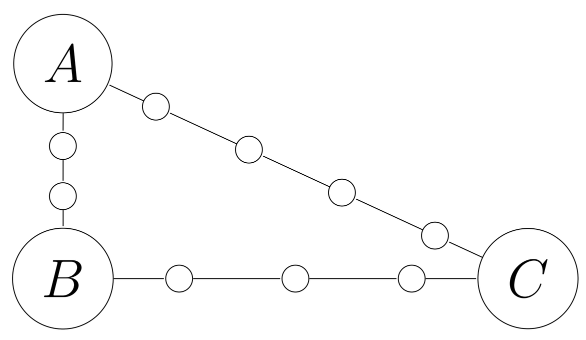
Other Target Parameters

- vertices (targets) may have variable attack durations
- vertices (targets) may have variable importances
- **edges (distances) may have variable lengths**

Variable Length of Edges

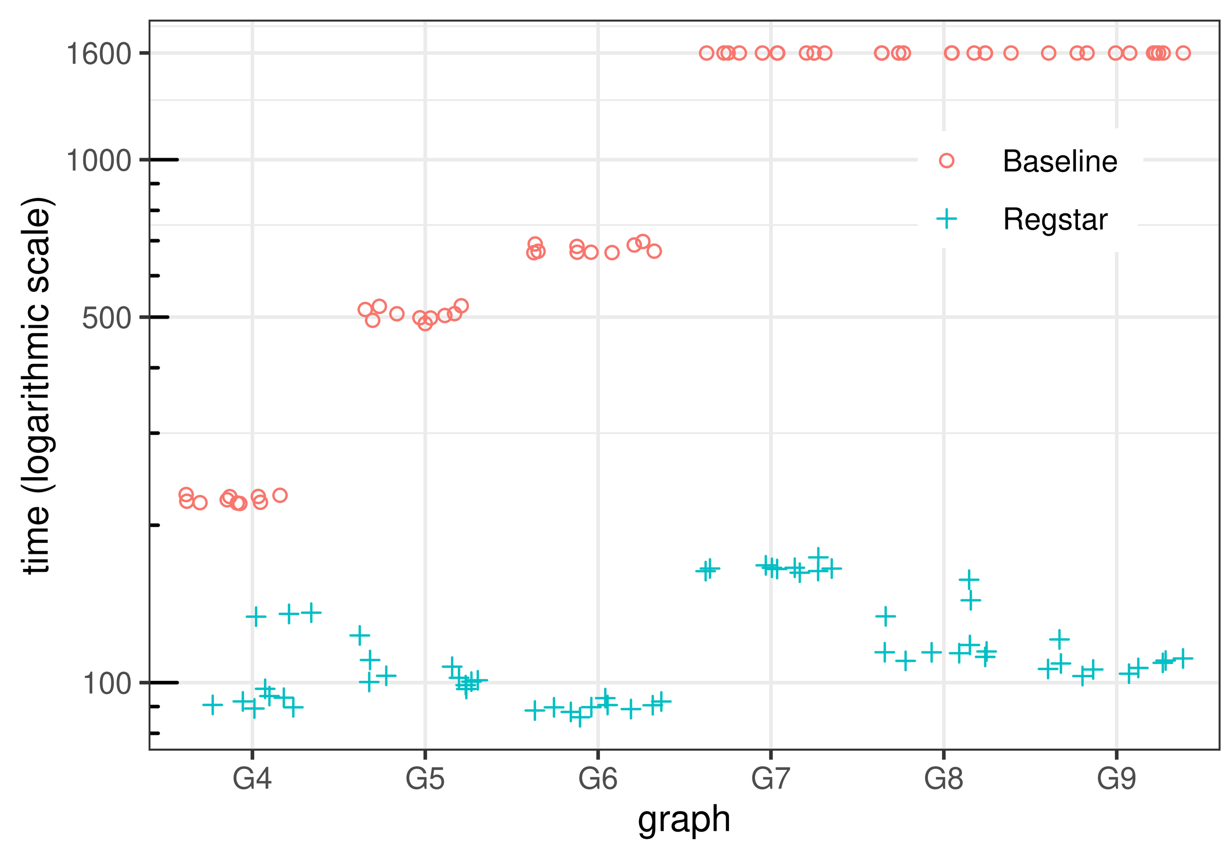
Contrary to the previous approach [1] using auxiliary vertices,





we come up with Regstar:

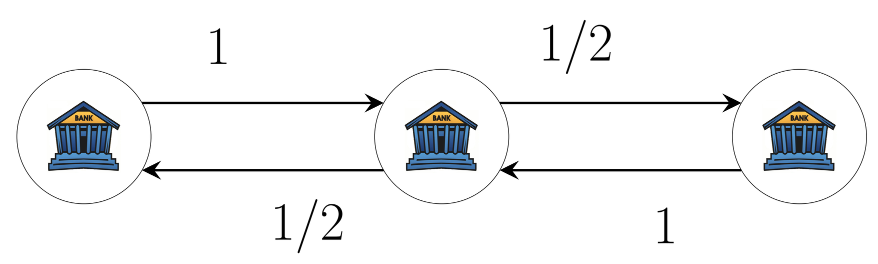
- based on **gradient descent**
- can deal with **imperfect detection**
- partly similar to Dijkstra's algorithm, but more **involved**
- i.e., **fast algorithm** for strategy synthesis



Types of Strategies

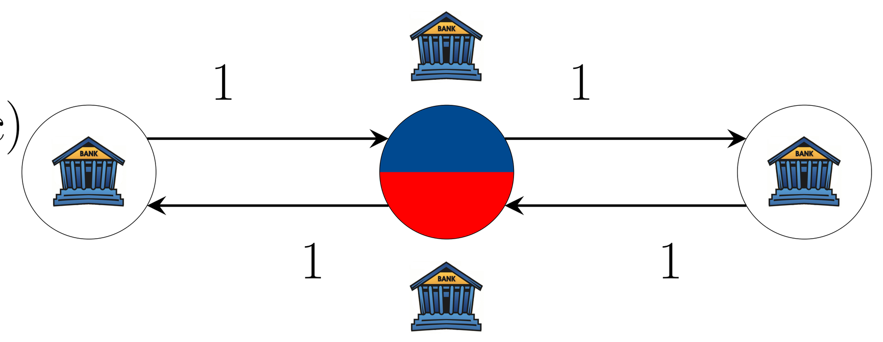
- type of general strategy is $\mathbf{V}^* \to \Delta(V)$
- not finitely representable
- Markovian (memoryless, positional) strategy: $\mathbf{V} \to \Delta(V)$
- k-Markovian strategy: $\mathbf{V}^\mathbf{k} \to \Delta(V)$ for a fixed $k \in \mathbb{N}$
- regular strategy: $\mathbf{V} \times \mathbf{M} \to \Delta(V \times M)$ for a fixed **finite set** M

(stochastic) Markovian



75 %

(deterministic) regular



100 %

Which type of strategies is sufficient?

	ε-optimal	optimal
Markovian	NO	NO
k-Markovian	NO?	NO
regular	**YES**	YES?

Real-word Example: Montreal ATMs

On the graph of Montreal ATMs, we study the impact of the **memory size** on the quality of the synthesized strategy and the execution time of Regstar. $m = 1$ stands for positional strategy.

m	Val_{best}	Val_{avg}	# of iterations	time [s]
1	64	57 ± 3	280 ± 30	5 ± 1
2	75	70 ± 2	684 ± 41	79 ± 8
3	80	77 ± 2	1045 ± 60	360 ± 58
4	81	79 ± 1	1346 ± 75	1250 ± 196

Acknowledgments

Research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-21-1-0189.
This work was supported from Operational Programme Research, Development and Education - Project Postdoc2MUNI (No. CZ.02.2.69/0.0/0.0/18_053/0016952).

References

[1] Klaška et al. Automatic synthesis of efficient regular strategies in adversarial patrolling games. *AAMAS 2018.*