Faculty of Mathematics and Information Science
Warsaw University of Technology

Evolution of Strategies in Sequential Security Games

Adam Żychowski, Jacek Mańdziuk

{a.zychowski, j.mandziuk}@mini.pw.edu.pl

AAMAS 2021

PROBLEM DEFINITION

Sequential Security Games with Stackelberg Equilibrium

Asymmetric two player game with imperfect information
Defender (D) commits to a certain strategy first, then Attacker (A) chooses strategy

Goal: maximize Defender's payoff

$$BR(\pi^D) = \arg\max_{\pi^A \in \Pi^A} U^A(\pi^D, \pi^A)$$

$$\arg\max_{\pi^D \in \Pi^D} U^A(\pi^D, BR(\pi^D))$$

π^D, π^A - Defender's/Attacker's strategy

U^D, U^A - Defender's/Attacker's payoff

MOTIVATION AND CONTRIBUTION

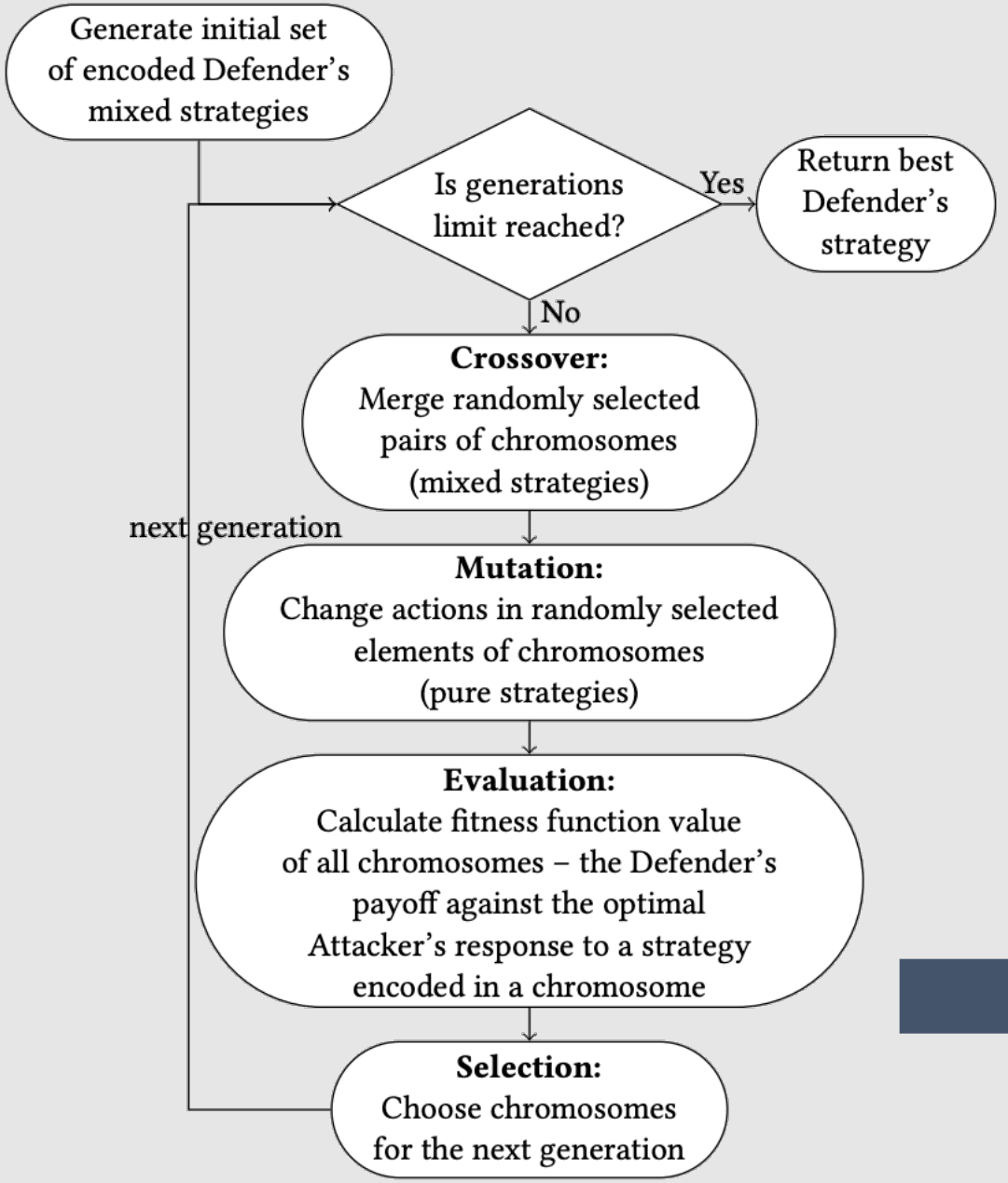
State-of-the-art solutions base on Mixed Integer Linear Programming (usually inefficient and domain dependent)

Finding Stackelberg Equilibrium is a kind of optimization problem – **evolutionary algorithms** is one of the most promising optimization methods

Creating a **general** Stackelberg Games solution framework based on evolutionary algorithms, easily **adaptable to various types of games**

An **anytime** approximation method for **time-critical** applications

ALGORITHM OVERVIEW



CHROMOSOME REPRESENTATION

Each chromosome represents Defender's mixed strategy – a set of pure strategies with their probabilities:

$$CH_q = \{(\pi_1^q, p_1^q), \ldots, (\pi_{l_q}^q, p_{l_q}^q)\}, \quad \sum_{i=1}^{l_q} p_{l_q}^q = 1$$

π_l^q - pure strategy (e.g. list of Defender's actions in consecutive time steps)

p_l^q - probability of the strategy π_l^q

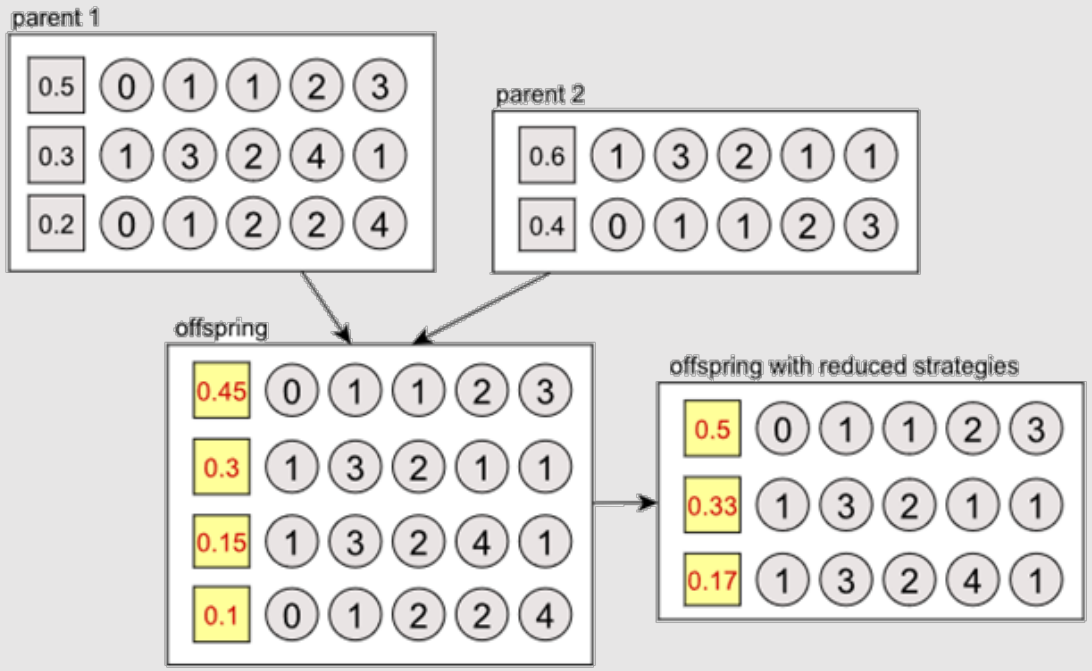
l_q - length of chromosome CH_q (the numer of pure strategies included in the mixed strategy represented by that chromosome)

Initial population contains random pure strategies (single strategy with probability equal to 1).

CROSSOVER

Crossover operation combines two chromosomes by merging their sets of pure strategies and halving their probabilities

After crossover each pure strategy (π_l^q) may be deleted with probability $1 - \frac{1}{2}p_j^q$



MUTATION

Mutation randomly changes one Defender's action in one of pure strategies starting from a randomly chosen time step to the last one

Each chromosome is mutated with *mutation rate* probability

Exploration of new areas of the search space

SELECTION

Fitness function - Defender's payoff in case of playing a mixed strategy encoded in the chromosome

Binary tournament - two chromosomes are randomly chosen and the one with a higher fitness value is promoted to the next generation with probability $p_s > 0.5$, otherwise the lower-fitted one is promoted

Some number of *elite* chromosomes (with the greatest fitness function value) ➔ unconditionally promoted to the next generation population

EXPERIMENTS

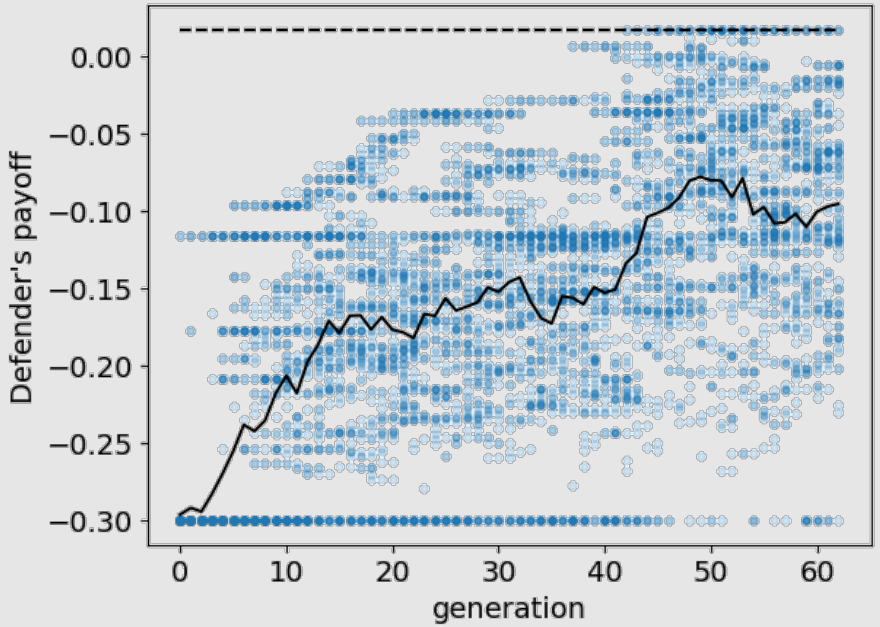
EASG parameters tuned based on a separate set of 50 games:

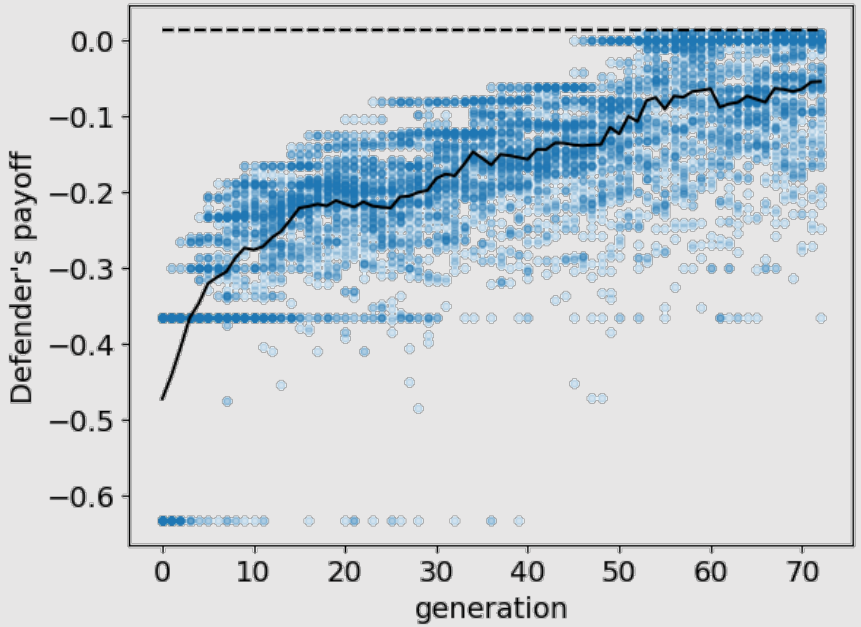
population size: 100, generations: 1000, mutation rate: 0.5, crossover rate: 0.8, selection pressure: 0.9, elite: 2

Experimental evaluation:

- 3 sets of multi-step games with variable characteristic: Warehouse Games, Secucrity Games, and FlipIt Games
- test were perfomed based on 300 game instances
- 4 algorithm aspects: convergence, results quality, stability, and time scalability

PERFORMANCE



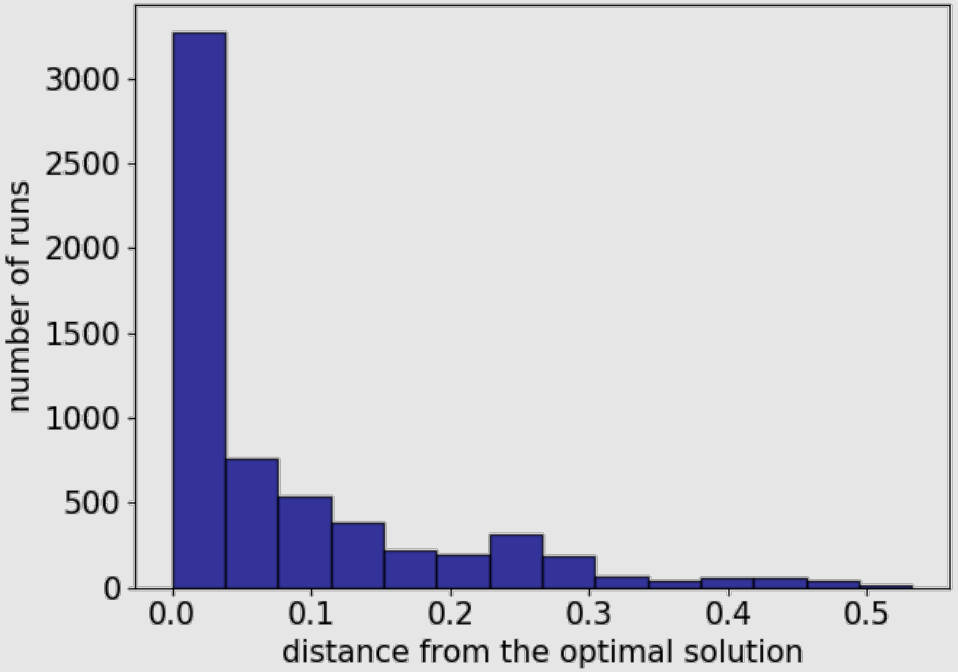


Mean Defender's payoff increases in time → the entire population moves towards the areas with higher payoff

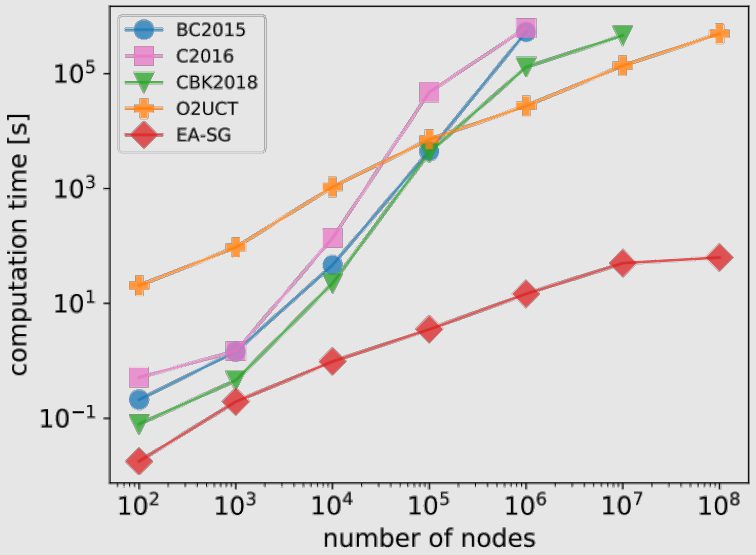
Low-payoff individuals exist in all generations → exploration of new strategies

RESULTS QUALITY

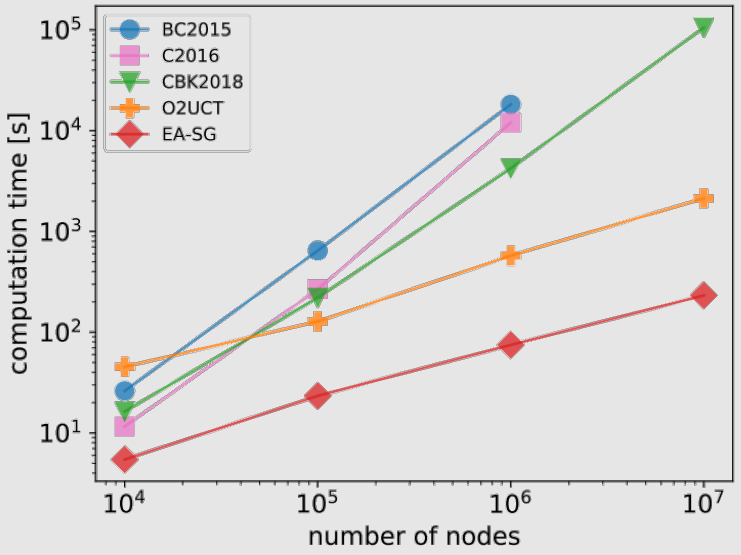
Games type	Fraction of games with optimal solution found	Mean difference between the optimal and EASG	The highest difference between the optimal and EASG
Warehouse Games	72%	0.0013	0.0127
Search Games	47%	0.0253	0.0955
FlipIt Games	73%	0.0087	0.0321



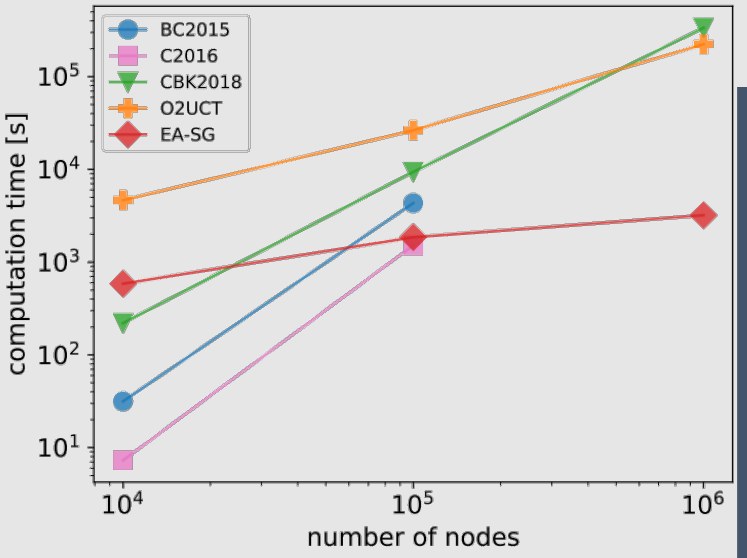
TIME SCALABILITY



Warehouse Games



Search Games



FlipIt Games

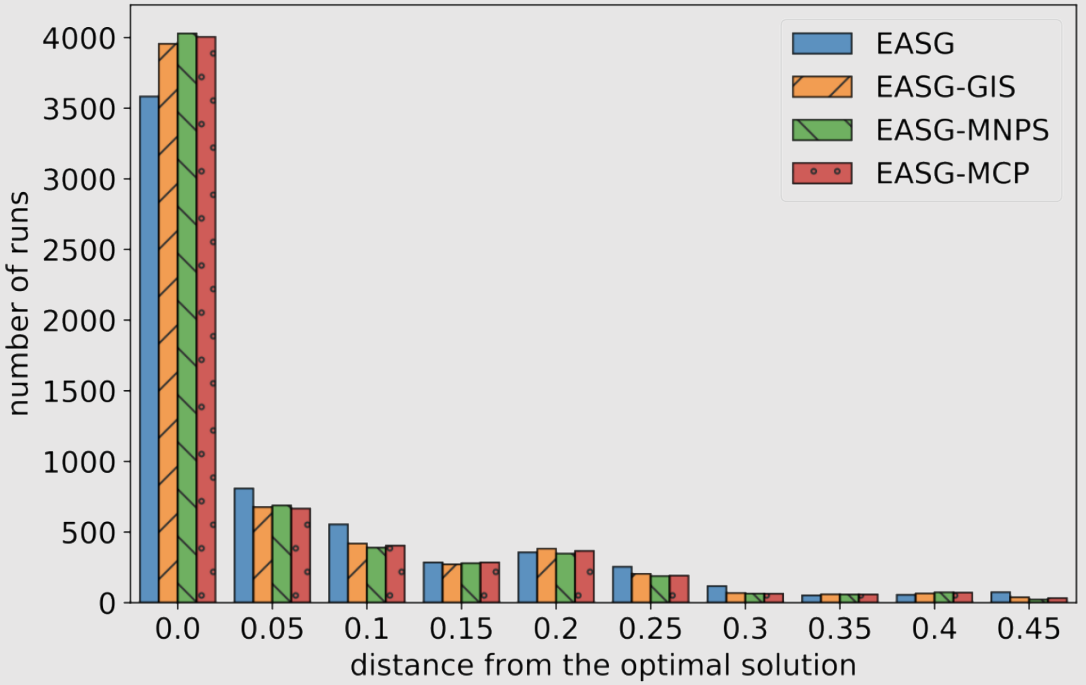
Time performance strongly depends on selected steering parameters – the possibility of establishing the expected balance between computation time and quality of results

The highest time efficiency among the tested methods

ABLATION STUDY

EASG formulation is intentionally generic so as to make the method widely applicable to various types of SGs.

The method can be tailored to SG domain which improves obtined results, e.g. adding local memetic optimization increases the rate of optimal solutions from 47% to 64% for Search Games.



SUMMARY

Evolutionary method which can be **easily adapted to various types of Security Games**

Efficient approximation method with **high stability** and **good results quality**

Capable of **solving larger and more complex sequential Security Games** than state-of-the-art methods

Iteration-based construction - **well suited for time-critical applications** (*anytime* method)