A MEMETIC APPROACH FOR SEQUENTIAL SECURITY GAMES ON A PLANE WITH MOVING TARGETS

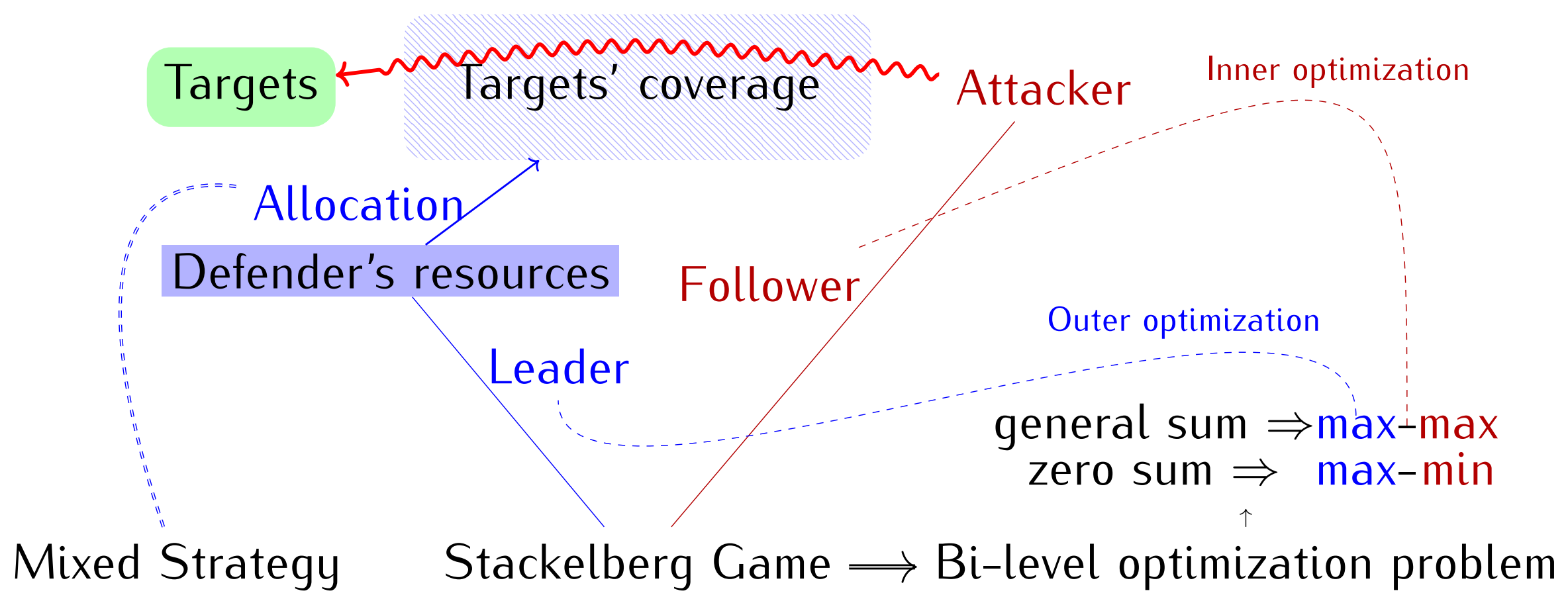
Jan Karwowski[1], Jacek Mańdziuk[1], Adam Żychowski[1], Filip Grajek[1], and Bo An[2]

[1]Warsaw University of Technology, Faculty of Mathematics and Information Science, POLAND
[2]Nanyang Technlogical University, School of Computer Science and Engineering, SINGAPORE

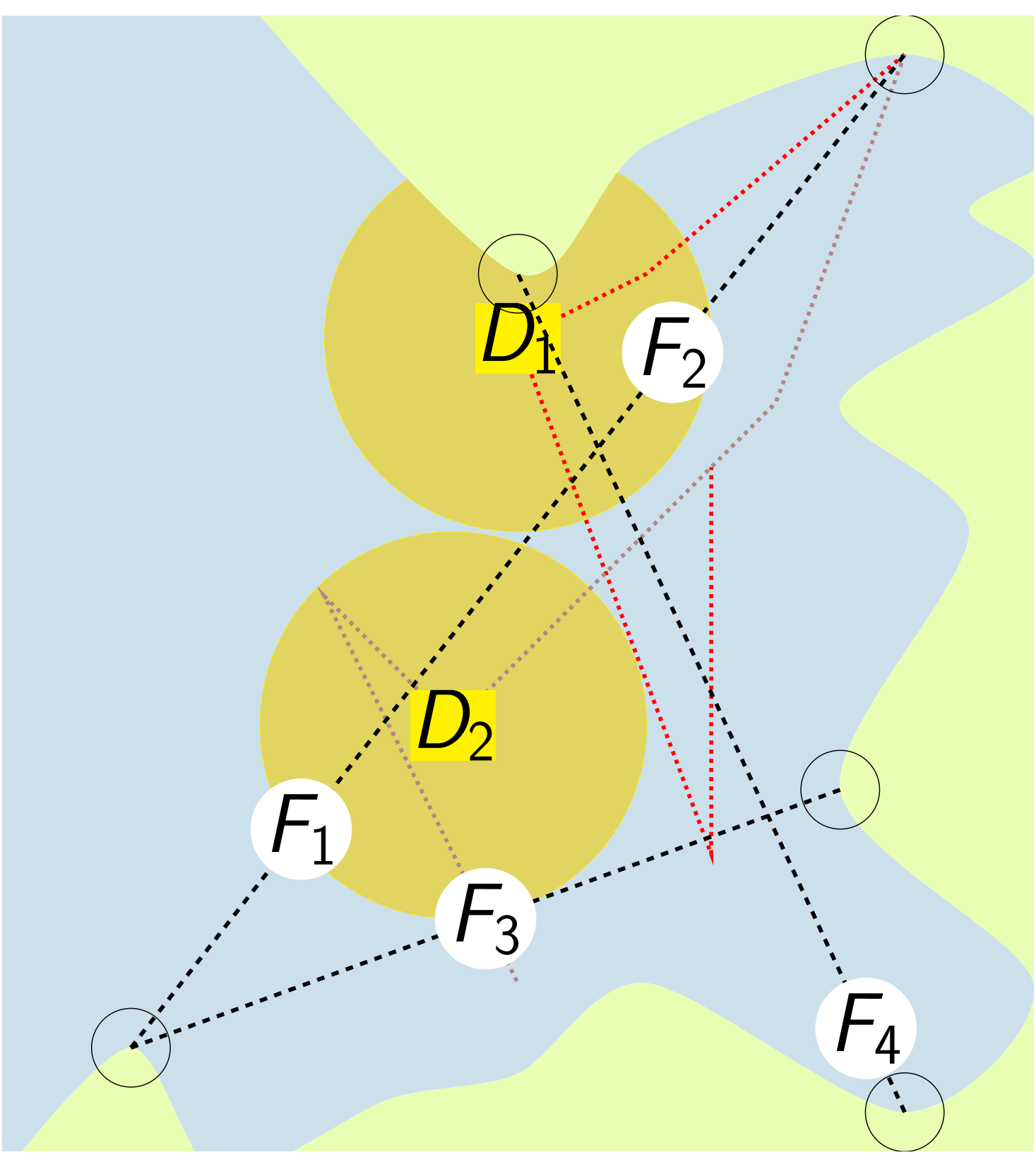
Thirty-Third AAAI Conference on Artificial Intelligence, January 27 — February 1, 2019, Honolulu, Hawaii, USA

★ Multi-Act Game • Good scalability • Repeatability • Anytime method ★

Security Games


Targets
Targets' coverage
Attacker
Inner optimization
Allocation
Defender's resources
Follower
Outer optimization
Leader
general sum ⇒ max–max
zero sum ⇒ max–min
Mixed Strategy
Stackelberg Game ⟹ Bi-level optimization problem

Game (with moving targets)



Time: game lasts several rounds

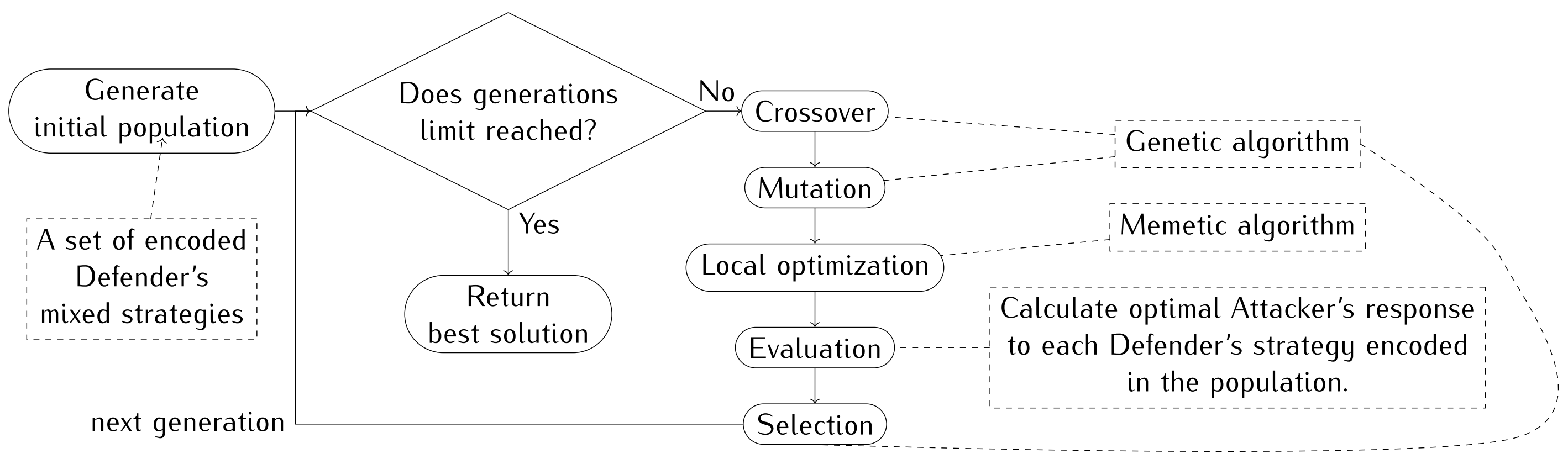
Ports: fixed locations

Ferries: single route, fixed schedule

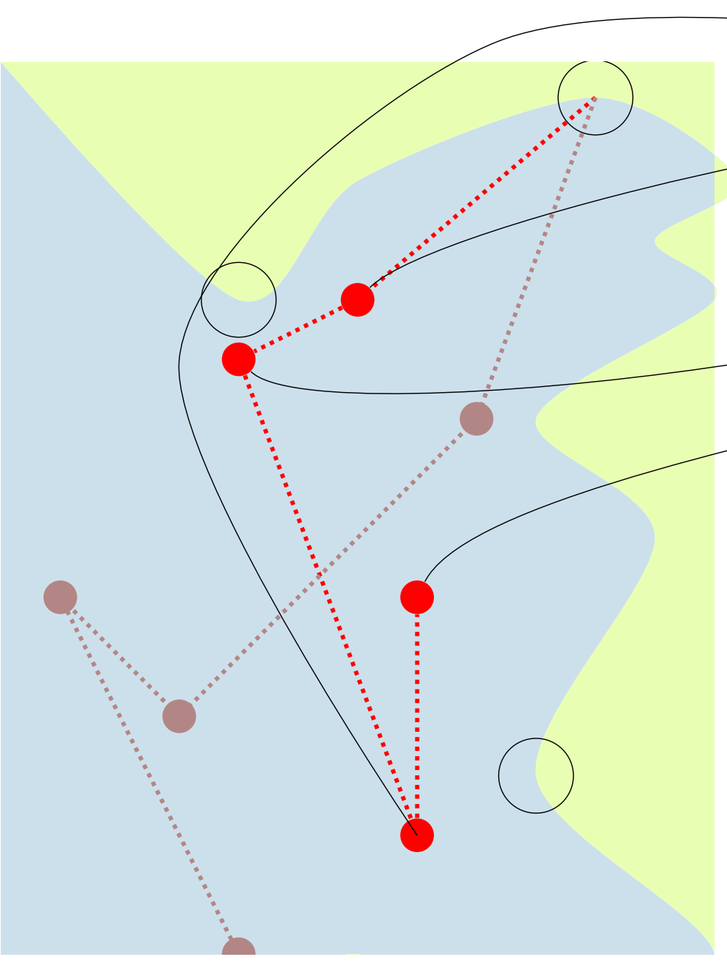
Defenders: protect radial area around them

Attacker: tries to attack a ferry, attack lasts several rounds

Memetic algorithm


Generate initial population
A set of encoded Defender's mixed strategies
Does generations limit reached?
No
Yes
Return best solution
Crossover
Mutation
Local optimization
Evaluation
Selection
next generation
Genetic algorithm
Memetic algorithm
Calculate optimal Attacker's response to each Defender's strategy encoded in the population.

Chromosome (Defender's strategy encoding)

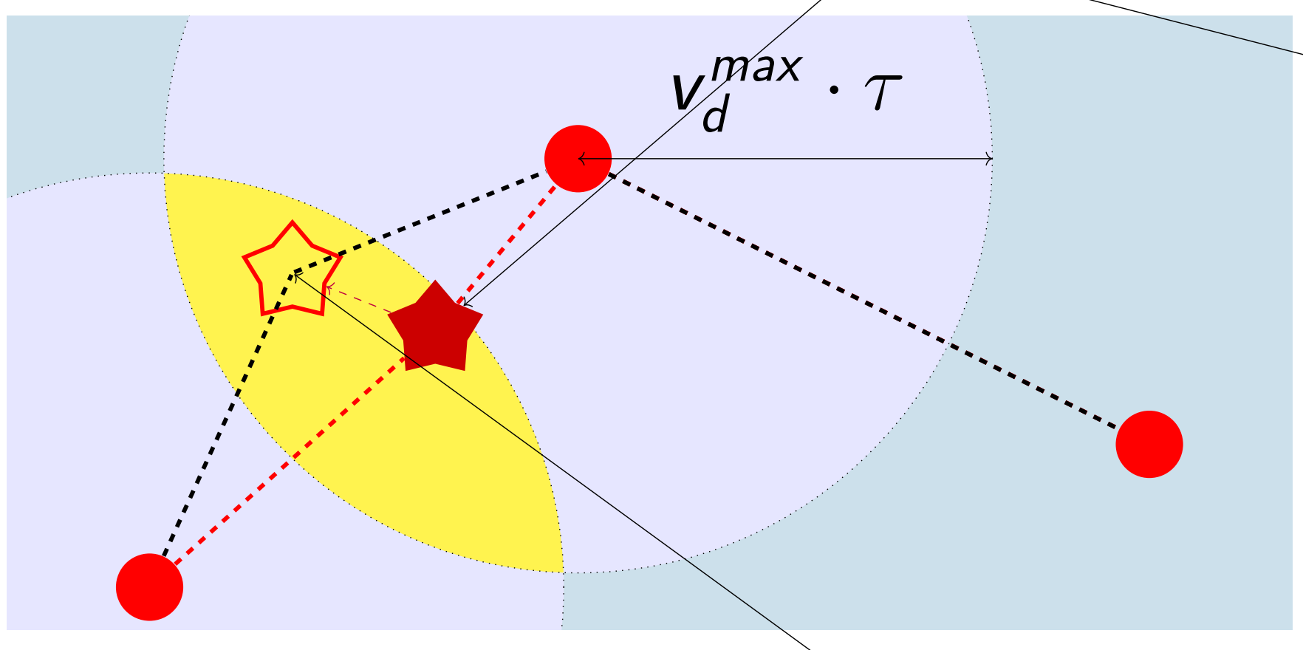


- $SS = [(D_1(t_1), D_1(t_2), \cdots, D_1(t_m)), (D_2(t_1), \cdots, D_2(t_m)), \ldots]$
- chromosomes may contain different numbers of SSs
- $CH_q = \{(\underbrace{SS_1^q}_{\text{pure strategy}}, \underbrace{p_1^q}_{\text{probability}}), \ldots, (SS_{l_q}^q, p_{l_q}^q)\}$, $\sum_{i=1}^{l_q} p_i^q = 1$

pure strategy   probability

Mutation

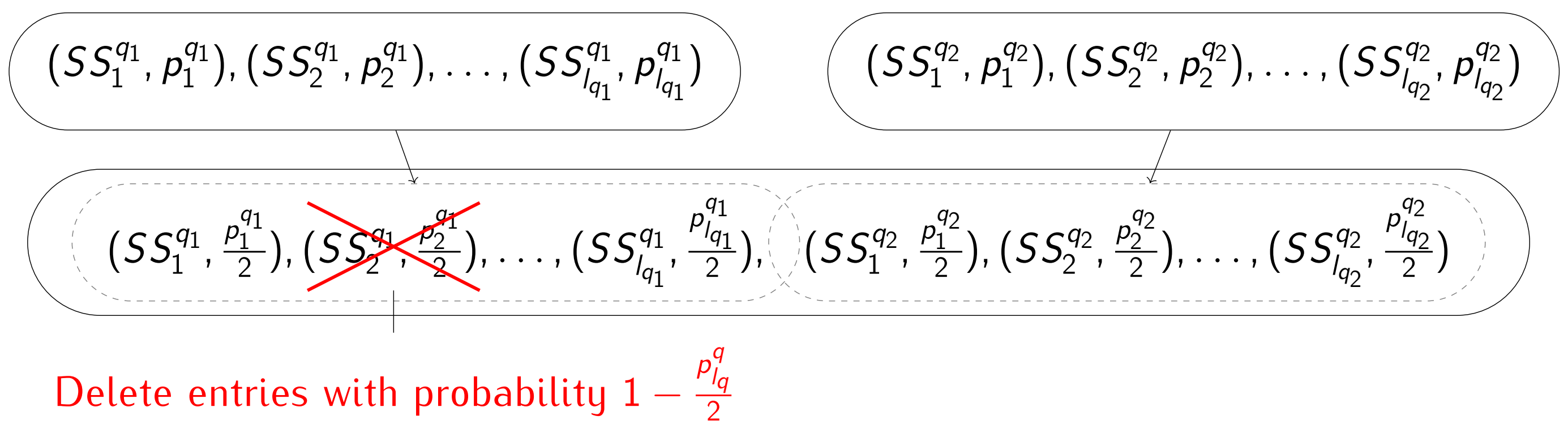
$CH = [\ldots, (p_i, (D_1(t_1), \boxed{D_1(t_2),} D_1(t_3), D_1(t_4)), \ldots), \ldots]$



Randomly chosen entry

$CH' = [\ldots, (p_i, (D_1(t_1), D_1'(t_2), D_1(t_3), D_1(t_4)), \ldots), \ldots]$

Crossover

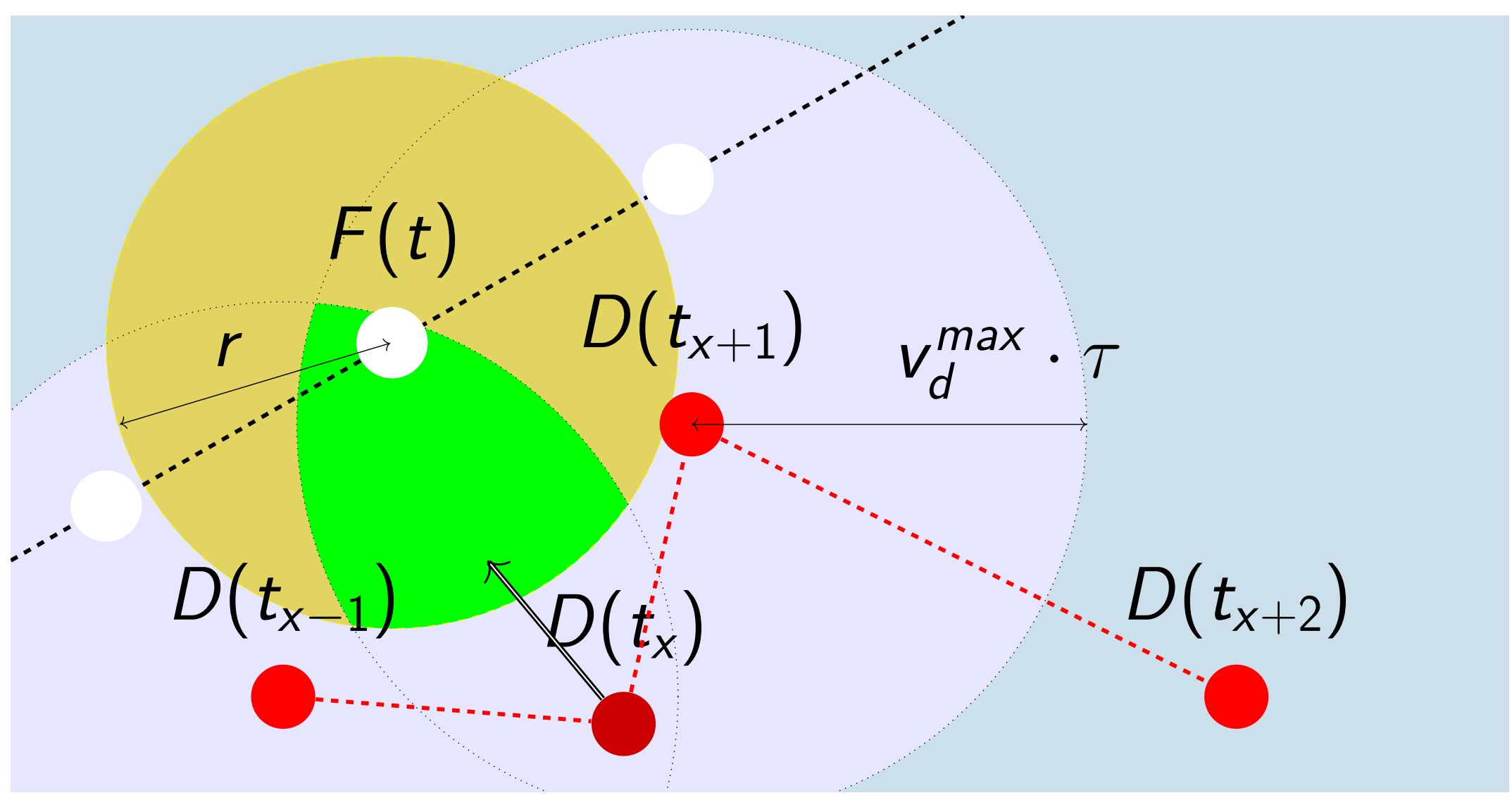


$(SS_1^{q_1}, p_1^{q_1}), (SS_2^{q_1}, p_2^{q_1}), \ldots, (SS_{l_{q_1}}^{q_1}, p_{l_{q_1}}^{q_1})$ $(SS_1^{q_2}, p_1^{q_2}), (SS_2^{q_2}, p_2^{q_2}), \ldots, (SS_{l_{q_2}}^{q_2}, p_{l_{q_2}}^{q_2})$

$(SS_1^{q_1}, \frac{p_1^{q_1}}{2}), \cancel{(SS_2^{q_1}, \frac{p_2^{q_1}}{2})}, \ldots, (SS_{l_{q_1}}^{q_1}, \frac{p_{l_{q_1}}^{q_1}}{2}), (SS_1^{q_2}, \frac{p_1^{q_2}}{2}), (SS_2^{q_2}, \frac{p_2^{q_2}}{2}), \ldots, (SS_{l_{q_2}}^{q_2}, \frac{p_{l_{q_2}}^{q_2}}{2})$

Delete entries with probability $1 - \frac{p_{l_q}^q}{2}$

Local optimization



Evaluation

$SS_q \longrightarrow$

$\mathbb{E}U_D^q$	payoff against optimal attacker's response
N^q	number of optimal attacker's pure strategies

$\longrightarrow$ Rank by $\mathbb{E}U_D^q$, break ties by N^q

Experiments

6150 random games

MA Parameters:

parameter	value
population size	2000
# generations	2000
mutation rate	0.2
mutation repeats	10
crossover rate	0.9
selection pressure	0.9
# elite chromosomes	2

Results baseline: SMOS-based program

$$\max \sum_f U^d(f, f)$$

$$0 \leq a - \sum_{g \in F} U^a(g, f) \leq (1 - q_f) \cdot M$$

$$\sum_{f \in F} q_f = 1,$$

$$c(i, t_k, f) = \sum_{j \in N(i)} fl_{((i,t_k),(j,t_{k+1}),f)} \quad \forall i \in Z, k \in 0, \ldots \tau - 1, f \in F$$

$$c(i, t_k, f) = \sum_{j \in N(i)} fl_{((j,t_{k-1}),(i,t_k),f)} \quad \forall i \in Z, k \in 1, \ldots \tau, f \in F$$

$$q_f \cdot m = \sum_{i \in Z} c(i, t_k, f) \quad k \in \{0, \tau\}$$

$$q_f \cdot DS(i) = c(i, 0, f) \quad \forall i \in S, f \in F$$

$$1 - m_1(g, f) M_2 \leq dpp(g, f) \leq 1$$

$$\sum_{(i,t) \in R(f)} c(i, t, g) - m_2(g, f) M_2 \leq dpp(g, f) \leq \sum_{(i,t) \in R(f)} c(i, t, g)$$

$$m_1(g, f) + m_2(g, f) = 1, \quad \forall f, g \in F$$

$$q_g - m_3(g, f) M_2 \leq dpp_{neg}(g, f) \leq q_g$$

$$1 - dpp(g, f) - m_3(g, f) M_2 \leq dpp_{neg}(g, f) \leq 1 - dpp(g, f)$$

$$m_3(g, f) + m_4(g, f) = 1, \quad \forall f, g \in F$$

$$\forall i, g, f\, m_i(g, f) \in \{0, 1\} \quad \forall f\, q_f \in \{0, 1\}$$

$$fl_{((i,t_k),(j,t_{k+1}),f)} \in \mathbb{R}, dpp(g, f) \in \mathbb{R}$$

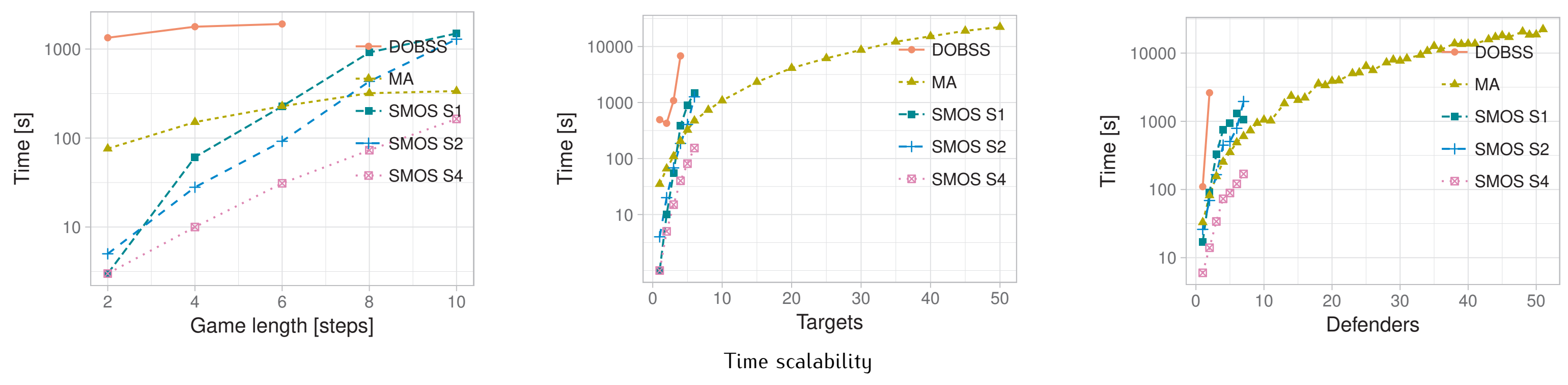
$$c(i, t_k, f) \in \mathbb{R} \quad \forall g, f\, dpp_{neg}(g, f) \in \mathbb{R}$$

$$U^a(g, f) = dpp_{neg}(g, f) U^{a+}(f) + dpp(g, f) U^{a-}(f)$$

$$U^d(g, f) = dpp_{neg}(g, f) U^{d-}(f) + dpp(g, f) U^{d+}(f)$$

Xinrun Wang et al. "Catching Captain Jack: Efficient time and space dependent patrols to combat oil-siphoning in international waters". In: *AAAI Conference on Artificial Intelligence*. 2018, pp. 208–215

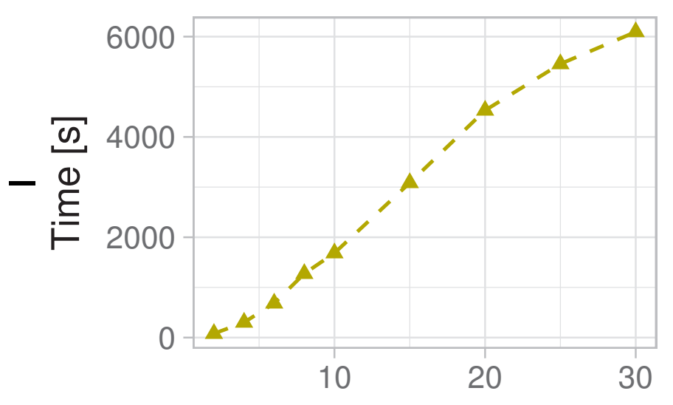
Results

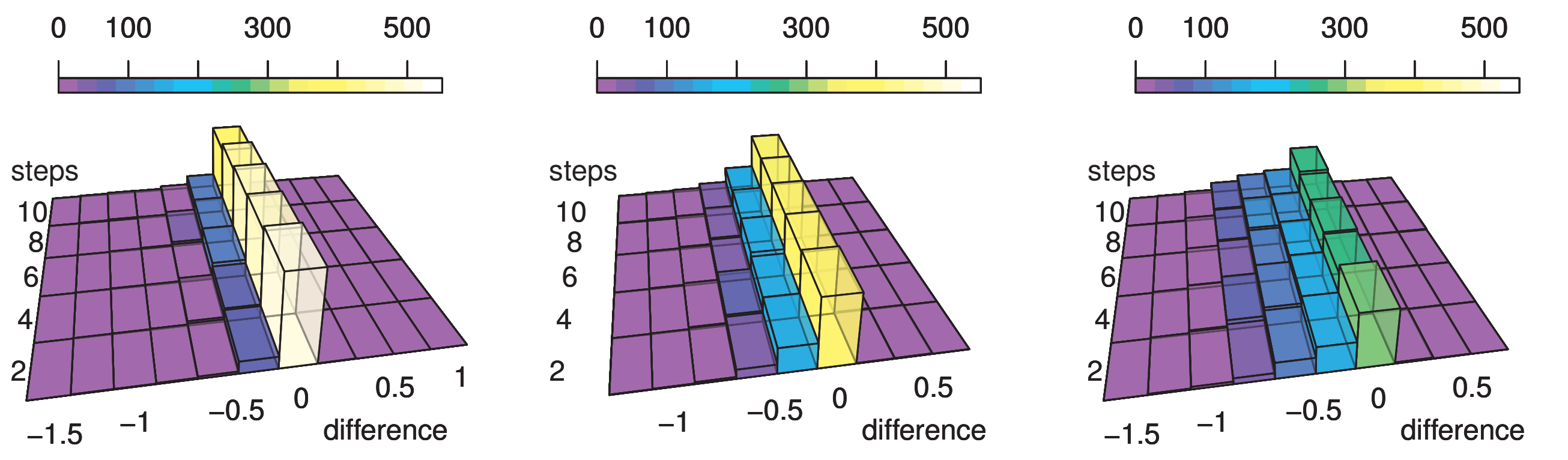


Time scalability

Close to linear scalability

Large instances:





Differences between Defender's utility values obtained with MA and SMOS for various settings of game abstraction