



PP-RAI'2024



# Hall of Fame in Coevolutionary Algorithm for Stackelberg Security Games

Adam Żychowski<sup>1</sup>, Jacek Mańdziuk<sup>1</sup>, Andrew Perrault<sup>2</sup>

<sup>1</sup>Faculty of Mathematics and Information Science, Warsaw University of Technology

<sup>2</sup>Department of Computer Science and Engineering, The Ohio State University

April 2024

# Security Games

- Two asymmetrical players: **Defender and Attacker**
- Each game is composed of  $m$  time steps.
- Each player chooses an action to be performed in each time step.
- A player's pure strategy  $\sigma_P$  ( $P \in \{D, A\}$ ) is a sequence of their actions in consecutive time steps:  $\sigma_P = (a_1, a_2, \dots, a_m)$ .
- **Defender commits to his/her strategy first.**
- **Attacker, knowing the Defender's strategy, chooses his/her strategy.**
- **Defender always commits to a mixed strategy.**

# Stackelberg equilibrium

**Stackelberg equilibrium:** a pair of players' strategies, for which strategy change by any of players leads to his/her result deterioration.

$$(\pi_D^*, R(\pi_D^*)) \in \Pi_D \times \Pi_A$$

$$\pi_D^* = \operatorname{argmax}_{\pi_D \in \Pi_D} U_D(\pi_D, R(\pi_D))$$

$$R(\pi_D) = \operatorname{argmax}_{\pi_A \in \Pi_A} U_A(\pi_D, \pi_A)$$

$G \in \{D, A\}$  - players (Defender, Attacker)

$\Pi_G$  - a set of player's  $G$  all mixed strategies

$U_G$  - payoff of player  $G$

**Goal:** find optimal Defender's strategy

# Real-life applications



Federal Air Marshal Service



US Coast Guard in Boston Harbor



Los Angeles Airport



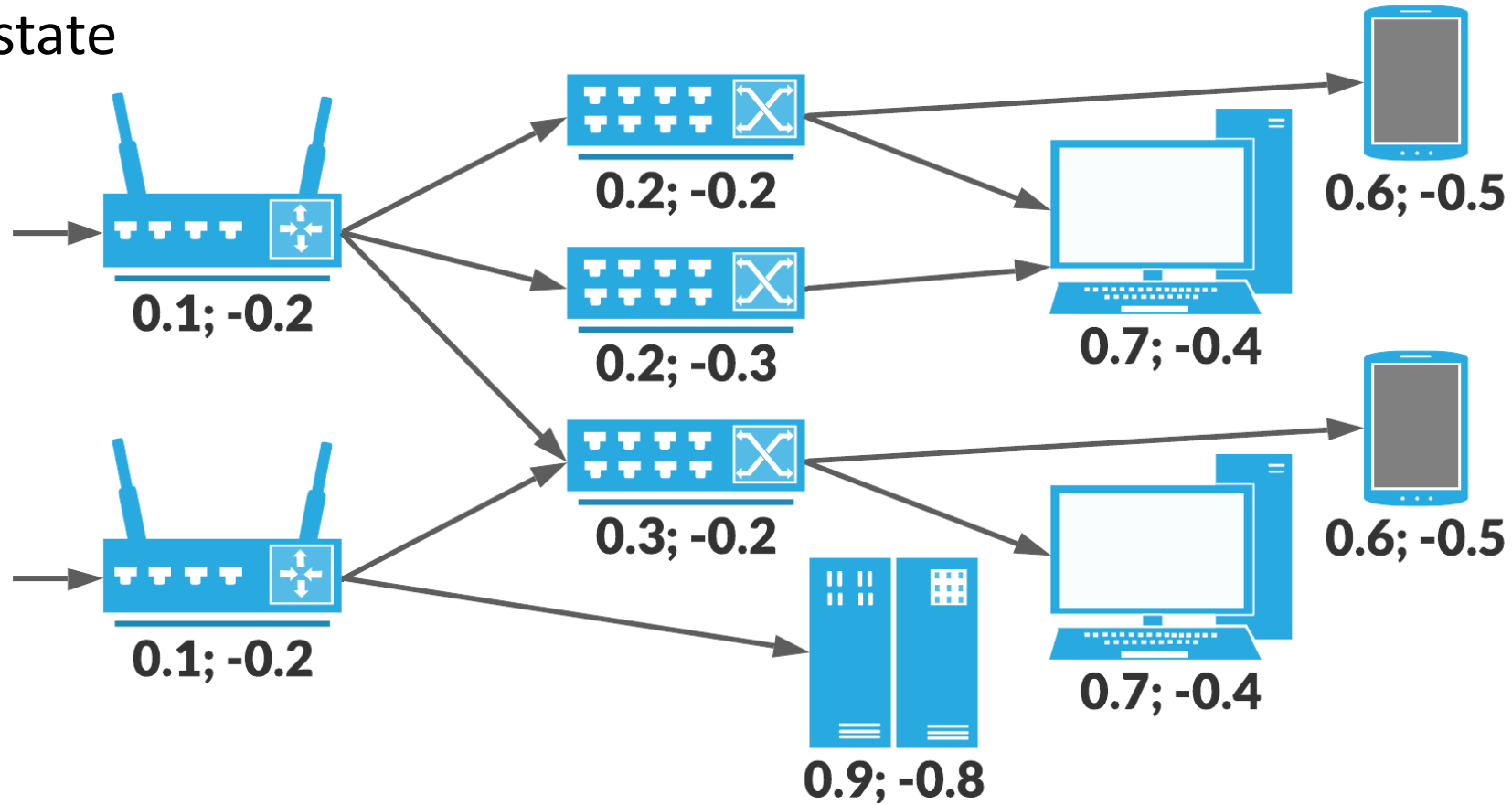
Poaching in Uganda



Tickets control in Los Angeles

# Example – Fliplt games

Initial game state



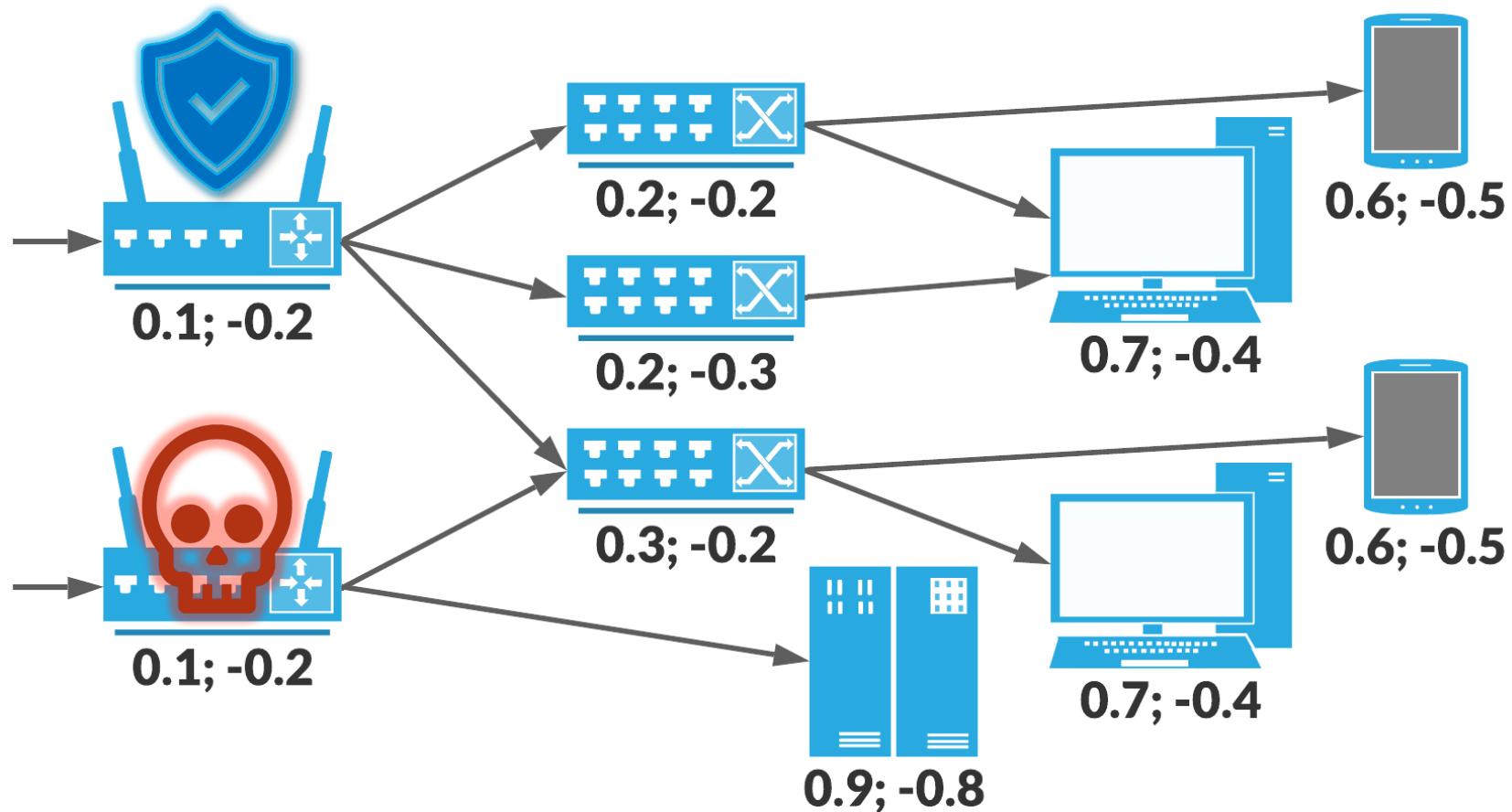
Defender payoff: 0



Attacker payoff: 0

# Example – Fliplt games

Game step 1



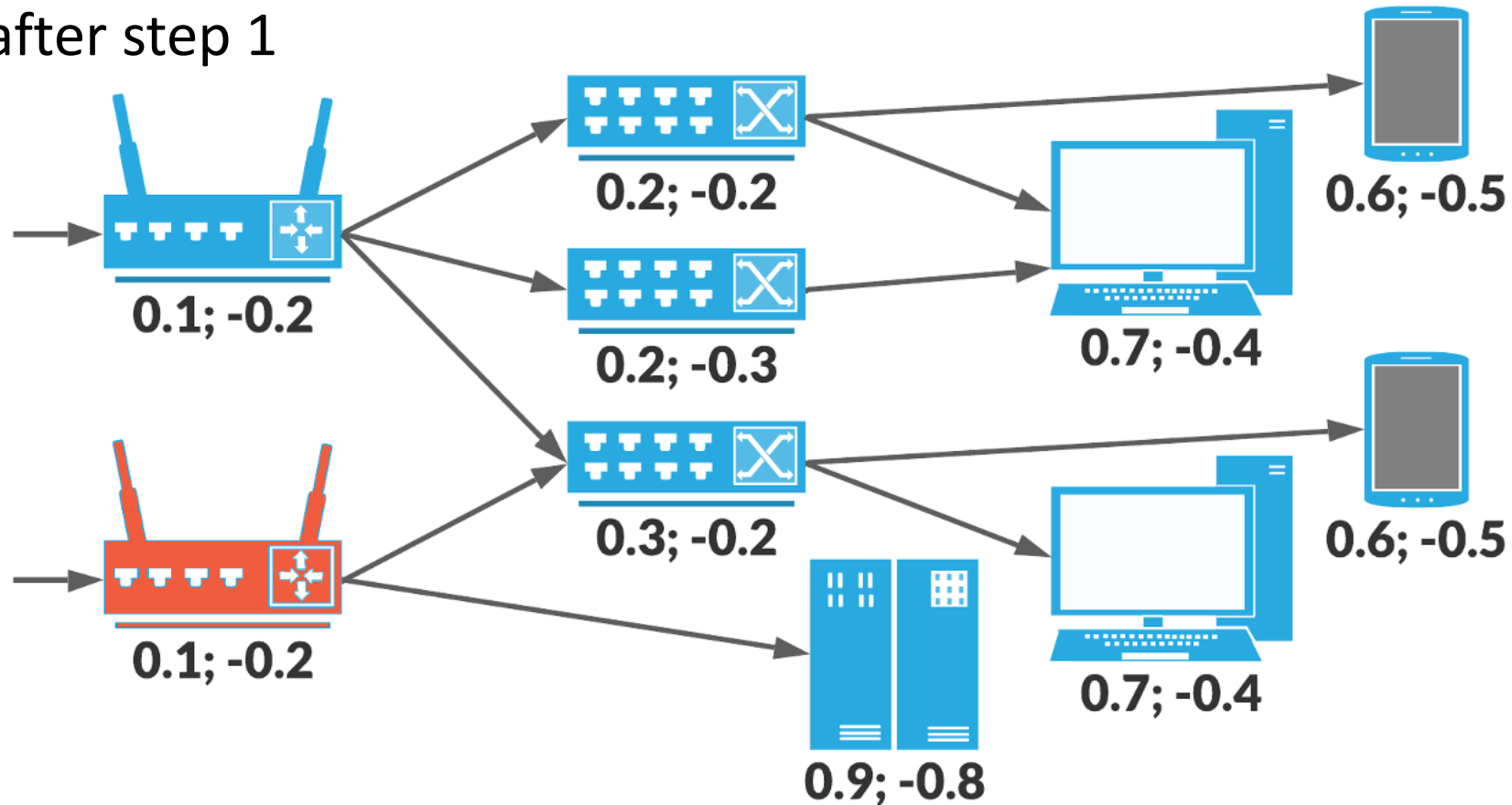
Defender payoff: -0.2



Attacker payoff: -0.2

# Example – Fliplt games

Game state after step 1



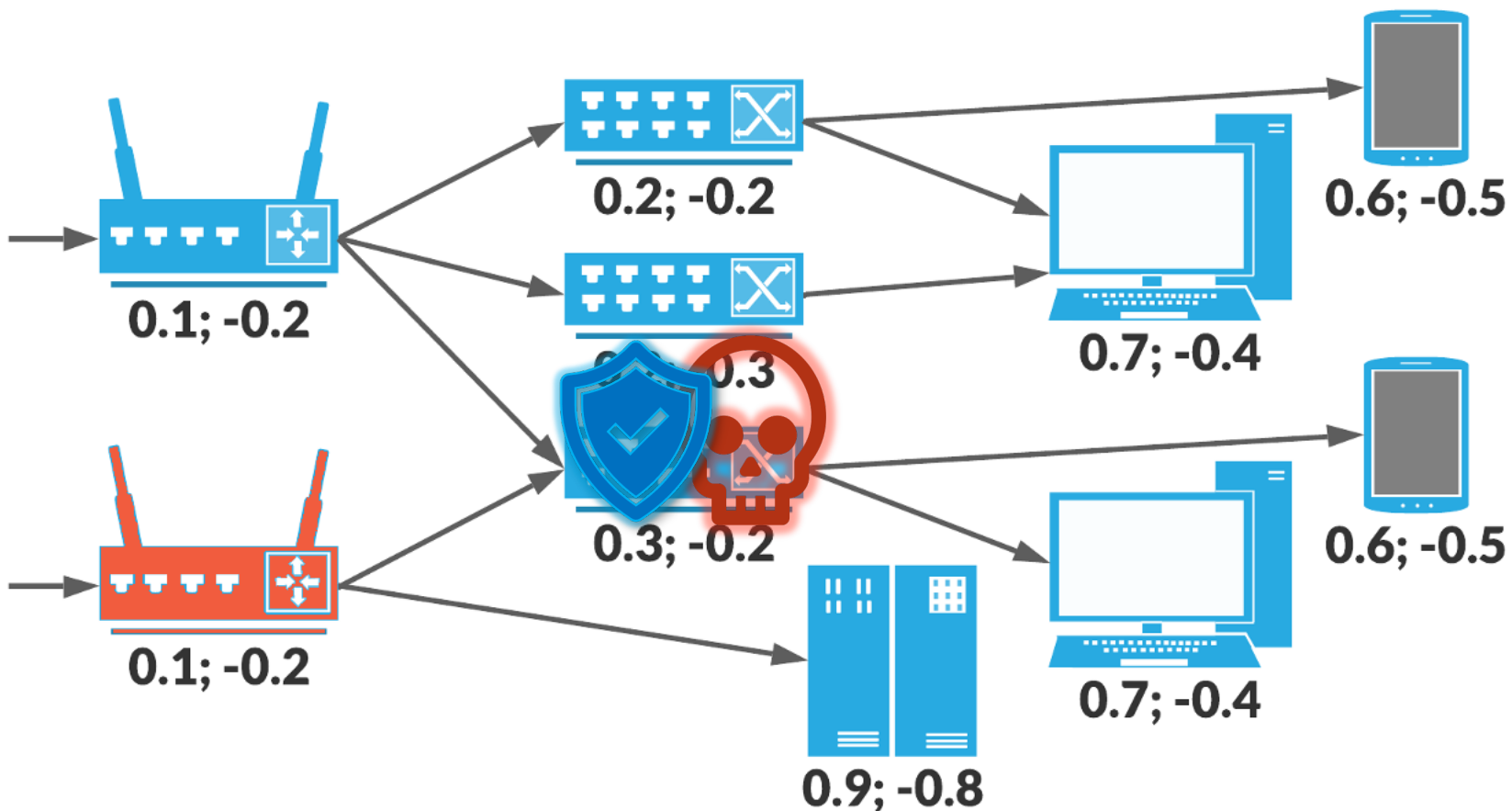
Defender payoff:  $-0.2 + 4.3 = 4.1$



Attacker payoff:  $-0.2 + 0.1 = -0.1$

# Example – Fliplt games

Game step 2



Defender payoff:  $4.1 - 0.2 = 3.9$

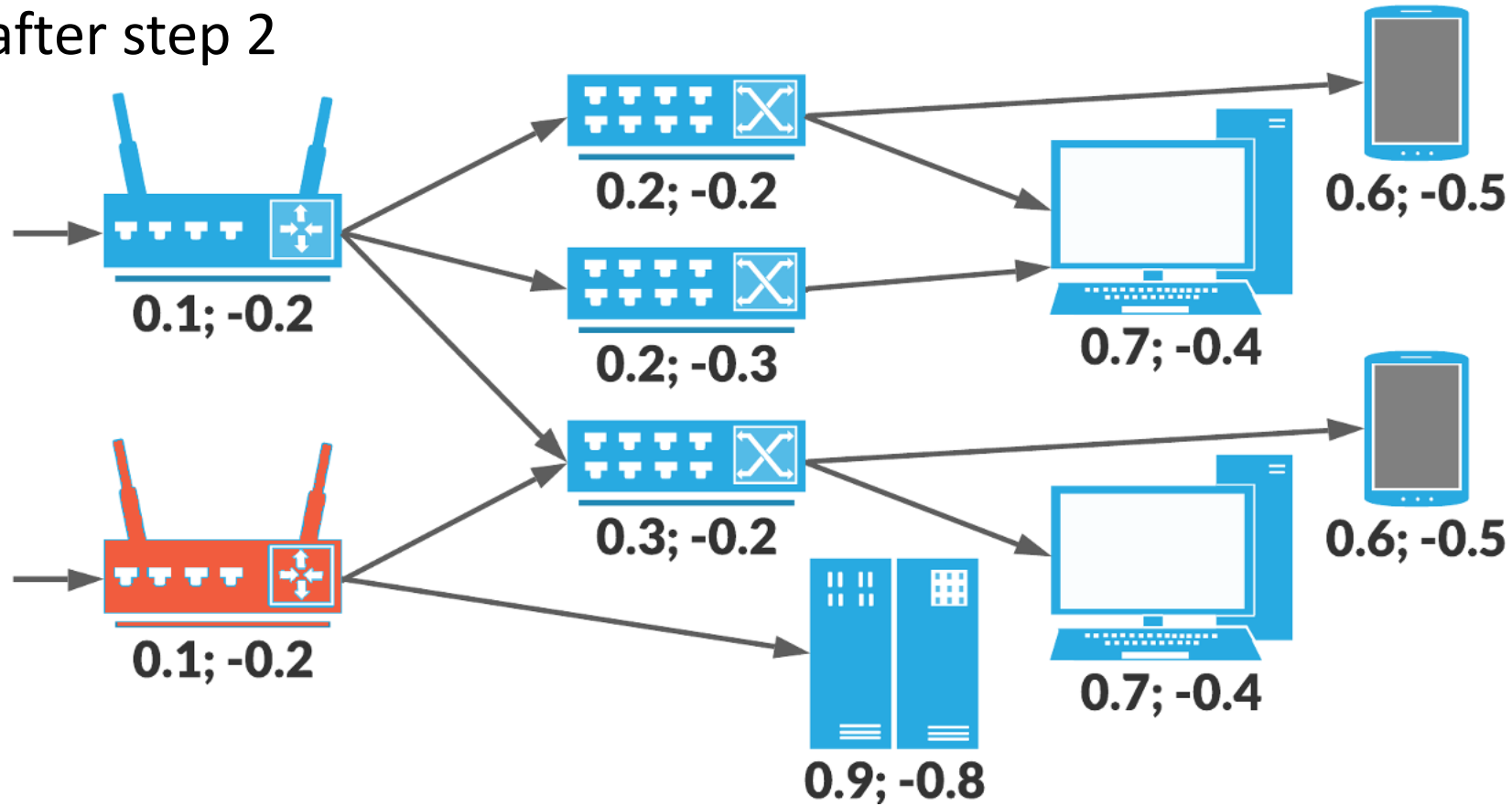


Attacker payoff:  $-0.1 - 0.2 = -0.1$



# Example – Fliplt games

Game state after step 2



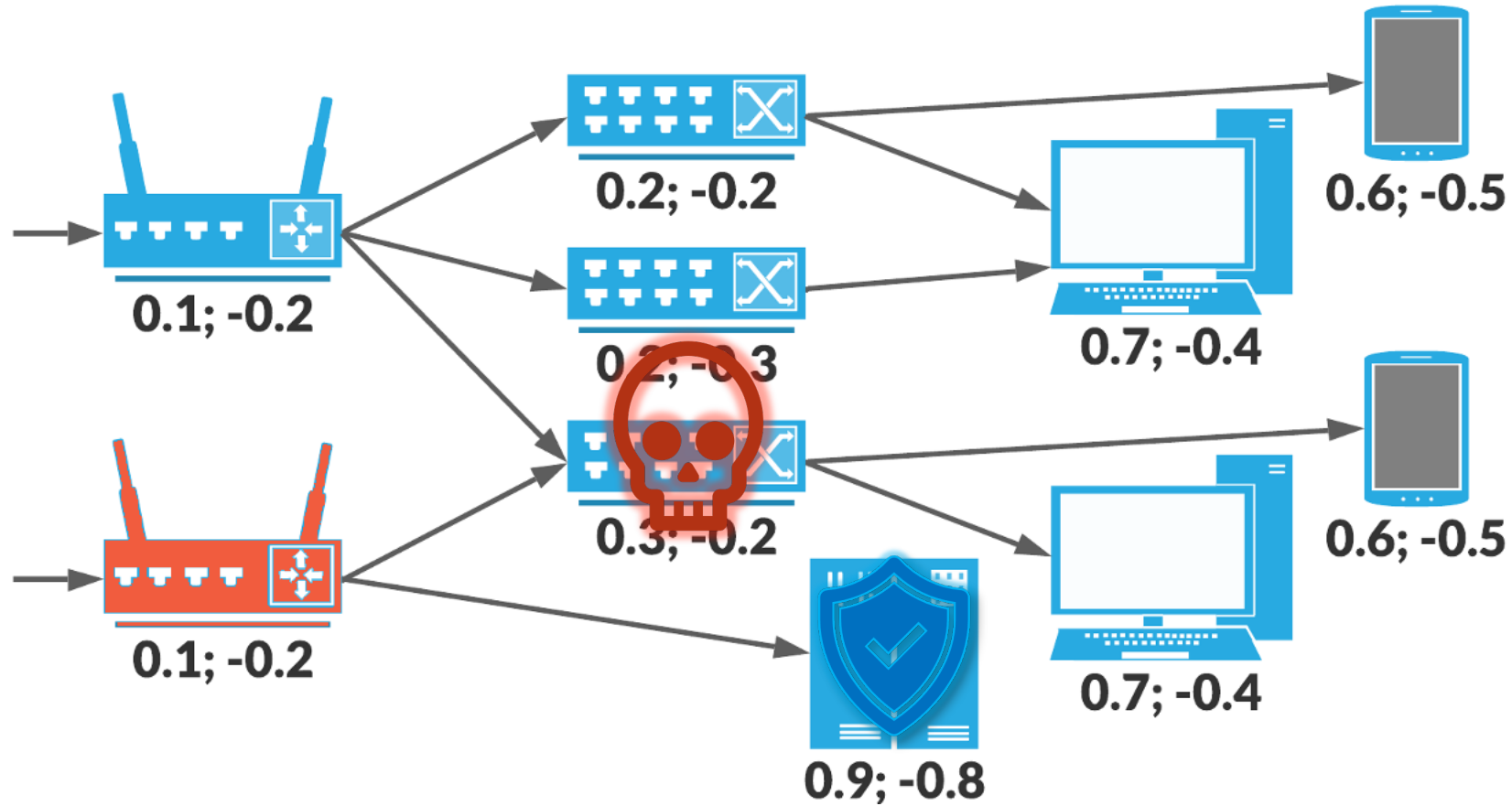
Defender payoff:  $3.9 + 4.3 = 8.2$



Attacker payoff:  $-0.1 + 0.1 = 0$

# Example – Fliplt games

Game step 3



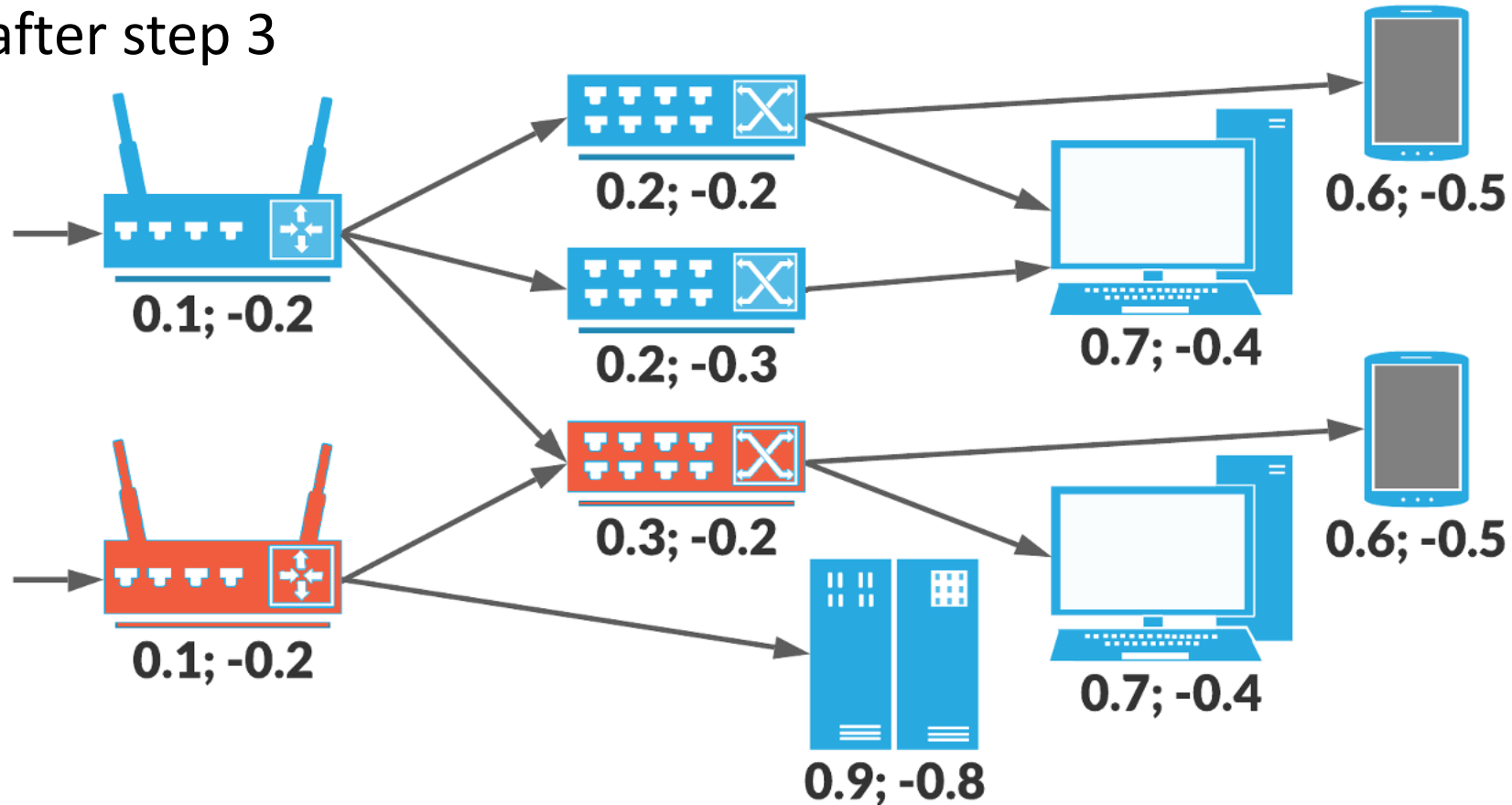
Defender payoff:  $8.2 - 0.8 = 7.4$



Attacker payoff:  $0 - 0.2 = -0.2$

# Example – Fliplt games

Game state after step 3



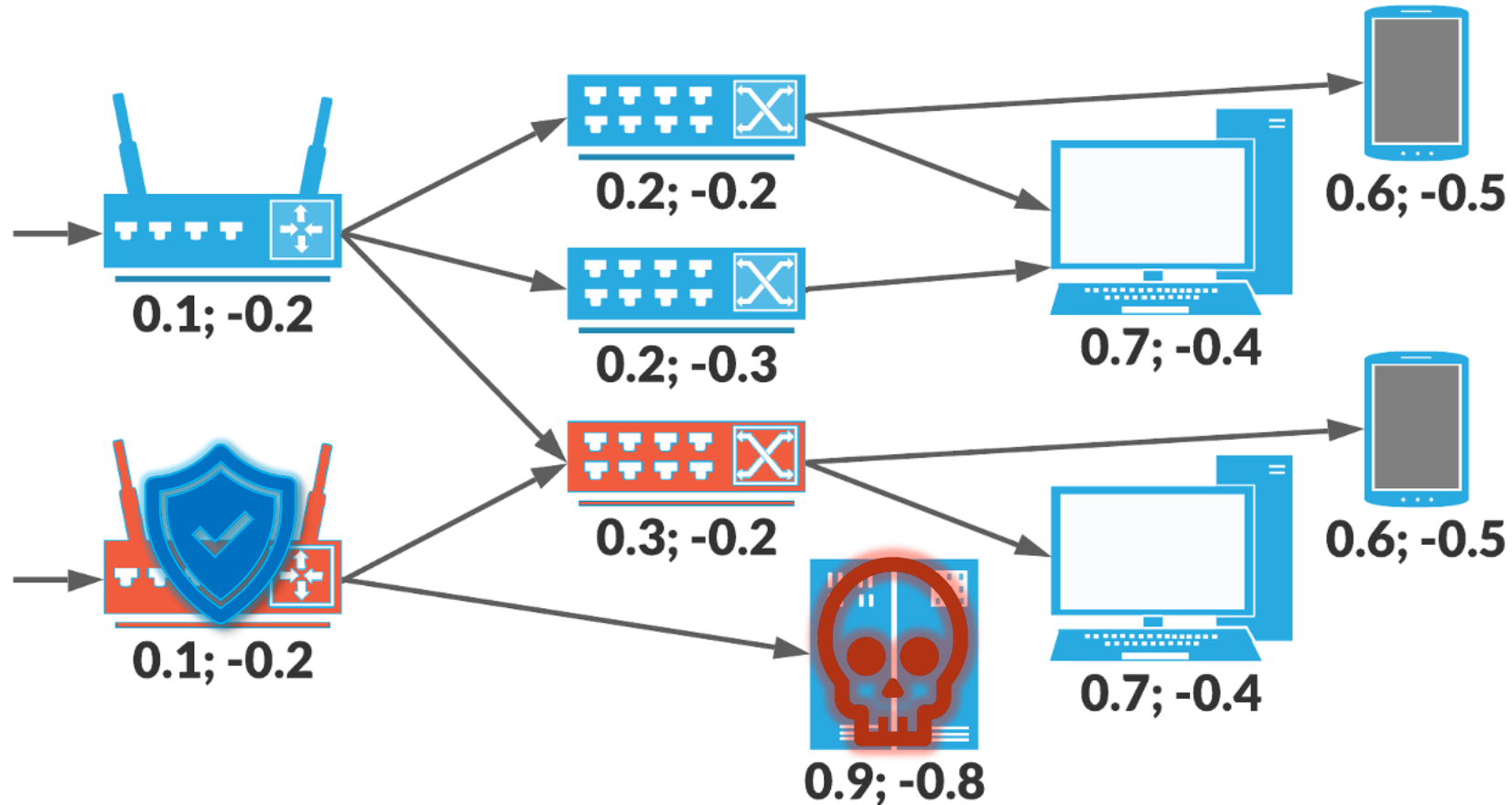
Defender payoff:  $7.4 + 4.0 = 11.4$



Attacker payoff:  $-0.2 + 0.4 = 0.2$

# Example – Fliplt games

Game step 4



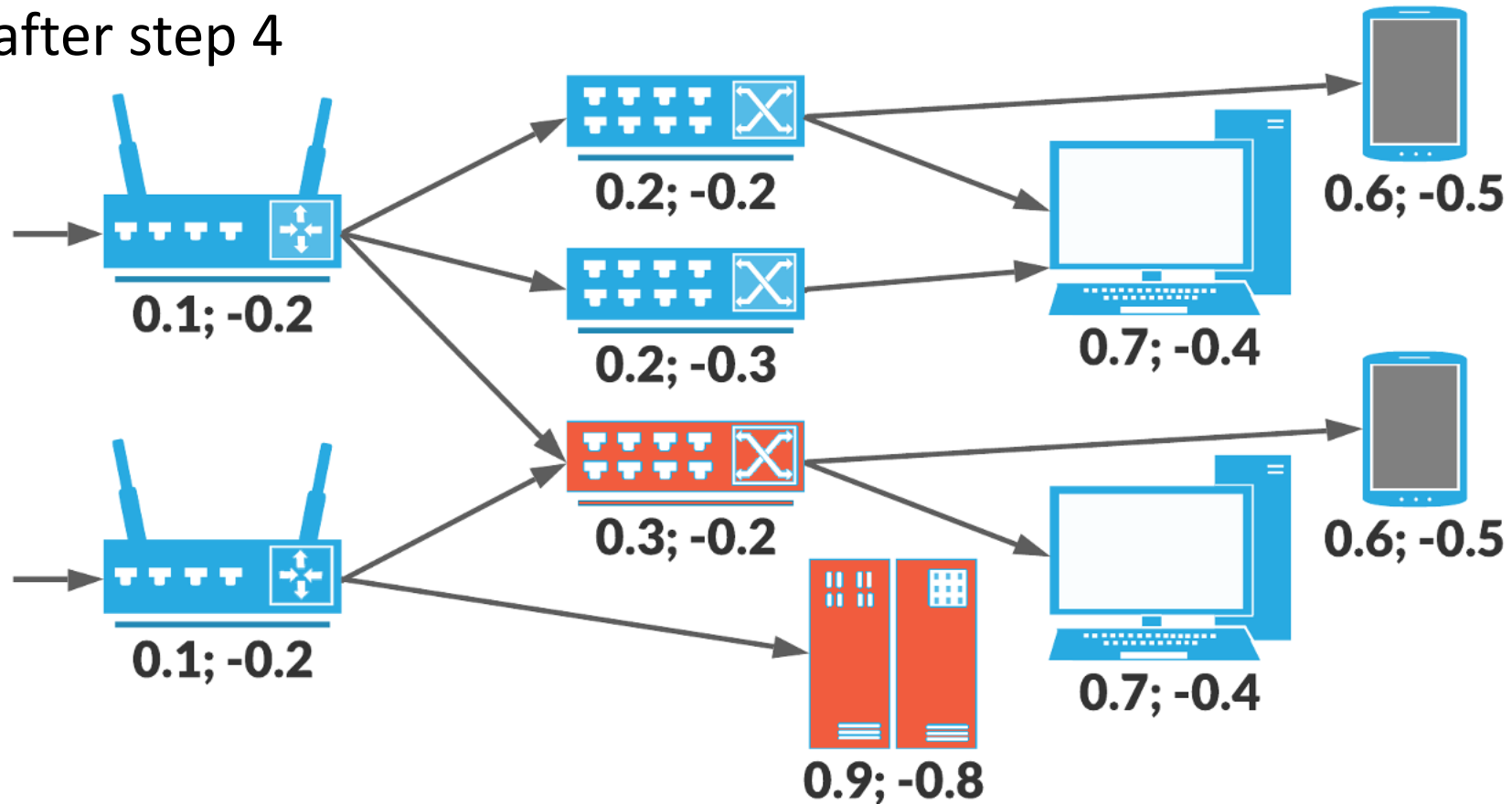
Defender payoff:  $11.4 - 0.2 = 11.2$



Attacker payoff:  $0.2 - 0.8 = -0.6$

# Example – Fliplt games

Game state after step 4



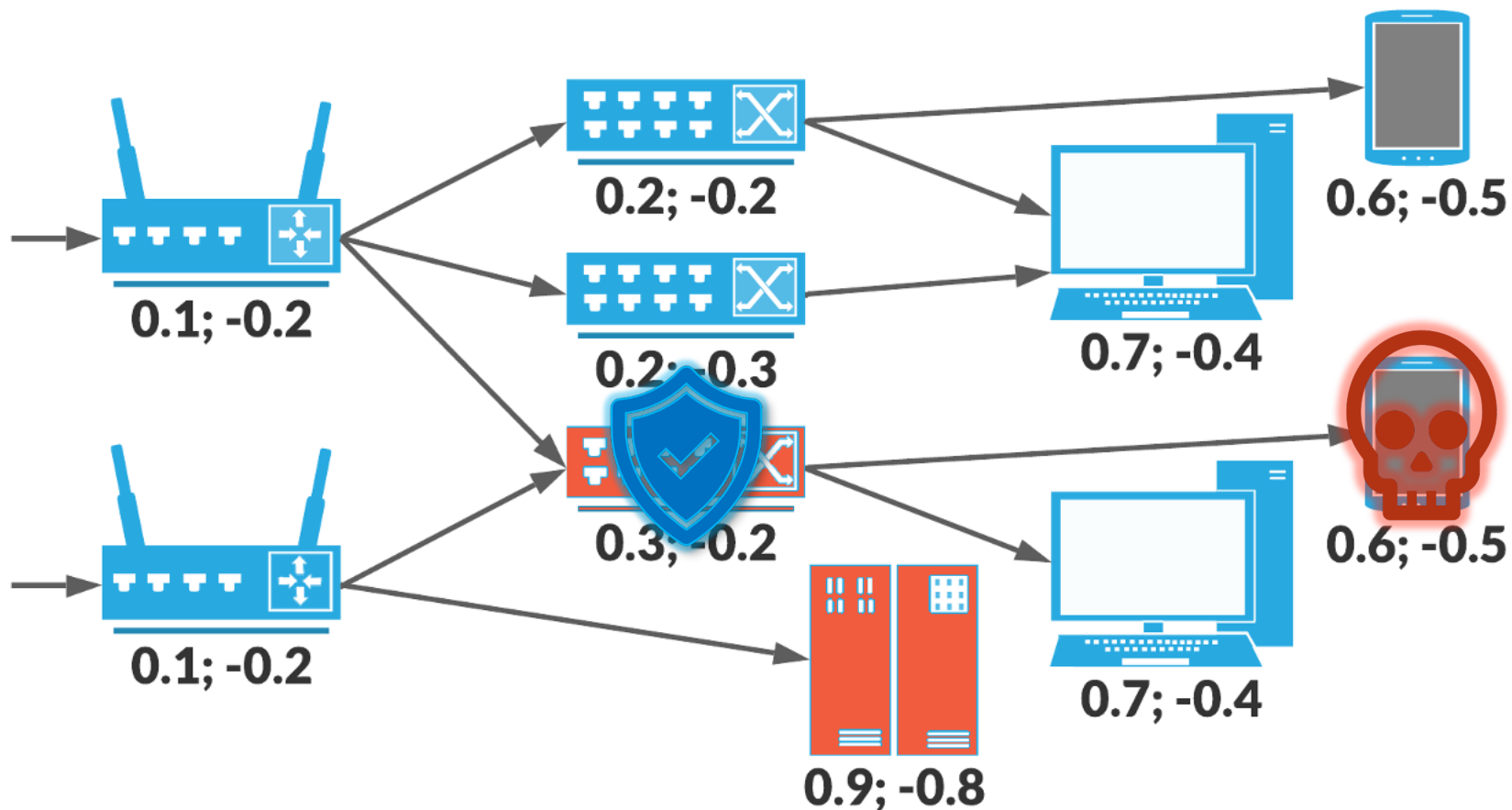
Defender payoff:  $11.2 + 3.2 = 14.4$



Attacker payoff:  $-0.6 + 1.2 = 0.6$

# Example – Fliplt games

Game step 5



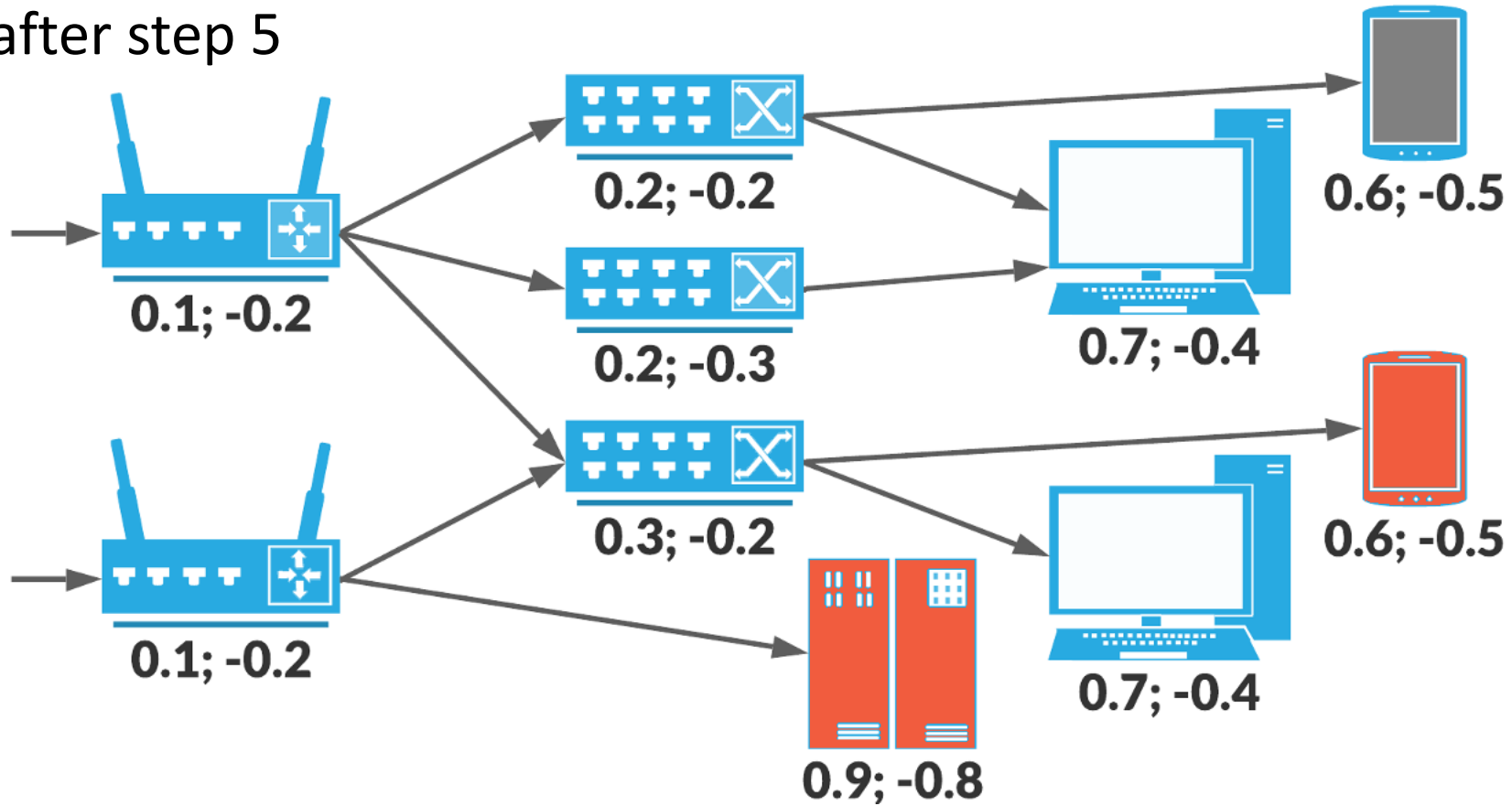
Defender payoff:  $14.4 - 0.2 = 14.2$



Attacker payoff:  $0.6 - 0.5 = 0.1$

# Example – Fliplt games

Game state after step 5

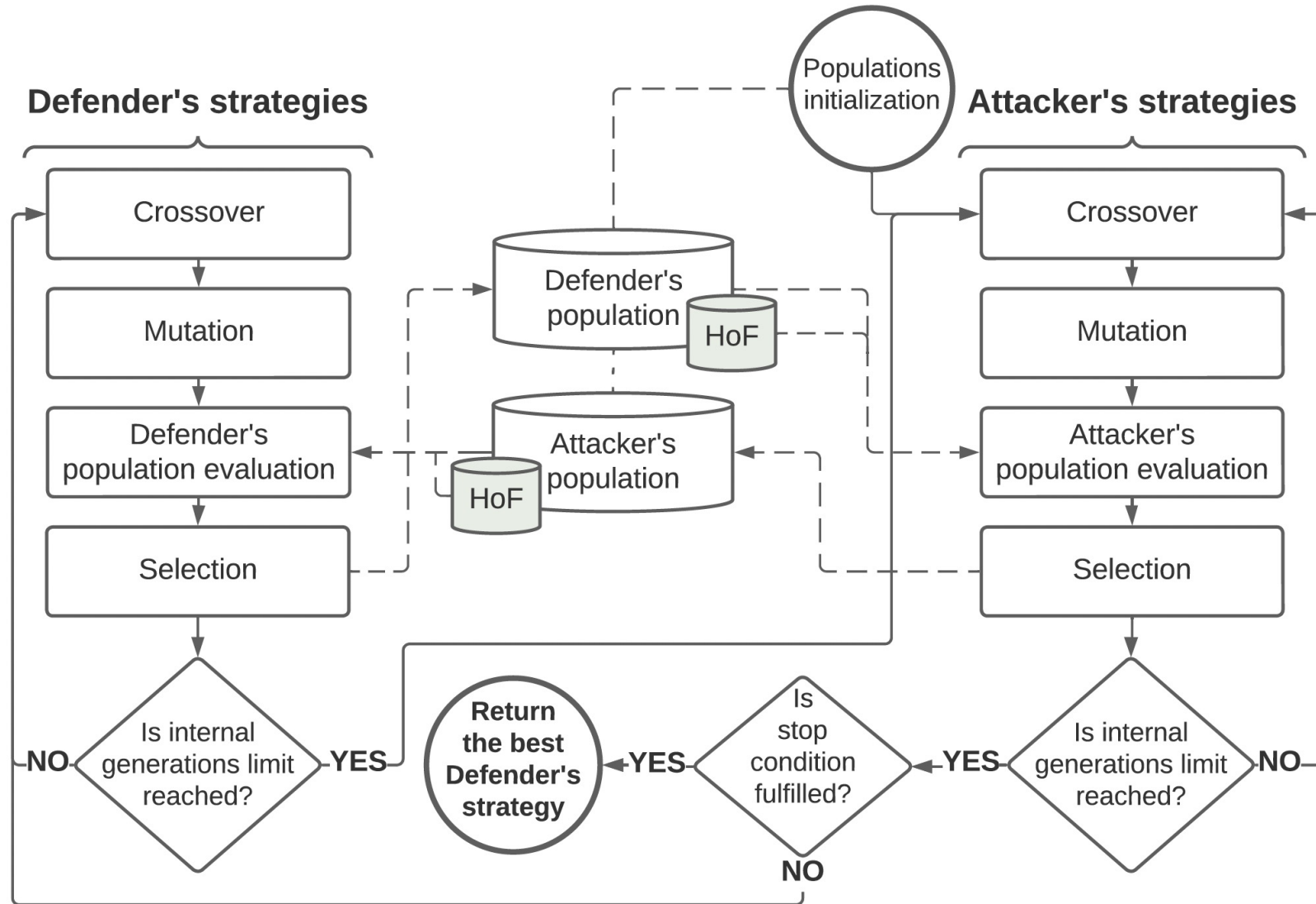


Defender payoff:  $14.2 + 2.9 = 17.1$



Attacker payoff:  $0.1 + 1.5 = 1.6$

# Coevolutionary Algorithm for Stackelberg Security Games (CoEvoSG)





# Hall of Fame

**Role:** Mechanism to retain and store best-performing individuals encountered during evolution.

**Common approach:** Traditional approach adds one highest-fitness individual per generation, potentially suboptimal for diversity.

**Our approach:** Calculates and adds to Hall of Fame mixed Nash equilibrium (mixture of strategies for both players).

**Evaluation enhancement:** Fitness function calculated against a merged set of Hall of Fame and population individuals.

# Results

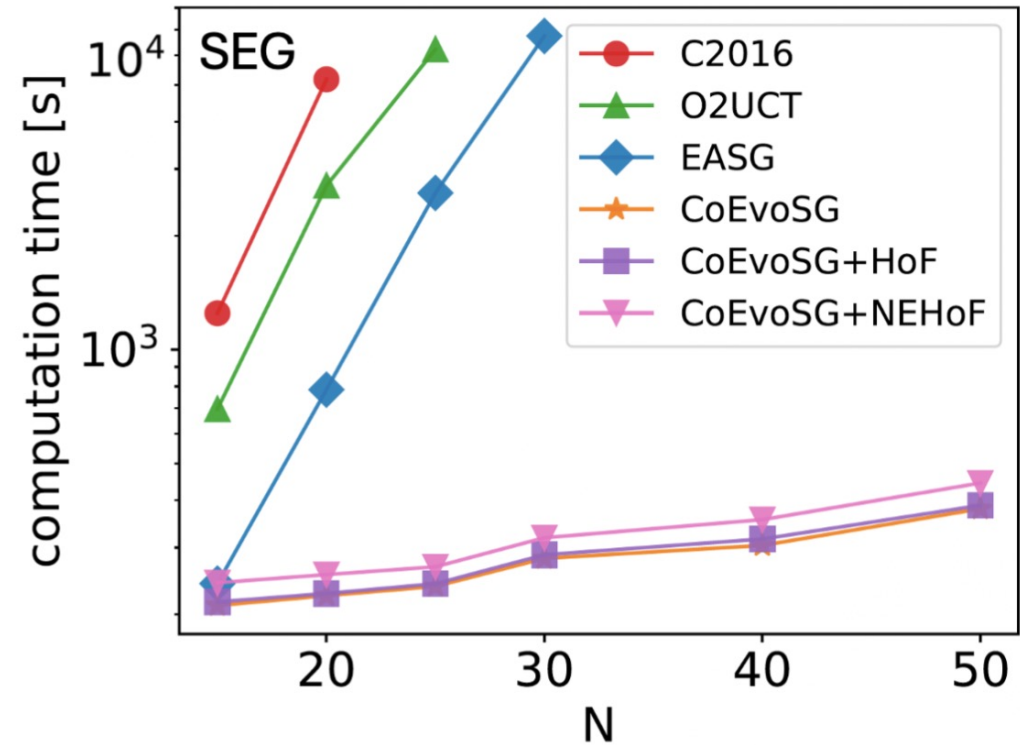
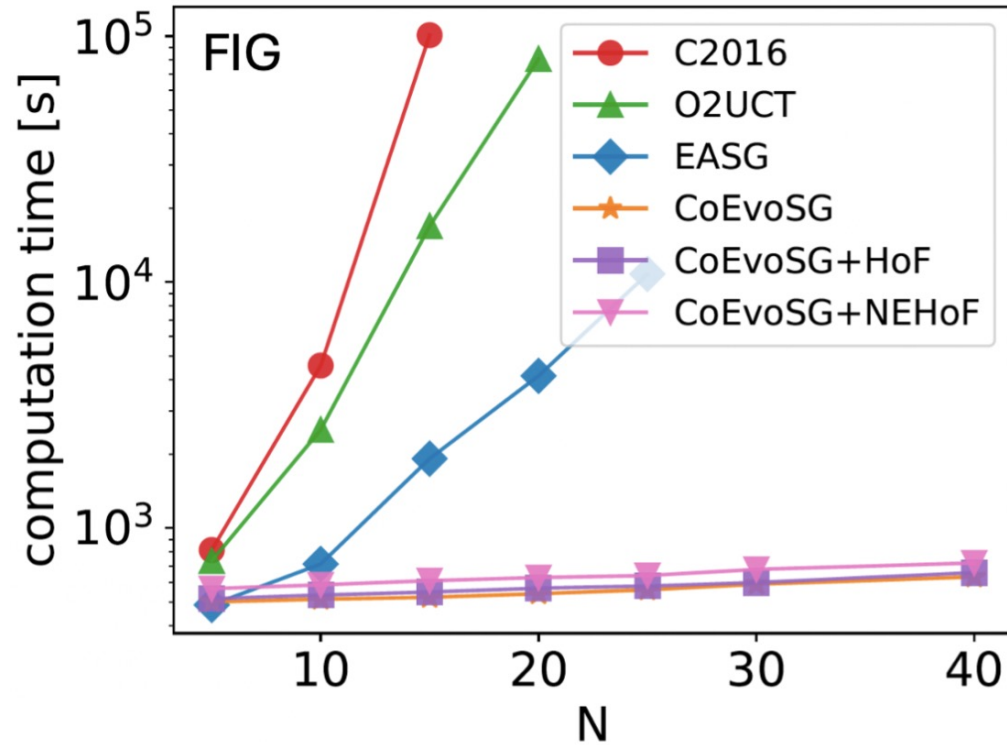
	C2016	O2UCT	EASG	CoEvoSG	CoEvoSG+HoF	<b>CoEvoSG+NEHoF</b>
5	0.890	0.887	0.886	0.886	0.886	0.887
10	0.854	0.848	0.847	0.845	0.845	0.849
15	0.811	0.805	0.802	0.798	0.801	0.806
20	-	0.779	0.780	0.772	0.775	0.776
25	-	-	0.754	0.746	0.751	0.754
30	-	-	-	0.730	0.732	0.735
40	-	-	-	0.722	0.726	0.733

**Table:** Averaged Defender's payoff with respect to game nodes for Fliplt games.

	C2016	O2UCT	EASG	CoEvoSG	CoEvoSG+HoF	<b>CoEvoSG+NEHoF</b>
15	0.122	0.116	0.115	0.115	0.115	0.116
20	0.117	0.107	0.106	0.101	0.104	0.106
25	-	0.119	0.117	0.115	0.116	0.119
30	-	-	0.136	0.135	0.135	0.135
40	-	-	-	0.150	0.152	0.156
50	-	-	-	0.139	0.144	0.146

**Table:** Averaged Defender's payoff with respect to game nodes for Search games.

# Computation times



**Figure:** Computation times with respect to game nodes ( $N$ ) for Flipt and Search games.

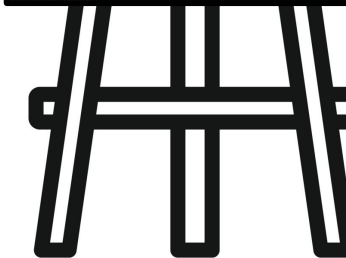
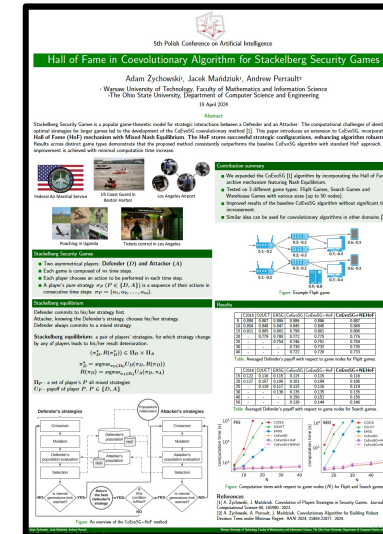
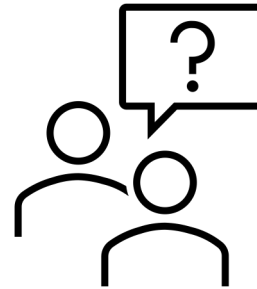
# Summary

- We expanded the CoEvoSG algorithm **by incorporating the Hall of Fame archive mechanism featuring Nash Equilibrium.**
- Tested on 3 different game types: *Fliplt Games*, *Search Games* and *Warehouse Games* with various sizes (up to 50 nodes).
- **Improved results of the baseline CoEvoSG algorithm** without significant time increasement.



Full paper

Thank you



Poster

Tomorrow 15:15

POSTERS 2 session, stand 25

