Deep Reinforcement Learning for Solving Combinatorial Games with Exponential Action Spaces

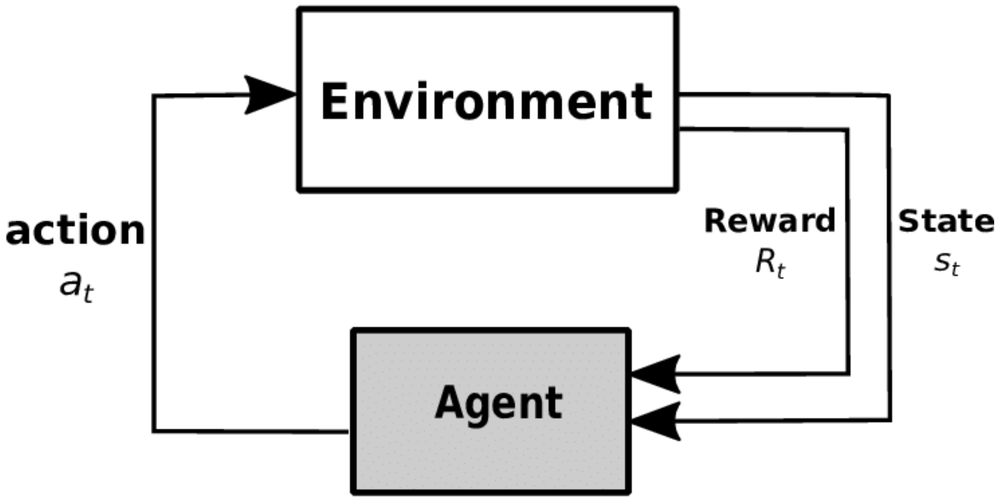
A Paper by Ben Burk and Wei Hu

Supervised by Lucia Moura and Yongyi Mao

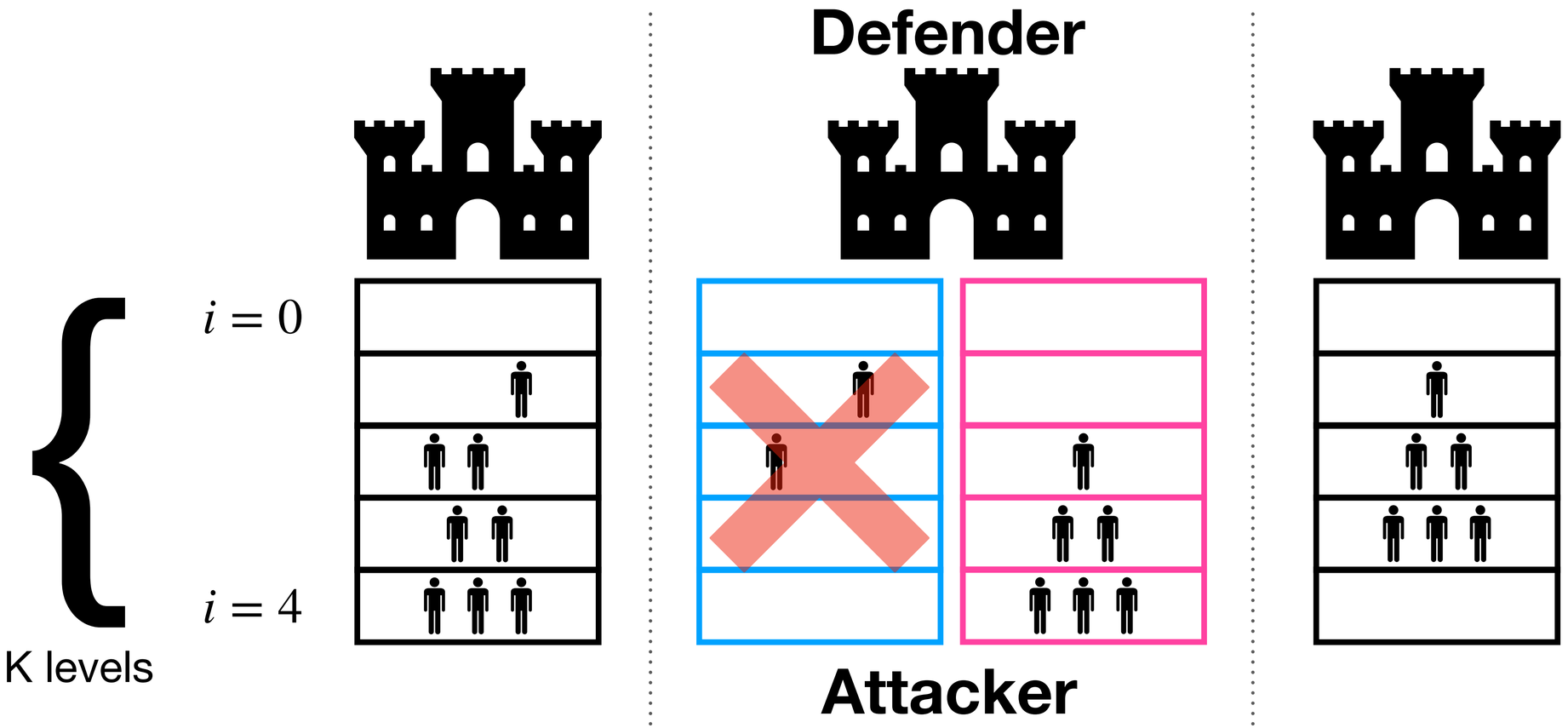
Reinforcement Learning

A Quick Refresher...

- Agent learns to perform a task by interacting with an environment
- Applied to Robotics, Chess, Go
- Agent takes actions based on the environment and receives rewards
- Agent learns to maximize expected cumulative reward



Attacker-Defender Game



- First described by Paul Erdos and Spencer Selfridge (1973)
- Large action space for the attacker
- Easy to make the game more difficult

Erdös, P., & Selfridge, J. L. (1973). On a combinatorial game. *Journal of Combinatorial Theory, Series A*, *14*(3), 298-301.

Attacker-Defender Game

Example game

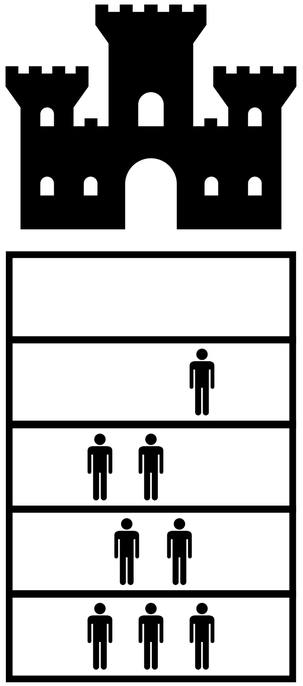
Attacker-Defender Game

Optimal Play

- Easy to define optimal play
- (Spencer, 1994) gives the optimal value function for each level:

$$v_*(k) = \frac{1}{2^{k+1}}$$

- The *potential* is the value of all the pieces on the board
- Represents the number of pieces that will make it the castle if playing against an optimal defender



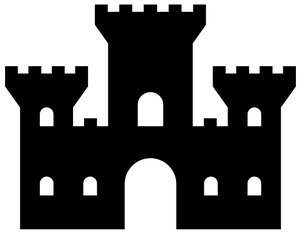
$$\text{potential} = \sum_{i=0}^{K} S[i]v(i) = \frac{3}{2^5} + \frac{2}{2^4} + \frac{2}{2^3} + \frac{1}{2^2} \approx 0.72$$

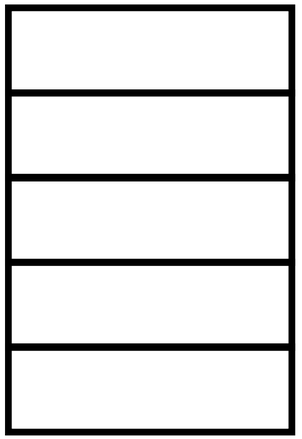
Spencer, J. (1994). Randomization, Derandomization and Antirandomization: Three Games. Theoretical Computer Science, 131(2), 415-429.

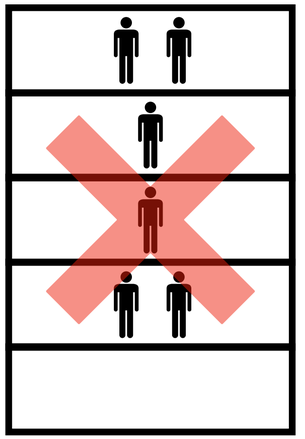
Theorem 2.3 Attacker

Optimal Play

- Theorem 2.3: When playing against an optimal defender, one optimal approach for the attacker is to make the value according to v_* of the two sets as close as possible







potential $= 0.75$

potential $= 0.75$

Attacker-Defender Game

Scoring the Game

- We modified the win condition of the original game
- The score is the number of pieces that reach the castle
- Allows the game to have many more pieces on the board
- Useful for evaluating model generalization

$$
\begin{aligned}
score &> \lfloor potential \rfloor \implies attacker\ wins \\
score &= \lfloor potential \rfloor \implies attacker\ draws \\
score &< \lfloor potential \rfloor \implies attacker\ loses
\end{aligned}
$$

Challenges

Training an Agent

- Action space of attacker grows exponentially with the number of levels and pieces
- (Raghu, Irpan, Andreas, Kleinberg, Le & Kleinberg, 2018) proposed linear reduction
- Attacker training was an afterthought with a simplified action space.

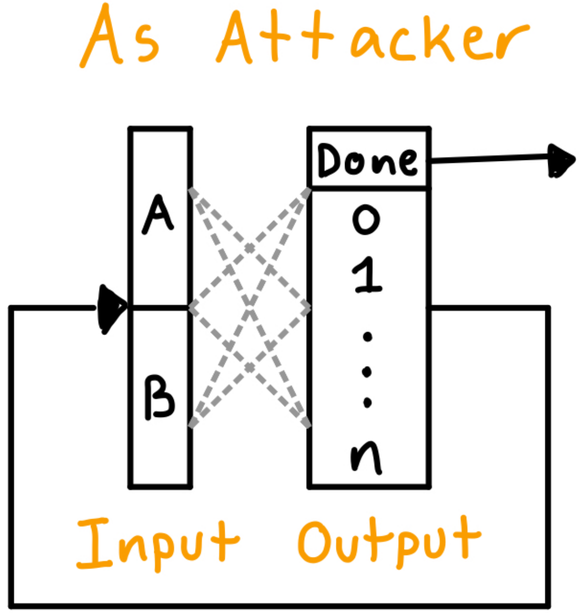
- We attempted several RL approaches to train an agent on the exponential action space
- Attempted Table-Based Q-learning, Deep Q-learning, Actor-Critic
- Failed to perform, and will not work for score-keeping games.

Raghu, M., Irpan, A., Andreas, J., Kleinberg, B., Le, Q., & Kleinberg, J. (2018, July). Can Deep Reinforcement Learning Solve Erdos-Selfridge-Spencer games?. In *International Conference on Machine Learning* (pp. 4238-4246). PMLR.

Our Solution

Micro-actions

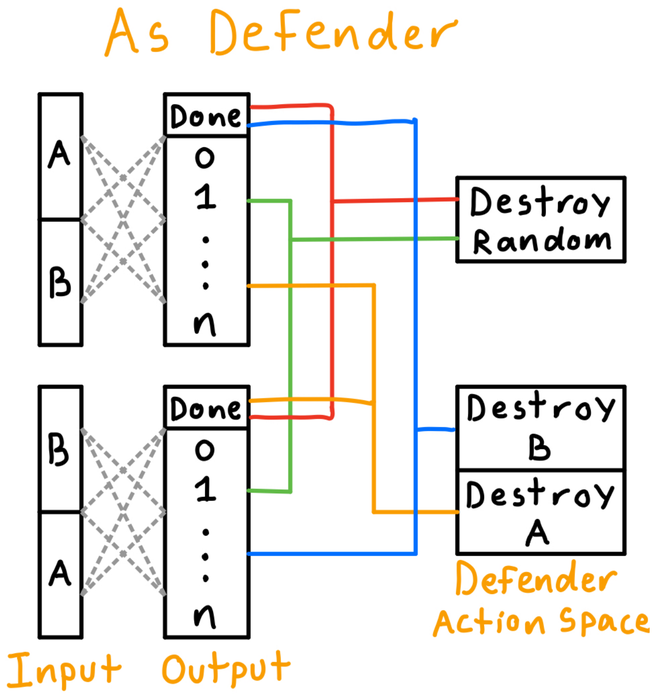
- Construct the partitions iteratively
- Augment action space with a "done" action, signifying end of turn
- We call these micro-actions



Our Solution

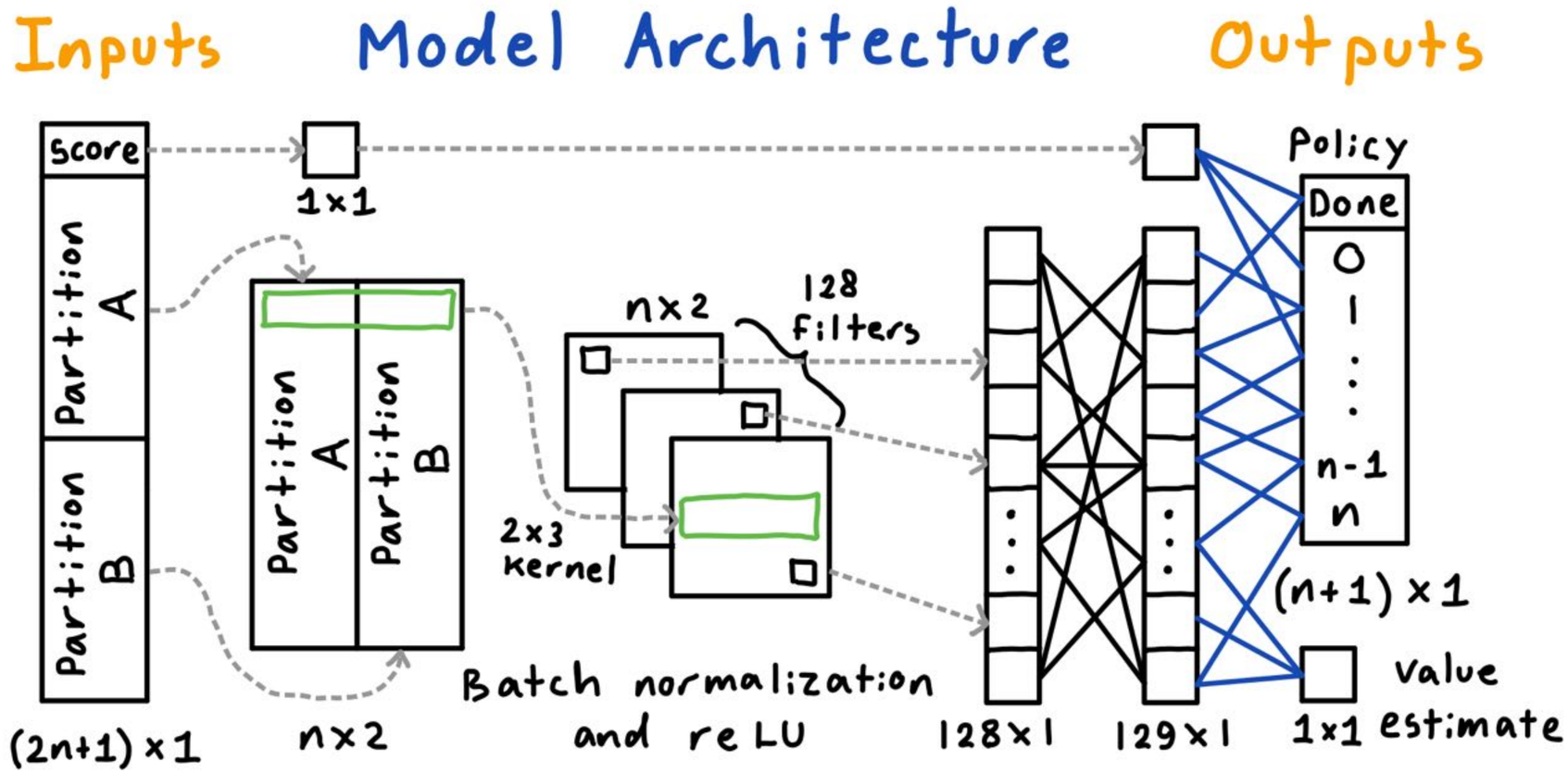
Unifying Attacker and Defender

- Same model can play as an Attacker or a Defender
- If a Defender moves a piece from one set to the other, it sees the set being taken from as more valuable
- If it does not move a piece, it sees the sets as having equal value
- Enabled agents to be trained via self-play



Our Solution

AlphaZero

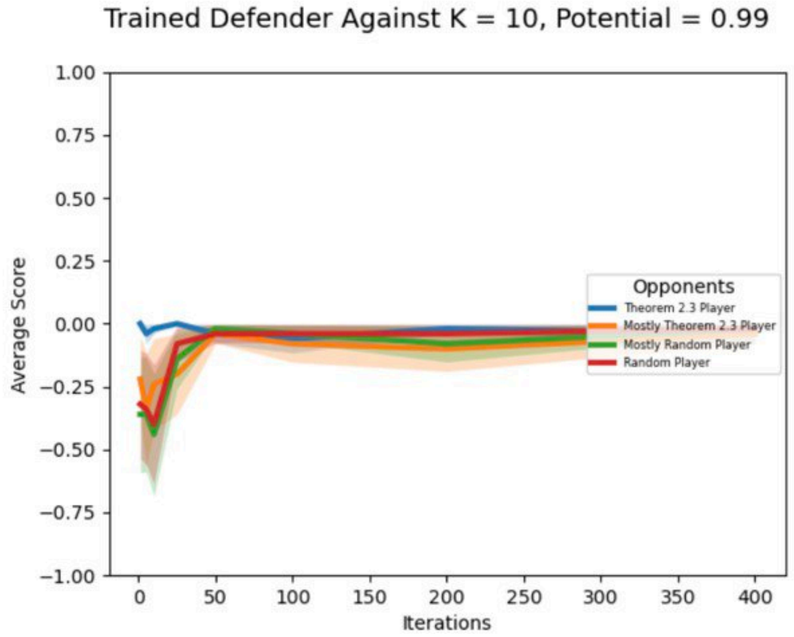


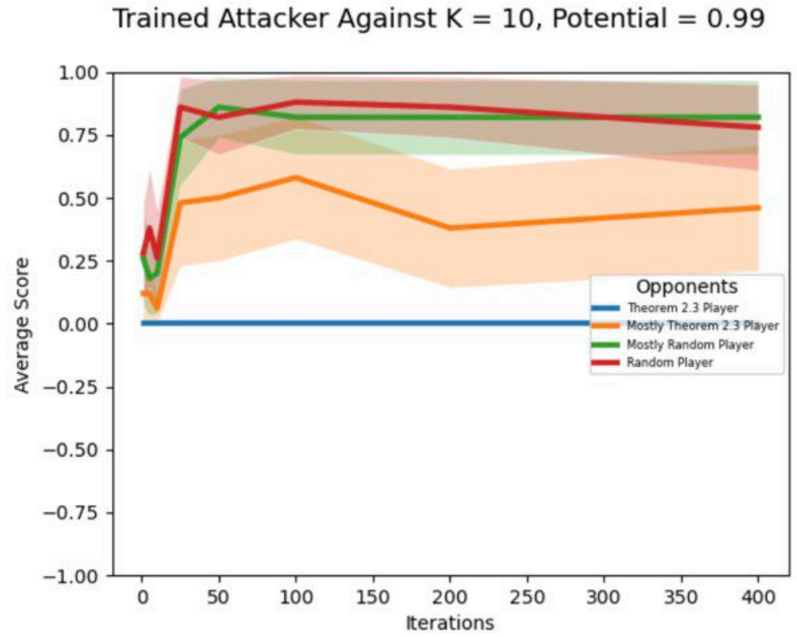
Our Solution

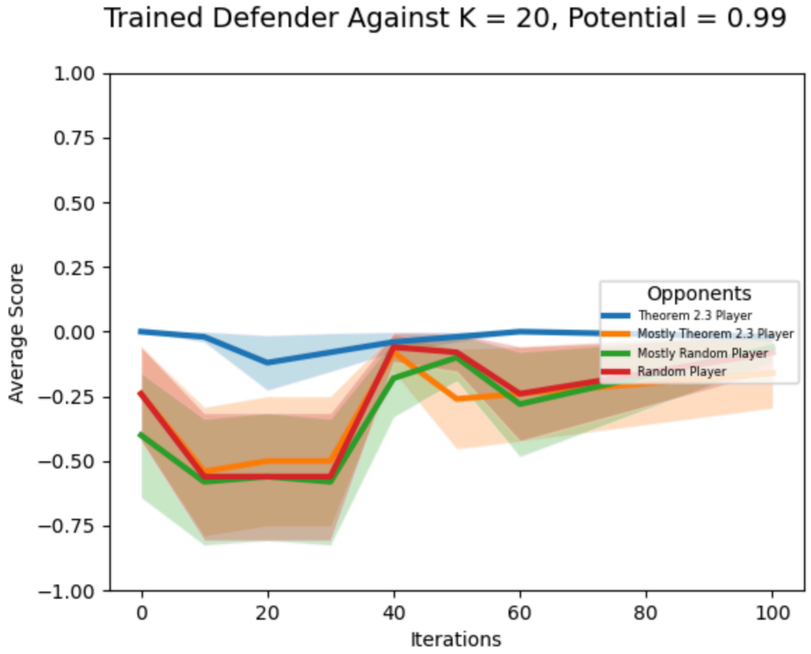
Micro-actions in Action

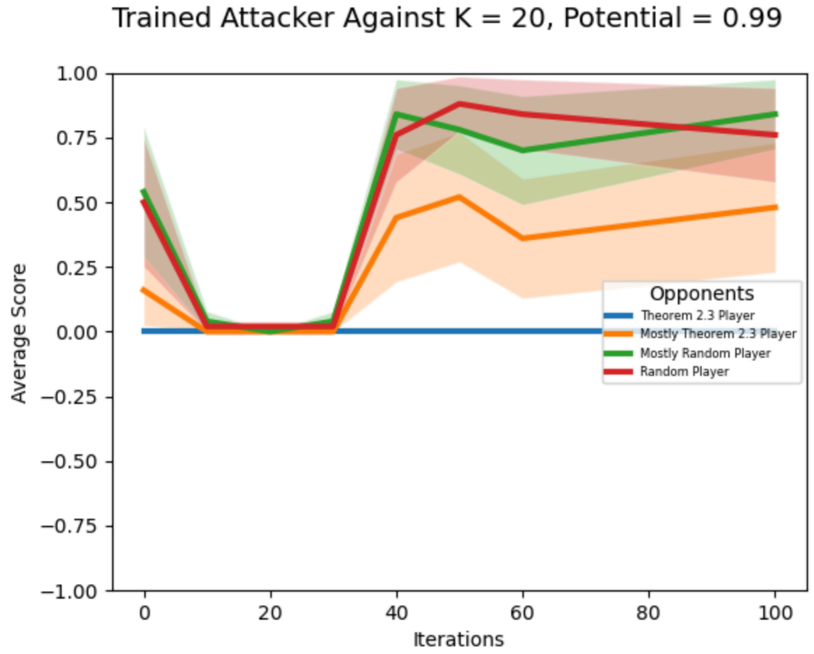
Results

Defending



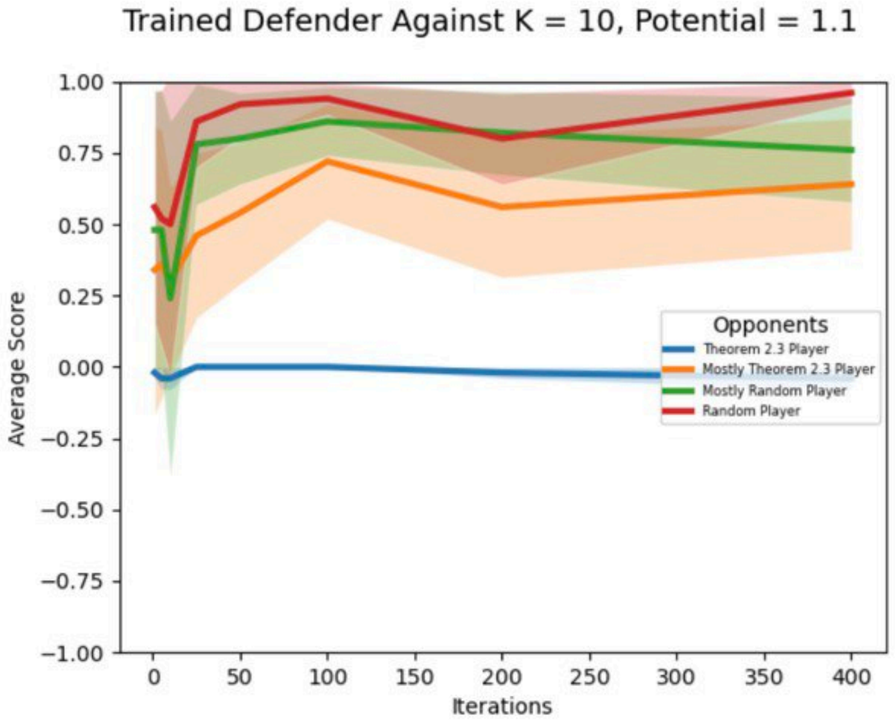


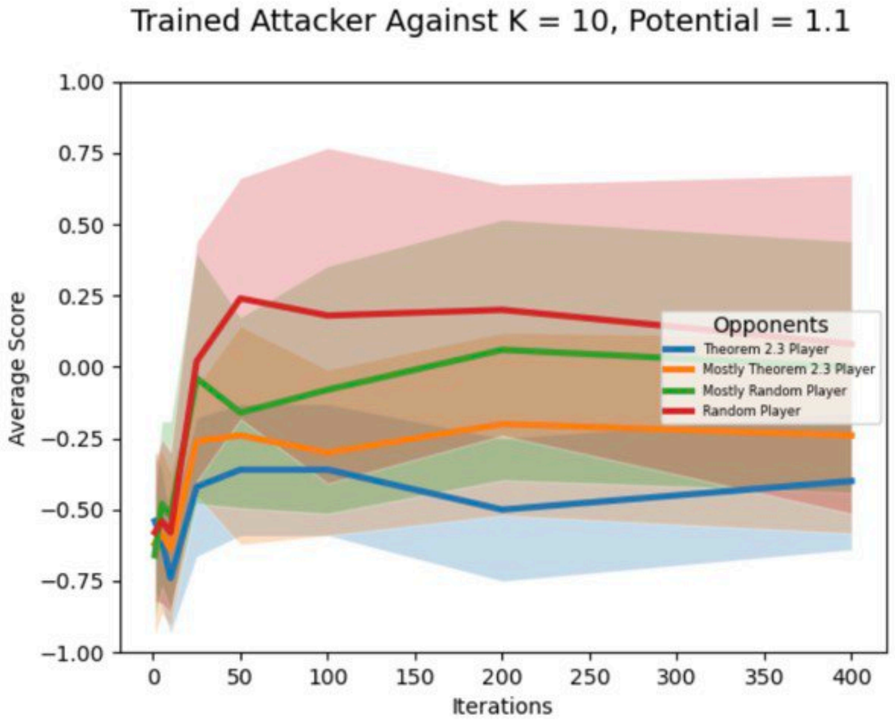




Results

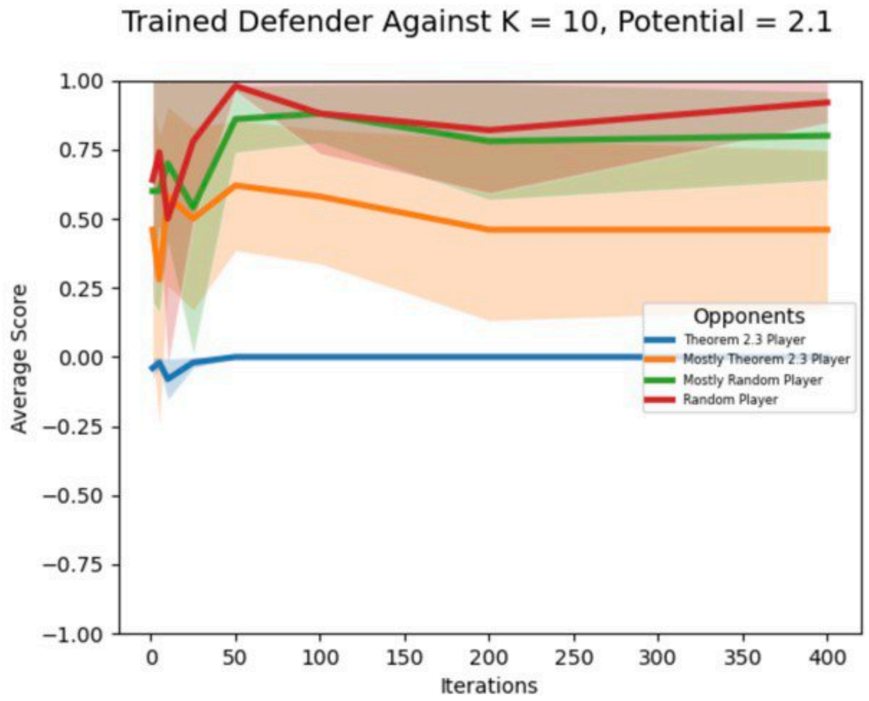
Attacking

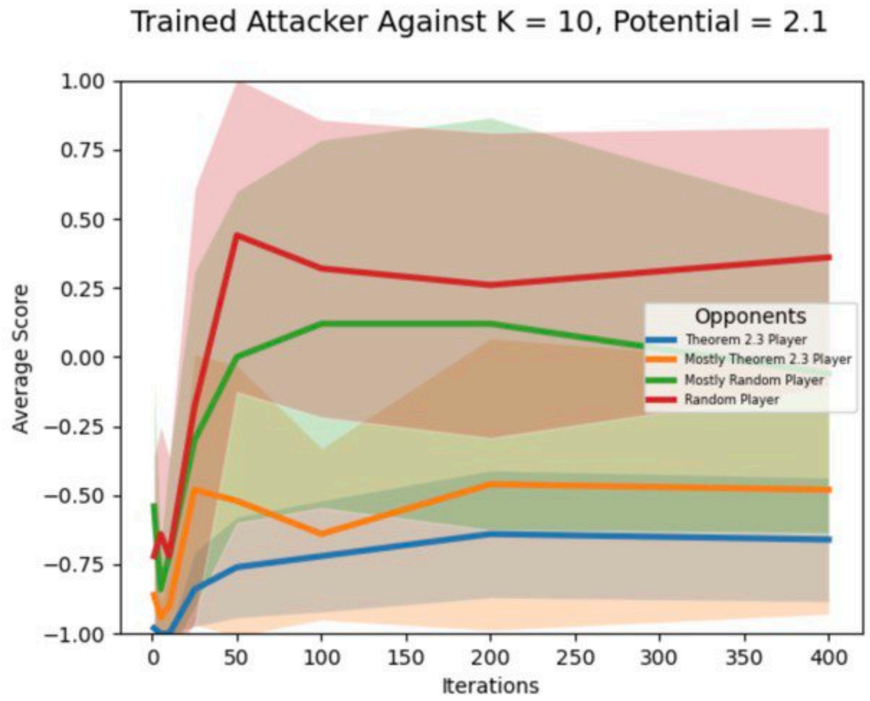




Results

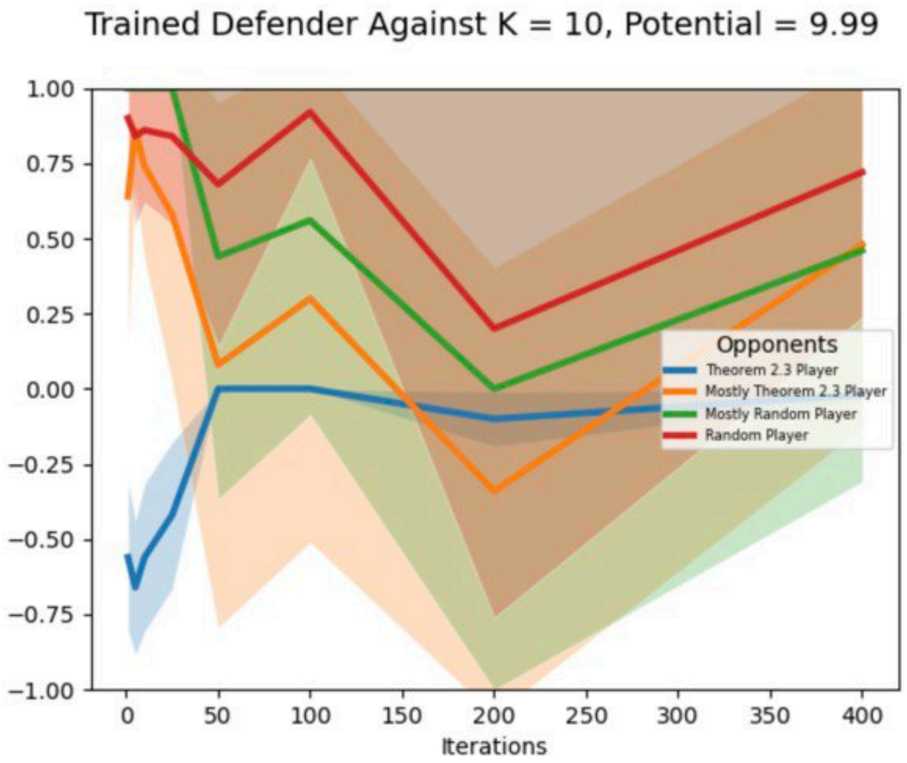
Generalization





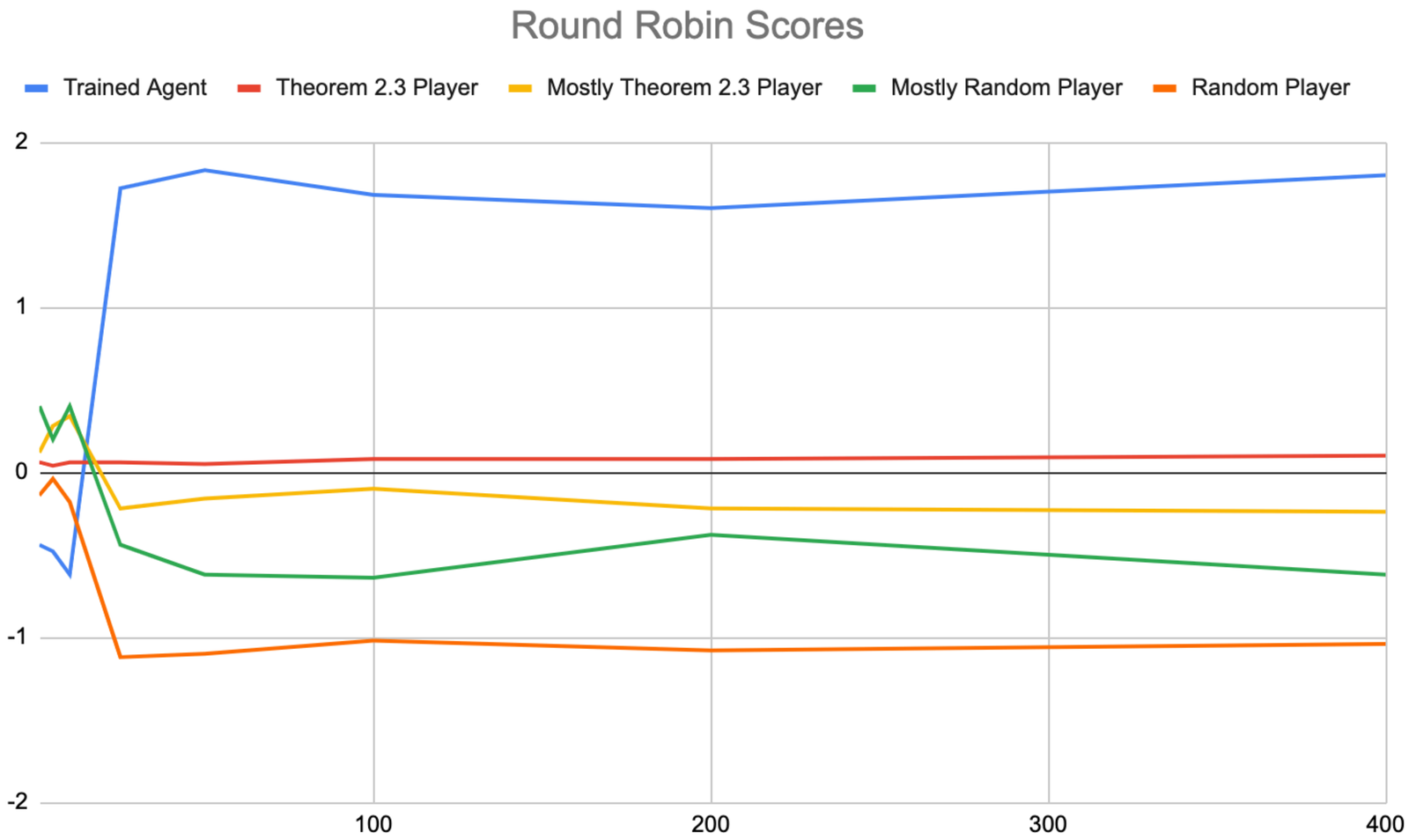
Results

Generalization



Suboptimal Play

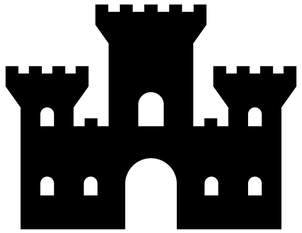
Exploiting Suboptimal Agents

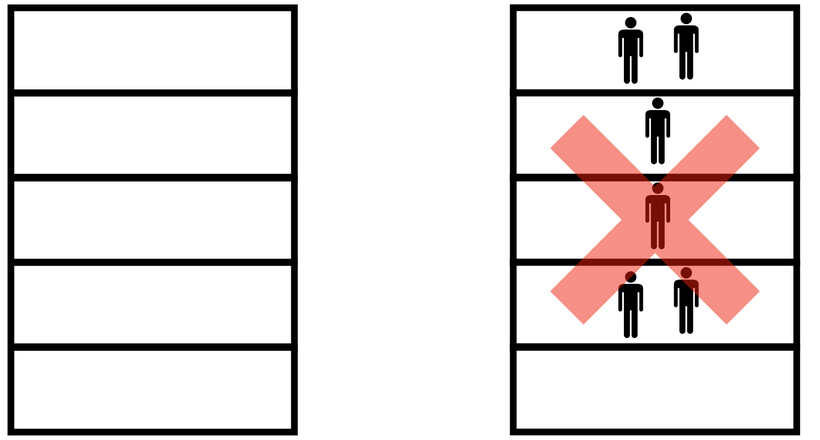


Suboptimal Agent

Playing Unbalanced

- Theorem 2.3: When playing against an optimal defender, one optimal approach for the attacker is to make the value according to v_* of the two sets as close as possible





potential $= 0.5$ potential $= 0.5$

Suboptimal Play

Theoretical Guarantees

- Optimal

$$v(i) = 2v(i+1)$$

- Nearsighted

$$v(i) > 2v(i+1)$$

- Farsighted

$$v(i) < 2v(i+1)$$

$$v_*(k) = \frac{1}{2^{k+1}}$$

We designed and proved algorithm for optimal play against nearsighted and farsighted opponents.

Conclusion

Contributions

- Generalized a family of classic combinatorial games, by considering a score-keeping variant.
- Designed and implemented a novel RL algorithm to handle exponential action spaces:
  - Attackers and Defenders can now both be trained.
  - We can train using true self-play, without expert knowledge.
  - We achieve strong generalization — good performance on score-keeping games.
- Proved theoretical guarantees for suboptimal play

Conclusion

Questions

- Thanks for listening!
- Thanks to Dr. Moura & Dr. Mao for their guidance and support.