PSN Game: Game-theoretic Prediction and Planning via a Player Selection Network

Extended Abstract

Tianyu Qiu
University of Texas at Austin
Austin, United States of America
tianyuqiu@utexas.edu

Eric Ouano
University of Texas at Austin
Austin, United States of America
eouano@utexas.edu

Fernando Palafox
University of Texas at Austin
Austin, United States of America
fernandopalafox@utexas.edu

Christian Ellis
University of Texas at Austin
Austin, United States of America
christian.ellis@austin.utexas.edu

David Fridovich-Keil
University of Texas at Austin
Austin, United States of America
dfk@utexas.edu

ABSTRACT

While game-theoretic planning frameworks are effective at modeling multi-agent interactions, they require solving large optimization problems where the number of variables increases with the number of agents, resulting in long computation times that limit their use in large-scale, real-time systems. To address this issue, we propose i) **PSN Game**—a learning-based, game-theoretic prediction and planning framework that reduces runtime by learning a *Player Selection Network* (PSN); and ii) a *Goal Inference Network* (GIN) that makes it possible to use the PSN in incomplete information games where agents' intentions are unknown. A PSN outputs a player selection mask that distinguishes influential players from less relevant ones, enabling the ego player to solve a smaller, masked game involving only selected players. By reducing the number of players in the game, and therefore reducing the number of variables in the corresponding optimization problem, PSN directly lowers computation time. The PSN Game framework is more flexible than existing player selection methods as it i) relies solely on observations of players' past trajectories, without requiring full state, action, or other game-specific information; and ii) requires no online parameter tuning. Experiments in both simulated scenarios and human trajectory datasets demonstrate that PSNs outperform baseline selection methods in i) prediction accuracy; and ii) planning safety. PSNs also generalize effectively to real-world scenarios in which agents' objectives are unknown without fine-tuning. By **selecting only the most relevant players** for decision-making, PSN Game offers a general mechanism for **reducing planning complexity** that can be seamlessly integrated into existing multi-agent planning frameworks.

KEYWORDS

Multi-agent systems, Key player selection, Game-theoretic planning

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/

ACM Reference Format:
Tianyu Qiu, Eric Ouano, Fernando Palafox, Christian Ellis, and David Fridovich-Keil. 2026. PSN Game: Game-theoretic Prediction and Planning via a Player Selection Network: Extended Abstract. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 3 pages. https://doi.org/10.65109/

1 INTRODUCTION

Most existing game-theoretic planning approaches [1, 3] focus on static scenarios with only a small number of agents (e.g., ≤ 5), where computational efficiency is rarely a concern. In contrast, scenarios involving many dynamic agents demand frequent replanning, making computational efficiency a critical bottleneck. In such settings, the number of optimization variables, typically proportional to the number of agents, can grow rapidly and lead to significant delays[1].

We propose PSN Game—a novel game-theoretic framework that learns a Player Selection Network (PSN) to reduce the computation time of game-solving for multi-agent trajectory prediction and planning. The PSN takes the interested agent and the surrounding agents' past trajectories as input, and outputs a player selection mask identifying the most influential agents. Furthermore, we build a Goal Inference Network (GIN) that infers other agents' objectives even when they are *a priori* unknown, making player selection practical even in incomplete information settings. Ultimately, the PSN, together with the GIN, constructs a smaller-scale masked game over the selected subset of agents, whose equilibrium solution can encode trajectory predictions and planned interactions (Fig. 1).

2 MASKED NASH GAME

Definition 1 (Player Selection Mask)**.** Suppose that the i^{th} player is the ego agent. Then, the player selection mask is denoted as $M^i := (m^{ij}) \in \{0,1\}^{N-1}$ for $j \in [N] \setminus \{i\}$, where

$$m^{ij} := \begin{cases} 1, & \text{Agent } j \in [N] \text{ is included in the game} \\ 0, & \text{Agent } j \in [N] \text{ is excluded from the game.} \end{cases} \tag{1}$$

Definition 2 (Masked Nash Game)**.** Given an open-loop Nash game $\Gamma(\mathbf{x}_0, \mathbf{f}; \theta)$, a masked Nash game for agent i is denoted as

[1]Prior work [1, 2] has shown that even in simple linear-quadratic games, computation time scales cubically with the total number of state and control variables for all agents, rendering game-theoretic planners impractical for large-scale use.

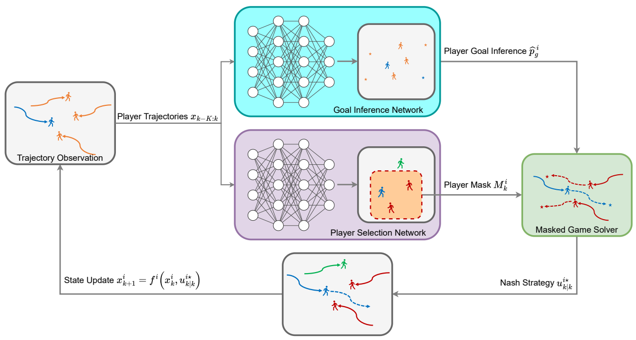

Figure 1: Overview of our game-theoretic prediction and planning framework via the Player Selection Network (PSN) and the Goal Inference Network (GIN).

$\Gamma^i(\tilde{\mathbf{x}}_0, \tilde{\mathbf{f}}; \tilde{\theta})$, where we define the set of unmasked agents (agents that are included in the game) $U^i := \{j \neq i \mid m^{ij} = 1\}$ and let

$$\tilde{\theta} := (\theta^j)_{j \in U^i}, \quad \tilde{\mathbf{x}}_k := (x_k^j)_{j \in U^i}, \quad \tilde{\mathbf{f}} := (f^j)_{j \in U^i},$$
$$\tilde{c}_{s,k}^i(\mathbf{x}_k, \mathbf{u}_k; \theta^i) := \sum_{j \in U^i} c_{s,k}^{ij}(\mathbf{x}_k, \mathbf{u}_k; \theta^i). \tag{2}$$

The masked game preserves only the specific parameters and states corresponding to those agents most relevant to agent i.

3 PSN GAME

Player Selection Network (PSN). We propose the Player Selection Network (PSN) to infer a mask M^i which balances performance with sparsity. Based on the available information, we present two variants of the network: PSN-Full, whose input is all agents' past states $\mathbf{x}_{0:K}$, and PSN-Partial , whose input is only a partial observation of state, i.e. $\{h(\mathbf{x}_k)\}_{k=0}^K$ for a known function h. Both variants' output is the inferred mask M^i.

Loss Function Design. We train two PSN variants: PSN-Prediction and PSN-Planning for prediction and planning tasks, respectively; their corresponding training loss functions are structured as:

$$\begin{aligned} L_{\text{Pred}} &= L_{\text{Binary}} + \sigma_1 L_{\text{Sparsity}} + \sigma_2 L_{\text{Similarity}} \\ L_{\text{Plan}} &= L_{\text{Binary}} + \sigma_3 L_{\text{Sparsity}} + \sigma_4 L_{\text{Cost}} \end{aligned} \tag{3}$$

where $L_{\text{Binary}} := \frac{1}{N} \sum_{j=1}^N m^{ij}(1 - m^{ij})$ encourages mask m^{ij} to converge to either 0 or 1. $L_{\text{Sparsity}} := \frac{\|M^i\|_1}{N}$ encourages agent i to consider fewer other agents. $L_{\text{Similarity}} := \sum_k \|h(\hat{\mathbf{x}}_k^i) - p_k^i\|_2$ encourages agent i to consider agent subsets which lead to Nash solutions that recover its observed trajectory, and $L_{\text{Cost}} := J^i(\hat{\mathbf{x}}^i; \mathbf{x}^{\neg i})$ encourages agent i to consider appropriate agents to minimize its game cost.

Goal Inference Network (GIN). To employ the PSN in more complicated scenarios where the the game information θ^i (in this work we refer θ^i as agent i's 2-D goal position) is incomplete, we introduce a data-driven goal inference network G_ϕ, which effectively infers the agent's goal $\hat{p}_g^i$ from the (partial) observation of their past trajectories $\{h(\mathbf{x}_k)\}_{k=0}^K$. The network is trained from the dataset $\mathcal{D}$, which contains the Nash equilibrium trajectories of all agents from solving a game with ground truth goals, by minimizing the mean goal inference error $L_{\text{Goal}} := \frac{1}{|\mathcal{D}| \cdot N} \sum_{d \in \mathcal{D}} \sum_{i \in [N]} \|p_{g,\text{ref}}^i - G_\phi(\mathbf{x}_{0:K})\|$ for each sample d. Assisted by G_ϕ, the PSN is readily adaptable to trajectory prediction and ego-centric planning tasks where the agents' goals are not known to the predictor/planner, by constructing and solving a masked game $\Gamma^i(\tilde{\mathbf{x}}_k, \tilde{\mathbf{f}}; \{\hat{p}_g^i\}_{i=1}^N, M_k^i)$ parametrized by the inferred goals.

Receding Horizon Application. We integrate the player selection network with a (differentiable) Nash game solver [5] in the receding time horizon for prediction and planning tasks (Algorithm 1).

Algorithm 1 Receding Horizon **Planning** via PSN

Input: PSN-Planning (Full, Partial), Goal Inference Network G_ϕ, receding horizon masked game $\Gamma^i(\tilde{\mathbf{x}}_k, \tilde{\mathbf{f}}; \{p_g^i\}_{i=1}^N, M_k^i)$, observation interval K, ego agent i's goal p_g^i.
Output: Agent i's receding horizon strategy $\mathbf{u}_{\text{RH}}^{i*}$.

```
// Infer the other agents' goals via G_ϕ.
{p̂_g^j}_{j=1, j≠i}^N ← G_ϕ({h(x_k)}_{k=0}^K) if {p_g^j}_{j=1, j≠i}^N not known,
    else {p̂_g^j}_{j=1}^N ← {p_g^j}_{j=1}^N.
while k ≥ K and player i not reaching his goal p_g^i do
    // Identify key agents via PSN.
    M_k^i ← PSN-Full(x_{k−K:k}) or PSN-Partial(p_{k−K:k}).
    // Solve Masked game for agent i.
    u_{k|k}^{i*} ← solving Γ_k^i(x̃_k, f̃; {p̂_g^i}_{i=1}^N, M_k^i).
    // Update agent i's state based on game dynamics.
    x_{k+1}^i ← f^i(x_k^i, u_{k|k}^{i*}) based on dynamics.
    k ← k + 1.
end while
```

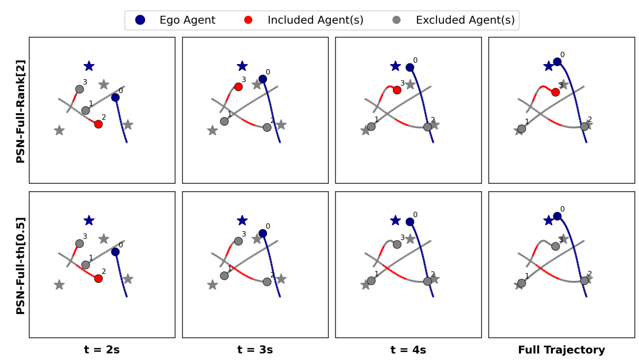

Figure 2: Trajectory visualization for PSN-Full-Rank (Top), PSN-Full-Threshold (Bottom) in a 4-agent planning scenario.

4 RESULTS AND CONCLUSIONS

We tested our PSN Game in a 4-agent navigation scenario (Fig 2). At each stage, PSN identifies the key agents (red) to the ego-agent (blue) and wipes out the less important ones (grey), enabling the ego agent to speed up in computation while still keeping safe during planning.

Conclusion. PSN Game offers two key advantages over prior selection methods. i) Flexible data usage: relying only on past trajectory data. It also eliminates the need for online parameter tuning. ii) State-of-the-art performances: by leveraging spatio-temporal patterns in past trajectories, PSN outperforms existing baseline selection methods in various metrics in multiple tasks.

More Results. For more approach details and empirical results in prediction and planning tasks with more agents, please refer to our technical report [4].

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under grants 2336840 and 2409535, and by the Army Research Laboratory under cooperative agreements W911NF-23-2-0011 and W911NF-25-2-0021.

REFERENCES

[1] David Fridovich-Keil, Ellis Ratner, Lasse Peters, Anca D Dragan, and Claire J Tomlin. 2020. Efficient iterative linear-quadratic approximations for nonlinear multi-player general-sum differential games. In *2020 IEEE international conference on robotics and automation (ICRA)*. IEEE, 1475–1481.

[2] Forrest Laine, David Fridovich-Keil, Chih-Yuan Chiu, and Claire Tomlin. 2023. The computation of approximate generalized feedback nash equilibria. *SIAM Journal on Optimization* 33, 1 (2023), 294–318.

[3] Tianyu Qiu and David Fridovich-Keil. 2024. Inferring occluded agent behavior in dynamic games from noise corrupted observations. *IEEE Robotics and Automation Letters* 9, 12 (2024), 11489–11496.

[4] Tianyu Qiu, Eric Ouano, Fernando Palafox, Christian Ellis, and David Fridovich-Keil. 2025. PSN Game: Game-theoretic Prediction and Planning via a Player Selection Network. *arXiv preprint arXiv:2505.00213* (2025).

[5] Max Muchen ["Sun"]. 2025. *LQRax: JAX-enabled continuous-time LQR solver.* https://github.com/MaxMSun/lqrax