Bayes-Adaptive Monte-Carlo Planning for Type-Based Reasoning in Large Partially Observable, Multi-Agent Environments

Extended Abstract

Jonathon Schwartz
The Australian National University
Canberra, Australia
jonathon.schwartz@anu.edu.au

Hanna Kurniawati
The Australian National University
Canberra, Australia
hanna.kurniawati@anu.edu.au

ABSTRACT

Designing autonomous agents that can interact effectively with other agents without prior coordination is an important problem in multi-agent systems. Type-based reasoning methods achieve this by maintaining a belief over a set of potential behaviours for the other agents. However, current methods are limited in that they assume full observability of the environment or do not scale efficiently to larger problems with longer planning horizons. Addressing these limitations, we propose Bayes-Adaptive Partially Observable Stochastic Game Monte-Carlo Planning (BAPOSGMCP) – a scalable online planner for Type-based reasoning in partially observable environments – which combines Monte-Carlo Tree Search with a novel meta-policy for selecting the best policy to guide search during planning. Through comprehensive evaluations we demonstrate that BAPOSGMCP is able to effectively adapt online to diverse sets of agents in large cooperative, competitive and mixed environments with up to 10^{14} states and 10^8 observations.

KEYWORDS

POSG; Type-Based Reasoning; Monte-Carlo Planning; Meta-Policy

ACM Reference Format:
Jonathon Schwartz and Hanna Kurniawati. 2023. Bayes-Adaptive Monte-Carlo Planning for Type-Based Reasoning in Large Partially Observable, Multi-Agent Environments: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

1 INTRODUCTION

A core research area in multi-agent systems is the design of agents that can interact effectively with other agents without prior coordination [2, 8, 21]. Type-based reasoning methods give agents this ability by maintaining a belief over a set *types* for the other agents [1, 4, 6, 7]. Each type is a mapping from the agent's interaction history to a probability distribution over actions, and completely specifies the agent's behaviour. If the set of types is sufficiently representative, type-based reasoning can lead to fast adaptation and effective interaction without prior coordination [3, 6].

Unfortunately, type-based reasoning introduces significant complexity into the decision making problem and finding scalable and efficient solution methods remains a key challenge. This is especially true in partially observable environments where the planning agent must reason about the type of the other agent, their interaction history, and the state of the environment, all at the same time. Several online planning methods based on Monte-Carlo Tree Search (MCTS) have shown promising performance in non-trivial partially observable problems [9, 12, 17]. However, so far these methods have only been demonstrated in settings where the other agent's type is known and in settings requiring relatively short planning horizons.

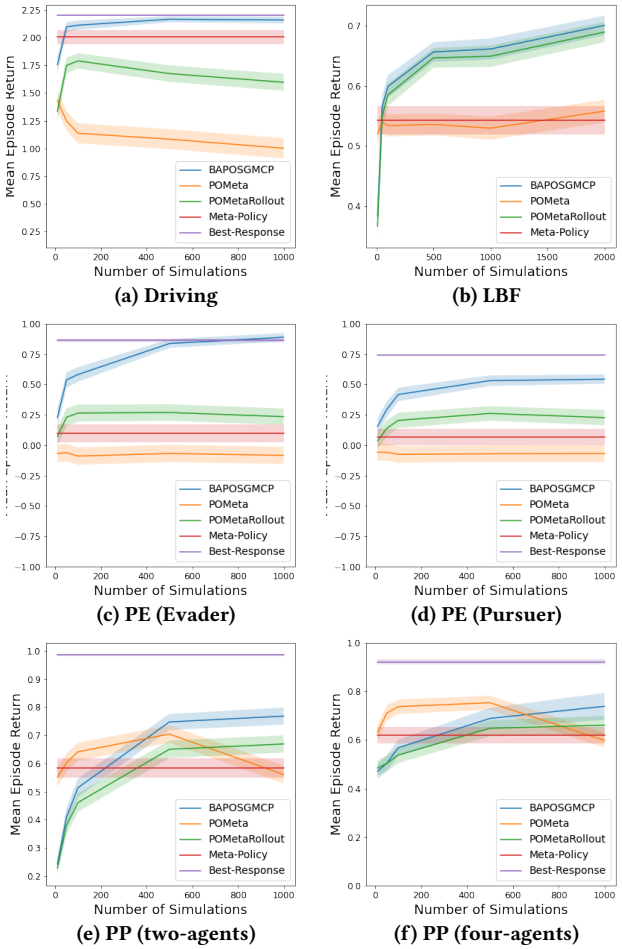
In this work we propose Bayes-Adaptive Partially Observable Stochastic Game Monte-Carlo Planning (BAPOSGMCP), an online planning algorithm for type-based reasoning in partially observable environments. BAPOSGMCP extends the PUCT algorithm [19], that uses a search-policy for guiding search, to the partially observable setting. For the search-policy, we introduce a novel meta-policy which is robust to the set of types of the other agents and is efficient to compute. We evaluate the proposed method on large competitive, cooperative, and mixed partially observable environments - the largest of which has four agents and on the order of 10^{14} states and 10^8 observations - and demonstrate that it is able to rapidly adapt and interact effectively without explicit prior coordination in complex environments.

2 BAPOSGMCP

We model the problem of type-based reasoning as a Partially Observable Stochastic Game (POSG) [11] where N agents act simultaneously in an environment. Each agent $i \in 1, ..., N$ acts according to their *policy* π_i, which is a mapping from their history h_i to a probability distribution over their actions a_i, and is equivalent to an agent's *type*. We are interested in finding a policy for the planning agent, denoted i, that maximizes it's expected sum of rewards, assuming that all other agents, collectively denoted $-i$, are using policies from a known fixed set of policies Π according to a distribution over this set ρ.

Our goal in this work is to find a scalable and efficient planning algorithm for our problem setting. To this end we present BAPOSGMCP. Like existing planners [9, 12, 17], BAPOSGMCP uses MCTS to calculate the planning agent's best action from its current belief b_i. However, it offers several important improvements over existing algorithms. Firstly, it incorporates the PUCT algorithm [19] for selecting actions during search, which can significantly improve planning efficiency by biasing search towards the most relevant actions according a *search policy*. This makes it possible to plan for longer horizons, as well as offers improved integration of the search-policy's value function for leaf node evaluation. To address the limitation of PUCT, namely that it relies on access to a good search-policy, the second improvement offered by BAPOSGMCP is

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.



(a) Driving (b) LBF

(c) PE (Evader) (d) PE (Pursuer)

(e) PP (two-agents) (f) PP (four-agents)

Figure 1: Mean episode return of BAPOSGMCP and baseline methods. Shaded areas show the 95% CI.

the use of a novel meta-policy as the search-policy. Our proposed meta-policy has the advantage that it can be efficiently generated from the policy set Π, and offers a robust prior since it considers performance across the entire set of other agent policies.

The proposed meta-policy σ_i is a mapping from the set of other agent joint policies to a distribution over the set of valid policies for the planning agent $\sigma_i : \Pi \rightarrow \Delta(\Pi_i)$, so that $\sigma_i(\pi_{i,k}|\pi_{-i,m}) = Pr(\pi_{i,k}|\pi_{-i,m})$ for $\pi_{i,k} \in \Pi_i, \pi_{-i,m} \in \Pi$. Where the set Π_i is the set of all individual policies for any of the other agents from the set Π. To generate the meta-policy we use an empirical game [13, 22, 23], which efficiently constructs an estimate of each policy's performance against each other policy in the set of policies Π. The meta-policy selects the policy from the set Π_i that maximizes performance against a given policy for the other agent, according to the empirical game's estimate.

We incorporate the meta-policy into MCTS to create BAPOSGMCP, which extends the POMCP [20] algorithm to planning with beliefs over history-policy-states and uses PUCT [16, 19] and the meta-policy for selecting actions from each belief during search. in BAPOSGMCP each belief is a distribution over the other agents' histories h_{-i}, their policies $\pi_{-i} \in \Pi$, and the environment state s. This transforms the problem into a type of POMDP [10], and allows us to apply MCTS based belief-tree planning to the problem. We improve the efficiency of planning by using the meta-policy to guide search via the PUCT algorithm. The meta-policy selects the policy to guide planning $\pi_i \in \Pi_i$ based on the planning agent's belief about the other agent's policy π_{-i}.

3 EXPERIMENTS

We evaluated BAPOSGMCP against baseline methods on one cooperative (Predator-Prey (PP) [15]), one competitive (Pursuit-Evasion (PE)[17, 18]), and two mixed (Driving [14], and Level-Based-Foraging (LBF) [3, 5]) environments. These environments add additional complexities to existing benchmarks [9, 12] and were chosen in order to assess BAPOSGMCP's ability across a range of domains that required both planning over many steps and reasoning about the other agent's behaviour. For each environment, we created a diverse set of policies Π which was used for the other agent policies during evaluations and also for the meta-policy σ_i and policy prior ρ. We compared BAPOSGMCP against a number of baselines, two of which are indicative of upper and lower bounds on the performance of BAPOSGMCP, while the other two test the benefits of different components of our approach.

We found that for all environments the performance of BAPOSGMCP improved with the number of simulations and given enough simulations BAPOSGMCP equaled or outperformed all non-upper bound baselines across all environments (Figure 1). Furthermore, in the Driving and PE (Evader) problems the performance converged towards the Best-Response upper-bound, suggesting BAPOSGMCP converges towards Bayes-optimal performance as the number of simulations increased. The importance of the different aspects of BAPOSGMCP- beliefs, search, and search tree - varied by environment, however using all three lead to overall best performance given enough planning time. The most significant improvements over the baselines were found in the PE (Evader) problem, which is the problem that required the longest horizon planning. Indicating the benefit of our approach for problems requiring longer planning look-ahead.

4 CONCLUSION

In this work we presented a scalable planning method for type-based reasoning in large partially observable environments. Our algorithm, BAPOSGMCP, offers two key contributions over existing planners. The first is the use of PUCT for action selection during search. The second is a new meta-policy which is used to guide the search. Through extensive evaluations we demonstrate BAPOSGMCP's ability to effectively adapt online to diverse sets of agents in large cooperative, competitive and mixed environments. Multiple avenues for future research exist, including extending BAPOSGMCP to handle continuous actions and observations, along with exploring alternative constructions of the meta-policy [13].

Code and Paper

The full paper and code are available at https://github.com/Jjschwartz/ba-posgmcp.

ACKNOWLEDGMENTS

This work is supported by an AGRTP Scholarship and the ANU Futures Scheme.

REFERENCES

[1] Stefano V. Albrecht, Jacob W. Crandall, and Subramanian Ramamoorthy. 2016. Belief and Truth in Hypothesised Behaviours. *Artificial Intelligence* 235 (2016), 63–94.
[2] Stefano V. Albrecht, Somchaya Liemhetcharat, and Peter Stone. 2017. Special Issue on Multiagent Interaction without Prior Coordination: Guest Editorial. *AAMAS* 31, 4 (2017), 765–766.
[3] Stefano V. Albrecht and Subramanian Ramamoorthy. 2013. A Game-Theoretic Model and Best-Response Learning Method for Ad Hoc Coordination in Multiagent Systems. In *AAMAS*. 1155–1156.
[4] Stefano V. Albrecht and Subramanian Ramamoorthy. 2014. On Convergence and Optimality of Best-Response Learning with Policy Types in Multiagent Systems. In *UAI*. 12–21.
[5] Stefano V. Albrecht and Peter Stone. 2017. Reasoning about Hypothetical Agent Behaviours and Their Parameters. In *AAMAS*. 547–555.
[6] Samuel Barrett and Peter Stone. 2015. Cooperating with Unknown Teammates in Complex Domains: A Robot Soccer Case Study of Ad Hoc Teamwork. In *AAAI*, Vol. 29.
[7] Samuel Barrett, Peter Stone, and Sarit Kraus. 2011. Empirical Evaluation of Ad Hoc Teamwork in the Pursuit Domain. In *AAMAS*. 567–574.
[8] Michael Bowling and Peter McCracken. 2005. Coordination and Adaptation in Impromptu Teams. In *AAAI*, Vol. 5. 53–58.
[9] Adam Eck, Maulik Shah, Prashant Doshi, and Leen-Kiat Soh. 2020. Scalable Decision-Theoretic Planning in Open and Typed Multiagent Systems. In *AAAI*, Vol. 34. 7127–7134.
[10] Piotr J. Gmytrasiewicz and Prashant Doshi. 2005. A Framework for Sequential Planning in Multi-Agent Settings. *JAIR* 24 (2005), 49–79.
[11] Eric A. Hansen, Daniel S. Bernstein, and Shlomo Zilberstein. 2004. Dynamic Programming for Partially Observable Stochastic Games. In *AAAI*. 709–715.
[12] Anirudh Kakarlapudi, Gayathri Anil, Adam Eck, Prashant Doshi, and Leen-Kiat Soh. 2022. Decision-Theoretic Planning with Communication in Open Multiagent Systems. In *UAI*. 938–948.
[13] Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Pérolat, David Silver, and Thore Graepel. 2017. A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning. *NeurIPS* 30 (2017).
[14] Adam Lerer and Alexander Peysakhovich. 2019. Learning Existing Social Conventions via Observationally Augmented Self-Play. In *AAAI/ACM Conference on AI, Ethics, and Society*. 107–114.
[15] Ryan Lowe, Yi I. Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. *NeurIPS* 30 (2017).
[16] Christopher D. Rosin. 2011. Multi-Armed Bandits with Episode Context. *Annals of Mathematics and Artificial Intelligence* 61, 3 (2011), 203–230.
[17] Jonathon Schwartz, Ruijia Zhou, and Hanna Kurniawati. 2022. Online Planning for Interactive-POMDPs Using Nested Monte Carlo Tree Search. In *IROS*. 8770–8777.
[18] Iris Rubi Seaman, Jan-Willem van de Meent, and David Wingate. 2018. Nested Reasoning About Autonomous Agents Using Probabilistic Programs. *arXiv preprint arXiv:1812.01569* (2018). arXiv:1812.01569
[19] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, and Thore Graepel. 2018. A General Reinforcement Learning Algorithm That Masters Chess, Shogi, and Go through Self-Play. *Science* 362, 6419 (2018), 1140–1144.
[20] David Silver and Joel Veness. 2010. Monte-Carlo Planning in Large POMDPs. *NeurIPS* 23 (2010), 2164–2172.
[21] Peter Stone, Gal A. Kaminka, Sarit Kraus, and Jeffrey S. Rosenschein. 2010. Ad Hoc Autonomous Agent Teams: Collaboration without Pre-Coordination. In *AAAI*, Vol. 24. 1504–1509.
[22] William E. Walsh, Rajarshi Das, Gerald Tesauro, and Jeffrey O. Kephart. 2002. Analyzing Complex Strategic Interactions in Multi-Agent Systems. In *AAAI Workshop on Game-Theoretic and Decision-Theoretic Agents*. 109–118.
[23] Michael P. Wellman. 2006. Methods for Empirical Game-Theoretic Analysis. In *AAAI*. 1552–1556.