GENERAL SUM STOCHASTIC GAMES WITH NETWORKED INFORMATION FLOW

Anonymous authors
Paper under double-blind review

ABSTRACT

Inspired by naturally emerging networks of reinforcement learning problems in operation sectors such as supply chains, epidemics and social networks, we present a stochastic game formulation with networked reward and transition dependencies for multi-agent reinforcement learning (MARL). Despite the rapid development of MARL algorithms, the majority of research efforts are motivated by settings where each pair of players is either collaborating or fully competing. However, in many large-scale networked systems, pairs of interacting players experience general sum-type objectives— players must collaborate for total profit maximization and locally compete for individual profit maximization. In contrast to popular MARL benchmark environments, stochastic games with general sum objectives have received relatively less attention, in part due to the lack of motivating application and toy examples. In this paper, we address the lack of motivating examples by presenting a networked stochastic game framework with pair-wise general sum objectives and relating the framework to operation sector systems such as supply chains. We discuss specific game features that distinguish this type of stochastic game from existing canonical stochastic games models, and specifically outline the impact of these features on the adaptation of popular multi-agent learning paradigms such as individual learning and centralized learning decentralized execution. We conclude with a two player supply chain to benchmark existing MARL algorithms and to contextualize the challenges at hand.

1 INTRODUCTION

With increasing population concentration in metropolitan regions around the world, operation sectors such as supply chain logistics (Baryannis et al., 2019), power grids (Zhang et al., 2018), and transportation (Haydari & Yilmaz, 2020) are collectively seeing the emergence of large, interconnected networks of self-interested decision-makers. In many of these networked systems, the efficiency of the overall network depends on the collective behavior of the self-interested participants. As machine learning techniques advance, the *components* of these large-scale networks turn to reinforcement learning approaches to optimize their operation. Typically, each component treats the environment as stochastic and uses *local* observations, insights, and rewards to optimize their individual decisions. The result is a network of competing reinforcement learning algorithms each with limited, sparse communication to the rest of the network and a lack of awareness of competing agents and their effect on the local algorithmic performance (Zhang et al., 2021).

MARL has be shown to be empirically successful in applications such as multi-robotic coordination, autonomous driving, among others. However, characterization of the *types* of multi-agent problems rarely extend beyond purely cooperative or purely competitive. On the other hand, problems in game theory are categorically grouped by information structure, reward structure and player dynamics into a rich taxonomy of game types (Başar & Olsder, 1998). Differentiating between game classes not only assists the classification of different approaches, but also enables better targeted algorithm development (Hopkins, 1999; Mazumdar et al., 2020). Motivated by the competitive reinforcement learning problem that models many operation sector systems, we propose a type stochastic games that explores more complex general sum objectives than the standard MARL examples (Terry et al., 2020; Baker et al., 2019).

In this paper, we propose both the networked stochastic game model and a high fidelity implementation of a supply chain to highlight the key features and challenges associated with this game setting. Within the supply chain game example, we demonstrate how the general sum player relationships differ from player relationships in existing models such as Hide and Seek (Baker et al., 2019), and highlight information access challenges that may prevent the successful implementation of state-of-art MARL algorithms to networked systems. By building our model around the key features common to many network systems in the operation sector, we hope to provide a base from which reinforcement learning algorithms can better cater to these types of applications. Finally, we summarize shortcomings in current solution approaches with respect to networked stochastic games as well as opportunities that can be exploited to improve algorithmic performance.

1.1 Related work

The dynamics of our networked stochastic game is similar to Qu et al. (2020), with the distinction that we model continuous state-action spaces and assume each player is pursuing independent objectives instead of a centralized reward. In the continuous state-action domain, extensions of both deep deterministic policy gradient (DDPG) (Lillicrap et al., 2015) and proximal policy optimization (PPO) (Schulman et al., 2017) have had the greatest success in the multi-agent setting. Notable algorithms include MADDPG Lowe et al. (2017), counterfactual multi-agent policy gradient Foerster et al. (2018), and multi-agent PPO (MAPPO) Yu et al. (2021). However, both counterfactual multi-agent policy gradient and MAPPO are designed for centralized rewards, and MADDPG was not applied to games where the pair-wise interaction between players is general sum.

2 Networked stochastic games

We consider a group of N players, each seeking to find an optimal policy for a Markov Decision Process (MDP) that is coupled with its competitors. The coupling between agents is specified by a graph $\mathcal{G} = ([N], \mathcal{E})$ where $[N] := \{1, \dots, N\}$. The vertices of $\mathcal{G}$ correspond to the players and an edge $(i, j) \in \mathcal{E}$ exists if players i and j are neighbors. The edges may be directed. The set of all neighbors conected to player i is given by $\mathcal{N}_i$.

More specifically, each player $i \in [N]$ seeks to find an optimal policy for the MDP $(\mathcal{S}_i, \mathcal{A}_i, R_i, P_i, \gamma_i)$ where each of the elements in the tuple are characterized as follows:

1. The space of internal states for player i is $\mathcal{S}_i \subseteq \mathbb{R}^{S_i}$.
2. The space of available actions to player i is denoted $\mathcal{A}_i \subseteq \mathbb{R}^{A_i}$. The action set $\mathcal{A}_i$ is state-independent, such that every action $a_i \in \mathcal{A}_i$ can be chosen from every $s_i \in \mathcal{S}_i$.
3. Player i's transition kernel is denoted $P_i : \mathcal{S}_i \times \mathcal{S}_i \times \mathcal{A}_i \times \mathcal{D}_i \mapsto [0, 1]$ and satisfies $\int_{\mathcal{S}_i} P_i(s, s', a; d)\partial s = 1, \forall\, d \in \mathcal{D}_i, a \in \mathcal{A}_i$, where $\mathcal{D}_i = \{(s_j, a_j) \in \mathcal{S}_j \times \mathcal{A}_j \mid \forall j \in \mathcal{N}_i\}$ is the set pf state and actions of i's neighboring players. $P_i(s_i, s_i', a_i; d_i)$ is player i's conditional probability of transition from state s_i' to s_i by taking action a_i if the neighboring players are in state-action d_i.
4. $R_i : \mathcal{S}_i \times \mathcal{A}_i \times \mathcal{D}_i \mapsto \mathbb{R}$ is the reward function and may be stochastic.
5. $\gamma_i \in (0, 1)$ is player's discount factor in time.

Each player chooses a stationary policy $\pi_i : \mathcal{S}_i \mapsto \rho(\mathcal{A}_i)$, where $\rho(\mathcal{A}_i)$ denotes the set of probability distribution over $\mathcal{A}_i$. The actions of player i's opponents is collectively denoted as $a_{-i} = (a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_N)$ and their policies denoted as $\pi_{-i} = (\pi_1, \dots, \pi_{i-1}, \pi_{i+1}, \pi_N)$.

Players use their policy π_i to maximize their infinite horizon expected return, conditioned on the stationary joint opponent policies π_{-i}, the initial state $s(0) = [s_1(0) \quad \dots \quad s_N(0)]$, and defined as follows:

$$J_i(\pi_i, \pi_{-i}) = \mathbb{E}_{s(0)}^{(\pi_i, \pi_{-i})}\Big[\sum_{t=0}^{\infty} \gamma_i^t R_i\Big(s_i(t), a_i(t), d_i(t)\Big)\Big]. \tag{1}$$

2.1 NASH EQUILIBRIUM

In a networked stochastic game, players unilaterally optimize their expected return J_i (1) using their own policy π_i. Players reach a *Nash equilibrium* when every player has optimized their own reward with respect to the opponents' joint policy.

Definition 1 (Nash equilibrium) *A joint policy $\pi^\star = (\pi_1^\star, \ldots, \pi_N^\star)$ is a Nash equilibrium if for all i, $\pi_i^\star \in \operatorname{argmax}_{\pi_i \in \Pi_i} J(\pi_i, \pi_{-i})$.*

3 SUPPLY CHAINS

Supply chains exemplify the type of systems that networked stochastic games model. We develop a detailed model here for a single-commodity supply chain with N chain components, i.e. the players. In AppendixA, we present additional detailed examples of networked stochastic game models for epidemics and competitive opinion dynamics in social networks.

We assume that each player is a selfish enterprise that is unilaterally trying to earn the largest share of the overall supply chain profit. However, all players must coordinate to turn raw material into profitable consumer goods. Players order raw material from its upstream suppliers and sells finished products to its downstream retail venues. We assume that each player experiences some lead time between ordering and receiving products and is aware that their listed product prices changes the market demand. Our mathematical model of player dynamics is equivalent to a price-endogenous, multi-period newsvendor with lead time in operations research (Dana Jr & Petruzzi, 2001).

3.1 STATES AND ACTIONS

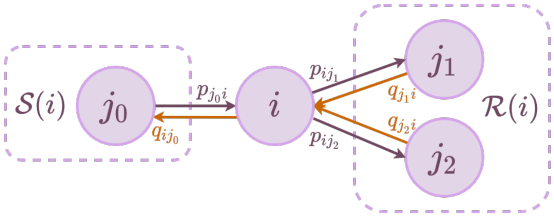


Figure 1: Player i and its neighbors in a supply chain game.

We consider a directed graph $\mathcal{G} = ([N], \mathcal{E})$ modeling the supply chain. The node set $[N]$ corresponds to the set of players and the edge set $\mathcal{E}$ corresponds to the supplier-retailer relationships. A directed edge exists from i to j if player i is player j's supplier and if player j is player i's retailer. Player i's suppliers collectively form the set $\mathcal{S}(i)$ and player i's retailers collectively form the set $\mathcal{R}(i)$.

State. Player i's internal state is $s_i = [c_i, \mu_i, x_i, y_i] \in \mathbb{R}_+^{|\mathcal{S}(i)|+|\mathcal{R}(i)|+\ell_i+1}$. The cost of purchasing unit product from i's suppliers is $c_i \in \mathbb{R}_+^{|\mathcal{S}(i)|}$. The anticipated demand from player i's retailers is $\mu_i \in \mathbb{R}_+^{|\mathcal{R}(i)|}$. The current stock level is $x_i \in \mathbb{R}_+$ and the incoming stock replenishment in the next ℓ_i time steps is $y_i \in \mathbb{R}_+^{\ell_i}$. At time t, $[y_i(t)]_n$ is the replenishment that arrives at time $t+n$.

Action. Player i chooses the quantity of raw material to buy and the price per unit product to sell, denoted as $[q_i, p_i] \in \mathbb{R}_+^{2N}$. When $j \in \mathcal{S}(i)$, q_{ij} is the quantity i orders from j, otherwise $q_{ij} = 0$. When $k \in \mathcal{R}(i)$, p_{ik} is the unit product price i offers to k, otherwise $p_{ik} = 0$.

We define some auxiliary variables to aid the definition of transition and rewards. The total demand player i receives is given by $\sum_{j\in[N]} q_{ji}$. The *gap* between total demand and player i's stock is given by $w_i = \max\{\sum_{j\in[N]} q_{ji} - x_i, 0\}$. When player i's total demand exceeds available stock, w_i is positively and will be evenly distributed among player i's retailers. Let d_{ij} be the *realized* units of products that i delivers to j. d_{ij} is computed as follows:

$$d_{ij} = \begin{cases} \max\left\{q_{ji} - \frac{w_i}{|\mathcal{R}(i)|}, 0\right\}, & x_i \geq \sum_{j\in[N]} q_{ji} \\ q_{ji} & x_i < \sum_{j\in[N]} q_{ji} \end{cases}, \quad \forall j \in \mathcal{R}(i).$$

3.2 STATE TRANSITIONS

Price and anticipated demand. At time t, the raw material price $c_i(t)$ is directly set by player i's suppliers and is equivalent to $\{p_{ji}(t),\ j \in \mathcal{S}(i)\}$. The anticipated demand $\mu_i(t)$ is computed by player i based on historical demand, $\mu_{ij}(t) = f_i(d_{ji}(0), d_{ji}(1), \ldots, d_{ji}(t-1))$, where f_i is a forecasting method employed by player i. Possible forecasting techniques are ARMAX (Thomassey, 2010) and machine learning (Bontempi et al., 2012).

Stock level and replenishment. The stock level at $t+1$ is given by $x_i(t+1) = \big(x_i(t) - \sum_{j\in\mathcal{S}(i)} d_{ji}(t)\big)_+ + [y_i(t)]_1$. The incoming stock replenishment transitions as $[y_i(t+1)]_k = [y_i(t)]_{k+1}$, for $k = 1, \ldots, \ell - 1$. The last incoming replenishment is the sum of all realized demands from player i's suppliers, $[y_i(t+1)]_{\ell_i} = \sum_{j\in\mathcal{S}(i)} d_{ji}(t)$.

3.3 REWARDS

Each player's reward is the total revenue subtracted by operation expenditure and material cost. Operation expenditure includes holding cost for current stock and loss of goodwill cost for unmet demands. We assume that the holding cost scales linearly with $h_i \in \mathbb{R}_+$ and the leftover stock level after transition. The loss of goodwill cost scales linearly with $w_i \in \mathbb{R}_+$ and the total unmet demand.

$$R_i(t) = \underbrace{\sum_{j\in\mathcal{R}(i)} p_{ij}(t)d_{ij}(t)}_{\text{Total Revenue}} - \underbrace{\sum_{k\in\mathcal{S}(i)} c_{ik}(t)d_{ki}(t)}_{\text{Total Cost}} + \underbrace{h_i\Big(x_i(t) - \sum_{j\in\mathcal{R}(i)} d_{ij}(t)\Big)_+}_{\text{Holding Cost}} + \underbrace{w_i\Big(\sum_{j\in\mathcal{R}(i)} d_{ij}(t) - x_i(t)\Big)_+}_{\text{Loss of Goodwill Cost}}. \quad (2)$$

3.4 PAIR-WISE GENERAL SUM COMPETITION

We can demonstrate the general sum nature of the supply chain games using a simple two player game. Suppose a two player supply chain game has zero lead time, raw materials prices $P(q_0)$, and consumer demands $Q(p_1)$. Players are configured as shown in Figure 2.

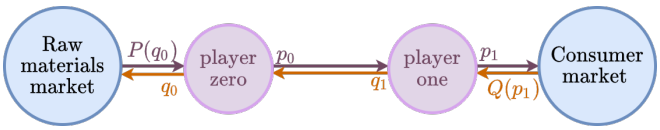


Figure 2: Two player single commodity supply chain.

With zero lead time, player rewards (2) simplify to rewards become

$$R_1 = p_1Q(p_1) - p_0d_{10} + h_1[s_1 - Q(p_1) + d_{10}]_+ + w_1[Q(p_1) - s_1]_+, \quad (3)$$

$$R_0 = p_0d_{10} - P(q_0)q_0 + h_0[s_0 - d_{10} + q_0]_+ + w_0[q_1 - d_{10}]_+, \quad (4)$$

where $d_{10} = \min\{q_1, s_0\}$ is the realized quantity transfer between players zero and one. The *zero sum component* of the rewards is p_0d_{10}; player zero tries to maximize it while player one tries to minimize it. The collaborative component is less obvious: when $h_0 + w_0 << p_0$, and $p_1 << h_1$, both players' rewards will increase if d_{10} increases.

This type of network-induced, mixed competitive-collaborative environment is not well addressed in multi-agent learning. Most MARL frameworks with competitive elements focus on teams of players where the team objectives directly conflict with other teams' objectives and players within a team share identical objectives (Terry et al., 2020), which is in fact a general sum game under simplifying assumptions. Whereas in networked stochastic games, each pair of neighboring players have objectives that are partially competitive and partially collaborative with one another. Due to its vast applicability to many networked systems in operation sectors, network stochastic games with pair-wise general-sum objectives deserve greater attention from the research community.

3.5 EFFECT OF INFORMATION AVAILABILITY ON MULTI-AGENT ACTOR CRITIC

Information availability is crucial to the applicability of MARL algorithms to networked stochastic games and has strong implications for the training and execution methods. We adapt the multi-agent actor-critic framework and describe three scenarios of information availability that arise naturally from supply chain and MARL paradigms.

3.5.1 SINGLE AGENT ACTOR CRITIC

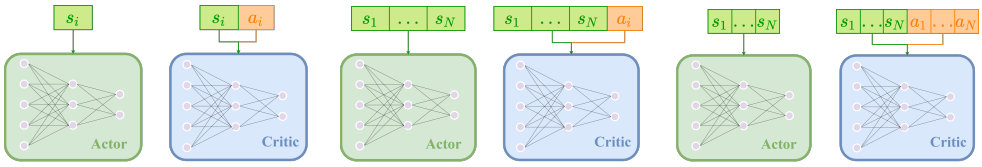
We first summarize single-agent actor critic methods to facilitate our discussion on the effect of information availability on multi-agent actor critic methods. Actor critic methods are policy gradient algorithms that solve a single agent MDP $(\mathcal{S}, \mathcal{A}, R, P, \gamma)$ by iteratively estimating the policy π, and the expected reward under a stationary π, defined as $Q^{\pi}(s,a) = \mathbb{E}^{\pi}_{s(0)}\left[\sum_{k=t}^{T} \gamma^t R(s(t), a(t))\right]$. The policy is approximated by an *actor* from a family of functions $\pi(\cdot, \theta) \mapsto \rho(\mathcal{A})$ parametrized by $\theta \in \mathbb{R}^{M_a}$, and the expected reward is approximated by a *critic* from a family of functions $\hat{Q}(\cdot, \cdot; \omega) \mapsto \mathbb{R}$ parametrized by ω. From initial parameter values $(\theta(0), \omega(0))$, actor critic gradients approximate the optimal policy $\pi^{\star}$ and the expected reward $Q^{\pi^{\star}}(s,a)$ by performing gradient steps as

$$\theta(t+1) = \theta(t) + \alpha \mathbb{E}^{\pi(\cdot,\theta)}\left[Q^{\pi(\cdot,\theta)}(s,a)\right]\Big|_{\theta=\theta(t)}, \tag{5}$$

$$\omega(t{+}1) = \omega(t){+}\beta \mathbb{E}^{\pi(\cdot;\theta(t))}\left[\left(R(s,a) + \gamma \mathbb{E}^{\pi(\cdot;\theta(t+1))}[\hat{Q}(s',a',\omega(t))] - \hat{Q}(s,a,\omega(t))\right) \nabla_{\omega} \hat{Q}(s,a;\omega(t))\right], \tag{6}$$

where $s = s(t)$, $s' = s(t+1)$, $a = a(t)$, $a' = a(t+1)$, and $\alpha, \beta > 0$ are the step sizes for the parametrization parameters (Parisi et al., 2019).

In the multi-agent extension, players each maintain their own actor and critic functions, but the required state and action inputs for (5) and (6) becomes $s(t) = (s_1(t), \ldots, s_N(t))$ and $a(t) = (a_1(t), \ldots, a_N(t))$. In networked systems such as supply chains, players tend to focus on competition and try to gain the upper-hand by withholding information from its neighbors (Khan et al., 2016). For actor critic methods, players are constrained to updating their critics using *internal* state-action information (s_i, a_i) and actors using *internal* state s_i. This information constraint is akin to solving a partially observable stochastic game. However, players may be persuaded to exchange information if it leads to beneficial outcomes. We consider two different multi-agent paradigms under different information access levels.



(a) Individual learning (b) Players share states (c) Players share states and actions

Figure 3: Actor-critic methods under different levels of information access.

3.5.2 INDIVIDUAL LEARNING

Assuming that players do not share any state-action information, *individual learning* dynamics is be akin to local actor-critics performed in a coupled environment. While individual learning has no convergence guarantee even in the case of full state-action information, the partial observability of joint state-action space adds an extra dimension of difficulty towards algorithm convergence.

3.5.3 CENTRALIZED LEARNING, DECENTRALIZED EXECUTION

A popular multi-agent learning paradigm is centralized learning for decentralized execution (CLDE) (Lyu et al., 2021), exemplified in both collaborative and competitive settings by Lowe et al.

(2017) and Foerster et al. (2018). Under the CLDE framework, players would be required to share their state-action information with all other players. In the supply chain scenario, state information includes player inventory and forecasts, and player action corresponds to information on the cost and prices charged among competitors. One can imagine that while inventory and demand forecast information warrant privacy concerns, cost and prices charged among competitors is much more classified and could be difficult for players to share. We therefore consider two levels of information sharing: level one where players share only states, and level two where players share both states and actions. The resulting actor critics are summarized in Figure 3.

4 TWO PLAYER, SINGLE-COMMODITY SUPPLY CHAIN

To demonstrate the performance of actor critic methods on networked stochastic games, we construct a simple two player, single-commodity example. In this example, two players - a supplier (player zero) and a retailer (player one) form a single-commodity supply chain between a raw materials market and a consumer market as shown in Figure 2.

Player objectives are defined in (2). We assume the raw material market is demand insensitive, such that $P(q_0) = 0.5$ per unit product. The consumer demand is price sensitive, such that $Q(p_1) = 10 - 2p_1 + 0.05\epsilon$, where $\epsilon \sim \mathcal{N}(0, 1)$ is a Gaussian noise. Holding cost coefficient is $h_0 = h_1 = 0.05$, goodwill coefficient is $w_0 = w_1 = 0.1$.

Traditionally in supply chain literature, the total profit of the entire supply chain is optimized. We can apply the actor critic method to solve the total profit scenario: the actor takes the joint state as the input, the critic takes the joint state-action as the input, and the reward being optimized is $\sum_{i=1}^{2} R_i(s_i, a_i; d_i)$. The resulting rewards and player actions are shown in Figure 4.

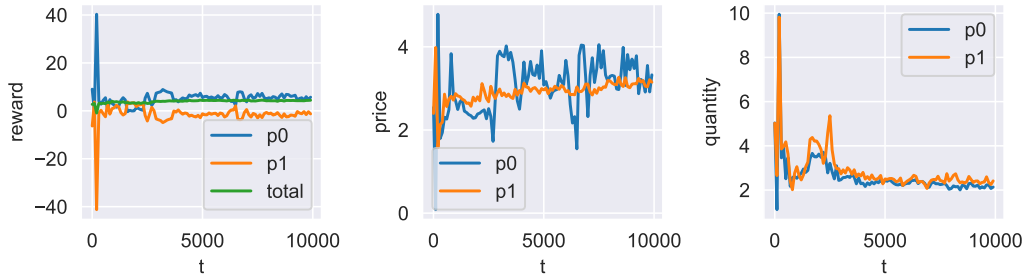


Figure 4: Learning trajectory of the centralized reward scenario with actor-critic method.

While the total reward is maximized, the profit distribution heavily favors player 0, and therefore discourages player 1 from participating in the centralized scheme. Next, we consider MADDPG with three different levels of information access: individual learning, sharing states during training and execution, and sharing states and actions during training and sharing states during execution (MADDPG). The results are shown in Figure 5.

We compare the three different informational access in terms of their optimality, stability, total chain throughput and inefficiency. Optimality and stability are typical training benchmarks to evaluate algorithm performance. Total chain throughput is the total unit products delivered to the consumer market. We can infer the total chain throughput by quantities ordered by both players. Inefficiency refers to the total amount of products produced by the chain but is not available to customers. In the two player supply chain, the inefficiency is characterized by $(q_0(t) - q_1(t))_+$.

In the individual RL setting, player rewards and policies both stabilize. However, player 1's reward is significantly lower than player 0's reward. Throughout the training process, player 0 consistently orders more products than player 1, implying inefficiency in the supply chain. Finally, the individual learning setting seem to achieve the greatest chain throughput among all three information settings.

The sharing state and action scenario also achieves stable learning dynamics, albeit at much lower profit for player 0, but greatly reducing player 1's losses. However, the gap of ordered quantities

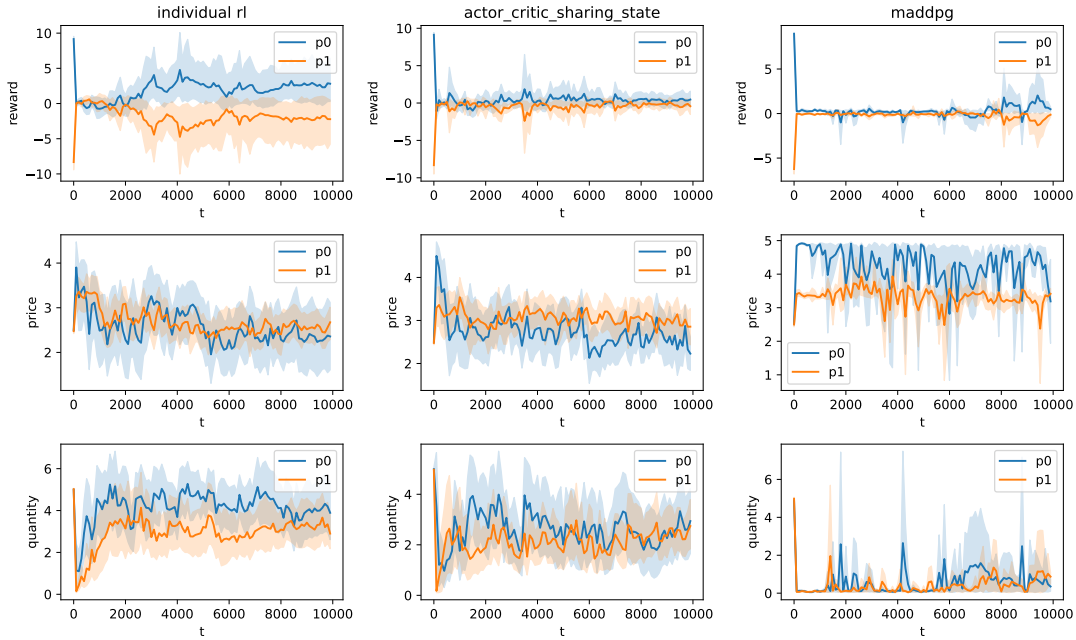


Figure 5: Learning trajectories of actor-critic methods under different information assumptions.

between the two players is significantly lower than the individual learning case, resulting in higher chain efficiency. This fact is quite intuitive - if player 0 can see player 1's inventory levels, player 0 can better predict player 1's demand and therefore order accordingly.

Finally, we note that the MADDPG setting with similar hyperparameters as the previous two scenarios did not converge, and appear to become increasingly more unstable around the 8000th episode. Furthermore, the chain throughput appears to be near zero through out the first 10000 episodes, which can be interpreted as both players had quit the chain and are not producing any products to meet consumer demand. The prices converging to exceptionally high values also reflects this.

5 Conclusion

In this paper, we addressed a key structure in emerging applications of MARL to large-scale operation management problems. We formulate these problems as networked stochastic games, and demonstrate how they apply to supply chains, epidemics, and opinion dynamics in social networks. We discussed some problems associated with solving these networked stochastic games with multi-agent reinforcement learning and discussed its implications for actor critic type algorithms.

References

Bowen Baker, Ingmar Kanitscheider, Todor Markov, Yi Wu, Glenn Powell, Bob McGrew, and Igor Mordatch. Emergent tool use from multi-agent autocurricula. *arXiv preprint arXiv:1909.07528*, 2019.

George Baryannis, Sahar Validi, Samir Dani, and Grigoris Antoniou. Supply chain risk management and artificial intelligence: state of the art and future research directions. *International Journal of Production Research*, 57(7):2179–2202, 2019.

Tamer Başar and Geert Jan Olsder. *Dynamic noncooperative game theory*. SIAM, 1998.

Gianluca Bontempi, Souhaib Ben Taieb, and Yann-Aël Le Borgne. Machine learning strategies for time series forecasting. In *European business intelligence summer school*, pp. 62–77. Springer, 2012.

James D Dana Jr and Nicholas C Petruzzi. Note: The newsvendor model with endogenous demand. *Management Science*, 47(11):1488–1497, 2001.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Ammar Haydari and Yasin Yilmaz. Deep reinforcement learning for intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 2020.

Ed Hopkins. A note on best response dynamics. *Games and Economic Behavior*, 29(1-2):138–150, 1999.

Mehmood Khan, Matloub Hussain, and Hussein M Saber. "Information sharing in a sustainable supply chain". *International Journal of Production Economics*, 181:208–214, 2016.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems*, 30, 2017.

Xueguang Lyu, Yuchen Xiao, Brett Daley, and Christopher Amato. Contrasting centralized and decentralized critics in multi-agent reinforcement learning. *arXiv preprint arXiv:2102.04402*, 2021.

Eric Mazumdar, Lillian J Ratliff, and S Shankar Sastry. On gradient-based learning in continuous games. *SIAM Journal on Mathematics of Data Science*, 2(1):103–131, 2020.

Simone Parisi, Voot Tangkaratt, Jan Peters, and Mohammad Emtiyaz Khan. Td-regularized actor-critic methods. *Machine Learning*, 108(8):1467–1501, 2019.

Guannan Qu, Yiheng Lin, Adam Wierman, and Na Li. Scalable multi-agent reinforcement learning for networked systems with average reward. *Advances in Neural Information Processing Systems*, 33:2074–2086, 2020.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

J. K Terry, Benjamin Black, Nathaniel Grammel, Mario Jayakumar, Ananth Hari, Ryan Sulivan, Luis Santos, Rodrigo Perez, Caroline Horsch, Clemens Dieffendahl, Niall L Williams, Yashas Lokesh, Ryan Sullivan, and Praveen Ravi. Pettingzoo: Gym for multi-agent reinforcement learning. *arXiv preprint arXiv:2009.14471*, 2020.

Sébastien Thomassey. Sales forecasts in clothing industry: The key success factor of the supply chain management. *International Journal of Production Economics*, 128(2):470–483, 2010.

Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative, multi-agent games. *arXiv preprint arXiv:2103.01955*, 2021.

Dongxia Zhang, Xiaoqing Han, and Chunyu Deng. Review on the research and practice of deep learning and reinforcement learning in smart grids. *CSEE Journal of Power and Energy Systems*, 4(3):362–370, 2018.

Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of Reinforcement Learning and Control*, pp. 321–384, 2021.

A APPENDIX

We provide two additional examples of systems that can be modeled as networked stochastic games.

Geophysical epidemic network. Networked stochastic games can model the spread of virus in a geophysical network. Consider a finite number of cities $[N]$ undergoing an epidemic outbreak where the virus is nonfatal and has no vaccines or cures. To minimize the outbreak, each city can choose to impose lockdowns of varying severity and reserve medical effort towards exclusively curing infected patients.

We model the cities as nodes on graph $\mathcal{G}$ and the transportation connections between cities as $e = (i, j) \in \mathcal{E}$. Each player has two internal states $s_i(t) = [h_i(t) \quad p_i(t)] \in \mathbb{R}^2$, where h_i and p_i represent the number of healthy individuals and patients in city i, respectively. Player actions are given by $a_i(t) = [\ell_i(t), \nu_i(t)]$, where $\ell_i \in [0, 1]$ is the level of lockdown in city i and $\nu_i \in [0, 1]$ is the medical effort dedicated towards curing infected patients. The rate at which healthy individuals become patients is given by $\Big(1 - \exp(-\ell_i(t) - \sum_{j\in\mathcal{N}_i} \alpha_{ij}\ell_j(t))\Big) h_i(t)$, where $\alpha_{ij} \in [0, 1]$ denotes the connectivity of cities i, j. The rate at which patients recover is given by $\delta_i \sim \mathcal{N}(\nu_i(t), \sigma_i)$. Each city's healthy and infected population evolve over time as

$$h_i(t+1) = \Big(1 - \exp\big(-\ell_i(t) - \sum_{j\in\mathcal{N}_i} \alpha_{ij}\ell_j(t)\big)\Big)h_i(t) + \delta_i(t)p_i(t) \tag{7}$$

$$p_i(t+1) = (1 - \delta_i(t))p_i(t) + \exp(-\ell_i(t) - \sum_{j\in\mathcal{N}_i} \alpha_{ij}\ell_j(t))h_i(t) \tag{8}$$

Each city receives a reward that is positively correlated with the number of healthy individuals and negatively correlated with the lockdown level and effort spent towards curing patients.

$$R_i(t) = h_i^2(t) - Q_{ii}\ell_i^2(t) - P_{ii}\nu_i^2(t),\ Q_{ii}, P_{ii} \in \mathbb{R}_+,\ \forall i \in [N].$$

Opinion dynamics in social networks. We consider a multi-agent consensus dynamic that models the spread of information through a social network with a leader(CITE). Consider a set of players $[N]$ connected on an undirected graph $\mathcal{G} = ([N], \mathcal{E})$, where player 1 is the leader, and $a_{ij} = a_{ji}$ is edge (i, j)'s weight. Each player's opinion in S different topics is represented by $x_i \in \mathbb{R}^S$. Players do not directly observe their own states, but instead observes the net error between his own opinion and his neighbors' opinions, $e_i = \sum_{j\in\mathcal{N}_i} a_{ij}(x_i - x_j)$. Based on e_i, players choose an action u_i, and see its opinions evolve in time t as

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t),\ \forall i \in [N].$$

Each player's reward is given by

$$R_i(t) = \sum_{j\in\mathcal{N}_i} (x_i(t) - x_j(t))^2 Q_{ii} + u_i(t)^2 R_{ii},\ Q_{ii},\ R_{ii} \in \mathbb{R}_+.$$