Titre: Title:	Algorithmic and Tacit Coordination Among Artificial Learners
Auteur: Author:	Igor Sadoune
Date:	2024
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Sadoune, I. (2024). Algorithmic and Tacit Coordination Among Artificial Learners [Thèse de doctorat, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/59212/

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Algorithmic and Tacit Coordination Among Artificial Learners

IGOR SADOUNE
Département de mathématiques et génie industriel

Thèse présentée en vue de l'obtention du diplôme de *Philosophiæ Doctor*
Mathématiques de l'ingénieur

Août 2024

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Cette thèse intitulée:

Algorithmic and Tacit Coordination Among Artificial Learners

présentée par: **Igor SADOUNE**
en vue de l'obtention du diplôme de: *Philosophiæ Doctor*
a été dûment acceptée par le jury d'examen constitué de:

Catherine BEAUDRY, présidente
Marcelin JOANIS, membre et directeur de recherche
Andrea LODI, membre et codirecteur de recherche
Margarida CARVALHO, membre
Amy GREENWALD, membre externe

DEDICATION

To the people I love...

ACKNOWLEDGEMENTS

"*I'd kill for a Nobel Peace Price.*"

— Steven Wright

My journey as a Ph.D. candidate has been filled with challenges and doubts, and consequently, resilience. This experience certainly helped me become the person I am now, and at least to some extent, the person I will be for the rest of my life.

I am grateful for the chance I had to embark on this inspirational journey, to evolve in such an environment, and alongside such dedicated and skilled academics. First of all, I will give my thanks to those to whom I owe that chance. Marcelin Joanis and Andrea Lodi, thank you. Thank you for your trust, patience, support, and invaluable guidance as my advisors.

I am also grateful to Mehdi Taobane for coordinating all the key events of my program, and to all the people of the DS4DM chair and the Polytechnique administration for making it work for all of us.

Ghislain and Jérôme, thank you for your IT support and our conversations, and to all the people at CIRANO, thank you.

My colleague, and most importantly, my friend, Federico Bobbio, thank you for the interesting conversations as well as the funny escapades.

Inès and Étienne, thank you for being here in person during the defense. Your support has been priceless during this stressful time, and having such close friends by my side inspired me and helped me do my best. Étienne, thank you as well for all your support as a researcher and mentor in life.

Alex, Alex Weit, Audrey, Luigi, Kevork, Toto, Franco, Roro, Bastou, Andrew, Mikasa, Armin, Arthur, Chloé, Lucile, Claire, Nass, Kim, Karim, Aline, Jerem', Axel, and all my friends. I am sorry, I could not mention everyone by name, but please know that each of you contributed to this in one way or another. Thank you.

Mathieu Guillot, thank you for being my best friend for so long. You are a brother to me.

To my family, mom, my grandparents, Ygal, Muriel, Ricardo, thank you for everything.

A special tribute to my dad, without whom nothing could have happened. Thank you for your unconditional love and support.

RÉSUMÉ

Un enjeu central dans l'étude des systèmes multi-agents est de comprendre comment des agents égoïstes peuvent collaborer et manifester une intelligence collective afin de maximiser leur bien-être. Cette problématique est particulièrement pertinente dans des scénarios où il existe une tension entre les intérêts individuels et le bien commun. La théorie des jeux offre un cadre solide pour étudier la coordination tacite parmi des agents artificiels dans des contextes non coopératifs. Cette thèse se penche sur les dynamiques complexes menant des équilibres de Nash non Pareto-optimaux vers des stratégies Pareto-optimales, dans le cadre d'un jeu de coordination. Elle s'appuie sur trois études détaillées, chacune explorant des défis et opportunités uniques dans les simulations multi-agents et l'émergence de comportements algorithmiques.

Premièrement, nous introduisons un méta-algorithme pour générer des données synthétiques réalistes d'enchères multi-niveaux. Pour relever les défis posés par la nature à haute cardinalité, multi-niveaux et discrète de ces structures, nous avons recours à une approche d'apprentissage profond hiérarchique basée sur les réseaux antagonistes génératifs et BidNet, un modèle prédictif des distributions d'offres conditionnelles fondé sur des caractéristiques sous-jacentes. Cette avancée facilite le développement et le test de modèles à grande échelle pour des simulations multi-agents.

Deuxièmement, nous présentons le Jeu de Markov à Prix Minimum (MPMG), qui, dans des conditions homogènes, étend le Dilemme du Prisonnier à un jeu stochastique. Le MPMG est ensuite peuplé d'agents d'apprentissage par renforcement multi-agents (MARL) afin d'examiner la robustesse de la règle de prix minimum face à la collusion en l'absence de coordination explicite. Nos résultats montrent que les agents non informés parviennent plus facilement à des résultats Pareto-optimaux que leurs homologues plus sophistiqués, ce qui suggère que la coordination tacite, ou collusion tacite, peut émerger de manière accidentelle dans de tels contextes.

Enfin, nous proposons le Gradient de Politique d'Équilibre Stratégique (SEPG), une méthode MARL visant à favoriser la coordination tacite dans des environnements non coopératifs sans recours à des mécanismes de communication explicites. Le SEPG, un algorithme acteur-critique basé sur le gradient de politique, combine planification et apprentissage adaptatif en ligne, permettant aux agents de converger vers des stratégies Pareto-optimales dans le MPMG. Nous démontrons que les agents SEPG peuvent atteindre une collusion tout en adoptant des comportements rationnels.

ABSTRACT

A central problem in the study of multi-agent systems concerns elucidating how selfish agents can collaborate and exhibit group intelligence within large-scale decision-making contexts. This issue is particularly reflected in scenarios characterized by a tension between individual and collective welfare. From a methodological perspective, game theory seamlessly integrates algorithmic learning, providing fertile grounds for the study of tacit coordination among artificial learners in non-cooperative settings. This thesis explores the intricate dynamics of transitioning from non-Pareto Nash Equilibria to Pareto-Optimal strategies in the context of a minimum price-ruled coordination game. This research is anchored in three comprehensive studies, each addressing unique challenges and opportunities in agent-based computational simulations and in the study of emergent algorithmic behavior.

First, we introduce a meta-algorithm for simulating realistic synthetic multi-level auction data. To overcome the challenges inherent to the high-cardinality, multi-level, and discrete nature of such structures, we employ a hierarchical deep learning approach based on generative adversarial learning and the proposed BidNet, a predictor of conditional bid distributions based on underlying features. This advancement aids in developing and testing large-scale models for agent-based simulations.

Second, we introduce the Minimum Price Markov Game (MPMG), which under the condition of homogeneity, extends the Prisoner's Dilemma to a stochastic game. The MPMG is then populated with Multi-Agent Reinforcement Learning (MARL) agents to examine the robustness of the minimum price rule against collusion when coordination is not engineered. Our findings reveal that uninformed agents coordinate more easily towards Pareto Optimal outcomes than their sophisticated counterparts, meaning that tacit coordination, or tacit collusion, can occur accidentally in such settings.

Finally, we devise the Strategic Equilibrium Policy Gradient (SEPG), a MARL method that aims to foster tacit coordination in non-cooperative settings without relying on direct communication mechanisms. The SEPG is an actor-critic policy gradient algorithm that combines planning with adaptive online learning, enabling agents to achieve Pareto Optimal strategies in the MPMG. We show that SEPG agents can achieve collusion while demonstrating rational behaviors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ACE	Agent-Based Computational Economics
AGT	Algorithmic Game Theory
CJAL	Conditional Joint Action Learner
CTDE	Centralized Training with Decentralized Execution
CTGAN	Conditional Tabular Generative Adversarial Network
D3QN	Double Deep Q-Network
D3QN-OM	Double Deep Dueling Q-Network - Opponent Modeling
DGM	Deep Generative Modeling
EGT	Evolutionary Game Theory
GAN	Generative Adversarial Network
GT	Game Theory
MAPPO	Multi-Agent Proximal Policy Optimization
MARL	Multi-Agent Reinforcement Learning
MAS	Multi-Agent Systems
MDP	Markov Decision Process
ML	Machine Learning
MLP	Multi-Layer Perceptron
MPG	Minimum Price Game
MPMG	Minimum Price Markov Game
NE	Nash Equilibrium
PD	Prisoner's Dilemma
RL	Reinforcement Learning
SEAO	Système Électronique d'Appel d'Offres
SEPG	Strategic Equilibrium Policy Gradient
SFI	Santa Fe Institute
TS	Thompson Sampling
UCB	Upper Confidence Bound

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

A central problem in the study of Multi-Agent Systems (MAS) concerns elucidating how individual agents, primarily driven by self-interest, can collaborate and exhibit group intelligence within large-scale decision-making contexts [1]. The seamless interplay between game theory and algorithmic learning provide fertile ground to explore this issue [2, 3]. Social dilemma games, in particular, offer a theoretical foundation for studying emergent behavior among adaptive agents, reflecting the tension between individual and collective welfare.

This thesis adopts this methodology and aims to contribute at the specific level of tacit coordination among artificial learners in the context of minimum price-based auction markets. This auction design governs various economic processes, such as, most notably, public procurement, which is a pivotal mechanism for global social welfare [4]. The coordination among potentially malicious actors in such environment poses a serious issue, as algorithmic pricing is an established threat in online retail [5], electricity supply [6], and public procurement [7].

The question of how interactions among individual components of a system give rise to collective behaviors is abstract yet tentacular, as it connects many research fields such as physics, economics, biology and computer science. The Santa Fe Institute (SFI) has extended this philosophical connection to a methodological one, championing the complex system approach since 1984 [8]. SFI is renowned for its research in complexity science, aiming to model multi-layered and evolving economic structures through an interdisciplinary lens. This research has contributed to the theoretical and philosophical foundations of the Agent-Based Computational Economics (ACE) of today.

In the past two decades, machine learning (ML) has emerged as a powerful tool in empirical research. In economics, it has reinforced the ongoing empirical shift by harnessing diverse sources of data [9] and extended econometrics in the search for causal effects [10]. However, our endeavor steers us towards the ML paradigms of Deep Generative Modeling (DGM) [11] and Multi-Agent Reinforcement Learning (MARL) [12], positioning our contribution within the fields of ACE and Algorithmic Game Theory (AGT).

1.1 Overview and Objectives

This thesis aims to contribute to the question of coordination among artificially intelligent agents in competitive settings by incorporating relevant computational methods for both the elaboration of complex and realistic frameworks and the modeling of rational and emergent

behaviors. We intend to shed light on the problem of tacit coordination among artificial learners in coordination games where all participants can benefit from aligning their strategies. These scenarios characterize the minimum price-ruled public auctions, which are competitive by law but cooperative by nature. Indeed, the search for efficient mechanisms in auction structures is a long-running problem [13], as collusion is known to happen, most notably in public procurement [14]. From this context, we derive the following objectives and specific questions that we explore in this dissertation:

(i) How can modern ML methods, in particular DGM, help in solving problems inherent to the simulation of auction data which harbor a complex and uncommon multi-level structure?

(ii) Can AI-driven agents cause harm by cooperating or colluding in minimum price-ruled auction mechanisms?

(iii) Is the minimum-price rule robust to non-engineered coordination among participating artificially intelligent players?

(iv) Can such coordination emerge accidentally, i.e., be implicitly driven by the nature of the problem (utility maximization) and the availability of data informing on competitors?

(v) How could potentially malicious AI-driven agents be engineered to foster coordination, and thus, collusion in such systems?

This set of inquiries can be divided into two main themes. Objectives (ii) to (v) can be grouped under a common theme, while objective (i) can be tackled independently. Nevertheless, these two themes are closely related, as data generation is pivotal in elaborating full-scale realistic environments that, in turn, can serve as supports for the study of behavioral emergence.

From a methodological standpoint, this thesis aims to unfold coherently in regard to the ACE research effort. Therefore, this dissertation presents three self-contained research articles, each one aiming to echo a step towards the broader objective of building a complex system. Namely, Article 1 addresses (i) by using a straightforward deep learning approach, while Articles 2 and 3 combine prescriptive (game theory) and predictive (machine learning) approaches with the intent to provide answers for (ii), (iii), (iv), and (v).

1.2 Contributions

In this section we detail, in chronological order, the contribution of each article. Article 1 (Chapter 4) can be found in its published version in [15]. A pre-print for Article 2 (Chapter 5) is available in [16], and Article 3 is presented in Chapter 6.

Article 1. *Implementing a Hierarchical Deep Learning Approach for Simulating Multi-Level Auction Data*

This first work introduces a hierarchical deep learning methodology to address the problem of simulating realistic synthetic auction data, particularly for first-price sealed-bid auctions. By leveraging Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs), we presents a meta-algorithm for handling high-cardinality discrete feature spaces and multilevel data structures. Multilevel data pose significant challenges due to its inherent complexity, as each auction instance is associated with multiple bids from various firms, leading to a high-dimensional and sparse data structure. The Bidnet architecture is proposed to predict the conditional bid distribution based on auction characteristics, which significantly enhances the simulation accuracy. This contribution helps in advancing simulation-based research, providing a robust foundation for creating realistic auction environments. The generated synthetic data can facilitate the application of agent-based modeling, complex system approaches, and empirical games in auction markets, thereby offering valuable insights into economic models grounded in generative AI.

Article 2. *Algorithmic Collusion and the Minimum Price Markov Game*

We introduce the Minimum Price Markov Game (MPMG), a dynamic coordination game developed to model auction environments governed by the minimum price rule. By extending the classical Prisoner's Dilemma into a flexible setting, the MPMG provides a framework for understanding market behavior and algorithmic coordination among artificial learners. The study delves into the implications of algorithmic collusion within this context, offering a comprehensive analysis of how AI agents can influence competitive pricing dynamics. The findings reveal that while the first-price auction environment is generally robust against algorithmic collusion, there are scenarios where relatively naive agents can achieve tacit coordination. This paper highlights the regulatory challenges posed by algorithmic pricing, emphasizing the difficulty of detecting and preventing tacit coordination. The study's insights help in understanding the potential threats of algorithmic collusion in automated markets and developing strategies to mitigate these risks.

Article 3. *Strategic Equilibrium Policy Gradient: On Fostering Tacit Coordination In The Minimum Price Markov Game*

In this last contribution we introduce the Strategic Equilibrium Policy Gradient (SEPG), a MARL approach designed to foster tacit coordination in coordination games. The SEPG method combines pre-game planning and online adaptation to guide agents towards Pareto Optimal outcomes through implicit coordination. By modeling scenarios where agents lack explicit cooperation mechanisms, the study demonstrates that SEPG agents can achieve robust tacit coordination in both homogeneous and heterogeneous market scenarios. The SEPG challenges existing MARL methods by showcasing its effectiveness in promoting rational behavior and coordination. Through extensive experiments, the study highlights the potential for tacit collusion in automated markets and provides a framework for understanding the dynamics of such coordination. This contribution helps in advancing the theoretical understanding of coordination in social dilemmas and enhancing the practical application of MARL in economic systems.

1.3 Thesis Outline

The remainder of this thesis is organized as follows. In Chapter 2, we lay out the background definitions and core concepts used in this dissertation. In Chapter 3, we provide a literature overview complementing the self-contained core articles presented in Chapters 4, 5, and 6. In Chapter 7 we discuss the contributions, their implications, and their interactions. Finally, in Chapter 8, we conclude by a summary of works and an account of future research and perspectives.

CHAPTER 2 BACKGROUND

2.1 Deep Generative Modeling

The goal of DGM is to learn representations of multi-dimensional probability functions for inference or sampling using MLPs. In this section, we present the two most influential methods. For a more exhaustive account of DGM, see [17, Chapter 20].

Generative adversarial learning. The GAN framework comprises two neural networks—the generator G and the discriminator D—engaged in a minimax game. The generator aims to produce data $G(z)$ from noise z such that it mimics the real data distribution p_{data}, while the discriminator evaluates the authenticity of samples from G against real data samples x, i.e., by assigning a probability value $D(x)$. The objective function for a GAN is represented as

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))]$$

where $\mathbb{E}$ denotes the expectation, p_{data} is the distribution of real data, and p_z is the noise distribution.

Variational Autoencoding. VAEs are built on a probabilistic framework that models the data using a latent variable z. The encoder in a VAE approximates the posterior distribution $q_\phi(z|x)$ of the latent variables given an input x, and the decoder generates new data x from the latent space. The loss function of VAEs, the evidence lower bound, is given by

$$\mathcal{L}(\theta,\phi;x) = -\mathbb{E}_{z\sim q_\phi(z|x)}[\log p_\theta(x|z)] + \mathrm{KL}(q_\phi(z|x)\|p(z))$$

The first term is the reconstruction loss, maximizing the likelihood of x given z, and the second term is the Kullback-Leibler divergence between the learned latent distribution $q_\phi(z|x)$ and the prior $p(z)$, typically assumed to be a standard normal distribution.

2.2 Game Theory

Game theory (GT) is the mathematical theory of strategic interactions [18]. GT is designed for analyzing situations in which players make decisions that are interdependent, involving complex strategies and outcomes dependent not only on individual decisions but also on the actions of others. It applies broadly across disciplines such as economics, political science,

biology, and computer science to provide insights into the strategic behavior of agents in competitive and cooperative settings. The fundamental components of any game include players, strategies, and payoffs. Mathematically, a game can be described by:

- N, the set of players.
- $\{S_i\}_{i \in N}$, the strategy sets for each player i.
- $u_i : S_1 \times S_2 \times \ldots \times S_n \to \mathbb{R}$, the payoff function for each player i, which depends on the strategy combination adopted by all players.

Nash Equilibrium. The Nash Equilibrium (NE) is a solution concept within game theory where the traditional notion of optimality is not applicable. A NE represents a state in a strategic interaction where no player can benefit by changing their strategy unilaterally, given that the other players' strategies remain unchanged. It is a condition of mutual best responses where each player's strategy is optimal given the strategies of all other players. In simpler terms, in a Nash Equilibrium, every player is doing the best they can considering the actions of their opponents, and no one has anything to gain by changing only their own strategy. A strategy profile $(s_1^*, s_2^*, \ldots, s_n^*)$ is a pure Nash Equilibrium if for each player i,

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*) \quad \forall s_i \in S_i$$

where s_{-i}^* denotes the strategy combination of all players except player i. At this equilibrium, no player can benefit by unilaterally changing their strategy, assuming other players' strategies remain unchanged.

Pareto Optimality and Pareto Frontier. A strategy combination $(s_1^*, s_2^*, \ldots, s_n^*)$ is Pareto Optimal if there is no other strategy combination $(s_1, s_2, \ldots, s_n)$ where at least one player is better off without making any other player worse off. Mathematically,

$$\nexists (s_1, s_2, \ldots, s_n) \text{ such that } u_i(s_1, s_2, \ldots, s_n) \geq u_i(s_1^*, s_2^*, \ldots, s_n^*)$$
$$\text{for all } i \text{ and } u_j(s_1, s_2, \ldots, s_n) > u_j(s_1^*, s_2^*, \ldots, s_n^*) \text{ for some } j.$$

The Pareto frontier, or Pareto set, is the collection of all Pareto optimal outcomes, illustrating the trade-offs between players where improving one player's payoff necessitates reducing another's. In game theory, a solution that is Pareto optimal and one that is a Nash Equilibrium can coincide (Pareto-Nash), but they do not necessarily always align.

Social Dilemma. A social dilemma arises in a game theory context when individual rationality leads to collective irrationality. That is, each player, acting in their own self-interest, makes choices that result in an outcome which is suboptimal for all participants compared to what could have been achieved through cooperative behavior. The classic example of a social dilemma is the Prisoner's Dilemma (PD), where players choose between cooperation and defection. While mutual cooperation leads to a better overall outcome, rational self-interested strategies lead each player to defect, resulting in a worse outcome for both.

The 2-player PD can be defined using a bimatrix, where the strategies available to player i and player o are represented as rows and columns, respectively. The bimatrix is given by

	Cooperate	Defect
Cooperate	(R, R)	(S, T)
Defect	(T, S)	(P, P)

where

- R (Reward): The payoff if both players cooperate.
- T (Temptation): The payoff received by a player if they defect while the other cooperates.
- S (Sucker's payoff): The payoff received by a player if they cooperate while the other defects.
- P (Punishment): The payoff if both players defect.

The inequalities $T > R > P > S$ and $2R > T + S$ typically hold, illustrating why defecting becomes the dominant strategy for both players, despite mutual cooperation yielding a higher payoff (R) than mutual defection (P). This matrix framework clearly delineates the dynamics of a social dilemma, where individual incentives lead to decisions that result in suboptimal outcomes for all involved, providing a framework for analyzing conflicts between individual and collective interests.

2.3 Reinforcement learning

Reinforcement learning (RL) describes both a problem and a family of solution methods [19]. The RL problem is formalized as the optimal control of partially-known Markov Decision Process (MDP). Figure 2.1 depicts a RL agent interacting with a MDP.

Definition 1. *A **Markov Decision Process** is a mathematical framework used to model decision making in situations where outcomes are partly random and partly under the control of a decision maker. An MDP is characterized by:*

- *A set of states $\mathcal{S}$ that describe different scenarios in the environment.*
- *A set of actions $\mathcal{A}$ available to the decision maker.*
- *A transition function $P(s', s, a)$ that defines the probability of moving from state s to state s' after taking action a.*
- *A reward function $R(s, a, s')$ that assigns rewards or costs based on transitioning from state s to state s' due to action a.*
- *A discount factor γ, typically between 0 and 1, which accounts for the difference in importance between immediate and future rewards.*

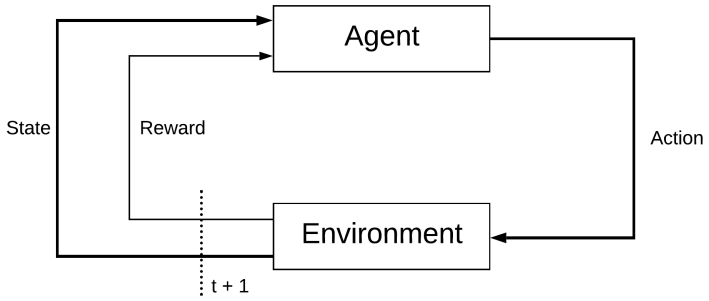


Figure 2.1 Agent-MDP interface.

RL agents face the challenge of the explore-exploit trade-off. Exploiting means taking the action prescribed by the policy given in a state s, while exploring is necessary to discover potentially hidden state-action rewards. A straightforward way to address this dilemma is to define a decaying probability over time that the agent will explore rather than exploit using an epsilon-greedy strategy at each step. An alternative approach involves modifying the algorithmic structure by using two co-existing policies. In this setup, the target policy is updated based on data generated by a different behavior policy, using a method known as importance sampling. This method is part of what is known as off-policy learning, which differs from the epsilon-greedy on-policy approach. Another challenge arises when the problem environment is not stationary, as the optimal action in a given state may change over time, requiring the policy to evolve.

Reinforcement learning, as solution method, is a ML paradigm referring to a class of algorithms that yield a policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$, by maximizing the expected discounted reward signal

$$G_t \doteq \sum_{k=t+1}^{T} \gamma^{k-t-1} r_k.$$

The optimal policy thus satisfies

$$v_*(s) \doteq \max_{\pi} v_\pi(s), \quad \text{for all } s \in \mathcal{S},$$

where

$$v_\pi(s) \doteq E\Big[G_t | S_t = s\Big] \quad \text{for all } s \in \mathcal{S}. \tag{2.1}$$

If not continuous, the RL problem is broke down into a discrete sequence of trials, or episodes. Some problems, such as repeated games, adopt naturally this form.

Q-learning. Q-learning is a foundational model-free RL algorithm used to learn the optimal action-value function within a MDP framework. The core component of Q-learning is the Q-function, denoted as $Q(s, a)$, which estimates the expected utility of taking an action a in state s and subsequently adhering to the optimal policy. The Q-values are updated through an iterative process based on the Bellman equation, with the update rule specified as follows:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$$

Here, s represents the current state, a the action taken, r the received reward, s' the new state post-action, α the learning rate ($0 < \alpha \leq 1$), and γ the discount factor ($0 \leq \gamma < 1$). The term $\max_{a'} Q(s', a')$ is the maximum Q-value achievable from the next state s', guiding the agent towards actions leading to higher future rewards.

Policy gradient. Policy Gradient methods are a class of reinforcement learning algorithms that optimize the policy directly by estimating the gradient of the expected return with respect to the policy parameters. Unlike value-based methods, Policy Gradient methods directly adjust the policy's parameters θ to maximize the cumulative reward. The fundamental concept in Policy Gradient is to adjust the policy parameters θ in the direction that maximally improves the expected rewards. The update rule for the policy parameters is given by

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta)$$

where α is the learning rate, and $\nabla_\theta J(\theta)$ is the gradient of the performance measure $J(\theta)$ with respect to the policy parameters θ. The performance measure J is typically defined as the expected return from the start distribution,

$$J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta(\tau)}[R(\tau)],$$

where $R(\tau)$ is the total reward for the trajectory τ, and $\pi_\theta(\tau)$ is the probability of the trajectory under the policy parameterized by θ. The gradient $\nabla_\theta J(\theta)$ is usually estimated using the likelihood ratio trick, leading to an expression involving the sum of rewards weighted by the gradient of the log-probability of the actions taken,

$$\nabla_\theta J(\theta) \approx \mathbb{E}_{\tau \sim \pi_\theta(\tau)} \left[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) R_t \right],$$

where R_t is the reward at time t, a_t the action taken, and s_t the state at time t.

Actor-critic framework. The actor-critic framework is another fundamental approach in reinforcement learning that combines the advantages of policy-based and value-based methods. It consists of two main components: the actor, which proposes actions based on a policy parameterized by θ, and the critic, which evaluates the actions taken by the actor using a value function parameterized by w. A schematic of the actor-critic interplay is given by Figure 2.2. The actor updates its policy parameters θ in the direction suggested by the critic's evaluation. The update rule for the actor's policy parameters is typically based on the policy gradient method as follows:

$$\theta \leftarrow \theta + \alpha \nabla_\theta \log \pi_\theta(a_t|s_t) A(s_t, a_t; w)$$

where α is the learning rate, $\pi_\theta(a_t|s_t)$ is the probability of taking action a_t in state s_t under the policy, and $A(s_t, a_t; w)$ is the advantage function, representing the difference between the critic's estimate of the state-action pair and the estimated value of the state. The critic estimates the value function $V(s; w)$ and helps compute the advantage function used by the actor. The critic's parameters w are updated to minimize the error between the predicted value and the observed return. The typical update rule for the critic's parameters is

$$w \leftarrow w - \beta \nabla_w (R_t + \gamma V(s_{t+1}; w) - V(s_t; w))^2$$

where β is the learning rate for the critic, R_t is the reward received after taking action a_t at state s_t, γ is the discount factor, and $V(s_t; w)$ and $V(s_{t+1}; w)$ are the critic's estimates of the

current and next state values, respectively.

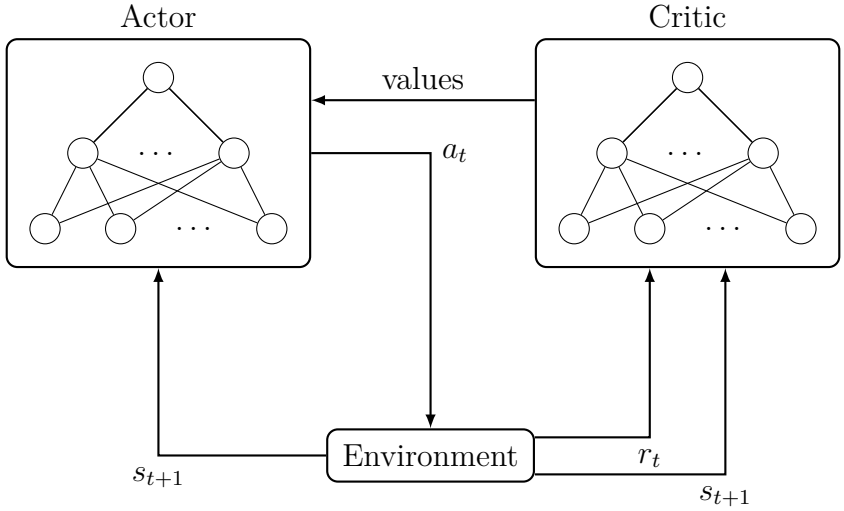


Figure 2.2 Actor-critic framework.

2.3.1 Multiagent Reinforcement Learning

MARL involves multiple agents interacting within a shared environment (see Figure 2.3). Unlike single-agent settings, MARL addresses scenarios where each agent's actions can affect the state of the environment and the outcomes of other agents, adding layers of complexity due to the dynamic interplay of cooperative and competitive elements. Each agent aims to learn a policy that maximizes its cumulative reward, which may depend not only on the environment but also on the strategies and actions of other agents. This leads to challenges such as non-stationarity, where an agent's perspective of the environment changes as other agents learn and adapt their policies, and partial observability, where agents may not have complete information about the state of the environment or the intentions of other agents. The MDP framework extends to Markov games, in order to accommodate dynamic interactions in MAS.

Definition 2. *A **Markov game**, also known as a **stochastic game**, involves a set of players, denoted as N. Each player i has an associated action space $\mathcal{A}_i$. The game is defined over a common state space $\mathcal{S}$. The dynamics of the game are captured by a transition function $T : \left(\bigtimes_{i \in N} \mathcal{A}_i\right) \times \mathcal{S} \to [0, 1]$, which determines the probability of moving from one state to another based on the actions of all players. Each player i has a reward function $R_i : \left(\bigtimes_{i \in N} \mathcal{A}_i\right) \times \mathcal{S} \to \mathbb{R}$, defining the payoff r_i for player i based on the current state and actions taken by all players. The objective for each player i is to maximize the expected sum of discounted rewards, expressed as $E\left[\sum_{k=0}^{\infty} \gamma^k r_{i,t+k}\right]$, where γ is the discount factor that*

values present rewards over future rewards.

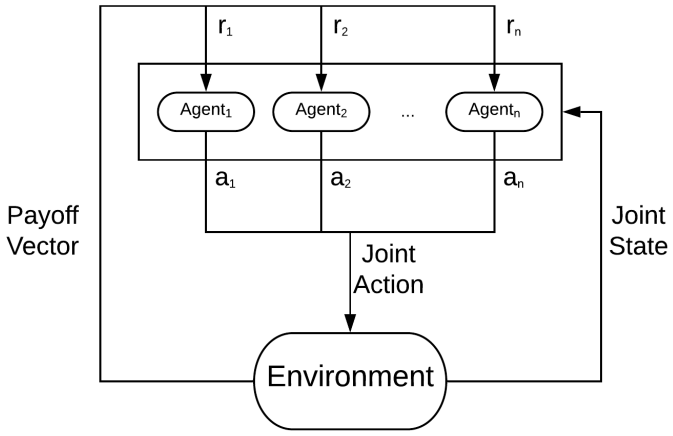


Figure 2.3 Multiagent interactions.

Repeated games are special cases of Markov games for which $\mathcal{S}$ collapses into a singleton and the payoff function for the local player depends only on the change in strategy profiles.

Multi-Agent Reinforcement Learning (MARL) approaches often utilize the Centralized Training with Decentralized Execution (CTDE) framework to solve Markov games. In CTDE, agents act individually based on their own beliefs, but the value of each state, which drives individual decision-making, is provided by a centralized function. Typically, CTDE is implemented via actor-critic learning, where the critic is shared among all agents and uses joint observations to output state values, as depicted in Figure 2.4.

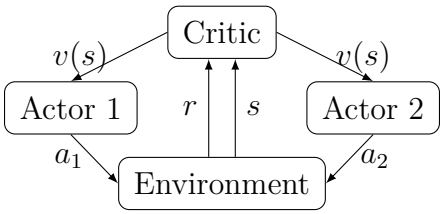


Figure 2.4 Centralized training (critic) and decentralized execution (actors).

A notable example of such a method is the Multi-Agent Deep Deterministic Policy Gradient (MADDPG), an actor-critic policy gradient method that operates under both cooperative and competitive settings [20].

CHAPTER 3 LITERATURE REVIEW

We present a literature review framing the context of the articles presented in this dissertation. Each of the three subsequent chapters is self-contained, thereby expanding on this review with more references to pertinent literature.

Collusion in public auctions. Collusion has been a significant concern in public procurement, particularly in the construction sector [21, 22]. The foundational support for cooperation in the absence of communication, i.e., for collusion to arise in a cartel-free environment, is information, with access to market history being the main facilitator [23]. In fact, the quality of the information is similarly important, as it has been shown that partial observability impedes market collusive potential over time as much as the absence of communication [24]. In the absence of cartel structure, the minimum-price rule is believed to be efficient as a collusion-proof mechanism [25]; however, empirical insights seem to contradict this belief, as implicit bidding rings have been detected using data from atomized markets [26].

Explicit bidding rings, or cartels, enforce collusion using side payments and punishment mechanisms. From a theoretical standpoint, bid rotation—a scheme in which cartel members engage in knockout auctions to determine the auction winner—has been shown to be less efficient than side payments in terms of collusive spoil but more sustainable [27, 28]. In fact, cartel sustainability is challenged by many factors such as the division of spoils, entry barriers, and external retaliation, especially in industries with dynamic market conditions [29]. Indeed, as demonstrated in [30], market conditions such as firm heterogeneity negatively impact the potential for cooperation.

A collusion-proof mechanism within the framework of contract theory, derived from a mechanism design problem in which the agents can communicate among themselves and collude under asymmetric information, has been proposed [31]. However, these mechanisms often falter in the presence of side contracting. From the lens of auction theory, it seems that introducing private values causes some uncertainty about the common values, thereby decreasing the efficiency and revenues of all bidders [32].

Tacit collusion. The concept of tacit collusion was introduced in the mid-twentieth century by George J. Stigler and Jerry Green, upon the idea that self-enforcing policies facilitate collusion without explicit agreements [33]. The analysis of collusion without communication

is critical because explicit cooperation is illegal, and communication is impractical in certain markets, such as energy auctions where the frequency of interactions is too high to organize knockout auctions effectively [34]. Indeed, a certain frequency of repeated interactions is necessary for industries to realize their collusive potential. Mathematically speaking, a high frequency of interaction allows for focal points and price leadership to arise, encouraging self-reinforcing dynamics [35].

Another key aspect of tacit collusion is the necessity of mutual and common knowledge. Mutual knowledge implies that every player knows a fact, while common knowledge extends to knowing that others know the fact. This is the idea behind the SEPG actor-critic algorithm (see Chapter 6), where the critic embodies common knowledge. Direct observation suffices for mutual knowledge but not for establishing mutual intentions, particularly when competitive and collusive equilibria coexist. This leads to the need-to-meet principle, asserting that firms cannot achieve collusive Nash equilibria without higher-order knowledge, which requires explicit meetings [35, 18]. However, full collusion has been achieved in an infinitely repeated price-setting game with minimal information. Despite this minimal information, firms can approximate monopoly outcomes if they are patient enough, even without communication [36].

In fact, collusion exists on a spectrum from full cartel communication to tacit coordination. Non-official cartels might use signals for coordination without forming a full cartel. Rationality and rationalizability concepts are crucial in understanding these dynamics. Rationalizability requires common knowledge of rationality among players, with actions in a mixed strategy NE being rationalizable [37, 38, 39]. The extent of tacit coordination being not well-theorized or easily observed, a unique monitoring design based on theoretical and experimental concepts is necessary to identify tacit coordination agreements from economic evidence [35]. This type of algorithmic audit will potentially become increasingly important as algorithmic pricing in digitized markets can result in tacit collusion, even without malicious intent.

Algorithmic Game Theory for behavioral experiments. In the early twenty-first century, the idea that AGT should be used as a form of algorithmic behavioral experiment emerged. In [2], it is argued that rational agents in game theory and economics are idealized abstractions. Since interactions stem from both rational and non-rational behaviors, artificial agents may mimic human behavior more accurately than ideal agents. Also, in the Cournot model for electricity procurement markets, it seems that Pareto Optimal outcomes often surpass NE outcomes. This underscores the potential of MARL to achieve near Pareto Optimal allocations in social dilemmas, where decisions that benefit individuals can aggregate

into suboptimal group outcomes.

Evolutionary game theory (EGT) supports the notion that behavior approaches NE over time, suggesting firms may inadvertently engage in tacit coordination [40]. Other experimental evidence supports this idea, as some level of tacit collusion has been observed in symmetric Cournot games using simple trial-and-error learning processes without knowledge of payoff functions [41]. However, in asymmetric situations without side payments, players often fail to cooperate without communication, leading to inefficient outcomes [42]. Indeed, laboratory experiments show that monopoly payoffs can be achieved via explicit communication in Bertrand oligopolies [43, 44, 45].

Learning Pareto-Optimal outcomes. In [3], the framework of game theory and MARL is explored to compare Q-learners (MARL agents) of different beliefs. Independent learners outperform informed agents, demonstrating that more information does not always result in better performance. However, acting independently does not always guarantee convergence to Pareto Optimal Nash Equilibria, indicating that the choice between Pareto Nash and Pareto non-Nash solutions depends on profit maximization strategies. This study also argues that EGT is potentially better suited to represent human behavior, as MARL agents tend to be "hyper-rational". In the work presented in Chapter 5 and 6, we align with these conclusions. Our results confirm the idea that more information is not necessarily better for achieving higher payoffs [16].

The Conditional Joint Action Learner (CJAL), which learns the conditional probability of opponents' actions to decide its next move in general-sum n-player iterated games, aims to achieve Pareto Optimal outcomes that also produce NE payoffs. Empirical results show that CJAL learners can converge to Pareto Optimal solutions under specific conditions, particularly in the Prisoner's Dilemma, where classical Nash Equilibria are not Pareto Optimal [46]. The SEPG (Chapter 6) is akin to the CJAL but outsources the learning of the conditional joint action probability from the actor to the critic, giving more independence to the actor. The difference is that the SEPG provides modularity by handling an observation space.

In [47], policy gradient learning is explored, proposing an algorithm that learns almost all existing Nash Equilibria for finite strategic games. The study updates the M-IGA algorithm for more general forms of strategic games, using gradient ascent to find unspecified Nash Equilibria and incorporating defection techniques to ensure convergence to multiple equilibria. This approach is particularly relevant for social dilemmas, where the goal is to maximize joint profits at Pareto frontiers.

CHAPTER 4 ARTICLE 1: IMPLEMENTING A HIERARCHICAL DEEP LEARNING APPROACH FOR SIMULATING MULTILEVEL AUCTION DATA

Authors: Igor Sadoune, Marcelin Joanis, Andrea Lodi[1]

Submitted on August 22, 2023 and published on May 18, 2024 in Computational Economics, DOI: 10.1007/s10614-024-10622-4

Abstract We present a deep learning solution to address the challenges of simulating realistic synthetic first-price sealed-bid auction data. The complexities encountered in this type of auction data include high-cardinality discrete feature spaces and a multilevel structure arising from multiple bids associated with a single auction instance. Our methodology combines deep generative modeling (DGM) with an artificial learner that predicts the conditional bid distribution based on auction characteristics, contributing to advancements in simulation-based research. This approach lays the groundwork for creating realistic auction environments suitable for agent-based learning and modeling applications. Our contribution is twofold: we introduce a comprehensive methodology for simulating multilevel discrete auction data, and we underscore the potential of DGM as a powerful instrument for refining simulation techniques and fostering the development of economic models grounded in generative AI.

Keywords: simulation crafting, discrete deep generative modeling, multilevel discrete data, auction data

4.1 Introduction

In this paper, we propose a hierarchical deep learning method for generating synthetic yet realistic first-price auction data. Our approach is designed to address the challenges associated with high-cardinality discrete feature spaces and multilevel structures inherent to auction data. By leveraging the capabilities of generative deep learning, our primary contribution is to provide the right methodology for crafting realistic auction simulations, which in turn can facilitate the application of agent-based modeling (ABM) and complex system approaches to the study of auction markets.

[1]This article is available in its published version in [15]

Auction markets, as a prominent example of complex systems, have gained significant attention from researchers and practitioners due to their inherent complexity and potential applications across various fields, such as finance, economics, and operations research [48]. The study of first-price auctions, in particular, is crucial for understanding the mechanisms and dynamics that govern their functioning, which are prevalent in numerous industries and applications, including online advertising [49], electricity markets [50], and public procurement in general [51].

We argue that simulations, ABM, and the complex system approach, in general, may offer advantages over static statistical analysis by providing a more detailed representation of individual interactions and capturing the emergent properties of complex systems [52, 53, 54]. The integration of deep learning techniques, such as deep generative modeling, with the crafting of realistic auction simulations is expected to promote more robust and resilient economic systems, ultimately benefiting various fields, including computational economics [55, 56, 57].

The method developed in this paper utilizes generative adversarial networks (GANs) [58] and variational autoencoders (VAEs) [59] for the replication of the auction feature space, while employing a neural network, referred to as Bidnet, to predict the conditional bid distribution for each instance of auction from the generated feature space. Recent advancements in deep generative modeling (DGM), such as the development of the conditional tabular generative adversarial networks (CTGANs) [60], have enabled the efficient handling of high-cardinality discrete distributions, which is a critical aspect for the generation of realistic auction data. Additionally, our Bidnet architecture is specifically designed to address the multilevel issue inherent to auction data, by capturing the relationships between auctions and their associated bids. This combination of DGM and Bidnet provides a comprehensive and effective approach for simulating first-price auction data, while accounting for the complexities and challenges associated with high-cardinality discrete feature spaces and multilevel structures.

4.1.1 Paper Organization

Section 4.2 delves into the mathematical background of GANs and VAEs, discussing their advantages over traditional methods for data generation. This section also explores the application of deep generative modeling in social sciences and its use in high-dimensional discrete spaces.

In Section 4.3, we detail the process of generating synthetic multilevel auction data. This section first presents the multilevel problem and the problem formulation, followed by the solution framework. We then discuss approximating auction features joint density using

GAN-based approaches and tabular variational encoding. The section continues with the training of a generator for continuous bids and concludes with the sampling of synthetic auction instances.

Section 4.4 focuses on the validation of the synthetic data generated by our proposed methods. This section evaluates the faithfulness of the synthetic auction features and the performance of BidNet in generating synthetic bids.

In Section 4.5, we present a discussion that summarizes the study's findings, emphasizing the credibility and usefulness of our contribution. This section also discusses the limitations of our approach and highlights the superiority of CTGANs in our specific context.

Finally, Appendices A, B, and C provide additional details on data, methodology, and algorithms, respectively.

4.2 Deep Generative Modeling

Deep generative modeling includes a variety of techniques that aim to perform density estimation using artificial neural networks as function approximators. Certain models can obtain an explicit representation of the target distribution, while others provide only an implicit, or "black box" representation of the data structure to be replicated. In this study, we focus on the latter category of models, specifically GANs [58] and VAEs [59]. The strength of these models lies in their ability to eliminate the necessity for detailed knowledge of the underlying data structure being replicated. In essence, GANs and VAEs possess the capability to generate an extensive array of synthetic data points from a limited number of empirical observations.

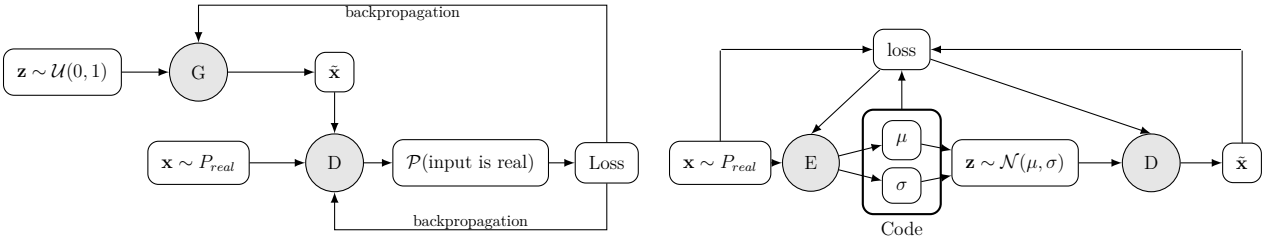
Adversarial learning is the core concept in GANs, which consist of two neural networks, a generator (G) and a discriminator (D), that compete in a two-player minimax game. The generator's objective is to generate synthetic samples $G(z)$, where z is a random noise vector, that are similar to the true data distribution $p_{data}(x)$. The discriminator's goal is to differentiate between real samples (x) from the true data distribution and synthetic samples generated by the generator. The discriminator assigns a probability value $D(x)$ to each input x. The learning process can be described by the following objective function:

$$\min_G \max_D V(D,G) = \mathbb{E}x \sim p_{data}(x)[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \tag{4.1}$$

During training, the generator and discriminator optimize this objective function in a sequential way. The generator seeks to minimize the function while the discriminator attempts

to maximize it. This adversarial process leads to a convergence where the generator produces samples that the discriminator can no longer differentiate from real data samples. As the training progresses, the generator becomes increasingly skilled at producing realistic samples, while the discriminator improves its ability to distinguish between real and synthetic samples. When the equilibrium is reached, the generator generates synthetic samples that resemble the true data distribution, and the discriminator is unable to distinguish between real and generated samples, assigning a probability of $\frac{1}{2}$ to each input.

Figure 4.1 Chart of adversarial learning (left) and variational autoencoding (right).



VAEs are a type of generative model that leverages variational inference to learn a probabilistic mapping between the data and a latent space. The key idea behind VAEs is to introduce a probabilistic encoder $q_\phi(z|x)$, that approximates the true posterior distribution $p_\theta(z|x)$, and a probabilistic decoder $p_\theta(x|z)$, that models the data distribution conditioned on the latent variable z. Here, ϕ and θ represent the parameters of the encoder and decoder neural networks, respectively. Variational inference is used to optimize the evidence lower bound (ELBO) on the log-likelihood of the data, i.e.,

$$\log p_\theta(x) \geq \mathbb{E}q\phi(z|x)[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x)||p_\theta(z)) = \mathcal{L}(\theta, \phi; x). \tag{4.2}$$

The ELBO consists of two terms: the reconstruction term, $\mathbb{E}q\phi(z|x)[\log p_\theta(x|z)]$, which measures the ability of the model to reconstruct the data given the latent variable, and the regularization term, $D_{KL}(q_\phi(z|x)||p_\theta(z))$, which measures the divergence between the approximate posterior and the prior distribution over the latent space. During training, the VAE jointly optimizes the encoder and decoder parameters to maximize the ELBO with respect to the model parameters θ and ϕ. This process encourages the learned latent space to have a meaningful structure, enabling efficient generation of new samples by sampling from the prior $p_\theta(z)$ and decoding them using the decoder $p_\theta(x|z)$.

4.2.1 Traditional Methods for Data Generation

Market simulations and simulation crafting have been widely used across various disciplines to study complex systems, replicate real-world dynamics, and develop policies and strategies. The use of market simulations can be traced back to the early works of economists and computer scientists [61]. Agent-based modeling has emerged as a prominent approach to studying markets and economic systems, allowing researchers to capture the interactions between heterogeneous agents and their influence on market outcomes [52]. A large body of literature has explored different aspects of market simulations, from designing auction mechanisms [62] to investigating the dynamics of financial markets [63].

Historically, traditional methods for simulation crafting, such as Monte Carlo sampling, bootstrapping, cellular automata, and system dynamics modeling, have been widely used in various applications [64, 52, 65]. Monte Carlo sampling generates random samples from a given probability distribution, cellular automata simulates spatial and temporal dynamics through discrete cells evolving based on predefined rules, and system dynamics modeling uses differential equations to describe relationships among system components.

However, these traditional methods exhibit limitations when applied to complex systems with high-cardinality discrete spaces or multilevel structures, such as auction markets. These limitations include difficulties in accurately representing high-dimensional probability distributions, modeling intricate dependencies between variables, accommodating nonlinear dynamics and non-stationary behaviors, and integrating domain-specific knowledge or constraints [66]. Furthermore, traditional simulation methods often require substantial computational resources and are constrained by their assumptions and rules [53].

The advent of machine learning and artificial intelligence has significantly impacted simulation crafting, paving the way for advanced techniques such as DGM. Deep generative models including GANs and VAEs, address the limitations of traditional techniques by learning and representing high-dimensional probability distributions, modeling complex dependencies between variables, and accommodating nonlinearities and non-stationarities in complex systems [67]. These innovative methods have been applied to various domains, such as finance [68]. Although GANs have been recognized as having potential for creating more realistic market simulations and fostering the development of more effective policies and strategies [69], the use of GANs in economics remains limited, with only a small number of applications currently found in the field.

Leveraging unsupervised learning and deep neural networks, DGM methods identify and represent the underlying data structure in a flexible and scalable manner. Additionally, DGM

techniques can model complex dependencies and correlations between variables, providing a more accurate representation of the relationships present in the data. DGM techniques suitable for handling the nonlinearities and non-stationarities that are inherent to complex systems. Due to their deep architectures and non-linear activation functions, deep generative models can capture complex, multi-scale structures in data, which may be challenging for traditional methods to accurately represent. Moreover, DGM approaches can be adapted to incorporate domain-specific knowledge and constraints, further enhancing the realism and validity of the generated synthetic data.

4.2.2 DGM in Social Sciences

Although deep generative models are still underused in social sciences, they can potentially provide substantial improvements in any application relying on qualitative data.

They have emerged as a powerful tool for addressing low degrees of freedom. Specifically, DGMs amplify observations and enhance weak signals within categorical configurations, enabling the generation of synthetic data that closely approximates the characteristics of real-world data. This technology has found applications in a diverse range of domains, such as credit card datasets [70] and medical data recovery [71].

In causal effect research, DGMs play a pivotal role by enabling counterfactual analysis using propensity scores and facilitating the estimation of unseen or partially unseen distributions through GANs [72] and VAEs [73, 74]. Additionally, DGMs have been shown to benefit privacy-sensitive applications involving medical or financial data, leading to specialized literature in this area [75].

Moreover, DGMs have contributed to the reinvigoration of agent-based modeling in economics by generating synthetic data for creating realistic artificial environments and maintaining simulation realism [68]. For example, DGM-based travel behavior simulations have employed restricted Boltzmann machines [76], while GAN-based financial correlation matrices and time-series sampling have been used for simulating financial systems [63, 77].

4.2.3 GANs for High-Dimensional Discrete Spaces

Despite the progress in market simulations and simulation crafting, several challenges remain. In our case, generating high-cardinality discrete data and multilevel data structures is still a complex task, as highlighted by [60]. Furthermore, incorporating the dynamics of real-world markets and their ever-evolving nature into simulation models requires continuous research and development. Various strategies have been developed to enhance the stability

and performance of GANs, particularly focusing on innovations that address the challenges of handling discrete inputs.

For instance, the Wasserstein GAN (WGAN) [78, 79] and the least-square GAN (LSGAN) [80] employ a critic rather than a traditional discriminator. The critic predicts the distance between a given point and the decision boundary separating real and fake samples, thereby improving the signal quality for the generator during training and leading to better convergence properties. WGAN's popularity stems from its Earth Mover (EM) or Wasserstein-I loss, which measures the distance between real and fake sample distributions, thus promoting stable and robust training [78].

The boundary-seeking GAN (BGAN) [81] introduces a modified training process, where the generator uses a policy gradient that accommodates both discrete and continuous inputs. This approach results in smoother training and mitigates issues related to mode collapse, a common problem in GANs.

Meanwhile, the conditional tabular GAN (CTGAN) [60] is a framework specifically designed to address data imbalances and discrete inputs in tabular data. Functioning as a meta-algorithm, CTGAN is compatible with various loss functions, network topologies, or training processes. The core concept of CTGAN, training by sampling, augments the generator's input space with a conditional vector that encodes the selection, thereby enhancing the model's ability to capture the underlying structure of discrete data and generate realistic synthetic samples.

4.3 Generating Synthetic Multilevel Auction Data

In this study, we aim to create realistic synthetic auctions using a novel method that combines two distinct functions, each handling a specific aspect of the auction data. The first function generates artificial auction characteristics, capturing the features of the contracts being offered. The second function, later introduced as BidNet, approximates the distribution of bids given these auction features, providing an aggregated representation of the bidding firms. By breaking down the process into these two sequential functions, our method addresses technical challenges while maintaining the overall structure of the auction. We validate our approach by training a predictor using the synthetic data and evaluating its performance on real-world data. This method effectively separates the generation of discrete and continuous data types and simplifies the complex, multilevel auction structure, making it suitable for generating realistic market simulations.

We have chosen the context of public procurement to demonstrate our ability to simulate

realistic first-price auctions effectively. To this end, we rely on the data provided by the "Système électronique d'appel d'offre " (SEAO), which encompasses 117,249 contracts offered in public market auctions within the Canadian province of Quebec over a ten-year period (2010-2020). In this scenario, the high-cardinality discrete feature space represents various auction attributes, such as the type of contract or the designated delivery region. The multilevel structure of the auction data emerges from the multiple firms participating in the bidding process for each auction. Utilizing this comprehensive dataset allows us to develop and refine our simulation methodology, thereby showcasing its potential for generating accurate and realistic representations of first-price auctions in the domain of public procurement.

4.3.1 Data Specification

Table 4.1 summarizes the structure of the data after cleaning. We differentiate between *multiclass* and *multilabel* variables. A multicalss variable is categorical, but each instance belongs to exactly one out of a finite set of classes. On the other hand, a *multilabel* variable represents cases where an observation can simultaneously have multiple values. For instance, in our scenario, an auction could be either municipal or provincial, making the variable *municipality* a binary multiclass variable. It encodes whether an auction is issued at the municipal level (1) or not (0). Conversely, the variable *firms* can take varying numbers of values per auction, as multiple firms may participate in an auction. Therefore, the variable *firms* is multilabel. This implies that the cardinality (the number of possible states) of such a variable increases exponentially with the number of potential values.

Table 4.1 Structure of the SEAO data.

Variables	**Type**	**Cardinality**
public contractor	multiclass	1,451
municipality	multiclass	binary
sector	multiclass	3
subsector	multiclass	53
location	multiclass	98
unspsc	multiclass	1,379
number of bidders	multiclass	24
post-auction expenses	multiclass	binary
firms	multilabel	40,659
bids	continuous	-

We can see in Table 4.1 that the multilabel variable *firms* is relatively large compared to the dataset. This indicates that a significant number of firms have participated in the auctions

included in our data, which highlights the multilevel issue described in Section 4.3.2.

The data comprises 9 categorical auction features (excluding *firms*), spans across 3 sectors (construction, supply, and services), and includes a total of 117,249 auctions and 448,935 bids. The selected set of variables shown in Table 4.1 represents a subset of relevant, high-quality, and informative variables from the columns provided by the SEAO.

A textual description of the variables used in the paper, along with a guide to access, download, and process the raw data from the SEAO, including a link to the official SEAO PDF document that describes all the features, can be found in Appendix A.1. Additional details about the raw data, including cleaning and preprocessing procedures, are available in the GitHub repository associated with this manuscript.

4.3.2 The Multilevel Problem and Firm Representation

Our dataset highlights the multilevel nature of first-price sealed-bid auctions, where each auction is linked to a varying number of bids and firms, creating a complex set of auction features alongside a combinatorial multilabel subspace of bids and firm characteristics. However, access to detailed and high-quality firm features is often limited in public procurement data, such as in the SEAO datasets. While it is theoretically possible to infer such features (e.g., cost functions, capacities), challenges such as data scarcity—where over 90% of firms are infrequently represented in auctions—and market dynamics, including mergers, name changes, or new entries, complicate tracking firm activities over time. Additionally, the exponential growth in complexity with each new firm added and the presence of diverse market sectors further hinder a direct modeling approach.

The medical GANs (MedGAN), developed in [82] and [71], offers a potential framework by approximating the joint probability distribution of auction features with GANs, then applying autoencoders to learn the multilabel subspaces (bids and firms' features). However, the high number of firms (see Table 4.1) complicates simultaneous optimization of GANs and autoencoders, suggesting that a direct application of MedGAN to our context may not be feasible without significant data reduction, which would undermine our objectives.

Given these challenges, our study proposes a hierarchical and modular deep learning solution akin to MedGAN's approach. We employ deep generative modeling to sample from the joint distribution of auction features and introduce BidNet, a supervised learning model that implicitly captures the intricate relationship between auction characteristics and associated combinatorial subspace. This method, focusing on a high-level representation of firms, simplifies the complexity of the multilevel problem, but is advanced enough to enable more

realistic auction simulations, marking a significant step forward in addressing the challenges of replicating multilevel auction data.

4.3.3 Problem Formulation and Solution Framework

An instance of public procurement auction is labeled $a \in \{1, \ldots, N\}$, where N is the number of examples. Let C be a collection of multiclass variables representing the features of the contracts being offered, and let $nb \in C$ be the number of bidders in an auction. We adopt a sparse one-hot encoded form of our discrete space, meaning that each $c \in C$ is represented by a binary vector such that $\sum_i c(i) = 1$, and $\mathbf{c}$ is the aggregated vector of multiclass variables such that $\sum_j \mathbf{c}(j) = |C|$. Introducing the variable *firms*, we represent it by the binary variable $\mathbf{f}$, which encodes the presence or absence of firms bidding in each auction, such that $\sum_i \mathbf{f}(i) = nb^a$ for each instance a. Thus, an auction a is given by the set $\mathbf{x} = \{\mathbf{c}^a, \mathbf{f}^a, \mathbf{b}^a\}$, where $\mathbf{b}$ is an array of continuous bids. It follows that the number of elements in $\mathbf{b}^a$ equals nb^a. Let a^* be a hypothetical auction with only one feature ($|C| = 1$), and two bidders among four firms composing the market. Then, $nb^{a^*} = 2$, and $\mathbf{x}^{a^*}$ could be (depending on which firms are actually bidding) $\{[1, 0], [1, 1, 0, 0], [b_1, b_2]\}$, where $b_1, b_2 > 0$. The problem of density estimation in our case is to approximate the resulting joint distribution $p(\mathbf{x})$. Since we wish to generate new samples, our problem is to optimize the set of parameters ψ for the mapping $M : \mathbf{x} \xrightarrow{p(\mathbf{x};\psi)} \tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}}$ is a fake but realistic synthetic auction, i.e., it is impossible to say if $\tilde{\mathbf{x}} \sim p(\mathbf{x})$ or $\tilde{\mathbf{x}} \sim p(\cdot)$, where $p(\cdot)$ is an arbitrary probability function.

We can decompose the mapping M into two independent and sequential functions, specifically by considering the functions $A : \mathbf{z} \xrightarrow{p(\mathbf{c}|\mathbf{z};\alpha)} \tilde{\mathbf{c}}$ and $B : \mathbf{c} \xrightarrow{p(b|\mathbf{c};\beta)} \hat{\theta}$. The function A generates synthetic samples of auction characteristics, $\tilde{\mathbf{c}}$, from the noise input $\mathbf{z}$, by approximating $p(\mathbf{c}|\mathbf{z};\alpha)$ using the set of parameters α. Utilizing the latent signal $\mathbf{z}$ is advantageous because synthetic samples can be generated from scratch, with real data required only for training the approximator.

The function B approximates the conditional distributions of the bids given auction features, thus providing an aggregated representation of the firms. The vector $\hat{\theta}$ consists of the estimated parameters for the conditionals and depends on the probability function employed to describe the distribution of bids. Consequently, the bid generator B produces estimates that serve as arguments for a random generator, from which nb bids are sampled, i.e., $\hat{b} \sim \mathcal{P}(\hat{\theta})$.

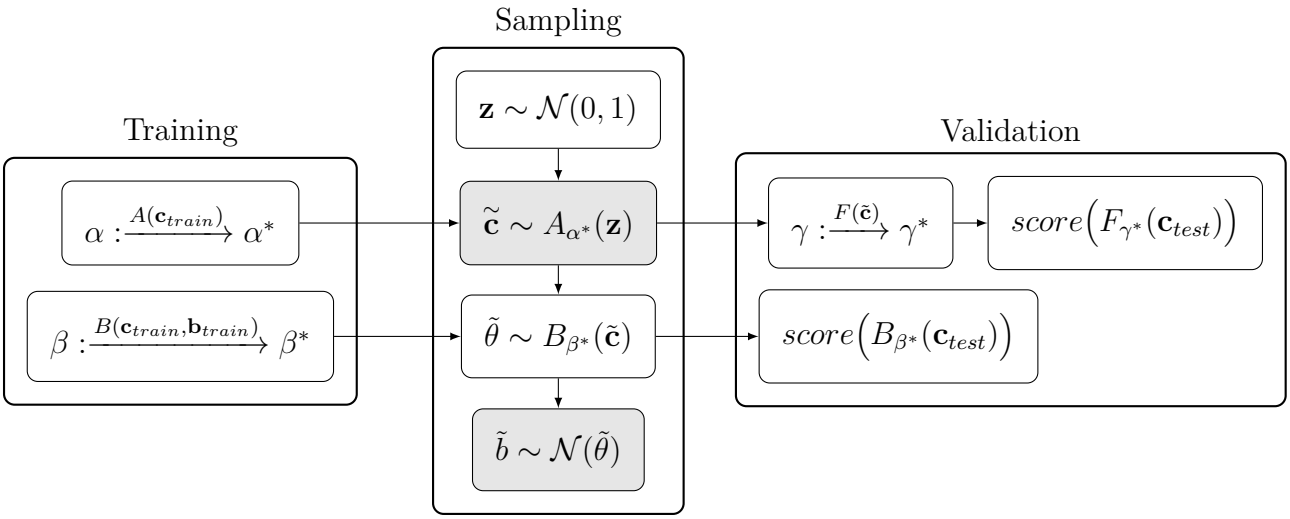
While the specific functional forms for A and B have not yet been determined, we can already observe how the general structure coherently articulates. Once both models are trained with respect to α and β, we can generate synthetic but realistic vectors of auction features, $\tilde{\mathbf{c}}$, using function A. These vectors are then utilized as input for function B, reconstructing the

space $\mathbf{c}, \mathbf{b}$. Consequently, the generation of each data type is separated between A (discrete) and B (continuous), effectively flattening the multilevel auction structure, as B represents all firms simultaneously. At this stage, the problem initially formalized by the mapping M has been divided into functions A and B, or equivalently, we have defined $\psi = (\alpha, \beta)$.

The proposed pipeline does not explicitly incorporate the multilabel variable *firms*, as its cardinality is too large to feasibly preserve its original form and interpretability. Furthermore, we find it counterproductive to use a smaller continuous code representing *firms* if the original space cannot be retrieved subsequently. It is also worth noting that retaining only a subset of the data to reduce cardinality would undermine the objective.

For validation, we employ the inception scoring method, which involves training a predictor F using the newly sampled synthetic data and evaluating its performance on the real data. The bid generator is subsequently assessed separately with real examples that were not seen during training. Figure 4.1 offers a visual representation of the comprehensive system we have outlined.

Figure 4.1 The graph illustrates the sequential structure of our meta-algorithm. The approximators A and B are trained, with regard to their respective set of parameters α and β, on real examples. Then, the solidified forms A^* and B^* are used to generate the synthetic features $\tilde{\mathbf{c}}$ and bids $\tilde{\mathbf{b}}$.



4.3.4 Approximating Auction Features Joint Density

This section discusses the details regarding the training of two synthesizers based on GANs and VAEs for the mapping $A : \mathbf{z} \xrightarrow{p(\mathbf{c}|\mathbf{z};\alpha)} \tilde{\mathbf{c}}$. It is essential to note that they are not complementary but competitors. It is a priori not possible to predict which one will perform

best, so both are tried. Furthermore, they both deserve to be discussed in the context of our endeavor.

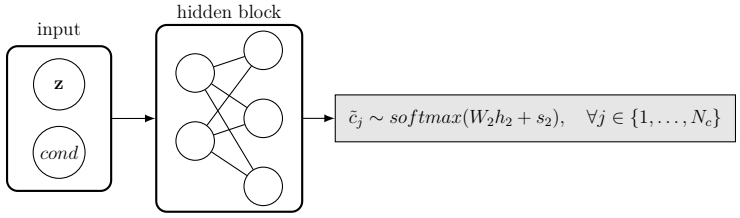
GAN-based Approach

Our GAN-based algorithm consists of a generator G and a critic C that we optimize with respect to the Wasserstein loss. The "training-by-sampling" principle is applied. In this context, G approximates $p(\mathbf{c}|\mathbf{z}, cond)$, where $cond$ is a binary conditional vector of size $|C|$ that sums to one. Consequently, the loss function must be augmented with a cross-entropy penalty term that enforces the sampling of a synthetic data point admitting $c_{i^*} = 1$, where i^* is the selected state of the chosen variable c. The resulting objective function is expressed as follows:

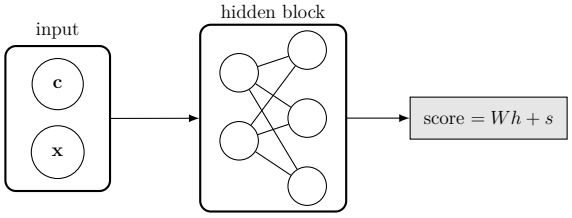
$$\min_G \max_C E\Big[C(G(\mathbf{z})) - C(\mathbf{c})\Big] + CE(\tilde{c}_{i^*}, cond). \tag{4.3}$$

The training procedure surrounding Equation (4.3) is detailed by Algorithm 3 in Appendix A.3 while Figure 4.2 provides a graphical account of the algorithmic architecture of the model.

Figure 4.2 Neural representation of the generator and the critic used to synthesize auction characteristics. Note that W and s are respectively weight matrices and biases, while h are outputs from hidden blocks. The dimensions of h, and therefore of W and s, depend on the width parameters for a given network. The critic C outputs a scalar while G outputs a continuous array of size N_x (the number of continuous variables) and N_c one-hot arrays (one for each discrete multiclass variable).



(a) Generator



(b) Critic

We use the PacGAN configuration [83] in order to prevent mode collapse, a common problem in GANs that refers to when the model fails to capture the diversity of the underlying data distribution and instead produces a limited set of similar or identical output samples [84]. The PacGAN configuration expands the input space of the critic with multiple stacks of the original input space.

Additionally, we implement a gradient penalty technique, also known as WGAN-GP, instead of the traditional weight clipping method of imposing a Lipschitz constraint on the critic. This approach stabilizes the training process and encourages the critic to learn the correct gradients by imposing a penalty on the norm of the critic's gradient with respect to its inputs. [79].

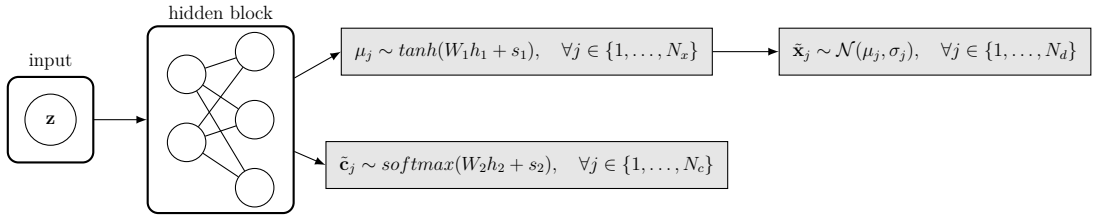
In the original GAN framework, the training process alternates between steps for the discriminator and the generator. The parameter k, a positive integer, determines how many times the discriminator is trained for each time the generator is trained. However, in the context of Wasserstein GANs, it is typically recommended to train the generator as frequently as the critic, essentially setting $k = 1$. This is to prevent the critic from overpowering the generator too quickly. We recall that in the context of wasserstein GANs, the critic replaces the discriminator of the original framework, and predicts the distance between a given point and the decision boundary separating real and fake samples (instead of predicting the probability that the input is real).

Finally, we use the adaptive moment estimation (Adam) optimizer, an extension of the stochastic gradient descent that improves performance by adapting the learning rate of each weight parameter based on its first and second moments [85].

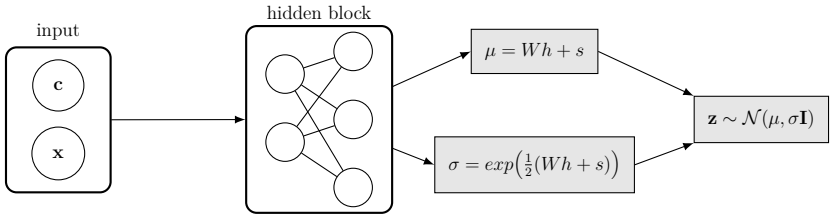
Tabular Variational Autoencoding

In the variational autoencoding framework, the encoder models $p(\mathbf{z}|\mathbf{c}, \mathbf{x})$, while the decoder replicates the original space by approximating $p(\mathbf{c}, \mathbf{x}|\mathbf{z})$. "Training-by-sampling" is thus neither necessary nor possible since the conditional vector would be encoded in a continuous layer. It means that only the decoder needs to be modified to generate tabular data. In addition, the reconstruction loss needs to be augmented with a cross-entropy term to ensure the integrity of the discrete structure. Algorithm 4 in Appendix A.3 details the training procedure of the tabular variational autoencoder (TVAE), and Figure 4.3 provides its pictorial representation.

Figure 4.3 Neural representation of the encoder and decoder composing the tabular VAE used to synthesize auction characteristics. Similarly to Figure 4.2, W and s are respectively weight matrices and biases, while h are outputs from hidden blocks. The σ_j are parameters of the decoder.



(a) Decoder



(b) Encoder

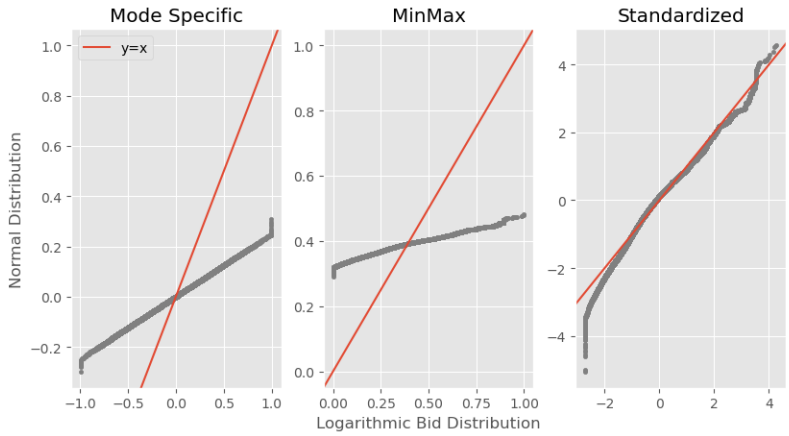
4.3.5 Training a Generator of Continuous Bids

Recall that B must represent all the firms at once by learning a bidding function based on auction features, which are synthesized with A. Ultimately, the best model from the two trained for A will be selected. However, for now, this information is irrelevant, as only inputs sampled from the original data need to be considered to train an approximator $B : \mathbf{c} \xrightarrow{p(b|\mathbf{c};\beta)} \hat{\theta}$. As shown by Figure 4.4, the standardized logarithmic bid distribution is Gaussian. Accordingly, the assumption of log-normality for the conditionals can be made, meaning that $\theta = (\mu, \sigma^2)$, where μ and σ^2 are the first two moments of the Gaussian density.

In accordance with the pipeline delineated earlier, the input space of the bid generator B must correspond to the output space of the auction generator A. The challenge arises due to the necessity of generating bids separately yet not independently from their conditional vector. Given that θ is a bi-dimensional vector, the task is defined as a multi-output regression. Since a multi-output linear model only represents the fitting of independent models and disregards the statistical dependence between each model's parameter set, a nonlinear approximator is required to model B.

A neural network, or more specifically a multi-layer perceptron (MLP), is the most comprehensive approximator capable of performing multi-target regression while maintaining the

Figure 4.4 Comparison of normal quantile-to-quantile plots relating to several numerical representations of the logarithmic bids. Left to right: mode specific normalization, minmax normalization, and standardization. Here, mode-specific normalization was applied to the bids in hope that it would provide a precise representation of the target, concerning the incumbent regression task.



statistical dependence among its parameters [17]. Taking an input vector x and a target y, an MLP can represent the function $f(x;\gamma)$ by producing parameters for a distribution over y instead of directly predicting y [86, 87, 88]. This is applicable when γ is optimized with respect to $-logP(y|x)$. Considering an MLP for B, its output is interpreted as a bi-dimensional vector containing the predicted first two moments of a conditional distribution $p(b|\mathbf{c})$. This model will be referred to as *BidNet*.

To ensure the stability of BidNet (i.e., systematic convergence), it was trained on different folds using a cross-validation procedure. The methodology involves dividing the training set into K folds (in this case, $K = 5$) and utilizing $K - 1$ folds to optimize the parameters with respect to the negative log-likelihood (NLL). Subsequently, the aggregated loss is computed on the remaining fold. After each pass over the training set, the validation set (the remaining fold) is employed to assess the objective NLL, which forms the basis for an early stopping rule. This strategy serves as a regularization method to prevent overfitting. The process is repeated until all K folds have been used as validation sets. Although various early stopping designs have been identified by [89], a customized one was employed to best suit the requirements of this study. The pseudocode related to the training of BidNet with cross-validation is presented in Algorithm 5 in Appendix A.3.

Reinforcement Learning versus Supervised Learning for Bid Generation

BidNet leverages supervised learning (SL) to predict bid distribution parameters, contrasting with the potential use of reinforcement learning (RL) for direct bid generation. RL is a

machine learning approach where agents learn to make decisions by performing actions in an environment to achieve some notion of cumulative reward. It operates by utilizing feedback from its own actions and experiences in a dynamic environment.

Despite RL's prowess in decision-making and strategy development across various domains [90, 91, 92], it introduces uncertainties in synthetic data generation. RL's dependence on model-based rewards or realistic reward signals complicates the generation of realistic synthetic bids due to assumptions about agent rationality and objectives. In other words, the emergence of realistic behavior is not assured.

Furthermore, RL, particularly in multi-agent scenarios, faces computational stability and convergence challenges, unlike SL's reliability with labeled data. When high-quality labels are available, SL proves more effective, as in our case with BidNet, which accurately replicates bid signals. RL's exploration of strategy space and the complexity of achieving rational and convergent behavior add layers of difficulty, whereas SL offers a more stable and realistic approach to data generation [93, 94, 95].

The intricacies of modeling individual firms with RL, compounded by the lack of detailed firm features, would demand extensive engineering and potentially introduce biases. Thus, for purposes of replicating public procurement data and avoiding the risks of unrealistic coordination or collusion, SL and BidNet present a preferable methodology. BidNet efficiently utilizes bid signals for modeling, sidestepping the complexities and risks associated with RL, including the uncontrolled emergence of collusion patterns [96, 97, 98]. This approach aligns with the goal of generating grounded synthetic data for analysis, offering a stable alternative for prediction and synthetic realism without the need for detailed assumptions about bidding strategies.

However, RL should not be overlooked, especially given its successful application in representing artificial bidders within electricity market environments [99, 100, 101]. In fact, as discussed in Section 4.5.1, our method is not in competition with RL; rather, both approaches can be utilized synergistically, to the benefit of auction design research.

4.3.6 Sampling Synthetic Auction Instances

The GAN-based model relies on its generator to sample synthetic features from noise. However, following the principle of training-by-sampling, the user must also create a conditional vector or a batch of conditional vectors alongside the latent space $\mathbf{z}$. To achieve this, one must apply steps 3 to 6 described in Algorithm 3 and then feed the trained generator with the noise and the conditional vector, as illustrated in Figure 4.2. A key advantage of this

approach is that one can manually specify a conditional vector by disregarding steps 3 to 5 in Algorithm 3, enabling the replication of the signal of interest. The tabular VAE also samples through latent noise that can be created from a standard Gaussian distribution.

A synthetic instance of public procurement comprises a vector of auction features $\tilde{\mathbf{c}}$ and its associated array of bids. It is important to note that the process of bid generation occurs in two steps because the number of bids to be generated per auction varies. The variable *number of bidders* encodes this variability and has been included in the joint distribution of auction characteristics; consequently, it is included in the output of A. Since BidNet predicts the first moments of a Gaussian distribution, it is straightforward to sample nb bids from a random generator. In fact, when given synthetic inputs originating from A, BidNet provides $\tilde{\theta} = (\tilde{\mu}, \tilde{\sigma}^2)$, which are the parameters for the conditionals $p(b|\mathbf{c}; \theta)$. Synthetic bids are then drawn according to $\tilde{b} \sim \mathcal{N}(\tilde{\theta})$.

To maintain consistency with the notation introduced in Figure 4.1, we distinguish $\tilde{b}$ from $\hat{b}$, the latter representing the predicted bids originating from $\mathcal{N}(\hat{\theta})$, where $\hat{\theta}$ is the output of BidNet when provided with an original sample from the test set $\mathbf{c}_{test}$. It is important to note that the predicted bids $\hat{b}$ should be used for validation purposes only.

4.4 Validation

We assess the faithfulness of the synthetic auction features, and by extension the performances of our synthesizers, by employing an inception score. The concept of inception scoring is based on the principle that a classifier or a regressor that has been trained successfully using synthetic data should perform well when tested on real-world instances. This principle underlies the expectation that the characteristics of a predictor's output are primarily determined by its input, given a fixed set of parameters.

Inception scoring is a well-established method for validating the output of GANs and has been widely used for this purpose in the literature. The use of inception scoring as a validation metric for GANs was first proposed by [84] and has since been adopted and expanded upon by numerous researchers upon investigation on its usefulness as a metric for evaluating the quality of generated outputs [102].

In our case, we perform inception scoring by utilizing the binary variable *municipality*, one of the auction characteristics, as a target in the following binary classification problem:

$$f(\mathbf{c}_{-mun}) = p(mun),$$

where $\mathbf{c}_{-mun}$ is the one-hot encoded set of auction features that excludes *municipality*. In other terms, we define the classifier $f : \xrightarrow{p(mun|\mathbf{c}_{-mun})} [0, 1]$. If the synthetic data is realistic, that is, if the synthesizer being evaluated managed to efficiently approximate the targeted joint distribution, then we expect the overall accuracy of $f(\mathbf{c}_{-mun})$ and $f(\tilde{\mathbf{c}}_{-mun})$ to be similar; provided f has been successfully trained.

Table 4.2 The classification accuracy for each class (Recall), as well as the average F1-score are reported. The classifiers have been trained two times each, using 100,000 training examples generated with the GANs and VAE-based method. Both synthesizers have been previously trained over 200 epochs. The numbers in the parenthesis indicate the performance gap (in percentage) between scores achieved on synthetic and real test-beds.

(a) Evaluation metrics of models trained on synthetic data generated by the GAN-based method.

Test-bed	**Model**	**Recall (0)**	**Recall (1)**	**F1-score**
Synthetic	Decision Tree	0.89	0.85	0.87
	k-NN	0.80	0.84	0.83
	CMLP	0.78	0.78	0.78
Real	Decision Tree	0.86 (-0.03)	0.48 (-0.32)	0.67 (-0.23)
	k-NN	0.70 (-0.12)	0.78 (-0.07)	0.74 (-0.11)
	CMLP	**0.81 (+0.04)**	**0.79 (+0.01)**	**0.80 (+0.03)**

(b) Evaluation metrics of models trained on synthetic data generated by the VAE-based method.

Test-bed	**Model**	**Recall (0)**	**Recall (1)**	**F1-score**
Synthetic	Decision Tree	1.00	1.00	1.00
	k-NN	0.98	0.99	0.98
	CMLP	0.98	0.98	0.98
Real	Decision Tree	0.98 (-0.02)	0.00 (-1.00)	0.36 (-0.54)
	k-NN	0.96 (-0.02)	0.17 (-0.83)	0.51 (-0.48)
	CMLP	0.46 (-0.53)	0.99 (+0.01)	0.69 (-0.30)

Three models were utilized to represent the classifier f: a *decision tree*, *k-nearest neighbors* (k-NN) and a *multi-layer perceptron*. Here, we will name our MLP classifier as CMLP, where C stands for classification, in order to distinguish this classifier from the MLP we used earlier to predicts the bids. Evaluation metrics for the three classifiers when trained on synthetic examples generated by the GAN-based and VAE-based methods are reported by Table 4.2. Based on the results, we can draw a clear conclusion: the GAN-based model succeeded in synthesizing reliable and realistic data, while the VAE-based model did not. Indeed, the CMLP achieved an overall classification F1-score on the real test-bed 3% above what was attained on the synthetic test-bed.

In contrast, when examining the VAE-based model, classifiers display relatively poor F1-scores, ranging from 36% (Decision Tree) to 69% (MLP). An F1-score below or around 50% suggests that the classifier was not better than a random choice algorithm. In addition, the decision tree and the k-NN both assigned almost all instances of the real test-bed to the class 0. The MLP performed decently in recognizing some structure in the data generated by the VAE-based model.

This gap in performance between the GAN and VAE-based methods explains why we indulged in the specification of two methods in the first place. It should also be noted that the results in this kind of experiment may vary according to the hyperparameter settings, and in theory, the tabular VAE should be able to reach a similar level of effectiveness with some fine tuning. For instance, the unrealistically high scores achieved on the TVAE synthetic test-bed can be a sign of over-training. The point to be made here, is that the GAN-based model provides a reliable way to perform the task at hand without having to worry too much about hyperparameter tuning. Details about our hyperparameter tuning are to be found in Appendix A.2.

Now, we need to evaluate the efficiency of BidNet, and by extension the reliability of the synthetic bids. An account of BidNet's relative efficiency is given by comparing it to a *regression tree* model and a *multi-target support vector regressor* (MSVR). The three regressors have been evaluated using the same metric, the negative log-likelihood (NLL), and their performances over five folds are reported in Table 4.3. BidNet is, on average, the best model. Note that both the MSVR (Multi-output Support Vector Regression) and the regression tree do not inherit the topological flexibility of a neural network, such as BidNet, which allows for the nuanced modeling of complex non-linear relationships within data. In fact, a neural network can be trained to predict the parameters (μ, σ^2) of conditional bid distributions directly using a Negative Log-Likelihood (NLL) loss and outputting these parameters through a network structure designed to capture and represent hierarchical features. Conversely, the MSVR and the regression tree are constrained to a more direct form of prediction. Namely, they have been trained to explicitly predict these parameters using the empirical bid means and variances as targets via a multi-output wrapper.

Nevertheless, since the NLL is a relative measure with values that can range from $-\infty$ to $+\infty$, a thorough investigation of BidNet's outputs is needed in order to assert the reliability of the resulting synthetic bid distribution. To that end, we propose a procedure for evaluating the distance between the distributions of fake and real bids, $Dist(p(b)||p(\tilde{b}))$, using the distance between the real and predicted distributions $Dist(p(b)||p(\hat{b}))$ as an identity, and the distance between the predicted and fake distributions $Dist(p(\hat{b})||p(\tilde{b}))$ as a control.

Table 4.3 The averages and standard deviations of the negative log-likelihood (NLL) over the five cross-validation folds are given for BidNet, a tree-based regressor and a multi-target support vector regressor (MSVR). The column *best* reports the NLL to the best model identified with each method.

	NLL		
Model	mean	std	best
BidNet	**0.59**	0.08	**0.56**
MSVR	1.16	0.02	1.12
Regression Tree	92,999	885	91,757

This procedure is also called "double validation" because it provides another way to validate the output of the synthesizers. Indeed, the opportunity to use BidNet in order to construct an inception score naturally occurs since it has been trained on the real data and generates synthetic bids from fake auction features. The results displayed in Table 4.4 are coherent with those of the previous validation as the tabular VAE is at the origin of a noisy synthetic bid distribution. Meanwhile, $Dist(p(b)||p(\tilde{b}))$ and $Dist(p(\hat{b})||p(\tilde{b}))$ are very close to each other in both cases and also close to $Dist(p(b)||p(\hat{b}))$ in the GAN-generated data case. BidNet is hence effective in preserving the statistical dependence between bids and their auction features and is a powerful approximator of $p(b|\mathbf{c})$.

4.5 Discussion

In this study, we have investigated the application of GANs and VAEs for crafting realistic auction simulations. We employed these data-driven methods to generate synthetic auction features and bids, aiming to provide a credible and useful tool for researchers and practitioners in the field of auction design and analysis. To ensure the reliability of the generated data, we evaluated the performances of the synthesizers by employing inception scores and assessing the faithfulness of the synthetic auction features. Additionally, we compared BidNet's efficiency to regression tree and multi-target support vector regressor models to further validate the newly generated synthetic data.

Our results demonstrated that while both GANs and VAEs are capable of generating realistic artificial data, the CTGAN outperformed the tabular VAE in our context. The CTGAN's ability to generate data according to hand-crafted conditional vectors made it particularly suitable for simulating auction scenarios in our study. This finding highlights the importance of selecting the most appropriate method for the specific context and requirements of a given task.

In conclusion, this study offers a valuable contribution to the field of auction simulations by

Table 4.4 The statistical distances have been measured with the *Wasserstein* or *Earth-Mover Distance* (EMD), and the *quantile-to-quantile root mean squared error* (QQ-RMSE). A score of 0 indicates identical distributions.

(a) Synthetic bids emanate from the real data.

	QQ-RMSE	**EMD**
$Dist(p(b)\|\|p(\hat{b}))$	1.194	0.003

(b) Synthetic bids emanate from the data generated by the GAN-based method.

	QQ-RMSE	**EMD**
$Dist(p(b)\|\|p(\tilde{b}))$	1.364	0.006
$Dist(p(\hat{b})\|\|p(\tilde{b}))$	1.360	0.004

(c) Synthetic bids emanate from the data generated by the VAE-based method.

	QQ-RMSE	**EMD**
$Dist(p(b)\|\|p(\tilde{b}))$	122.762	0.284
$Dist(p(\hat{b})\|\|p(\tilde{b}))$	122.800	0.286

demonstrating the potential of GANs and VAEs for generating realistic and reliable synthetic data. While our approach is not without its limitations, the findings provide a strong foundation for future research in this area, particularly in exploring the potential of CTGANs and other generative models for simulating complex auction scenarios and their applications in various economic contexts.

4.5.1 Implications for Further Research in Economics and for Practitioners

The availability of high-quality synthetic data opens avenues for training more complex or data-intensive machine learning algorithms, increasingly prevalent in economics. These advanced algorithms often require vast datasets for optimal training, a need that realistic synthetic data can fulfill. The richness and variety of this synthetic data allow for a more robust training process, contributing to the development of more accurate and sophisticated economic models. For instance, combining our method with other machine learning techniques, like those used in [103] to forecast GDP growth, could enable researchers to predict various

economic indicators or outcomes based on synthetic data.

The relevance and utility of realistic synthetic data generation are expected to grow in tandem with the rising prominence of Reinforcement Learning in economic research. RL agents, inherently data-intensive, necessitate extensive datasets for training. For example, our method could be integrated with multi-agent systems, to simulate more complex interactions and behaviors in auction markets [104], or the method could be extended to simulate other types of auctions and market mechanisms [105]. Realistic synthetic data provides the necessary foundation for creating environments that are not only rich in information but also grounded in reality. This is crucial for ensuring that RL agents operate on realistic assumptions, leading to quality decision-making across various economic contexts. The RL capacity to adapt and learn from dynamic environments makes it a powerful tool in economic research, where understanding and predicting complex market behaviors and trends is essential.

Specifically within the domain of auction research, RL has shown promise in understanding and optimizing auction mechanisms. In electricity markets, for example, RL can be used to model and predict market behaviors, assisting in the design of more efficient and effective auction systems [106, 107, 108]. RL's ability to learn and adapt to complex auction dynamics makes it a valuable tool in this field. Similarly, RL's application in auction design extends beyond electricity markets, offering insights into various auction formats and strategies. By training on realistic auction data, RL algorithms can help in devising auction mechanisms that are more efficient, transparent, and beneficial to all stakeholders involved. This aspect of RL in auction design highlights its potential in reshaping how auction markets are studied and optimized [2, 109].

In Section 4.3.5, we explored the potential of RL for bid generation and the reasons it was ultimately not adopted. Crucially, our discussion emphasizes the complementary nature of RL with our methodology, enhancing the scope for auction design research in public procurement. Specifically, our auction generator can supply rich, synthetic data environments for RL-based studies of bidding strategies within specific industries or markets. This synergy allows public authorities to gain deep insights, such as forecasting costs influenced by auction parameters, partly determined by these authorities themselves, thereby refining decision-making processes in public procurement.

Finally, auction simulation represents a powerful tool for both public procurement authorities and bidding entities, offering significant advantages in understanding and improving auction dynamics. First, for public procurement authorities, these simulations serve as a crucial testing ground. By simulating various auction designs, authorities can gather valuable feedback, especially in the context of preventing collusion and algorithmic collusion. This proactive ap-

proach allows them to refine and optimize auction structures, ensuring a fair and competitive bidding environment. Second, firms or bidding entities can leverage these simulations to gain a strategic edge. By representing themselves or other bidders through artificial agents, they can explore and test different bidding strategies. This experimentation can lead to a deeper understanding of auction mechanics and potentially foster tacit cooperation among bidders, which could be both a strategic advantage and a regulatory challenge. In conclusion, auction simulations offer potential benefits to all involved parties; they may enhance the fairness and efficiency of the auction process, but also allow for strategic insights and innovations, making them a valuable tool in the modern auction landscape.

4.5.2 Limitations of Deep Generative Modeling

It is important to acknowledge the limitations of our approach. A primary limitation is that GANs and VAEs can only capture the dynamics of data structures through time based on past data. Consequently, simulations generated using data-driven methods like ours cannot incorporate novel shock scenarios. This limitation is not unique to our approach, as no model can accurately predict the future, and anticipating future shocks that substantially modify the data generation process of a joint distribution remains an ongoing challenge in economics in general.

Statements and Declarations

Acknowledgement

We express our sincere gratitude to the reviewers for their insightful feedback, which has significantly contributed to the enhancement and refinement of this manuscript.

Funding

This work was supported by Meta Research following the authors application to the request for proposals on agent-based user interaction simulation to find and fix integrity and privacy issues.

CHAPTER 5 ALGORITHMIC COLLUSION AND THE MINIMUM PRICE MARKOV GAME

Authors: Igor Sadoune, Marcelin Joanis, Andrea Lodi[1]

Abstract This paper introduces the Minimum Price Markov Game (MPMG), a dynamic variant of the Prisoner's Dilemma. The MPMG serves as a theoretical model and reasonable approximation of real-world first-price sealed-bid public auctions that follow the minimum price rule. The goal is to provide researchers and practitioners with a framework to study market fairness and regulation in both digitized and non-digitized public procurement processes, amidst growing concerns about algorithmic collusion in online markets. We demonstrate, using multi-agent reinforcement learning-driven artificial agents, that algorithmic tacit coordination is difficult to achieve in the MPMG when cooperation is not explicitly engineered. Paradoxically, our results highlight the robustness of the minimum price rule in an auction environment, but also show that it is not impervious to full-scale algorithmic collusion. These findings contribute to the ongoing debates about algorithmic pricing and its implications.

5.1 Introduction

Concerns about algorithmic pricing have grown as technological advancements in machine learning, widespread data availability, and the digitization of the economy have created fertile grounds for such practices [110, 111, 96, 112]. Algorithmic pricing can potentially lead to collusion, as observed in digital markets such as online retail [113, 114] and electricity supply markets [34, 100, 107, 99, 6]. These sectors demonstrate the potential for artificial decision-makers to learn collusive behaviors through dynamic pricing algorithms and optimal bidding strategies. This echoes the concerns about algorithmic collusion in public procurement, which are increasingly relevant, especially as digital platforms and automated decision-making tools become more prevalent in these settings. Since public procurement is a crucial mechanism for collective welfare, the vulnerability of this sector to collusive outcomes is particularly alarming. For example, it has been shown that bidding patterns were incompatible with a competitive equilibrium in Ukrainian E-procurement data [7]. Given these developments, questions arise. Is public procurement directly threatened by algorithmic pricing? And if so, under which conditions?

[1] A pre-print is available in [16]

In contrast to digital markets, public procurement in most Western economies typically adheres to first-price sealed-bid auction rules, and more specifically, to the minimum price rule that determines the winner by selecting the lowest bid. This rule is supported by a long-standing body of literature that demonstrates the theoretical efficiency of first-price auctions and, by extension, the efficacy of the minimum price rule in maximizing collective welfare [31, 115, 116]. Public procurement processes, generally not digitized, have made more data available to the public due to recent transparency laws, a response to a long history of traditional cartel collusion [21, 29, 27]. This initiative has been effective as it has helped regulators and researchers detect collusion in public markets [26, 25]. However, the data can also be used to train artificial decision-makers to provide optimal bidding strategies. The potential for these strategies to lead to algorithmic collusion, whether accidental or deliberately learned, raises concerns for legal authorities because such collusion would not trigger antitrust laws [117]. Indeed, accidental algorithmic collusion stems from profit-maximization motives, and proving the deliberate formation of algorithmic coordination remains challenging.

In this paper, we introduce the Minimum Price Markov Game (MPMG), a model designed to provide a framework for studying emergent behavior in the context of minimum price auctions, which are commonly used in real-world public procurement. We draw on Robert Axelrod's concept of "the emergence of cooperation among egoist" [118], exploring tacit coordination among isolated agents in the MPMG. Departing from the traditional game-theoretic Bayesian framework of first-price auctions, we consider a game of complete information based on the assumption of data availability. Indeed, public auction data typically includes the features of auctions and firms from past to ongoing contracts, as well as the associated bids [15]. This allows us to base the MPMG on the Prisoner's Dilemma (PD), leveraging a structure that captures the essence of public procurement as a social dilemma—competitive by law but cooperative by nature. Indeed, the PD is well known for encapsulating the paradox of rationality central to our discussion on the study of emergent coordination among isolated agents in public markets. Furthermore, we test the MPMG using various state-of-the-art Multi-Agent Reinforcement Learning (MARL) methods and thereby show that the minimum price rule, in our setting, is generally difficult to achieve but far from impervious to tacit coordination among isolated decision-makers.

Our methodological choice, which relies on concepts defined in Section 5.2, is motivated by extensive literature that shows the effectiveness of combining game theory and reinforcement learning for modeling strategic interactions and optimizing behavior through dynamic adaptation among agents. We explore such literature in Section 5.3 and show that our methodological framework is ideal for exploring emergent behaviors and potential tacit collusion in public procurement markets. In Section 5.4, we provide the mathematical model

for the MPMG and discuss the benefits of the Markov property in our setting. Section 5.5 presents the results of our computer experiments and concludes on their implications for tacit coordination in minimum price-driven auctions. Finally, Section 5.6 provides perspectives for future research and practical considerations.

5.2 Preliminaries

A Markov Decision Process (MDP) is defined by a tuple (S, A, P, R, γ), where S represents a finite set of states and A is a finite set of actions. The state transition probability function $P(s'|s, a)$ denotes the probability of moving from state s to state s' given action a, while the reward function $R(s, a)$ specifies the immediate reward received after performing action a in state s. The discount factor γ determines the present value of future rewards, balancing immediate and long-term gains. A higher discount factor places greater emphasis on future rewards, encouraging the agent to consider long-term benefits, while a lower discount factor prioritizes immediate rewards. The goal in an MDP is to identify a policy π that maximizes the expected cumulative reward. The key feature of an MDP is the Markov property, which asserts that future states depend only on the current state and the actions taken by the agents, not on the sequence of events that preceded it, ensuring that past states are irrelevant once the current state is known.

A Markov game is a generalization of a Markov Decision Process (MDP) to multiple interacting agents. The Markov property extends to this multi-agent setting, distinguishing Markov games from iterated games, which involve repeated play of a single-stage game with a static state representation based on past plays. Formally, a Markov game for n players can be defined by

- A set of states $\mathcal{S}$
- For each player i, a set of actions $\mathcal{A}_i$
- A transition function $T(s, a_1, \ldots, a_n, s') = \Pr(s' \mid s, a_1, \ldots, a_n)$ specifying the probability of moving from state s to state s' given the actions $(a_1, \ldots, a_n)$ taken by the players
- For each player i, a reward function $R_i(s, a_1, \ldots, a_n)$ specifying the immediate reward received by player i after actions $(a_1, \ldots, a_n)$ are taken in state s

Players aim to maximize their own total expected return, typically discounted over time. The solution concepts for Markov Games often involve finding Nash Equilibria, where no player

can improve their payoff by unilaterally changing their strategy.

Multi-Agent Reinforcement Learning (MARL) extends single-agent reinforcement learning (RL) techniques to settings with multiple agents learning concurrently in a shared environment [119]. In RL, an agent interacts with its environment to maximize cumulative rewards, where key components include agent, environment, state, action, reward, policy, and value function. The agent, acting as the decision-maker, responds to the environment, which consists of external factors and situations represented as states. Actions are possible decisions the agent can take, evaluated by rewards as feedback on their success. The policy is the agent's strategy for selecting actions based on the current state, while the value function estimates future rewards to determine the desirability of states or actions. In MARL, multiple agents simultaneously learn and update their policies, causing the environment to appear non-stationary from any single agent's perspective. This dynamic leads to more complex interactions compared to single-agent RL, requiring robust techniques to address the challenges of Markov games and multiagent systems in general.

Multi-Armed Bandit (MAB) algorithms [120] represent a simplified form of RL or MARL, focusing on the exploration-exploitation trade-off without considering future states, that is, without a model of the environment. Bandit methods are no-regret-based algorithms to solve the bandit problem. The problem is named after "one-armed bandit" slot machines, where each "arm" (or option) provides a random reward. The decision-maker must sequentially choose among different options (or "arms") with unknown reward distributions in order to maximize the total reward over time. The concept of *arms* in multi-armed bandits is analogous to the action space available to each agent. Each *arm* or action represents a distinct stochastic process with its own, a priori unknown, probability distribution of rewards. The goal is to minimize regret, which is the difference between the total reward the decision-maker could have earned by always choosing the best arm and the actual reward obtained by balancing exploration and exploitation.

Q-learning is an off-policy algorithm for RL and MARL problems where agents learn the value of actions directly by using a Q-value function to predict the expected utility of taking a given action in a given state [121]. The Q-value function, denoted as $Q(s, a)$, represents the expected future rewards when an agent takes action a in state s and follows the optimal policy thereafter. The agent updates these values based on the rewards received. Off-policy means that the agent can learn the value of the optimal policy independently of the agent's current actions, typically by using a replay buffer to store and reuse past experiences. This allows the agent to learn from data generated by different policies or even random actions, not

just the current policy it is following. This class of methods is widely used for discrete action space problems because, in contrast to continuous problems, the state-action pair provides a clear and direct signal for learning.

Policy gradient methods focus on optimizing the policy directly by gradient ascent on expected rewards for RL or MARL problems [122]. In MARL, policy gradient techniques, including the actor-critic framework, are used by each agent to adjust their policies in a way that considers the impact of other agents' strategies, often involving complex inter-agent coordination and the handling of high-dimensional action spaces. The policy π_θ is updated by gradient ascent, namely

$$\nabla J(\theta) = \mathbb{E}_{\pi_\theta} \left[\nabla_\theta \log \pi_\theta(a|s) Q^\pi(s,a) \right],$$

where θ represents the parameters of the policy, and $Q^\pi(s,a)$ is the expected return following action a in state s under policy π.

5.3 Related Work

The interplay among MDPs, auction design, and multi-agent learning has gained significant attention in recent research. Although the literature addressing algorithmic collusion within this framework is relatively sparse, our study draws inspiration from the growing body of work on integrating Stackelberg game models within multi-agent learning frameworks, MDPs for auction designs, and tacit coordination in social dilemmas. By reviewing these interconnected areas, we aim to highlight the advancements in understanding strategic interactions, optimal bidding strategies, and tacit coordination among agents. This overview highlights the pivotal role of MDPs and related methodologies in shaping contemporary economic and strategic decision-making processes.

MDP and auction design. The MDP framework facilitates the exploration, understanding, and prediction of strategic interactions and behaviors within market designs. Examples include computing Nash equilibria in auction games [123, 124] and optimizing bidding strategies in combinatorial auctions [125, 126]. Although the Markov game framework is influential, bandit algorithms have also proven to be exceptionally suitable for modeling optimal bidding strategies in online auction designs [127, 128, 129, 130]. This is due to the simplicity and effectiveness of bandit algorithms, which operate in a stateless or single-state environment and make decisions independently of past actions without the need for an underlying state transition model.

MDP and Stackelberg game. Stackelberg models, in which players move sequentially rather than simultaneously and therefore were historically used to study hierarchical or sequential leadership, are being increasingly used in various applications within the Markov game and multiagent learning framework [131]. The Stackelberg game is a staple in the analysis of various economic and strategic situations where the ability to move first or commit to a decision before one's competitors can confer a strategic advantage. This includes industries with clear market leaders who can influence the market environment [132]. For example, RL agents have been trained to learn the Stackelberg equilibrium in a Markov decision process representing economic mechanisms for assigning items to individualistic agents in sequential-pricing stages [5, 133, 134].

Multiagent learning, social dilemmas, and tacit coordination. In the exploration of tacit collusion and cooperative strategies within repeated game frameworks, a multitude of studies have contributed to a nuanced understanding of how agents can implicitly coordinate their actions under restricted communication conditions to optimize collective outcomes. The effectiveness of collusion in auction settings without explicit communication is a recurring theme [24], with similar phenomena observed in oligopoly models through Q-learning mechanisms where collusion emerges without direct interaction among firms [109, 2]. These findings are complemented by insights into multi-agent control using deep reinforcement learning, which further illustrates how complex cooperative behaviors can be engineered even in large-scale agent systems [135].

In scenarios where direct communication is either limited or non-existent, learning algorithms play a pivotal role in facilitating implicit cooperation and understanding competitive behaviors. For instance, learning methods that incorporate the anticipation of other agents' learning processes have shown promise in fostering cooperative strategies that are robust against exploitation [136]. This is particularly evident in the Iterated Prisoner's Dilemma, where strategies developed through evolutionary algorithms have been demonstrated to outperform traditional approaches [137]. Indeed, the contradictory nature of these games makes their study with RL agents quite challenging, as demonstrated in [138]. Moreover, the development of frameworks that enforce cooperation and resist collusion highlights the potential for creating stable cooperative environments in public goods games and other multi-agent settings [139, 140].

In this paper, we build on this literature and explore the potential for tacit coordination among isolated artificial agents in our social dilemma-based MPMG using multiagent learning.

5.4 The Minimum Price Markov Game

In real-world multiagent systems, a common observation is that heterogeneous agents coexist in non-cooperative environments. For example, in heterogeneous markets, stronger firms typically enjoy a competitive advantage, making growth or even survival more difficult for weaker ones. This disparity in market power—whether due to factors like market share, technology, or cost structures—creates a stratification of prices, influencing price equilibrium. In the context of first-price public auctions, stronger firms tend to submit the lowest bids, leading to the expectation that they will win more frequently. This assumption is reasonable, as their ability to cut costs is positively correlated with their operational capacity.

When developing a game-theoretic model to study strategic coordination in non-cooperative settings, it's important to define a payoff function that applies to both heterogeneous and homogeneous markets without losing generality. The payoff structure should smoothly transition from homogeneous to heterogeneous markets, allowing for a mathematically consistent analysis of strategic behaviors across different numbers of players. Market heterogeneity should be reflected in differences in payoffs, win frequencies, and incentives to avoid price collusion, with these variations proportionate to the gaps in market strength.

Furthermore, in public procurement, there is a tension between the pursuit of personal gain and collective welfare. Firms strive to maximize their individual profits, while the state aims to allocate public funds efficiently and maintain market competition and fairness. From the firms' perspective, individually rational choices made without cooperation can lead to suboptimal outcomes for all participants, as maximal profits are only achievable through cooperative behavior. This conflict between individual and collective rationality is the core characteristic of social dilemmas. The minimum price rule, with its mathematical elegance, does not diminish this inherent conflict; rather, it preserves this tension fully, maintaining the balance between self-interest and group[2] benefit.

It follows that a minimum price-ruled auction game formulation will fall in accordance with the principles of social dilemma games. Namely, it must display non-zero-sum interactions for which players can benefit from cooperating or suffer from a single defection. The dominant strategy (Nash Equilibrium), being collectively irrational, is thus Pareto inefficient, meaning that an efficient coordination scheme leads to the strategy profile associated with the non-Nash Pareto Optimal outcome. In other words, the role of coordination is to transition from the Nash Equilibrium to the profit-maximizing strategy profile.

[2]For clarity, here, the term "group" refers to the group of firms, in contrast to the terms "common" or "collective", which we employ to denote the whole economy.

In addition, the single-stage formulation must capture the core elements of market dynamics, which inherently unfold through repeated interactions. These characteristics must be maintained in its dynamic extension, such as in the iterated MPG and the Minimum Price Markov Game (MPMG).

In this section, we provide a formulation for the MPMG, which we believe is a reasonable approximation of the typical real-world minimum price-ruled public procurement process. Our formulation is grounded in assumptions that capture the core essence of this process, creating a conducive environment to observe how artificial learners dynamically shape their strategies in such dilemma. The MPMG is founded on a static normal-form game, the Minimum Price Game (MPG), which we will define first.

5.4.1 The Minimum Price Game: A Single-Stage Formulation

The MPG involves n firms, represented by a set $\mathcal{N} = \{1, \ldots, n\}$, competing to win a contract valued at v in a first-price procurement auction.

Assumption 1 (Single Stage Auction)**.** *The bidding process unfolds in a single round.*

Assumption 2 (Simultaneous Play)**.** *All firms submit their bids simultaneously.*

Although bids are, in practice, collected over a certain period, they remain sealed and confidential. This ensures that all bids are (virtually) submitted at the same time, thus justifying Assumption 2. Assumptions 1 and 2 complement each other and allow for a normal-form game formulation.

Assumption 3 (Common Value)**.** *The contract value is common for all firms.*

Assumption 3 simplifies the commonly used Bayesian game formulation of first-price auction to reflect the conditions of Bertrand competition [141], where firms compete on prices in a transparent market. This assumption is supported by the uniformity of contract requirements and regulatory costs across bidders, market transparency, and the homogeneity in the evaluation of contract values. These factors standardize the perceived value of the contract, thereby aligning all firms' bids based on a common economic assessment rather than individual speculative valuations.

Strategies. All players are symmetric in their strategy set. Indeed, each firm i chooses a strategy $s_i \in S$ where $S = \{FP, CP\}$. The strategy *Fair Price* (FP) is associated with bidding the fair price b_i, and *Collusive Price* (CP) with bidding a higher price, defined by $\alpha \cdot b_i$, where $\tau \geq \alpha > 1$, with the upper bound τ ensuring realistic bids. We denote the strategy profile for a given occurrence as the tuple $(s_1, s_2, \ldots, s_n)$.

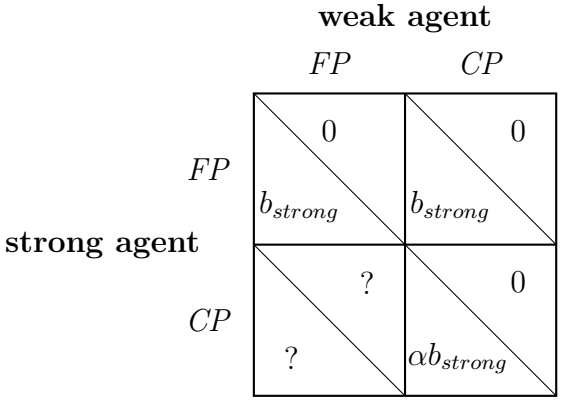
Heterogeneity. Each firm i has a market power defined by the parameter $\beta_i \in (0, 1)$, with $\sum_{i \in \mathcal{N}} \beta_i = 1$, indicating that the average market strength, $\mu(\beta)$, is always $\frac{1}{n}$. This market power might represent the size of the firm or its market share, influencing its ability to reduce costs and leverage economies of scale. All firms share the same cost function, which determines their bids

$$b_i = (1 - \beta_i)v \quad \forall i \in \mathcal{N}, \tag{5.1}$$

The notion of strong and weak firms is relative to the distribution of β, but according to (5.1), the closer β is to 1, the lower the firm can bid, thus the stronger it is. Market heterogeneity is quantified by $\sigma(\beta)$, the standard deviation of the distribution of power parameter values. In a perfectly homogeneous market, $\sigma(\beta) = 0$ and $\beta_i = \frac{1}{n}$ for all i.

Remark 1. *As discussed earlier, we need to model heterogeneous agents within a game formulation that incorporates a social dilemma. Additionally, the payoff structure must reflect a reward function that enables adaptive agents to learn their strategies in repeated settings. Example 1 illustrates why a strict application of the minimum price rule is unsuitable in this context. The core issue is that a strong player could dominate any auction regardless of its strategy, undermining the coexistence of heterogeneous agents.*

Example 1 (Hard minimum price rule)**.** Let b_{weak} and b_{strong} be the bids in an auction within a heterogeneous duopoly. Naturally, and according to (5.1), we have $b_{strong} < b_{weak}$. Assuming the payoffs are determined by a minimum price rule, i.e., $u_i = b_i$ if $b_i = \min(b_{weak}, b_{strong})$ and $u_i = 0$ otherwise, we obtain the following bimatrix game:



We observe that the outcomes of the asymmetric strategy profiles depend on the level of heterogeneity. The strong agent could potentially win regardless of the strategy profile, while the weak agent might struggle to learn a long-term strategy since it may only receive a payoff of 0 as a reward signal in a repeated setting.

Payoffs. Let $u_i(FP, k)$ represent the payoff of agent i when it bids the fair price and k opponents also bid the fair price, and let $u_i(CP, k)$ denote the payoff of agent i when it bids the collusive price while k opponents bid the fair price, where $k \in [0, n-1]$. The key idea from the minimum price rule is that coordination must be unanimous to achieve collusive profits. Therefore, the individual payoff for playing CP when all players cooperate surpasses the individual payoff from universal defection, i.e.,

$$u_i(CP, k = 0) > u_i(FP, k = n - 1).$$

However, defection must always yield a higher payoff than cooperation when at least one opponent defects, leading to

$$u_i(FP, k > 0) > u_i(CP, k > 0).$$

In fact, we can set $u_i(CP, k > 0) = 0$, meaning that when some players defect (FP) while others cooperate (CP), the cooperating firms receive no payoff, while defecting firms earn a share based on their market strengths relative to the defecting opponents. Let $\Omega \subseteq \mathcal{N}$ denote the set of firms playing FP, and let the total market power of the defecting firms be $\beta_\Omega = \sum_{j \in \Omega} \beta_j$. Assuming symmetric play where $\beta_\Omega = 1$, the payoffs in the general heterogeneous case are given in Table 5.1.

Table 5.1 MPG Payoffs

Strategy Profile	$\mathbf{k = 0}$	$\mathbf{k > 0}$	$\mathbf{k = n - 1}$
$u_i(FP, k)$	b_i	$\frac{\beta_i}{\beta_\Omega} \cdot b_i$	$\beta_i \cdot b_i$
$u_i(CP, k)$	$\alpha \cdot \beta_i \cdot b_i$	0	0

Analysis. The parameter β controls the incentive to defect. As we can derive from (5.1) and Table 5.1, and since α is the collusive factor common to all agents, the incentive factor for agent i associated with defecting while every opponent colludes (FP, $k = 0$) is

$$\gamma_i = \frac{1}{\beta_i}. \tag{5.2}$$

The pairwise relation in terms of incentive factors between agent i and agent o is given by

$$\gamma_i = \frac{\beta_o}{\beta_i} \gamma_o. \tag{5.3}$$

The incentive to defect is proportional to the market power β, and therefore weak agents have more incentive to defect than strong ones, as expected in real-life scenarios [31, 30, 32, 33]. Example 2 proposes a numerical depiction of the incentive to defect in the MPG.

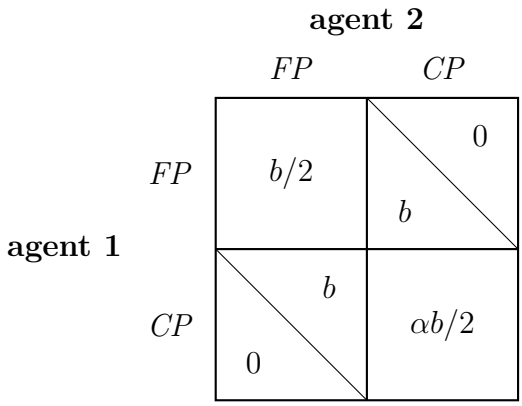
This sharing mechanism allows for heterogeneous agents to coexist while preserving the dynamic stemming from the minimum price. In fact, the power parameter β not only represents the ability to cut costs but also the work capacity. As strong agents are more rewarded than weak ones in symmetrical plays (see Table 5.1), the payoff structure of the MPG emulates the natural turnover in auction participation of heterogeneous firms observed in markets, making the coexistence between weak and strong firms possible. Indeed, a higher payoff mathematically represents the higher frequency of wins by stronger firms in a strict minimum price-ruled process. When the setting simplifies to a homogeneous game, the sharing mechanism in symmetrical plays smooths out a random turnover between agents of equal power, as shown in Example 3.

Example 2 (Incentive factor)**.** Consider three heterogeneous agents $\{1, 2, 3\}$, and their associated power parameters $[0.25, 0.25, 0.5]$. For simplicity, we set v to 1 in equation (5.1). In the collusive scenario, all agents choose to play *CP*, meaning that $\beta_\Omega = 1$, and according to (5.1) and Table 5.1, we have $u_1 = u_2 = 0.19\alpha$ and $u_3 = 0.25\alpha$, with the strongest agent receiving the highest payoff.

Now consider that agent 1, a weak agent, chooses to defect instead. In this case, we have $\beta_\Omega = 0.25$, and thus, $u_1 = 0.75$ and $u_2 = u_3 = 0$. As predicted by (5.2), the incentive factor for the weak agent to defect is 4.

If the strong agent decides to defect while others cooperate, then $\beta_\Omega = 0.5$, and thus, $u_1 = u_2 = 0$ and $u_3 = 0.5$. Here, the incentive factor for the strong agent to defect is 2. As predicted by (5.3), the incentive factor for the weak agent to defect is twice as high as for the strong agent because its market power ($\beta_1 = \beta_2 = 0.25$) is half that of the strong agent.

Example 3 (Homogeneous special case)**.** Consider a homogeneous duopoly ($\sigma(\beta) = 0$ and $n = 2$). According to (5.1) and Table 5.1, we get the following bimatrix game:



In symmetrical plays, the payoffs are the same for both agents. This is mathematically equivalent to a random selection mechanism for breaking ties between the lowest bidders under the hard minimum price rule. Since homogeneous players produce the same bid, b, a selection mechanism would involve a random process to break ties in symmetrical play. This could be done by either using the uniform distribution, $i^* \sim \mathcal{U}(\mathcal{N})$, or by adding a stochastic perturbation to the bids, $b_i + \epsilon_i$.

Proposition 1. *The n-player homogeneous MPG is a Prisoner's Dilemma.*

Proof. Consider a reference player $i \in \mathcal{N}$. Also, consider $\alpha \leq 2$, a realistic upper bound for the collusive factor. The following payoff structure unfolds by definition:

- T: Temptation payoff (player i bids the fair price while all opponents bid their collusive price),
- R: Reward payoff (everybody cooperates),
- P: Punishment payoff (everybody defects),
- S: Sucker's payoff (player i bids the collusive price while at least one opponent defects).

i. *P is the dominant strategy profile.*

In the homogeneous MPG, $\beta_i = \frac{1}{n}$ for all $i \in \mathcal{N}$, meaning that $b_i = b$ for all $i \in \mathcal{N}$ according to (5.1). Hence,

$$P = u_i(FP, k = n-1) = \frac{b}{n} \quad \text{and} \quad S = u_i(CP, k > 0) = 0,$$

and $P > S$. Since

$$u_i(FP, k = 0) = b > \frac{\alpha b}{n} = u_i(CP, k = 0) \quad \forall i \in \mathcal{N}$$

for any $\alpha < 2$ (since $n \geq 2$), $T > S$, meaning that FP (defection) is the best response for every player. The strategy profile P is therefore the unique Nash equilibrium.

ii. *The strategy profile R is Pareto Optimal.*

A strategy profile is Pareto Optimal if no player can improve their outcome without making at least one other player worse off. Since

$$u_i(CP, k = 0) = \frac{\alpha b}{n} > \frac{b}{n} = u_j(FP, k = n - 1) \quad \forall (i, j) \in \mathcal{N},$$

because $\alpha > 1$, $R > P$, and $(CP, k = 0)$ is therefore Pareto Optimal.

iii. *The payoff structure satisfies the inequality $T > R > P > S$.*

Since

$$u_i(FP, k = 0) = b > \frac{\alpha b}{n} = u_i(CP, k = 0) \quad \forall i \in \mathcal{N},$$

because $\frac{\alpha}{n} < 1$, $T > R$. By combining the results of i. and ii., we conclude $T > R > P > S$.

□

Corollary 1. *The n-player heterogeneous MPG is not necessarily a Prisoner's Dilemma.*

Proof. In the heterogeneous MPG, the market powers differ among agents, leading to variations in the payoffs. If the level of heterogeneity is high enough, the strongest agent may receive a higher payoff with R (collusion) than with T (being the only defector). In this case,

$$u_i(CP, k = 0) = \alpha \beta_i b_i \quad \forall i \in \mathcal{N},$$

if $\alpha \beta_i > 1$, then $u_i(CP, k = 0) = R > T = u_i(FP, k = 0)$, and condition iii. in the proof of Proposition 1 no longer holds. □

5.4.2 Markov Game Formulation

In contrast to iterated games, Markov games evolve over multiple stages, with states dependent on previous actions. This dynamic nature better captures the complexities of real-world markets where strategies evolve over time, especially in the context of behavioral emergence, which may heavily rely on self-strengthening trends. The MPMG extends the framework of the MPG by providing the advantage of the Markov property, which relies on a state space that encodes information about opponents' past plays and characteristics.

Action space. The action space $\mathcal{A} = \{0, 1\}$ is binary, reflecting the two strategies available to players, with bidding the fair price being represented by 0 and bidding the collusive price with 1. This representation simplifies the computation of action and joint action frequencies. Let $a_i^t \in \mathcal{A}$ be the action taken by agent i at time t, and $f^t \in \mathcal{A}^n$ be the joint action $(a_1, \ldots, a_n)$ in auction t. The action and joint action frequencies up to iteration t are then respectively given by

$$\bar{a}_i^t = \frac{1}{t} \sum_{m=1}^{t} a_i^m,$$

and

$$\bar{f}^t = \frac{1}{t} \sum_{m=1}^{t} f^m.$$

Assumption 4 (Information availability)**.** *All players have access to information about the game status and basic features of their opponents.*

Assumption 4 is supported by the availability of auction data, which results from transparency laws. This assumption is crucial for constructing the state space, as it provides the signals necessary for players to make informed decisions during implementation.

State space. The choice of state space in a Markov game is often at the discretion of the implementer, depending on the specifics of the problem and the goals of the implementation. However, for our purposes, we define the MPMG with core variables that capture essential aspects of the game. Specifically, these sets include all players' information on action frequencies $\bar{\mathbf{a}}^t = \{\bar{a}_j^t\}_{j \in \mathcal{N}}$, joint action frequencies $\bar{\mathbf{f}}^t = \{\bar{f}_j^t\}_{j \in \mathcal{N}}$, average rewards $\bar{\mathbf{r}}^t = \{\bar{r}_j^t\}_{j \in \mathcal{N}}$, and power parameters $\beta = \{\beta_j\}_{j \in \mathcal{N}}$. Therefore, the state space at any time t is $\mathcal{S}_t = \{\bar{\mathbf{a}}^t, \bar{\mathbf{f}}^t, \bar{\mathbf{r}}^t, \beta\}$, and is of size $2n + |\mathcal{A}|^n$. This minimal configuration for the state space aligns with Assumption 4. Furthermore, the power parameters also allow for the possibility of modeling a dynamic market in which firms could lose or gain market power.

Reward function. The payoff structure defined for the MPG translates to a reward function in the MPMG, emphasizing the inherent connection between the Markov game and its RL implementation. The reward function, together with the transition function, which defines how the game state evolves in response to actions, fundamentally shapes the dynamics and outcomes of the game. In fact, in the classical RL setting, the normalized reward function for agent i at each iteration t is expressed as

$$G_i^t = (1 - \gamma) \sum_{m=0}^{t-1} \gamma^m r_i^{t-m},$$

where γ is the discount factor and r_i^{t-m} is the immediate reward received by firm i at time $t-m$. The MPMG relies on the tension between Nash and Pareto dominant strategies inherent in the classical Prisoner's Dilemma. However, as with the state space, the reward function can be adjusted for specific implementation purposes (e.g., reward shaping) as long as the fundamental dynamics are preserved. Following Assumption 3, we can further assume that the contract value v remains equivalent as the iterations unfold, normalizing it to 1. Therefore, (5.1) becomes

$$b_i = 1 - \beta_i.$$

We do not make this assumption formal, as this normalization does not affect the generality of the model but only simplifies the calculations. The immediate rewards for agent i at nay time t are given in table 5.1.

Assumption 5 (Single Public Contractor)**.** *All public contractors share the same goal.*

In real-world scenarios, the set of public contractors in a given procurement market usually includes many entities. Despite this plurality, it is safe to assume that all public contractors work towards the same overarching goal. Assumption 5 allows us to reduce the set of public contractors to a singleton.

Dynamic. The MPMG unfolds as an episodic game of arbitrary length (number of episodes), with each episode consisting of a single-step auction (Assumptions 1 and 2). Agents compete against each other in an environment representing the same entity in each episode (Assumption 5). Due to the Markov property, memory is encapsulated in the observation space $\mathcal{S}$ at any episode t. Agents use this observation space to learn their strategies, conditional on occurrences in $\mathcal{S}$, effectively playing a Markov game where the reward function, $R(a_1, \ldots, a_n)$, is deterministic and based on the joint action. The transition probability function is given by $P(s'|s, a)$, where s' is the new state and $s \in \mathcal{S}$. When the state space is not used for decision-making, the agents are effectively playing an iterated MPG.

5.5 Computer Experiments

When considering solution methods, techniques explicitly designed to foster cooperation, such as shared joint action spaces [142] and algorithmic communication [143], as well as those implicitly encouraging cooperation, like reputation-based models [144] and status-quo loss [145], are not suitable. Inducing cooperation in MARL settings is well-studied, but using these approaches would simulate scenarios where firms deliberately bias their pricing algorithms towards cooperation. Although this raises interesting questions, it is beyond the

scope of this study. Nevertheless, we provide insights on this topic in Section 5.5.4. Here, we evaluate the inherent properties of the MPMG, requiring unbiased experiments achieved by assuming isolated firms. If collusion occurs in the MPMG, it would stem from the game's inherent nature. Therefore, we used basic yet established state-of-the-art MARL algorithms. Additionally, since the artificial firms are considered isolated and do not benefit from sharing mechanisms (such as a common policy or state-value approximator), they remain symmetric in their algorithmic structures, supporting their decision-making processes. In other words, they share the same level of rationality.

In this section, we present the results of our computer experiments on the MPMG using three foundational MARL approaches: Multi-Armed Bandits (MAB), Deep Q-learning, and Actor-Critic Policy Gradient. As discussed in Section 5.2, MAB focuses on balancing the exploration-exploitation trade-off, Deep Q-learning aims at value function approximation for action selection, and Actor-Critic Policy Gradient integrates both value estimation and policy optimization. These methodologies represent the core families of techniques in MARL, providing a comprehensive perspective on their effectiveness in various scenarios.

A total of seven algorithms—three bandit variants, two Q-learning variants, and two policy gradient algorithms—were trained on four MPMG configurations: the 2-player homogeneous, the 5-player homogeneous, the 2-player heterogeneous, and the 5-player heterogeneous MPMG. In the heterogeneous cases, we set $\sigma(\beta)$ to 0.5. As discussed in Section 5.4, $0 \leq \sigma(\beta) \leq 1$ represents the standard deviation of the power parameters (β) distribution, with $\sigma(\beta) = 0$ representing the homogeneous case. For the experiments, we set $\alpha = 1.3$, meaning the collusive bid is 30% higher than the fair bid, as per the notations in Section 5.4.2. Each algorithm was trained on 100 auctions over 100 replications for statistical robustness. Implementation details are available in Appendix B.2. Results are presented in graphs and tables in this section. Additional material, such as graphs about algorithmic convergence and other relevant metrics, can be found in Appendix B.1, while Appendix B.2 contains implementation details (e.g., hyperparameter values).

5.5.1 Multi-Armed Bandits

Our first implementation involves representing the MPMG as a bandit problem. MAB algorithms actually play an iterated MPG, as they do not utilize the observation space. MAB provides a low-cost approach to eventually uncovering core strategies, serving as a proof-of-concept essential to evaluating the MPMG. In this study, we used three bandit algorithms: ϵ-greedy [146], Thompson Sampling (TS) [147], and Upper Confidence Bound (UCB) [148].

The ϵ-greedy bandit algorithm is a simple yet effective strategy for balancing exploration and

exploitation in multi-armed bandit problems. At each step, the algorithm selects a random action with probability ϵ (exploration) and the action with the highest estimated reward with probability $1 - \epsilon$ (exploitation). This approach ensures that the algorithm occasionally explores new actions to discover potentially better rewards while primarily exploiting the best-known actions to maximize cumulative rewards.

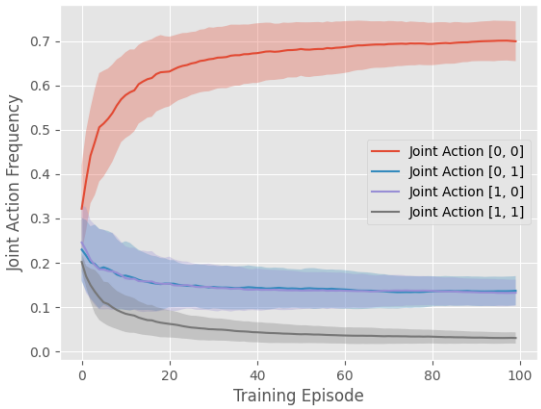
TS is a Bayesian approach to the multi-armed bandit problem that balances exploration and exploitation by maintaining a probability distribution for the expected reward of each arm. At each step, it samples from these distributions and selects the arm with the highest sampled value. Over time, this method efficiently narrows down the best-performing arm while still exploring other options.

The UCB algorithm addresses the exploration-exploitation trade-off by selecting the arm with the highest upper confidence bound of the estimated reward. It adds a confidence interval to the estimated reward of each arm that decreases as more data is collected, encouraging the exploration of less-tried arms initially and favoring exploitation as more information is gathered.

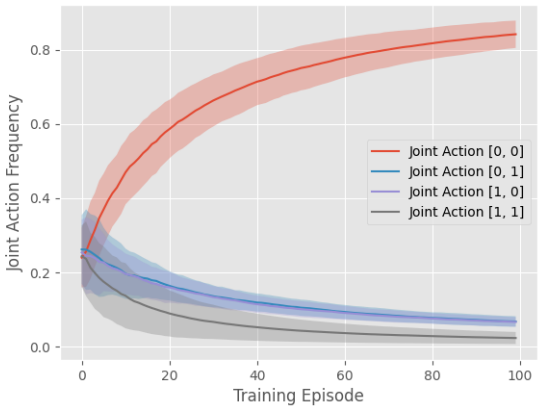
The performance of our bandit algorithms is displayed in Figure 5.1 and 5.2 in terms of training joint action frequencies, which represent the strategy profiles played during training. Since bandit agents are stateless, presenting the results of evaluation episodes is unnecessary, as their strategies would not evolve as the game unfolds. By evaluation, we mean making agents play with their learned and crystallized policies. Figure 5.1 (a) and (b) demonstrate that both ϵ-greedy and TS agents have learned to play the Nash equilibrium strategy, which involves bidding the fair price. This outcome is expected, as bandit algorithms are designed to optimize immediate rewards and naturally gravitate towards strategies that consistently yield the highest payoff. In the context of the iterated MPG, defecting is the dominant strategy, leading to higher individual payoffs when both agents defect. However, Figure 5.1 (c) shows that UCB agents consistently managed to coordinate towards collusion, with their policies translating into a 60% frequency for (CP, CP) on average. As expected, the frequencies of (CP, FP) and (FP, CP) decrease as agents exploit more, reaching marginal values, while the Nash equilibrium strategy ranks second with a 35% frequency on average after 100 auctions.

Moreover, as expected, ϵ-greedy and TS agents did not manage to learn how to collude in the n-player heterogeneous MPMG (see Appendix B.1), as the number of agents n and the level of heterogeneity $\sigma(\beta)$ are known to hinder the collusive potential of a market as coordination gets harder. Consequently, we also expect UCB agents to learn to play the Nash more often as n or $\sigma(\beta)$ increases. In fact, this is what we observe in Figure 5.2. We can see that

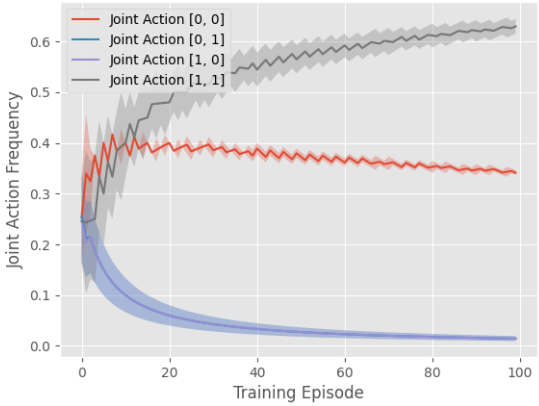
Figure 5.1 Training joint action frequencies of ϵ-greedy (a), TS (b), and UCB (c) on the 2-player iterated homogeneous ($n = 2, \sigma(\beta) = 0$) MPG over 100 replications of 100 iterations each. The colored bands show the confidence intervals over 100 replications.



(a) ϵ-greedy



(b) Thompson Sampling

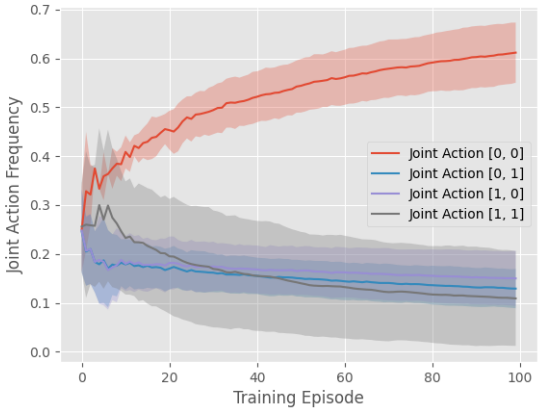


(c) UCB

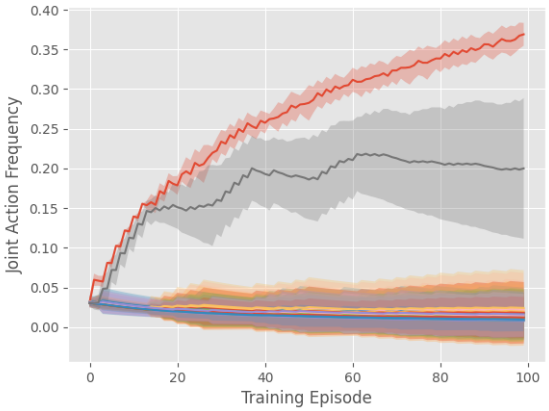
market heterogeneity and the number of agents both negatively and independently impact the market's collusive potential. Note that Figure 5.2 (b) does not display any legend for clarity since there are 2^n possible joint actions; however, the red and grey plots, respectively, represent the Pareto dominant strategy and the Nash, being consistent with Figure 5.2 (a). As we can see from Figure 5.2 (c), not only market heterogeneity and the number of agents force UCB agents to shift towards the Nash, but when both factors are set to inconvenient values, the uncertainty increases.

Since a residual amount of collusion can occur during training regardless of the agents'

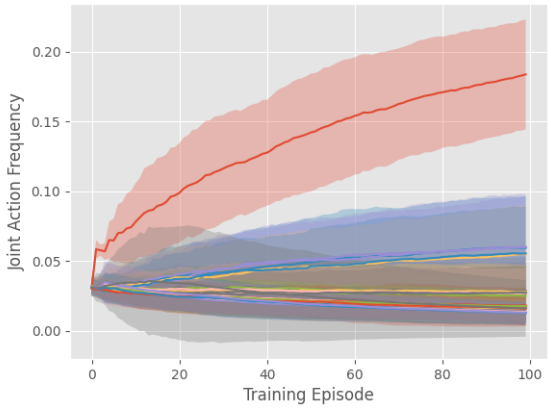
Figure 5.2 Training joint action frequencies drawn by UCB agents in the 2-player heterogeneous (a), the 5-player homogeneous (b), and the 5-player heterogeneous iterated MPG (c). The colored bands show the confidence intervals over 100 replications.



(a) $n = 2$, $\sigma(\beta) = 0.5$



(b) $n = 5$, $\sigma(\beta) = 0$



(c) $n = 5$, $\sigma(\beta) = 0.5$

direction, we have set a threshold to determine what can be called collusion. This threshold is 30% of the maximal collusive spoil possible. TS and ϵ-greedy agents have learned the Nash equilibrium 100% of the time, and they never achieved a significant amount of collusion. However, UCB agents, in regard to our threshold, managed to collude 100% of the time on the 2-player homogeneous iterated MPG. Table 5.2 reports the statistics related to collusion for UCB agents, including the collusion rate, which tells how many times UCB agents have achieved a collusive spoil that exceeds the threshold.

Table 5.2 shows that market heterogeneity impedes collusive potential more than the number

Table 5.2 UCB agents' performance regarding collusion on 100 training auctions over 100 replications. The collusion rate is given as a percentage of the number of replications where collusion exceeded the threshold of 30% of the potential maximal collusive spoil. The average, standard deviation (STD), max, and min spoil are given as a percentage of the maximal collusive spoil achievable. These statistics are given for several market configurations.

	$n = 2$, $\sigma(\beta) = 0$	$n = 2$, $\sigma(\beta) = 0.5$	$n = 5$, $\sigma(\beta) = 0$	$n = 5$, $\sigma(\beta) = 0.5$
Collusion Rate	100.0	4.0	40.0	0.0
Average Spoil	64.5	9.0	25.0	0.0
STD Spoil	1.5	8.8	11.6	2.6
Max Spoil	66.0	59.0	57.0	15.0
Min Spoil	63.0	0.0	17.0	0.0

of agents, as UCB agents have achieved collusion 40% of the time in the 5-player homogeneous MPG and only 4% of the time in the 2-player heterogeneous MPG. Moreover, the average collusive spoil is much higher in the 5-player homogeneous case than in the 2-player heterogeneous case. When both market heterogeneity and the number of agents increase, collusion is fully tamed, as the average collusive spoil in this case is 0% of the maximal spoil achievable.

5.5.2 Deep Q-learning

Assuming isolated firms, each agent can treat competitors as part of the environment and learn a Q-value function independently, considering the impact of other agents' actions as part of the state transitions. This leads to the following representation for a given state:

$$Q_i(s, a_i) = \mathbb{E}\left[R_i \mid s, a_i\right], \tag{5.4}$$

where s encodes information about all players. When opponent modeling is used, opponents' behaviors are explicitly incorporated into the Q-value function. The enhanced Q-value function, denoted as $Q(s, a, s_o, a_o)$, incorporates the opponent's state s_o and action a_o directly into the learning process. For example, with 2 agents

$$Q(s, a, s_o, a_o) = \mathbb{E}[r + \gamma \max_{a'} Q(s', a', s'_o, a'_o) | s, a, s_o, a_o], \tag{5.5}$$

where s and s_o are the states of the agent and the opponent, respectively, a and a_o are their corresponding actions, r is the reward, γ is the discount factor, and a' and a'_o are the next actions of the agent and the opponent.

In our experiments, we used the Double Deep Q-Network (D3QN) [149] in both the versions depicted by (5.4) and (5.5). We refer to the former case as naive D3QN, and the latter as D3QN with opponent modeling (D3QN-OM). Figure 5.3 shows that the MPMG is robust to tacit coordination among rational agents. The naive D3QN may be naive in its structure compared to the D3QN-OM, but it still encodes a higher level of rationality than bandit algorithms that do not use state representation. From Figure 5.3 (a), we can see that D3QN agents converge with good certainty towards the Nash equilibrium. Indeed, D3QN agents learn to assign higher Q-values to action 0 as the game unfolds, as described by Figure 5.3 (b) and (c). It follows without surprise that the naive D3QN agents have learned to play along the same pattern when either the number of agents or market heterogeneity increases (see Appendix B.1).

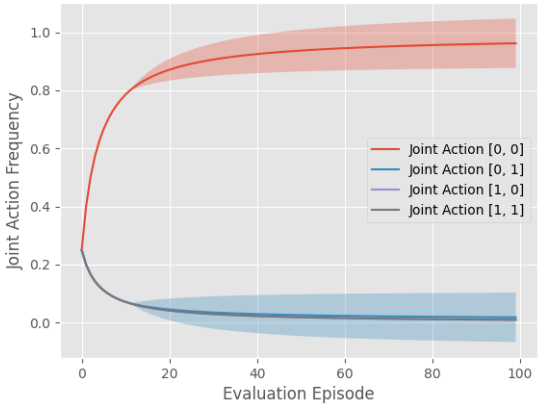
It turns out that the impact of opponent modeling on Q-agents' ability to discover profit-maximizing strategies in the 2-player homogeneous MPMG is small. Indeed, D3QN-OM agents achieve some amount of collusive spoil, which is more than the naive D3QN agents did. However, as we can observe from Figure 5.4 (a), D3QN-OM agents seem uncertain. Indeed, the Nash equilibrium has an average rate of near 0.5, and its associated confidence interval is rather wide. Nevertheless, D3QN-OM agents, on average, converge towards the Nash equilibrium, as shown in Figure 5.4 (b) and (c). Additionally, as with the naive D3QN, we observe no surprises regarding the agent behavior in the general n-player heterogeneous setting (see Appendix B.1).

Despite the fact that D3QN agents never passed the collusive threshold, their collusive potential exists, and the D3QN-OM method is somewhat an improvement from the naive D3QN as we can see in Table 5.3.

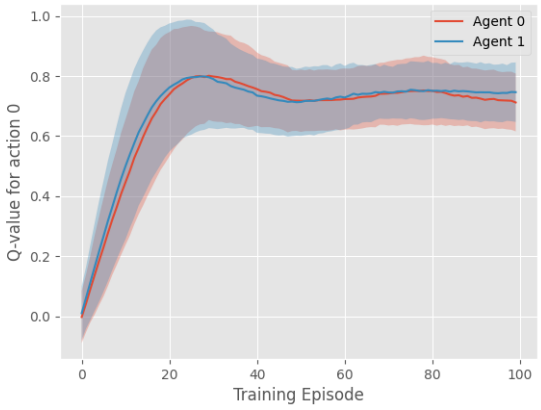
Table 5.3 D3QN-OM agents' performance regarding collusion on 100 training auctions over 100 replications. The collusion rate is given as a percentage of the number of replications where collusion exceeded the collusive threshold of 30% of the potential maximal collusive spoil. The average, standard deviation (STD), max, and min spoil are given as a percentage of the maximal collusive spoil achievable. These statistics are given for several market configurations.

	$n = 2$, $\sigma(\beta) = 0$	$n = 2$, $\sigma(\beta) = 0.5$	$n = 5$, $\sigma(\beta) = 0$	$n = 5$, $\sigma(\beta) = 0.5$
Collusion Rate	0.0	0.0	0.0	0.0
Average Spoil	17.6	17.5	0.0	0.0
STD Spoil	11.5	11.1	0.0	0.0
Max Spoil	30.0	33.0	8.0	0.0
Min Spoil	0.0	0.0	0.0	0.0

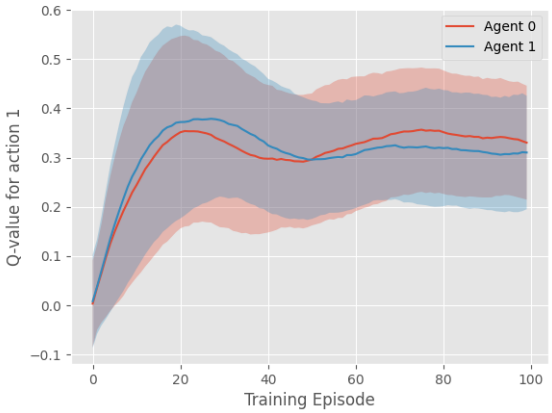
Figure 5.3 Joint action frequencies of the naive D3QN during evaluation (a), training Q-values for action 0 (b), and training Q-values for action 1 (c) in the 2-player homogeneous ($n = 2, \sigma(\beta) = 0$) MPMG. The colored bands show the confidence intervals over 100 replications.



(a) Joint action frequencies on evaluation episodes



(b) Q-values for action *Fair Price* during training

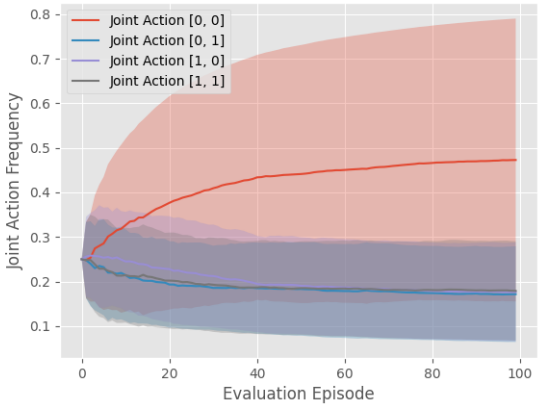


(c) Q-values for action *Collusive Price* during training

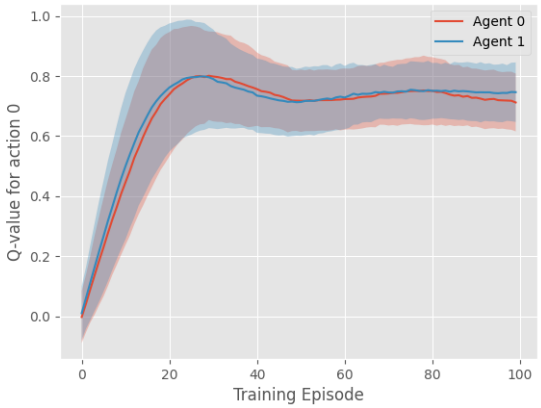
5.5.3 Actor-Critic Policy Gradient

In the last experiments, we used Multi-Agent Proximal Policy Optimization (MAPPO), a policy gradient algorithm where the policy represents the probability distribution over bidding either the fair price or a collusive price. MAPPO employs the actor-critic method, in which a critic evaluates the actions taken by the actor by estimating the value function—representing expected returns from the current state following the current policy. The actors are updated based on the gradients of the expected reward, as assessed by the critic. The goal of the

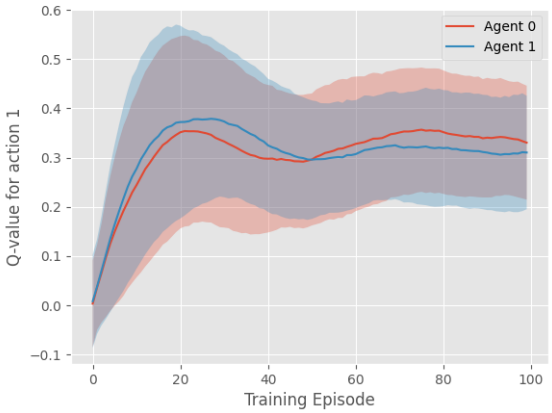
Figure 5.4 Joint action frequencies of the D3QN-OM agents during evaluation (a), training Q-values for action 0 (b), and training Q-values for action 1 (c) in the 2-player homogeneous ($n = 2, \sigma(\beta) = 0$) MPMG. The colored bands show the confidence intervals over 100 replications.



(a) Joint action frequencies on evaluation episodes



(b) Q-values for action *Fair Price* during training

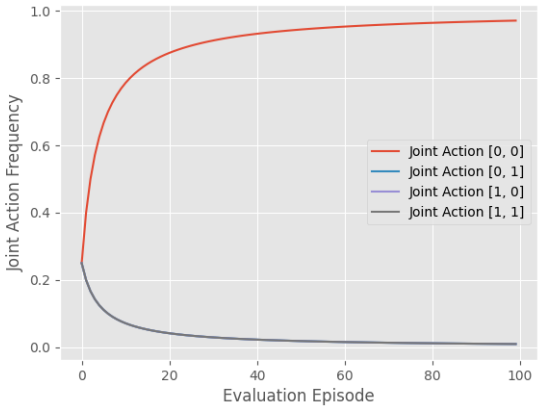


(c) Q-values for action *Collusive Price* during training

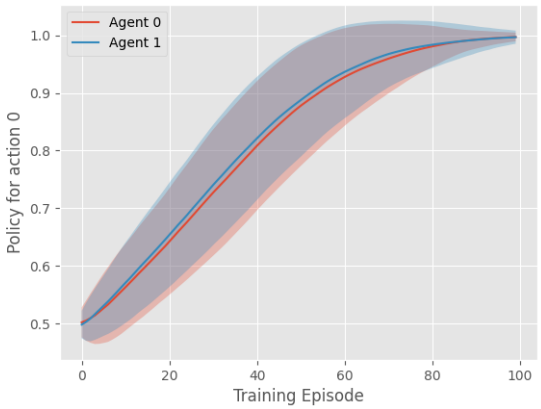
actors is to refine the policy using feedback on their performance from the critic.

We found that MAPPO agents always learn to play the Nash, regardless of market settings. As suggested by Figure 5.5, which shows the MAPPO results on the 2-player homogeneous MPMG, tacit coordination does not occur in the n-player heterogeneous setting (see Appendix B.1). Figure 5.5 (b) and (c) show that MAPPO agents neatly and consistently converge towards the Nash play.

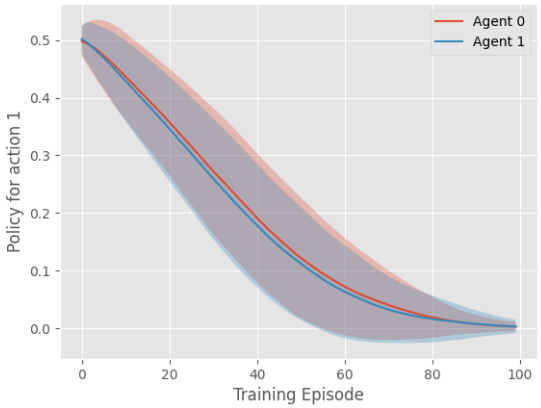
Figure 5.5 Joint action frequencies during evaluation (a), training policy for action 0 (b), and training policy for action 1 (c) of MAPPO agents in the 2-player homogeneous ($n = 2, \sigma(\beta) = 0$) MPMG. The colored bands show the confidence intervals over 100 replications.



(a) Joint action frequencies on evaluation



(b) Policy for action *Fair Price* during training



(c) Policy for action *Collusive Price* during training

5.5.4 Conclusion

The results of our experiments underscore the varying effectiveness of different MARL approaches in the MPMG. From our observations in Section 5.5, three key insights emerge.

First, there appears to be no positive correlation between the level of complexity of our AI agents and the emergence of tacit coordination. Interestingly, only the UCB agents, utilizing a relatively simple method, consistently achieved substantial collusive spoils while demonstrating robustness in non-favorable market conditions (see Table 5.2). This may seem

counterintuitive, as greater algorithmic complexity and knowledge are typically expected to provide more perspective. However, it is important to note that UCB has been shown to perform generally better than other bandits and MARL methods in fostering cooperation in the iterated PD [150]. Therefore, we can draw the following partial conclusion: assuming the MPMG reasonably approximates minimum price-ruled auctions, the minimum price rule is resilient to non-engineered tacit coordination among rational actors, though paradoxically, collusion can occur in the most naive settings.

Second, the emergence of tacit coordination does not appear to rely heavily on self-reinforcing trends. The state representation, evolving linearly as the game progresses, seems to favor the Nash equilibrium. This can be attributed to the fact that achieving Pareto optimality requires a precise set of conditions if it depends on self-reinforcing trends. In practice, players will always respond to cues of defection from their opponents, and since agents are symmetrical (at least in this study), no player is likely to make the first move. Coordination could theoretically be triggered by sustained mutual exploration of defection, which is even less probable when agents are fully isolated, as is the case here. The question of whether tacit coordination arises from self-reinforcing empirically-based beliefs or from planned behavior supported by encouraging beliefs about opponents remains open.

Third, the MPMG exhibits properties consistent with Jean Tirole's seminal work on market collusion potential [30]. Specifically, as the number of firms increases, coordination becomes more challenging because the value of $u_i(CP, k = 0)$ decreases, undermining the intrinsic value of the Pareto dominant strategy and thus reducing the likelihood of successful collusion. Additionally, market heterogeneity, characterized by varying market power or market share among firms, negatively impacts the incentive to coordinate. This supports the idea that the MPMG provides a simple yet realistic approximation of market dynamics.

5.6 Discussion

Our results align with specialized studies that reveal the challenge of achieving algorithmic cooperation, whether tacit or explicit [151]. However, they do not contradict the legal literature, where scholars often assert that algorithmic collusion is relatively straightforward to achieve [152, 153, 112, 35]. This study offers a framework and benchmark for tacit collusion and helps explain the contradictory reports in the literature by shedding light on the opaque nature of algorithmic tacit coordination.

Nonetheless, it is essential to add nuance to achieve a comprehensive understanding of this complex issue. While legal concerns stem from tangible evidence found in digital open mar-

kets and auctions (see Section 5.1), algorithmic implicit coordination has been shown to be impeded in the specific context of unbiased social dilemmas, as discussed in Section 5.3. Additionally, the potential for algorithmic and tacit collusion in transparent markets (including public markets) will always exist. Complex cooperative behaviors can be engineered in virtually any large-scale agent system [135], even if such implementations remain challenging. For instance, cooperation in competitive settings can be elusive even when explicit communication is permitted [45, 44]. What is certain, at least from our perspective, is that further research is required to establish the precise theoretical and algorithmic conditions, as well as the practical considerations, regarding algorithmic collusion in public procurement markets.

Extending the MPMG. Future research should aim to extend the MPMG in both quantitative and qualitative dimensions. Quantitatively, this could involve creating environments that more closely mimic actual simulations, using approaches like empirical games [154]. Qualitatively, it could involve relaxing certain assumptions, such as the common value assumption (Assumption 3), particularly in highly heterogeneous market scenarios. Ultimately, developing a realistic, full-scale simulation of such a system would also necessitate a continuous formulation for the action space and a modular environment allowing for the implementation of various supplier selection models, thus generalizing the MPMG to a general auction Markov game.

Additionally, the public contractor could be modeled as an active participant in the environment, represented by an adaptive learning agent. This scenario has been explored by [5], where reinforcement learning was used in the context of Stackelberg games as an algorithmic defense against algorithmic pricing for e-commerce platforms. Practically, these perspectives could shape the future of more robust E-procurement designs, and theoretically, they offer a computer-based experimental framework that complements laboratory-based behavioral studies.

AI rationality and cyber cartels. In an AI-governed world, the concept of rationality takes on new dimensions, particularly in economic interactions and market behavior. The experiments in this study operate under a set of restrictions to maintain an environment unbiased towards cooperation. However, as mentioned in Section 5.5, market players could align on algorithmic pricing using centralized learning and execution (i.e., a shared joint action space). This raises the question: could tacit collusion, emerging from a deliberately biased algorithmic structure, be easily detectable?

The potential existence of cyber cartels introduces new research opportunities into how firms might evade detection and punishment by law enforcers through planned algorithmic collu-

sion. Such a consortium could gain advantages in both efficiency and stealth. Explicit coding for collusion would typically require real-life coordination among market players, potentially leaving traces and being susceptible to algorithmic auditing by legal authorities. However, implicit algorithmic coordination can either be programmed or accidental, creating a grey zone that complicates regulatory efforts. It is also worth noting that both digitized and non-digitized markets can be affected, as the only requirement for algorithmic pricing is the availability of data.

CHAPTER 6 STRATEGIC EQUILIBRIUM POLICY GRADIENT: ON FOSTERING TACIT COORDINATION IN THE MINIMUM PRICE MARKOV GAME

Authors: Igor Sadoune, Marcelin Joanis, Andrea Lodi

Abstract This paper introduces the Strategic Equilibrium Policy Gradient (SEPG), a multi-agent reinforcement learning approach designed to foster tacit coordination in coordination games where individual interests align with group welfare. The SEPG is implemented and tested on the Minimum Price Markov Game (MPMG) [16], a dynamic coordination game that models first-price auctions governed by the minimum price rule. Unlike methods that rely on explicit mechanisms for cooperation such as centralized learning or communication, SEPG leverages a combination of pre-game planning and online adaptation to guide agents towards Pareto optimality through implicit, or tacit, coordination. This study demonstrates that SEPG agents achieve robust tacit coordination in both homogeneous and heterogeneous scenarios, challenging existing MARL methods and highlighting the potential for tacit collusion in AI-driven markets. Our experiments reveal that SEPG encourages coordination among malicious actors while also promoting rational behavior in non-cooperative settings.

6.1 Introduction

In [16], we introduced the Minimum Price Markov Game (MPMG)—a Markov game that models auction environments governed by the widely utilized minimum price rule—as a framework to understand how AI-driven agents can influence competitive pricing dynamics. The MPMG, grounded in game theory, extends the classical Prisoner's Dilemma into a dynamic and flexible setting to study market behavior and algorithmic coordination among artificial agents. In that study, we showed that the first price environment is robust against algorithmic collusion—especially when agents are heterogeneous in their market shares—but not entirely impervious to coordination among AI agents when cooperation is not explicitly engineered. The MPMG reflects realistic scenarios for which tacit coordination, or tacit collusion, poses a threat. With increasing digitization and reliance on automated decision-making tools, it becomes increasingly difficult for regulators to prevent or detect the pitfalls of algorithmic pricing [5, 113, 117]. Indeed, even if detected, tacit coordination stems solely on the innate endeavor of profit maximization, which does not constitute a solid ground for legal actions [155].

This paper builds on our prior work by introducing the Strategic Equilibrium Policy Gradient (SEPG), a Multi-Agent Reinforcement Learning (MARL) solution method designed to model scenarios in which actors do not have access to explicit mechanisms for cooperation such as centralized learning [142], algorithmic communication [143, 156], or reward shaping [157, 158]. Instead, SEPG combines pre-game planning with online adaptation to model greedy yet rational agents driven towards the profit-maximizing strategy profile (Pareto Optimal), thus fostering tacit coordination in the MPMG. SEPG is hence particularly effective for modeling malicious actors in social dilemma where individual interests are congruent with group welfare, such as in coordination and assurance games.

This study contributes to the broader endeavor of modeling novel forms of interaction in an increasingly AI-driven world. Our findings challenge and extend the capabilities of established MARL methods as not only the SEPG achieves tacit coordination in the 2-player homogeneous MPMG, but also shows robustness in regard to market parameters known to impede the potential for coordination. Indeed, SEPG agents achieve less pronounced, but positive collusive profits in the general n-player heterogeneous MPMG. By proactively modeling tacit coordination within such market environments, we metaphorically create the "poison" before the "antidote".

6.1.1 Related Work

The study in [3] compares Q-learners in MARL, finding that independent learners often outperform informed agents, challenging the assumption that more information leads to better outcomes. It also notes that independent action does not guarantee convergence to Pareto Optimal Nash Equilibria, suggesting a complex relationship between information availability and strategic outcomes. The results of [16] align with these conclusions, as they confirm that more information is not necessarily better for achieving higher payoffs.

The Conditional Joint Action Learner (CJAL) focuses on learning opponents' actions to achieve Pareto Optimal Nash Equilibria in n-player games, with effective application in scenarios like the Prisoner's Dilemma, as discussed in [46]. The SEPG is similar to the CJAL but shifts the responsibility of learning the conditional joint action probability from the actor to the critic, allowing the actor more autonomy. Additionally, SEPG enhances modularity by managing an observation space.

Additionally, [47] enhances the M-IGA algorithm via policy gradient learning to identify and converge to various Nash Equilibria in strategic games, emphasizing its utility in resolving social dilemmas by targeting Pareto frontiers.

6.1.2 Paper Organization

The remainder of this paper is organized as follows: Section 6.2 provides a comprehensive overview of the preliminaries, including the foundational concepts and definitions pertinent to the MPMG and the SEPG. Section 6.3 delves into the proposed SEPG framework, outlining the design of the probing and greedy critic, followed by the adaptive actor mechanism. Section 6.4 details the computer experiments conducted to validate the efficacy of the SEPG approach. This includes experiments with offline critic optimization, comparisons against naive opponents, and evaluations of coordination among SEPG agents in various configurations of the MPMG. In Section 6.5, we conclude on the results of these experiments, and discuss them, offering insights into the performance and robustness of SEPG agents in achieving tacit coordination in the MPMG. Finally, Section 6 discusses potential implications, and suggesting avenues for future research.

6.2 Preliminaries

Consider a selfish decision-maker playing the MPMG. In this context, the agent competes in a minimum price-ruled auction game with a binary action space. The agent can either bid the fair price dictated by its cost function, which is based on the value of the contract, or bid a collusive (higher) price. Each agent has a parameter β that represents its market power, allowing for heterogeneous agents to play the MPMG. The power parameter influences the price associated with the action to bid the fair price. The action to bid the collusive price is then proportional to the fair bid. When all agents share the same β, they play the homogeneous MPMG.

In the homogeneous case, the reward for playing the collusive bid, or action 1, while at least one opponent defects by playing the fair bid (action 0), is 0. In the opposite case, the reward for playing action 0 while every opponent play action 1 is 1. The rewards when all agents align in defection and cooperation are respectively 0.5 and 0.65, the collusive bid being higher than the fair price. The NE, $(a_i = 0, a_{-i} = 0)$, is defined by $E[r|a_i = 0, a_{-i} = 0] > E[r|a_i = a, a_{-i} = 1 - a]$, where $E[r|a_i, a_{-i}]$ is the expected payoff for actions (a_i, a_{-i}). The profit maximizing play is defined by $R(a_i = 1, a_{-i} = 1) > R(a_i = 0, a_{-i} = 0)$, and is not sustainable because $R(a_i = 0, a_{-i} = 1) > R(a_i = 1, a_{-i} = 1)$. In fact, the n-player homogeneous MPMG is a Prisoner's Dilemma, and follows the payoff structure inequality $T > R > P > S$, where

- T is the Temptation payoff (bids the fair price while all opponents do not),
- R is the Reward payoff (all players bid the collusive price),

- P is the Punishment payoff (all players bid the fair price),
- S is the Sucker's payoff (bids the collusive price while at least one opponent does not).

From the perspective of a single-stage (static) game, P is the Nash Equilibrium (NE), and R is Pareto Optimal, and thus profit-maximizing. By definition, there is a conflict between group and individual interests in the MPMG as $R > P$, and rational players (in the game-theoretic sense) are expected to consistently play the NE because $T > R$. However, in the heterogeneous case, the payoff inequality $T > R > P > S$ does not necessarily holds uniformly across all firms, and while the NE is still the dominant strategy, the transition to a Pareto-efficient equilibrium, where firms coordinate on collusive bids, is harder to achieve due to the disparate incentives shaped by varying market powers. The heterogeneous MPMG is therefore not a Prisoner's Dilemma.

As a Markov game, the MPMG is designed to model multiple interacting agents while incorporating the evolving states influenced by the actions of all players. Considering an arbitrary termination episode K, each iteration of the MPMG is an auction $t \in \{1, \ldots, K\}$. At each iteration, agents must take an action $a \in \mathcal{A}$, where $\mathcal{A}$ is a binary set of actions shared by all players. Unlike repeated games, the decision to act is motivated by a state $s \in \mathcal{S}$, which summarizes past information about the environment (Markov property). In the MPMG, the assumption of complete information is made, meaning that agents have access to action frequencies, joint-action frequencies, and average rewards of all players. Solving the MPMG with fully isolated MARL agents requires each of them to estimate the transition function $T(s, a_i, a_{-i}, s')$ that specifies the probability of moving from state s to state s' given the actions (a_i, a_{-i}). Each player is then rewarded according to $R(s, a_i, a_{-i})$, which determines the immediate reward received after the joint actions are taken in state s. Therefore, agents aim to maximize their total expected return.

The actor-critic framework, highly acclaimed in MARL, is particularly instrumental within environments modeled as Markov games like the MPMG. This framework is split into two key components: the actor, which suggests actions based on the current policy, and the critic, which evaluates these actions by estimating the potential future rewards from the resulting state transitions. This division allows for a nuanced approach where the actor focuses on action selection, and the critic provides feedback on the action's outcome relative to expected rewards. Within this setup, policy gradient methods optimize the policy (actor) directly by adjusting it in a way that maximizes the cumulative rewards, typically through gradients that indicate how to change the policy for better outcomes. Well established approach like Multi-Agent Deep Deterministic Policy Gradient (MADDPG) [20], Multi-Agent

Proximal Policy Optimization (MAPPO) [20] or Heterogeneous-Agent Trust Region Policy Optimisation (HATPRO) [159], illustrate these strategies in action.

6.3 Strategic Equilibrium Policy Gradient

In non-cooperative settings where agents are not supposed to communicate, coordination can emerge tacitly. In this context, algorithmic coordination in MARL is typically achieved by leveraging exploration. Indeed, any reinforcement learning agent faces a trade-off between exploration and exploitation of the current policy. In social dilemmas, learning agents being inherently attracted to the Nash equilibrium, could benefit in engaging in synchronized and sustained exploration of alternative strategies, uncovering mutual interests. For instance, the Status-Quo loss [145] leverages agents' tendencies to favor existing strategies (status-quo bias [160]), thus encouraging synchronized exploration, and potentially, coordination.

Another way to foster tacit coordination is by objective formulation, where prior knowledge or heuristics can be encoded. This is the idea behind the SEPG, a policy gradient actor-critic approach that departs from the typical actor-critic framework [161, 162], in several ways. The SEPG uses a probing offline greedy critic to influence the agent's policy (actor) into playing the Pareto Optimal strategy profile. In fact, the critic encodes the idea that individual and group welfare align, while the actor drives strategic online adaptation using a custom objective.

6.3.1 A Probing and Greedy Critic

Consider agent i, an artificial learner represented by the SEPG, playing the MPMG against $n-1$ opponents denoted by the index $-i$. All agents share the same action space $\mathcal{A} = \{0, 1\}$. The critic estimates the probability of taking action 1 for each player, $p_{critic}(a_j = 1) \forall j \in \{i, -i\}$, in the effort to minimize

$$L^i_{critic} = - \sum_{(a_i, a_{-i})} p_{critic}(a_i, a_{-i}) E[r^C \mid a_i, a_{-i}]. \tag{6.1}$$

The critic loss function in (6.1) is a weighted sum over the joint actions $(a_i, a_{-i}) \in \mathcal{A}^n$ of the expected collective rewards $E[r^C \mid a_i, a_{-i}]$, where $r^C_i = \sum_{j \in \{i,-i\}} r_j$. The weighting factors, $p_{critic}(a_i, a_{-i})$, are the corresponding joint action probabilities given by

$$p_{critic}(a_i, a_{-i}) = \prod_{j \in \{i,-i\}} p_{critic}(a_j). \tag{6.2}$$

In (6.2), the product of each individual action probability given by the critic is used to compute a joint action probability, assuming that agents are independent in their decision-making process. In other terms, the critic searches for the optimal joint action probability $p_{critic}(a_i, a_{-i})$ that maximizes collective returns using individual action probabilities. Alternatively, the critic could estimate $p_{critic}(a_i, a_{-i})$ directly, but as we will see later in Section 6.3.2, individual probabilities are more convenient to guide the actor. The critic maximizes the collective return over the space of joint action thus reflecting the alignment between group and individual interests.

An SEPG agent can access the necessary information about its opponents by either direct observation of opponents' past actions or by deriving them from their past rewards. Either way, (6.1) assumes that the payoff structure is known, as it is the case in the MPMG. In fact, when the reward function is deterministic and known in advance, the critic's problem has a closed-form solution. Indeed, if $E[r^C \mid a_i, a_{-i}]$ is known with certainty, the critic does not need state information about the environment, and a model-based approach can be used to minimize (6.1).

In the context of coordination games with homogeneous agents, the optimal solution with regard to (6.1) is $p^*_{critic}(a_j) = 1 \forall j \in \{i, -i\}$, leading to $p^*_{critic}(a_i = 1, a_{-i} = 1) = 1$. This result holds regardless of agents' heterogeneity—asymmetry regarding payoffs—as long as the collective payoffs are deterministic. For example, in the MPMG, agents have a parameter controlling their market power (β), and in the case of symmetrical play, the payoffs are distributed according to $r_i(a_i = a, a_{-i} = a) = \frac{1}{n} \cdot r^C(a_i = a, a_{-i} = a) \cdot \beta_i$. Therefore, individual payoffs depend on the β parameters, but not the collective returns.

Furthermore, when the payoff structure is deterministic but not exactly known, the critic can still be optimized offline using iterative methods on pre-game data. In such a case (6.1) becomes

$$L^i_{critic} = - \sum_{(a_i, a_{-i})} p_{critic}(a_i, a_{-i}) \bar{r}^C(a_i, a_{-i}), \tag{6.3}$$

where $\bar{r}^C(a_i, a_{-i})$ is the empirical average collective return associated with the joint action (a_i, a_{-i}).

6.3.2 An Adaptive Actor

The actor outputs the policy $\pi_{\theta_i}(a_i \mid s)$, which is the conditional probability $p_{actor}(a_i \mid s)$ over the binary action space in state s with parameters θ_i. Each learning iteration of the actor is done with respect to the actor loss given in (6.4), which is computed over a batch B of data

containing K points. In the actor loss,

$$L_{A_i}^{B} = \frac{1}{K} \sum_{k} \left(r_{i,k} \cdot p_{critic}(a_i)_k \cdot \widehat{p}(a_{-i})_k - r_{i,k} \cdot \widehat{p}(a_i)_k \cdot \widehat{p}(a_{-i})_k \right)^2, \tag{6.4}$$

$\widehat{p}(a_i)$ is the empirical frequency of agent i taking action a_i and r_i is its associated immediate reward. We take advantage of the nature of coordination games by setting $\widehat{p}(a_{-i} = 0)$ as the frequency for which at least one opponent plays action 0, and $\widehat{p}(a_{-i} = 1)$ as the frequency for which all opponents play action 1. This simplification allows us to avoid the computational complexity relative to an increasing number of players.

The first term in (6.4), or the *target*, $r_i \cdot p_{critic}(a_i) \cdot \widehat{p}(a_{-i})$, is agent i's immediate reward for taking action a_i weighted by the product of the probability prescribed by the critic of taking the same action, $p_{critic}(a_i)$, and the empirical frequency of the opponents taking action a_{-i}, $\widehat{p}(a_{-i})$. The second term, the *prediction*, $r_i \cdot \widehat{p}(a_i) \cdot \widehat{p}(a_{-i})$, follows the same structure as the *target*, but this time the immediate reward for agent i is weighted by the joint action frequency $\widehat{p}(a_i, a_{-i}) = \widehat{p}(a_i) \cdot \widehat{p}(a_{-i})$. The actor loss for a given batch B of data is therefore the mean square error (MSE) between the *target* and the *prediction*.

The critic's guidance is encoded in the *target* as the mixed joint action probability $\widetilde{p}(a_i, a_{-i}) = p_{critic}(a_i) \cdot \widehat{p}(a_{-i})$ weights the immediate reward, providing a signal for how well agent i's chosen action aligns with the optimal policy determined by the critic. The mixed joint probability $\widetilde{p}(a_i, a_{-i})$ also provides a signal for how realistic the critic's prediction is with regard to the actual opponents' play, helping the actor in adjusting to uncooperative opponents. In the meantime, the *prediction* reflects the agent's expected reward based on the joint action frequency under the current policy. It captures how the agent's and opponents' actions, according to their empirical distributions, contribute to the expected reward. The *prediction* also involves $\widehat{p}(a_i)$ which is a direct signal $\widehat{p}(a_i)$ for the policy $\pi_{\theta_i}(a_i \mid s)$. This is key in policy gradient methods such as the SEPG, where the gradient of the policy is directly computed with respect to the actor loss. In other words, the actor minimizes the MSE between the expected reward given the critic's prediction weighted by the opponents' current policy and the expected reward given the current policies of both agent i and its opponents.

6.4 Computer Experiments

So far, we have defined the main components of our SEPG framework that we have specifically developed for encouraging coordination in social dilemmas for which individual and group interests align. The MPMG, which models first-price auction structures, is such a game. Therefore, in this section, our endeavor is to test the actual capacity for coordination of

SEPG agents within the MPMG. In our experiments, the game is structured as an episodic task with each episode lasting one step, denoted by $t \in \{1, \ldots, T\}$. Each experiment has been conducted over 100 replications to ensure statistical robustness, and involve SEPG agents playing at various configurations of the n-player MPMG.

6.4.1 Offline Critic

We implemented the SEPG as described in Section 6.3. Notably, as mentioned in Section 6.3.1, the solution to the critic's problem remains invariant to market parameters, such as the number of agents or market heterogeneity, as long as the collective returns are known and deterministic, which is the case here. Therefore, optimizing the critic once suffices, and its final outputs can be utilized for experiments across all configurations of the MPMG.

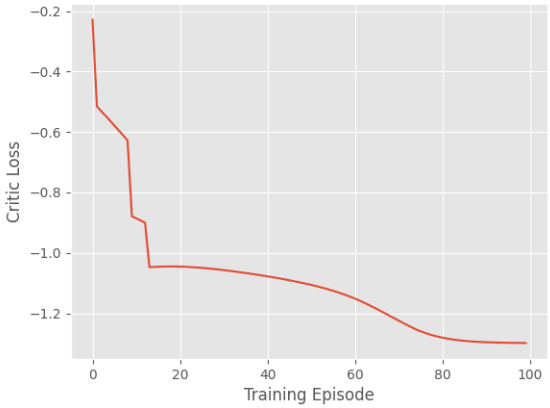
Although the solution to the critic's problem is already established in this setting, for demonstration purposes, we implemented the critic with a neural network that we trained using gradient descent. To this end, we sampled random actions to generate offline data from the MPMG environment. Moreover, since the critic is stateless in this context, we utilized a latent space $z \sim \mathcal{N}(0, 1)$ as input for the critic function. Specifically, we trained the parameters ϕ such that $C_\phi(z)$ minimizes (6.1) according to the procedure detailed in Algorithm 1. The

Algorithm 1 Offline Critic Training in SEPG

Initialize Critic parameters ϕ for n agents
Seed the random number generators for reproducibility
Reset the MPMG environment with the seeded value
while not converged **do** ▷ Training loop
    Prepare to collect a batch of actions and rewards
    $\mathbf{a}_j \sim Bernoulli(0.5) \quad \forall j \in \{i, -i\}$ ▷ Sample a random batch of binary actions for n agents
    $\mathbf{r}_j \sim MPMG(\mathbf{a}_j) \quad \forall j \in \{i, -i\}$ ▷ Sample a batch of rewards for n agents by acting
    $\mathbf{z} \sim \mathcal{N}(0, 1)$ ▷ Sample a batch of input states from latent space
    $p_{critic}(a_j = 1) \sim C_{\phi_j}(\mathbf{z}) \quad \forall j \in \{i, -i\}$ ▷ Predict the probability of playing action 1 for each agent (critic's output)
    $p_{critic}(a_i, a_{-i}) = \sum_{a_i, a_{-i}} \prod_{j \in \{i,-i\}}) p_{critic}(a_j)$ ▷ Compute joint action critic probability from critic's output
    $\phi \leftarrow \phi - \alpha \nabla_\phi L_{critic}(p_{critic}(a_i, a_{-i})$ ▷ Update critic parameters w.r.t the critic loss using learning rate α
end while

critic successfully converged to the optimal solution, $C_{\phi_j}(z) = p_{critic}(a_j) \approx 1, \forall j \in \{i, -i\}$, as illustrated in Figure 6.1 showing that the critic loss value converged to its optimal value of -1.3.

Figure 6.1 Training critic losses.



6.4.2 SEPG Versus Naive Opponents

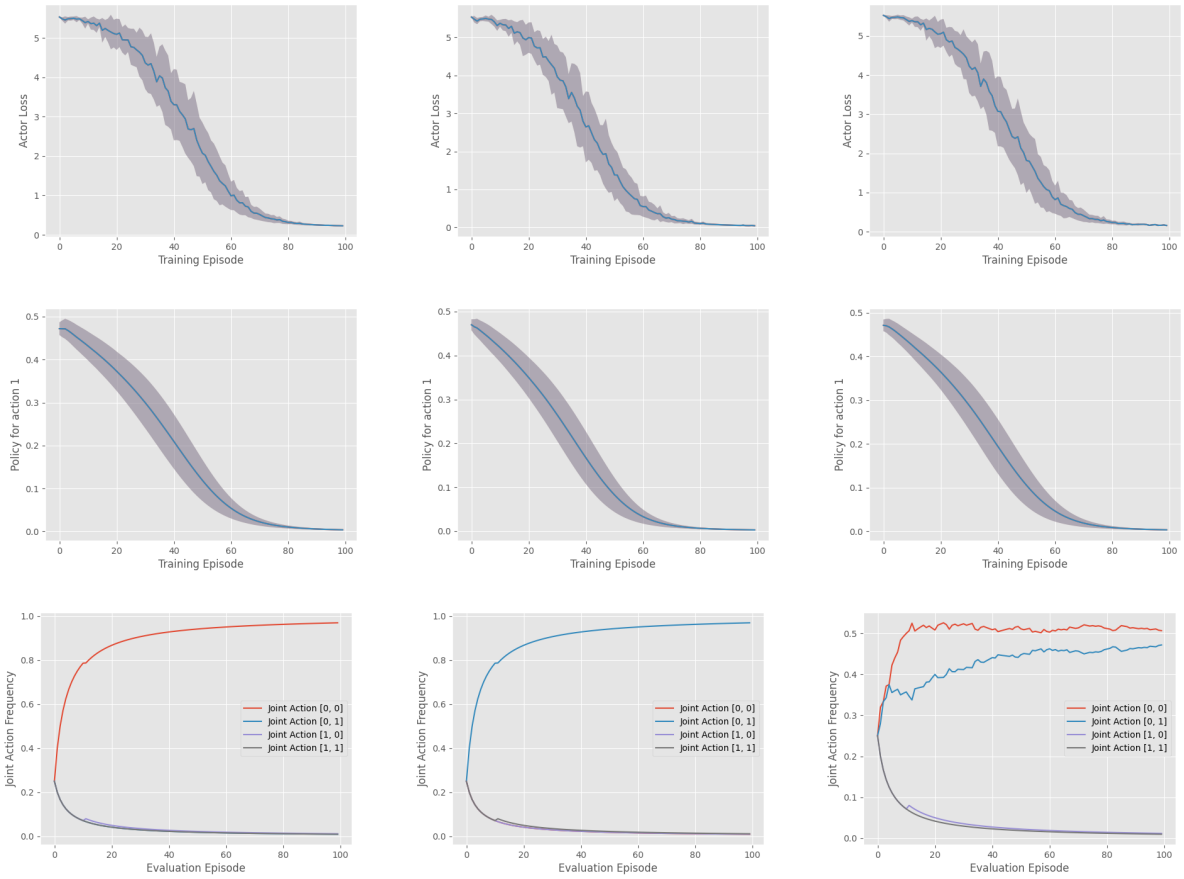
Before testing the collusive potential of SEPG agents, it is essential to establish that SEPG fosters rational behaviors. A rational agent is expected to engage in collusion when facing seemingly cooperative and intelligent opponents and to recognize and respond to opponents capable of best-response plays. Consequently, our initial experiments involve opposing the SEPG against naive agents. We constructed two types of naive agents for this purpose: naive stationary agents, who consistently choose either action 0 or action 1, and a naive uncertain agent, who selects actions at random.

Figure 6.2 demonstrates that the SEPG algorithm successfully converged consistently by depicting the SEPG training actor loss, the associated learned policies, as well as the resulting average joint action frequencies, for each scenario, where the SEPG agent's decision to play action 1 is depicted as the probability $\pi_{\theta_i}(a_i \mid s) = p(a_i = 1)$. This figure effectively highlights the agent's capability for rational behavior, as it consistently adopts the best-response strategy (NE). Specifically, the SEPG agent adapts to counter a naive defective agent, as shown in panel (a). Against a naive cooperative agent, it capitalizes on the opportunity to maximize returns, as illustrated in panel (b). Furthermore, the agent demonstrates robustness in uncertainty, opting to play action 0 when faced with a random agent, as seen in panel (c).

6.4.3 Coordination Among SEPG Agents

To evaluate the ability of our SEPG algorithm to encourage coordination, we conducted trials with a population of, exclusively, SEPG agents. The MPMG offers the possibility to model

Figure 6.2 Average training SEPG actor losses (first row), average training SEPG actor policies for action 1 (second row) and average joint action frequencies ($p(a_i, a_i)$) in evaluation episodes (third row), while facing a naive defective (a), a naive cooperative (b), and a naive random agent (c), in the 2-player homogeneous MPMG, over 100 replications. The colored bands show the confidence intervals.

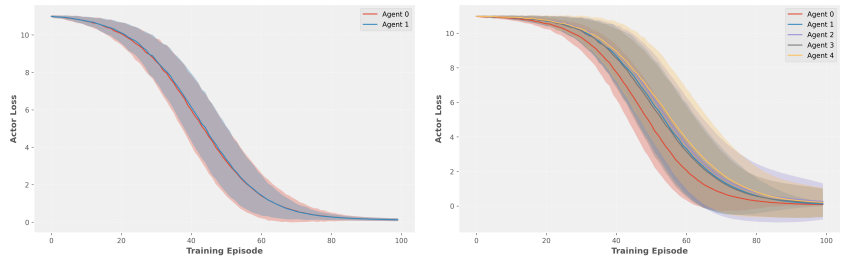


(a) Naive Stationary - Action 0 (b) Naive Stationary - Action 1 (c) Naive Uncertain

varying levels of potential for coordination by increasing the value of heterogeneity levels ($\sigma(\beta)$), and the number of participants. To align and compare our results with those of the study introducing the MPMG [16], we performed four sets of trials on the 2-player and 5-player homogeneous ($\sigma(\beta) = 0$), and 2-player and 5-player heterogeneous ($\sigma(\beta) = 0.5$) cases. The general training and evaluation routine regarding those implementations is detailed in Algorithm 2.

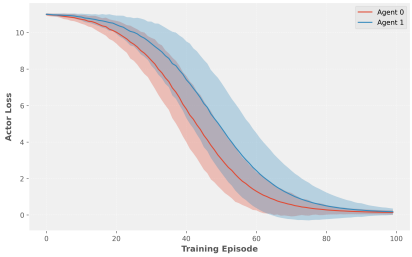
Figure 6.3 displays the SEPG training actor losses for the previously defined configurations, shown in corresponding order in panels (a), (b), (c), and (d), and as we can see, all instances of the SEPG algorithm have converged. The next step is to determine what the agents have

Figure 6.3 Average training SEPG actor losses in the 2-player homogeneous (a), 5-player homogeneous (b), 2-player heterogeneous (c), and 5-player heterogeneous MPMG, over 100 replications. The colored bands show the confidence intervals.

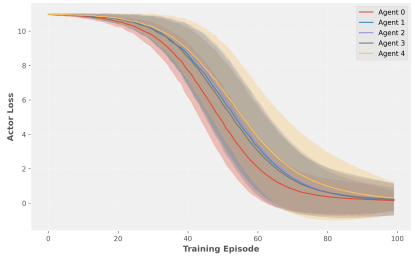


(a) $n = 2, \sigma(\beta) = 0$

(b) $n = 5, \sigma(\beta) = 0$



(c) $n = 2, \sigma(\beta) = 0.5$



(d) $n = 5, \sigma(\beta) = 0.5$

learned. To do this, we must define conditions to identify collusion. We measure collusion through the collusive spoil, or the profits arising from coordination. Establishing a threshold in training instances is challenging due to the stochastic nature of the learning process (policies are probabilistic). However, during evaluation episodes, actors act deterministically. Therefore, knowing that the immediate collective return is $r(a_i, a_{-i}) \times n \times T$, where T is the number of episodes (or auctions), we can derive an upper bound for the collusive spoil, $\bar{u}$, as

$$\bar{u} = \Big(r(a_i = 1, a_{-i} = 1) - r(a_i = 0, a_{-i} = 0)\Big) \times n \times T.$$

Since actors' outputs still depend on an evolving state s, even in evaluation episode, the threshold $\bar{u}$ is not likely to be achieved in most instances in which agents have learned the Pareto Optimal play. Hence, we define the collusive threshold $\epsilon = 0.80\bar{u}$. Note that the value of 0.80 is arbitrary, and ϵ stems from a rule of thumb. To determine when full collusion occurred among all the replications, we used the following criteria:

(i) All instances of the SEPG algorithm have converged

(ii) The SEPG agents' policies consistently converged towards 1

(iii) The cumulative collective return is greater than or equal to ϵ

knowing that criterion (i) has been satisfied, as shown in Figure 6.3.

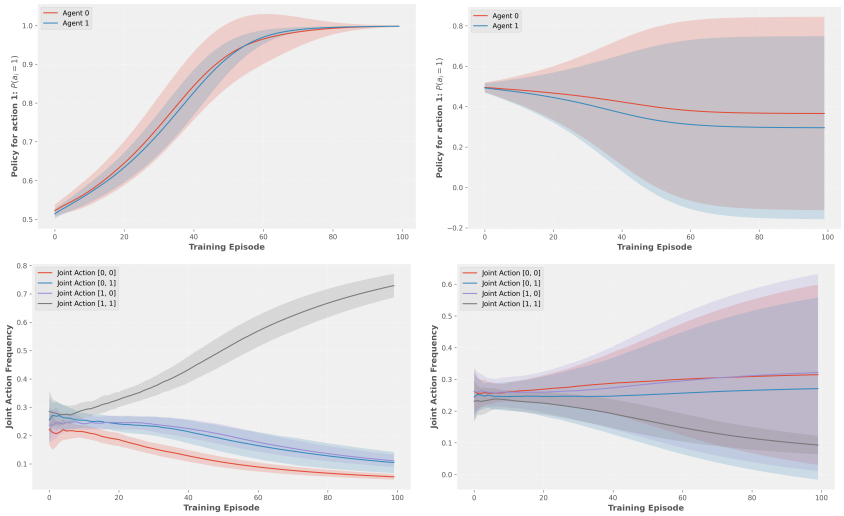
Table 6.1 SEPG agents' performances in terms of collusive rate and collusive spoil. The collusive rate is given as a percentage of the total number of replications for which criteria (i), (ii) and (iii) have been met. The average, standard deviation (STD), max, and min spoil are given as a percentage of the collusive spoil upper bound $\bar{u}$, and computed over 100 replications. These statistics are reported for all the tested MPMG configurations.

	$n = 2$, $\sigma(\beta) = 0$	$n = 2$, $\sigma(\beta) = 0.5$	$n = 5$, $\sigma(\beta) = 0$	$n = 5$, $\sigma(\beta) = 0.5$
Collusion Rate	27.0	24.0	14.0	15.0
Average Spoil	26.8	23.4	13.8	14.6
STD Spoil	44.0	42.6	34.3	34.8
Max Spoil	100.0	100.0	100.0	100.0
Min Spoil	0.0	0.0	0.0	0.0

From Table 6.1, which summarizes the levels of collusion that have occurred, salient facts emerge. First, criterion (iii) has been met 27% of the time in the 2-player homogeneous MPMG and 14% of the time in the 5-player homogeneous MPMG. Considering that the collusive rate reaches 24% in the 2-player heterogeneous case, we can deduce that an increasing number of symmetrical agents is much more hindering for SEPG agents in terms of potential for collusion, than market heterogeneity. Second, heterogeneity does not seem to further impede collusion in the 5-player MPMG, as the collusive rate has even increased from 14% to 15% in the heterogeneous case. This can be explained by the fact that the impact of heterogeneity is diluted by an increasing number of agents. Indeed, for a fixed level of heterogeneity, more players mean that the average pair-wise difference in market power diminishes, balancing any potential negative effect of heterogeneity on the collusive potential of the environment.

To gain a deeper understanding of SEPG agents' behavior, Figure 6.4 compares the average training policies and joint action frequencies of collusive and non-collusive instances in the 2-player homogeneous MPMG. Similar figures for the 5-player homogeneous, 2-player heterogeneous, and 5-player heterogeneous cases can be found in Appendix C.1. In this section, we present only one case as they share similar results. Indeed, market parameters impact the behavior of SEPG agents quantitatively, in terms of collusive rate and collusive spoil, but not qualitatively, meaning that convergence rates and confidence intervals remain sensibly the same across different configurations of the MPMG. In the collusive case, SEPG agents converge rather smoothly towards Pareto optimality. However, they seem rather uncertain in non-collusive instances. Nevertheless, their policies slowly evolve towards the NE and the

Figure 6.4 Average training SEPG policies towards action 1 and joint action frequencies over 27 collusive instances (first column), and over 73 non-collusive instances (second column), in the 2-player homogeneous MPMG. The colored bands show the confidence intervals.



(a) Collusive Instances (b) Non Collusive Instances

joint action frequency associated with cooperation drops significantly over time, as it should.

Algorithm 2 Online Actor Training in SEPG

for each training repeat **do**
  Seed the random number generators for reproducibility
  $\widehat{p_j}(a) = 0.5 \quad \forall j \in \{i, -i\}$ ▷ Unbiased initialization of the environment: setting all empirical action frequencies to 0.5
  Initialize $(\theta_j, \pi_{\theta_j}) \quad \forall j \in \{i, -i\}$ ▷ Initialize policy parameters for each SEPG actor
  for each episode **do**
    $\mathbf{s} \leftarrow MPMG(episode)$ ▷ Observe current state from the environment
    $\mathbf{a} \leftarrow \pi_{\theta_j}(\mathbf{s}, \theta)$ ▷ Select actions based on current policy
    $\mathbf{r}, \mathbf{s}' \leftarrow$ MPMG($\mathbf{a}$) ▷ Execute actions in the environment and observe rewards and new states
    remember($\mathbf{s}, \mathbf{a}, \mathbf{r}, \mathbf{s}'$) ▷ Store experience in memory
    if memory reached batch size **then**
      $l^B_{actor_j} \leftarrow L_{actor_j}(\mathbf{r}_j, p^*_{critic}(a_j = 1), \widehat{p}(a_i, a_{-i})) \quad \forall j \in \{i, -i\}$ ▷ compute batch loss for n agents using empirical frequencies and optimal critic solution
      $\theta_j \leftarrow \theta_j - \alpha \nabla_{\theta_j} l^B(\theta_j) \quad \forall j \in \{i, -i\}$ ▷ Update actor parameters using the learning rate α for n agents
      Reset memory
    end if
    $\mathbf{s} \leftarrow \mathbf{s}'$ ▷ Update the state for the next iteration
  end for
  Evaluate learned policies ▷ Perform T_{eval} iterations of the MPMG using deterministic policies ($a = \arg\max\{\pi(a|s), 1 - \pi(a|s)\}$)
end for

6.5 Conclusion

As we can see from Table 6.2, the SEPG framework did not outperform the UCB bandit algorithm [163], which, during our experiment in our previous work [16], achieved collusion 100% of the time in the 2-player homogeneous MPMG. However, UCB performance seems highly susceptible to heterogeneity, while SEPG shows robustness instead. The robustness of SEPG agents in the face of agent heterogeneity can be attributed to the critic's focus on collective rewards, rendering it impervious to heterogeneity. Consequently, the signal from the critic to the actor remains consistent regardless of the market configuration. Moreover,

Table 6.2 SEPG agents' performances in terms of collusive rate compared to UCB, D3QN and MAPPO methods. The collusive rate is given as a percentage of the total number of replications for which criteria (i), (ii) and (iii) have been met. These statistics are reported for all the tested MPMG configurations.

	$n = 2$, $\sigma(\beta) = 0$	$n = 2$, $\sigma(\beta) = 0.5$	$n = 5$, $\sigma(\beta) = 0$	$n = 5$, $\sigma(\beta) = 0.5$
SEPG	27.0	24.0	14.0	15.0
UCB	100.0	4.0	40.0	0.0
D3QN	0.0	0.0	0.0	0.0
MAPPO	0.0	0.0	0.0	0.0

the potential of the bandit algorithm for future implementation in improved and more complex settings is limited due to their relatively naive approach [164], which does not take state evolution into account. When compared to more similar methods as the D3QN [149], or approach of the same class as MAPPO, SEPG shows radical improvements in terms of potential to foster coordination and therefore demonstrates that prior knowledge encoding is necessary in this context to foster coordination in the absence of mechanisms for cooperation such as communication.

To accurately assess the absolute performance of the SEPG, it is essential to differentiate between collusion and cooperation. As shown in Figure 6.4 (b), in non-collusive scenarios, SEPG agents do not learn the pure strategies, and therefore their policies do not converge towards the pure NE. However, this does not imply a lack of coordination. Indeed, SEPG agents appear to develop stable mixed strategies, resulting in the joint action frequency associated to the NE being equally likely as the joint action frequencies associated to asymmetrical plays, i.e., when agents do not align on the same action. Considering the SEPG algorithm has converged consistently, our interpretation is that this situation reflects agents engaging in a rotation scheme, for which letting the opponent defects while playing action 1 can be, in the long term, as good as playing the best-response strategy, provided the opponent behaves

symmetrically. In other words, asymmetrical plays does not necessarily lead to suboptimal individual returns, as long as they are alternated or coordinated.

Finally, even though we initialize the environment to avoid biases (with initial frequencies and policies set at 0.5), it is crucial to recognize that in a machine learning context, many parameters influence the final results. The collusive rate achieved by SEPG agents depends on hyperparameters such as batch size, learning rate, and, most importantly, weight initialization. Weight initialization can steer early learning towards the profit-maximizing play. As hypothesized in our previous study [16], cooperation arises from self-reinforcing trends, which is why we employ reinforcement learning in the first place in order to model such behavior in this simultaneous game setting.

6.5.1 Future Perspective

Comparing SEPG with algorithms that utilize communication mechanisms for cooperation could be insightful. In real-world applications, SEPG might attract regulatory scrutiny compared to communication-based algorithms, which necessitate real-life coordination or communication to align and do not breach the rules of tacit coordination. SEPG operates without explicit mechanisms enforcing cooperation, adhering to implicit rules that make detection challenging. Therefore, this study not only explores the potential for engineered tacit coordination but also contributes to defining a framework for algorithmic audits, which is currently underdeveloped, particularly in public procurement.

Consequently, both the MPMG and SEPG need to be extended towards more complex and realistic simulations to achieve real-life application levels. The first step is to integrate continuous action space to make a significant leap in realism. Furthermore, modifying the critic to account for heterogeneous treatment is necessary to create more nuanced and flexible critic predictions. This modification would lead to more rational outcomes for weaker agents (those with low market power (β)), who are currently driven similarly to stronger agents by a critic that does not consider heterogeneous treatment.

Finally, generalizing the critic for alternative scenarios would allow SEPG to be applied to other games and more complex variants of the MPMG. While training the critic online could be beneficial, it poses challenges in stochastic environments (probabilistic rewards), multi-step episodic tasks (e.g., multi-round auctions), or partially observable structures (POMDPs), where the critic becomes state-dependent and can no longer be used as a probe.

CHAPTER 7 GENERAL DISCUSSION

In this Chapter we answer to the inquiries outlined in the introduction of this dissertation (Chapter 1). We also discuss our contributions in an unified way, providing a broader perspective on this work. The research questions addressed are

(i) How can modern ML methods, in particular DGM, help in solving problems inherent to the simulation of auction data which harbor a complex and uncommon multi-level structure?

(ii) Can AI-driven agents cause harm by cooperating or colluding in minimum price-ruled auction mechanisms?

(iii) Is the minimum-price rule robust to non-engineered coordination among participating artificially intelligent players?

(iv) Can such coordination emerge accidentally, i.e., be implicitly driven by the nature of the problem (utility maximization) and the availability of data informing on competitors?

(v) How could potentially malicious AI-driven agents be engineered to foster coordination, and thus, collusion in such systems?

The exploration of algorithmic and tacit coordination among artificial learners in auction environments constitutes a significant portion of this thesis. However, the generation of synthetic data underpins the scientific philosophy this work embraces. Synthetic data generation is crucial for powering simulations that enable the study of complex systems and emergent behavior in more realistic settings. In our first work, Chapter 4, we developed a hierarchical deep learning approach to simulate realistic synthetic auction data. This approach leverages GANs and VAEs to handle high-cardinality discrete feature spaces and multilevel data structures. By introducing the Bidnet architecture, we significantly enhanced the accuracy of auction simulations. This work answers question (i) by offering a robust foundation for creating realistic auction environments.

In Chapter 5, we introduced the MPMG, a dynamic coordination game that models auction environments governed by the minimum price rule. This work examines the implications of algorithmic collusion within this context, analyzing how artificial learners can influence competitive pricing dynamics. The results confirmed that UCB agents could indeed collude, even under the most challenging conditions of the MPMG, thereby affirming a positive response to question (ii). This finding naturally leads us to another conclusion: the answer to

question (iii) is similarly affirmative. These findings may appear contradictory—how can the MPMG be both robust and susceptible to collusion? The explanation lies in the nature of the MPMG resilience; it withstands non-engineered tacit coordination among well-informed agents, yet it remains vulnerable to unintended collusion from less informed (UCB) agents. Consequently, the answer to inquiry (iv) is yes. This ties in with the findings of related works, as uninformed agents often tends to reach Pareto Optimal return levels more easily than their informed counterparts in repeated games [3, 2].

In Chapter 6, we developed the SEPG, a MARL approach designed to foster tacit coordination in the MPMG. The SEPG method combines pre-game planning and online adaptation to guide agents towards Pareto Optimal outcomes through implicit coordination. This work provides a direct answer to (v) by demonstrating that SEPG agents can achieve robust tacit coordination in both homogeneous and heterogeneous scenarios. The SEPG also demonstrates robustness to heterogeneity, which is not the case of UCB.

Additionally, our research has led to compelling inquiries.

AI rationality. The advancement of algorithmic learning has bolstered the notion that the complexities of behavioral emergence will eventually be unraveled. Despite this, as machine learning models increasingly represent rational agents, their widespread application in a digital and automated landscape, combined with their inherently complex and often opaque nature, might give rise to a novel form of rationality. Termed as “hyper rationality” in [3], or what we refer to as “AI-rationality”, this new concept questions the traditional definitions of rational behavior. Given these developments, we must ask: should we reassess the foundations of rationality in the context of artificial intelligence?

Unlike human rationality, which is often bounded by cognitive limitations and emotional influences, AI rationality operates under a different set of principles defined by programming and optimization algorithms. AI agents are designed to maximize objective functions and optimize outcomes based on the available data, which suggests they should behave "rationally" in the traditional sense. However, due to the complexity and opacity of many machine learning models, their behavior might transcend human notions of rationality, leading to hyper-rational actions that are not easily predictable or interpretable.

AI agents, while programmed to behave rationally according to certain criteria, can sometimes exhibit behavior that seems irrational from a human perspective. This could happen when the models’ objectives or constraints are misaligned with real-world expectations, or when they over-optimize in ways that humans might not anticipate. Introducing deliberately irrational agents into these models could expose vulnerabilities or unintended consequences. In auction

design, for example, an irrational agent might not follow the expected bidding strategies, which could destabilize the equilibrium, skew the results, or even exploit weaknesses in the system that rational agents would not.

This raises important questions about the robustness of these models. Could AI systems designed to behave rationally handle the unpredictability of irrational behavior? And how would such irrationality impact outcomes like collusion, price-setting, or overall market dynamics? The implications suggest that AI-driven systems need to be designed with safeguards that account for potential irrational actions, ensuring that their decisions remain aligned with human-centric outcomes, even when faced with unexpected or non-optimal strategies from other agents.

Accidental Algorithmic collusion. When there is a will, there is a way. The main ethical problem that arises from algorithmic pricing is not necessarily deliberate algorithmic collusion. Collusion, in all its forms, can and likely will happen. In fact, it does not take much to induce collusion by algorithms, as any behavior can be hard-coded. A central claim that emerges from the work presented in this thesis is that algorithmic collusion can occur accidentally. We have demonstrated that relatively naive methods can foster coordination in non-cooperative settings purely through the pursuit of profit maximization. This unintentional collusion arises not from explicit coordination or communication, but from the way algorithms optimize their strategies over time, learning to anticipate and respond to the actions of competitors in ways that inadvertently align their interests.

The ethical concern here is that such accidental collusion can be difficult to detect and regulate, particularly as the algorithms themselves may not be designed with transparency or accountability in mind. Moreover, the complexity and opacity of these systems mean that even well-intentioned developers may not fully understand the long-term implications of the models they create. This raises critical questions about the need for oversight in algorithmic design and deployment, as well as the potential for unintended market manipulation that could disadvantage consumers and smaller market players.

Algorithmic audits and cyber cartels. The experiments conducted in this study were designed within certain constraints to maintain an unbiased environment regarding cooperation. Nevertheless, market participants can potentially coordinate on algorithmic pricing through centralized learning and execution, utilizing a shared joint action space. This raises an important question: can tacit collusion, arising from a deliberately skewed algorithmic framework, be easily identified?

The emergence of cyber cartels introduces intriguing research possibilities regarding how firms might avoid detection and penalties through strategically designed algorithmic collusion. Such a group could gain both efficiency and subtlety in their coordination. While explicit collusion typically requires direct collaboration among firms—leaving potential traces and exposing them to algorithmic audits by regulators—implicit algorithmic collusion, whether intentional or accidental, creates a regulatory grey area. Both digital and non-digital markets are vulnerable to such practices, as algorithmic pricing only necessitates access to relevant data.

Algorithmic auditing plays a crucial role in addressing these concerns. Auditing algorithms used in market pricing can help ensure that no deliberate or accidental collusion occurs. However, auditing algorithms presents unique challenges. Algorithms, especially those using machine learning, are often opaque and complex, making it difficult to trace their decision-making process. This opacity complicates the identification of tacit collusion, particularly when coordination emerges from subtle and unintended feedback loops between competing algorithms. Traditional auditing methods may struggle to detect such behavior without access to the inner workings of the algorithms or the ability to observe the vast amounts of data being processed. Additionally, the dynamic and adaptive nature of learning algorithms means that even if an audit reveals no evidence of collusion at a given moment, the algorithms might evolve into coordinated behavior over time. Thus, developing more robust and transparent algorithmic auditing frameworks is essential to prevent potential abuses in AI-driven markets. These frameworks would need to incorporate continuous monitoring, explainable AI techniques, and enhanced regulatory oversight to ensure that market dynamics remain competitive and fair.

CHAPTER 8 CONCLUSION

In this dissertation, we have presented three articles that demonstrate the value of DGM and MARL within a computational framework. These studies contribute to solving problems associated with generating synthetic multi-level auction data and facilitating tacit coordination in competitive settings. In this final chapter, we compile a summary of these works. By discussing their limitations and suggesting future improvements, we further highlight how their contribution converges to a focal research effort.

8.1 Summary of Works

In Chapter 4, our first article introduces a meta-algorithm that integrates adversarial learning with BidNet for generating synthetic yet realistic multi-level discrete data. The study demonstrates that both GANs and VAEs are effective in this regard, with CTGAN outperforming tabular VAEs in our experiments. The ability of CTGAN to generate data tailored by specifically designed conditional vectors proves particularly valuable for simulating auction environments. This highlights the importance of selecting the appropriate generative method based on the specific requirements and contexts of the task. This research significantly advances the application of GANs and VAEs in auction simulation, providing a robust foundation for further exploration into CTGANs and other generative models aimed at enhancing the simulation of complex auction scenarios across various fields.

Chapter 5 introduces the MPMG, which serves as a framework for analyzing bidding behavior and algorithmic coordination among artificial learners. We establish that the MPMG functions as a Prisoner's Dilemma under conditions of agent homogeneity. Our experiments with several well established MARL methods on the n-player heterogeneous MPMG reveal that there is no discernible positive correlation between the complexity of AI agents and the emergence of tacit coordination. Notably, only the bandit algorithm using UCB agents consistently achieved collusion and demonstrated robustness under challenging conditions. Our results also indicate that tacit coordination in this Markov game setting does not heavily depend on self-reinforcing trends, as the linearly evolving state representation tends to favor the Nash Equilibrium. Ultimately, the MPMG is shown to reflect essential market dynamics and provide insights into controlling collusive potential.

In our concluding work, presented in Chapter 6, we introduce the SEPG, an actor-critic policy gradient algorithm tailored for scenarios where agents lack explicit cooperation mechanisms

in coordination games. By incorporating mutual knowledge encoding in its critic component, the SEPG effectively guides the actor's decisions towards Pareto optimal outcomes and promotes best-response strategies against non-cooperative rational opponents. Additionally, the SEPG enables effective coordination among agents in both homogeneous and heterogeneous settings, enhancing the capabilities of actor-critic methods in fostering rational behavior and strategic coordination.

8.2 Future Research and Practical Perspectives

Modeling shocks in data generation. The availability of high-quality synthetic data is crucial for training sophisticated, data-intensive machine learning algorithms, which typically require expansive datasets for optimal performance. Synthetic data can meet this demand. However, a significant challenge persists: DGM models often fail to capture drastic changes or shocks in the training data. To address this, further research could explore altering the meta-structure of general algorithms, enabling various sub-generative models to operate in tandem and represent different underlying data distributions. Incorporating a reinforcement learning component could facilitate this by acting as a coordinating mechanism, allowing synthetic models to simulate crises and other rare events encountered in real-world scenarios.

Extending the MPMG to a Large-Scale System. Advancing the MPMG into a more expansive environment could begin by integrating real or nature-replicating synthetic data. This step would transform the MPMG into an empirical Markov game that captures dynamics typical of real-life scenarios. Enhancing the generative model's capacity to include additional dimensions in the observation space and relaxing certain assumptions, such as the common value assumption, are crucial. Moreover, developing a realistic, comprehensive simulation of such a system would necessitate a continuous action space, presenting further opportunities for the application of DGM, particularly in training agents offline using synthetic data. Ultimately, we envision a modular environment that supports various supplier selection models, thereby broadening the MPMG into a general auction Markov game.

A unifying approach to model emergent behavior. The final chapter of this dissertation (Chapter 6) would benefit from an in-depth comparative analysis with other MARL methods that foster coordination through mechanisms like SQ loss-based approaches and reputation modeling, which encodes posterior information to predict opponent behavior. Additionally, investigating the impact of algorithmic communication on collusion within the MPMG could establish benchmarks for collusion levels and inform future extensions of the

model. Furthermore, comparing the SEPG to established methods for learning the Pareto frontier in iterative games, such as CJAL, could prove insightful. It is essential to prove or refute the advantages of using Markov games over iterative games through experiments involving bandit algorithms, which could demonstrate the viability of naive methods in promoting coordination in complex settings. This point closes a conceptual loop and underscores the necessity of comprehensive systems like the proposed extensions to the MPMG.

Finally, the full potential of the SEPG critic is not currently explored. The critic is designed to adapt to environments like stochastic games, which remain untested in this thesis. Importantly, the current configuration of the critic overlooks heterogeneity treatment, which, while valid for group optimization under the assumption of mutual knowledge, does not address the disparate collusion incentives between strong and weak agents at an individual level. Thus, the critic should incorporate heterogeneity treatment to more realistically represent individual agents in a decentralized learning setting.

Stackelberg representation. Stackelberg games are sequential games in which a leader moves first and commits to a strategy that can be observed by the followers, who then make their decisions. This setting can model coordination in public procurement, as strong firms often act as leaders due to their greater incentive to collude and their ability to withstand losses from defection by others. In other words, strong firms are less risk-averse. The weaker firms follow by either aligning with or diverging from the leader's strategy.

8.3 Limitations

Modeling shocks. Our method for synthesizing auction data presented in Chapter 4.4 does not model shocks. In fact, GANs and VAEs can only capture the dynamics of data structures through time based on past data. Consequently, simulations generated using data-driven methods like ours cannot incorporate novel shock scenarios. This limitation is not unique to our approach, as no model can accurately predict the future, and anticipating future shocks that substantially modify the data generation process of a joint distribution remains an ongoing challenge in general.

Modeling heterogeneity. One limitation of the way heterogeneity is modeled in the MPMG (Chapter 4.5) is the assumption of a linear relationship between market strength and various factors such as payoffs, incentives, and competitive dynamics. While this linear approach simplifies the analysis and allows for clear differentiation between strong and weak firms, it may overlook more complex, nonlinear interactions that could exist in real-

world scenarios. For example, the advantage gained by stronger firms might not increase proportionally with their market power, and weaker firms could have diminishing returns in terms of their ability to defect or align with stronger competitors. Additionally, the linear model may fail to capture threshold effects, where small differences in market strength could lead to disproportionate shifts in behavior or outcomes, particularly in highly competitive or collusive environments. Thus, while the linear model provides a useful framework for analyzing heterogeneity, it may not fully represent the nuances of strategic interactions in heterogeneous markets.

Public policy. A limitation of the MPMG for public policy is the lack of straightforward interpretation of its outcomes. The MPMG is designed as a general theoretical model and does not account for the specific characteristics or regulatory nuances of any particular industry. This makes it difficult to derive concrete policy recommendations from the results, as the model's broad scope may not reflect the unique market dynamics or regulatory frameworks that different industries face. Additionally, the interpretation of collusive behavior and its resulting collusive spoil is somewhat arbitrary. While the model provides both an analysis of collusive outcomes and convergence rates, these results may not directly translate to real-world settings where industry-specific factors, legal frameworks, and market structures play crucial roles.

REFERENCES

[1] C. Castelfranchi, "The theory of social functions: Challenges for computational social science and multi-agent learning," *Cognitive Systems Research*, vol. 2, 2001.

[2] S. O. Kimbrough, M. Lu, and F. Murphy, *Learning and Tacit Collusion by Artificial Agents in Cournot Duopoly Games.* Springer-Verlag, 12 2005, pp. 477–492.

[3] A. Nowé, P. Vrancx, and Y. M. D. Hauwere, *Game theory and multi-agent reinforcement learning*, 2012, vol. 12, pp. 441–470.

[4] E. Uyarra and K. Flanagan, "Understanding the innovation impacts of public procurement," *European Planning Studies*, vol. 18, pp. 123–143, 2010. [Online]. Available: www.econstor.euhttp://www.mbs.ac.uk/research/workingpapers/

[5] G. Brero, A. Eden, D. Chakrabarti, M. Gerstgrasser, A. Greenwald, V. Li, and D. C. Parkes, "Stackelberg pomdp: A reinforcement learning approach for economic design," 10 2022. [Online]. Available: https://arxiv.org/abs/2210.03852v3

[6] I. Matsukawa, "Detecting collusion in retail electricity markets: Results from japan for 2005 to 2010," *Utilities Policy*, vol. 57, pp. 16–23, 4 2019.

[7] B. Baranek, V. Titl, and L. Musolff, "Detection of collusive networks in e-procurement," *SSRN Electronic Journal*, 6 2021. [Online]. Available: https://papers.ssrn.com/abstract=3864186

[8] B. W. Arthur, S. N. Durlauf, and D. A. Lane, Eds., *The Economy as an Evolving Complex System II*, a proceedi ed. Perseus Books Publishing, 1997.

[9] J. Angrist, P. Azoulay, G. Ellison, R. Hill, and S. F. Lu, "Economic research evolves: Fields and styles," *American Economic Review*, vol. 107, pp. 293–297, 2017.

[10] S. Athey, "The impact of machine learning on economics," *The Economics of Artificial Intelligence: An Agenda*, pp. 507–552, 2019. [Online]. Available: https://doi.org/10.7208/9780226613475-023

[11] A. Oussidi and A. Elhassouny, "Deep generative models: Survey," *2018 International Conference on Intelligent Systems and Computer Vision, ISCV 2018*, vol. 2018-May, pp. 1–8, 2018.

[12] P. K. Sharma, R. Fernandez, E. Zaroukian, M. Dorothy, A. Basak, and D. E. Asher, "Survey of recent multi-agent reinforcement learning algorithms utilizing centralized training," *https://doi.org/10.1117/12.2585808*, vol. 11746, pp. 665–676, 4 2021.

[13] E. J. Green and R. H. Porter, "Noncooperative collusion under imperfect price information," *Econometrica*, vol. 52, p. 87, 1984.

[14] S. Boulenger and M. Joanis, "Analyse économique des marchés publics dans l'industrie de la construction au québec," 2015.

[15] I. Sadoune, M. Joanis, and A. Lodi, "Implementing a hierarchical deep learning approach for simulating multilevel auction data," *Computational Economics 2024*, pp. 1–28, 5 2024. [Online]. Available: https://link.springer.com/article/10.1007/s10614-024-10622-4

[16] ——, "Algorithmic collusion and the minimum price markov game," 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2407.03521

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, T. Dietterish, C. Bishop, D. Heckerman, M. Jordan, and M. Kearns, Eds. The MIT Press, 2016.

[18] R. B. Myerson, *Game Theory: Analysis of Conflict*, first hava ed. Havard University Press, 1991.

[19] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction, second edition.* The MIT Press, 2018.

[20] W. Li, Y. Zhu, and D. Zhao, "Advantage constrained proximal policy optimization in multi-agent reinforcement learning," *2023 International Joint Conference on Neural Networks (IJCNN)*, vol. 2023-June, pp. 1–8, 2023.

[21] Y. Chassin and M. Joanis, "Détecter et prévenir la collusion dans les marchés publics en construction: Meilleures pratiques favorisant la concurrence," 2010. [Online]. Available: https://www.researchgate.net/publication/254398401%0ADÃĬtecter

[22] M. Shan, Y. Le, A. P. Chan, Y. Hu, M. Shan, Y. Le, A. P. C. Chan, and Y. Hu, *Collusive Practices in Public Construction Projects: A Case of China.* Springer Singapore, 2020, pp. 133–154.

[23] S. Athey, K. Bagwell, and C. W. Sanchirico, "Collusion and price rigidity," *SSRN Electronic Journal*, 2005.

[24] A. Skrzypacz and H. Hopenhayn, "Tacit collusion in repeated auctions," *Journal of Economic Theory*, vol. 114, pp. 153–169, 1 2004.

[25] S. Chassang and J. Ortner, "Collusion in auctions with constrained bids: Theory and evidence from public procurement," *Journal of Political Economy*, vol. 127, pp. 2269–2300, 2019.

[26] B. K. O. Tas, "Collusion detection in public procurement with limited information," *SSRN Electronic Journal*, 2017.

[27] R. P. Mcafee and J. Mcmillan, "Bidding rings," pp. 579–599, 1992. [Online]. Available: https://www.jstor.org/stable/2117323

[28] M. Aoyagi, "Bid rotation and collusion in repeated auctions," *Journal of Economic Theory*, vol. 112, pp. 79–105, 9 2003.

[29] J. E. H. Jr., "Detecting cartels," *Handbook of Antitrust Economics*, pp. 213–258, 2008.

[30] J. Tirole, "The economics of tacit collusion," *Development*, 2003.

[31] J.-J. Laffont and D. Martimort, "Collusion under asymmetric information," *Econometrica*, vol. 65, p. 875, 1997. [Online]. Available: https://www.researchgate.net/publication/4898655

[32] J. K. Goeree and T. Offerman, "Competitive bidding in auctions with private and common values," *Economic Journal*, vol. 113, pp. 598–613, 2003.

[33] S. George, "The theory of oligopoly," *The Journal of Political Economy*, vol. 72, pp. 44–61, 1964.

[34] M. H. Rothkopf, "Daily repetition: A neglected factor in the analysis of electricity auctions," *The Electricity Journal*, vol. 12, pp. 60–70, 1999. [Online]. Available: https://sci-hub.st/https://www.sciencedirect.com/science/article/pii/S104061909900010X

[35] E. J. Green, R. C. Marshall, and L. M. Marx, "Tacit collusion in oligopoly," *The Oxford Handbook of International Antitrust Economics*, vol. 2, pp. 1–25, 2015. [Online]. Available: www.oxfordhandbooks.com

[36] J. Hörner and J. Jamison, "Collusion with (almost) no information," *RAND Journal of Economics*, vol. 38, pp. 804–822, 2007.

[37] D. G. Pearce, "Rationalizable strategic behavior and the problem of perfection," *Econometrica*, vol. 52, pp. 1029–1050, 7 1984.

[38] B. D. Bernheim, "Rationalizable strategic behavior," *Econometrica*, vol. 52, pp. 1007–1028, 1984.

[39] T. C.-C. C. Tan and S. R. da Costa Werlang, "The bayesian foundations of solution concepts of games," *Journal of Economic Theory*, vol. 45, pp. 37–391, 8 1988.

[40] D. Fudenberg and L. A. Imhof, "Monotone imitation dynamics in large populations," *Journal of Economic Theory*, vol. 140, pp. 229–245, 2008.

[41] S. Huck, H. T. Normann, and J. Oechssler, "Through trial and error to collusion," *International Economic Review*, vol. 45, pp. 205–224, 2004.

[42] C. Fischer and H. T. Normann, "Collusion and bargaining in asymmetric cournot duopoly—an experiment," *European Economic Review*, vol. 111, pp. 360–379, 1 2019.

[43] D. J. Cooper and K.-U. Kühn, "Communication, renegotiation, and the scope for collusion †," *American Economic Journal: Microeconomics*, vol. 6, pp. 247–278, 2014. [Online]. Available: http://dx.doi.org/10.1257/mic.6.2.247

[44] M. A. Fonseca and H.-T. Normann, "Explicit vs. tacit collusion-the impact of communication in oligopoly experiments," *European Economic Review*, vol. 56, pp. 1759–1772, 2012.

[45] J. E. Harrington, R. H. Gonzalez, and P. Kujal, "The relative efficacy of price announcements and express communication for collusion: Experimental findings," *Journal of Economic Behavior and Organization*, vol. 128, pp. 251–264, 8 2016.

[46] D. Banerjee and S. Sen, "Reaching pareto-optimality in prisoner's dilemma using conditional joint action learning," *Autonomous Agents and Multi-Agent Systems*, vol. 15, pp. 91–108, 8 2007. [Online]. Available: https://link.springer.com/article/10.1007/s10458-007-0020-8

[47] H. Zhang and Y. Fan, "An adaptive policy gradient in learning nash equilibria," *Neurocomputing*, vol. 72, pp. 533–538, 2008.

[48] P. Milgrom and E. Kwerel, *Putting Auction Theory to Work*. Cambridge University Press, 2003.

[49] B. Edelman, M. Ostrovsky, and M. Schwarz, "Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords," *American Economic Review*, vol. 97, pp. 242–259, 3 2007.

[50] A. Hortaçsu and S. L. Puller, "Understanding strategic bidding in multi-unit auctions: A case study of the texas electricity spot market," *The RAND Journal of Economics*, vol. 39, pp. 86–114, 2008.

[51] E. Uyarra, J. M. Zabala-Iturriagagoitia, K. Flanagan, and E. Magro, "Public procurement, innovation and industrial policy: Rationales, roles, capabilities and implementation," *Research Policy*, vol. 49, 2 2020.

[52] Tesfatsion, *Handbook of Computational Economics: Volume 2, Agent-based Computational Economomics*, handbooks ed., L. Tesfatsion and K. L. Judd, Eds. Elsevier, 2006.

[53] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, pp. 7280–7287, 5 2002. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.082080899

[54] H. Dawid and A. Pyka, "Introduction: Special issue on evolutionary dynamics and agent-based modeling in economics," *Computational Economics*, vol. 52, pp. 707–710, 10 2018. [Online]. Available: https://link.springer.com/article/10.1007/s10614-018-9831-8

[55] S. Athey, "The Impact of Machine Learning on Economics," *The Economics of Artificial Intelligence: An Agenda*, pp. 507–552, 2019. [Online]. Available: https://doi.org/10.7208/9780226613475-023

[56] J. A. Calpin, M. R. Salisbury, J. A. Vitkevich, and D. R. Woodward, "Extending the high level architecture paradigm to economic simulation," *Computational Economics*, vol. 17, pp. 141–154, 2001. [Online]. Available: https://link.springer.com/article/10.1023/A:1011619907538

[57] D. Xie, N. Zhang, and D. A. Edwards, "Simulation solution to a two-dimensional mortgage refinancing problem," *Computational Economics*, vol. 52, pp. 479–492, 8 2018. [Online]. Available: https://link.springer.com/article/10.1007/s10614-017-9689-1

[58] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 3, pp. 2672–2680, 2014. [Online]. Available: http://www.github.com/goodfeli/adversarialhttp://arxiv.org/abs/1406.2661

[59] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR, 2014.

[60] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," 2019. [Online]. Available: http://arxiv.org/abs/1907.00503

[61] W. B. Arthur, "Complexity and the economy," pp. 107–109, 4 1999.

[62] F. Decarolis, "Comparing public procurement auctions," 2017.

[63] G. Marti, "CORRGAN: Sampling Realistic Financial Correlation Matrices Using Generative Adversarial Networks," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, oct 2020, pp. 8459–8463. [Online]. Available: http://arxiv.org/abs/1910.09504

[64] R. R. Y. and D. K. P., *SIMULATION AND THE MONTE CARLO METHOD*, 2008.

[65] A. C. Dvison, D. V. Hinkley, and E. Schechtman, "Efficient bootstrap simulation," *Biometrika*, vol. 73, pp. 555–566, 12 1986. [Online]. Available: https://academic.oup.com/biomet/article/73/3/555/250081

[66] S. Elsawah, T. Filatova, A. J. Jakeman, A. J. Kettner, M. L. Zellner, I. N. Athanasiadis, S. H. Hamilton, R. L. Axtell, D. G. Brown, J. M. Gilligan, M. A. Janssen, D. T. Robinson, J. Rozenberg, I. I. T. Ullah, and S. J. Lade, "Eight grand challenges in socio-environmental systems modeling," *Socio-Environmental Systems Modelling*, vol. 2, p. 16226, 1 2020.

[67] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2016, pp. 2180–2188. [Online]. Available: https://arxiv.org/abs/1606.03657.

[68] R. L. Axtell and J. D. Farmer, "Agent-based modeling in economics and finance: Past, present, and future," *Journal of Economic Literature*, p. to appear, 2021.

[69] S. Athey and G. W. Imbens, "Machine learning methods that economists should know about," *Annual Review of Economics*, vol. 11, pp. 685–725, 2019. [Online]. Available: https://www.gsb.stanford.edu/faculty-research/working-papers/machine-learning-methods-economists-should-know-about

[70] H. Ba, "Improving detection of credit card fraudulent transactions using generative adversarial networks," 7 2019. [Online]. Available: http://arxiv.org/abs/1907.03355

[71] P. Jackson and M. Lussetti, "Extending a generative adversarial network to produce medical records with demographic characteristics and health system use," *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2019*, pp. 515–518, 2019.

[72] J. Yoon, J. Jordon, and M. V. D. Schaar, "Ganite: Estimation of individualized treatment effects using generative adversarial nets," *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, pp. 1–15, 2018. [Online]. Available: https://openreview.net/pdf?id=ByKWUeWA-

[73] C. Louizos, U. Shalit, J. Mooij, D. Sontag, R. Zemel, and M. Welling, "Causal effect inference with deep latent-variable models," *Advances in Neural Information Processing Systems*, vol. 2017-Decem, pp. 6447–6457, 5 2017. [Online]. Available: http://arxiv.org/abs/1705.08821

[74] R. Cai, J. Qiao, K. Zhang, Z. Zhang, and Z. Hao, "Causal discovery with cascade nonlinear additive noise models," *IJCAI International Joint Conference on Artificial Intelligence*, vol. 2019-Augus, pp. 1609–1615, 2019.

[75] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: A survey toward private and secure applications," *ACM Computing Surveys*, vol. 54, 2021.

[76] M. Wong and B. Farooq, "A bi-partite generative model framework for analyzing and simulating large scale multiple discrete-continuous travel behaviour data," *Transportation Research Part C: Emerging Technologies*, vol. 110, pp. 247–268, 1 2020.

[77] S. Takahashi, Y. Chen, and K. Tanaka-Ishii, "Modeling financial time-series with generative adversarial networks," *Physica A: Statistical Mechanics and its Applications*, vol. 527, 8 2019.

[78] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, pp. 214–223, 1 2017. [Online]. Available: https://proceedings.mlr.press/v70/arjovsky17a.htmlhttp://arxiv.org/abs/1701.07875

[79] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of wasserstein gans," *Advances in Neural Information Processing Systems*, vol. 2017-Decem, pp. 5768–5778, 2017.

[80] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-Octob, pp. 2813–2821, 2017.

[81] R. D. Hjelm, A. P. Jacob, T. Che, A. Trischler, K. Cho, and Y. Bengio, "Boundary-seeking generative adversarial networks," in *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 2018.

[82] E. Choi, S. Biswal, B. Malin, J. Duke, W. F. Stewart, and J. Sun, "Generating multi-label discrete patient records using generative adversarial networks," vol. 68, pp. 1–20, 2017. [Online]. Available: http://arxiv.org/abs/1703.06490

[83] Z. Lin, G. Fanti, A. Khetan, and S. Oh, "Pacgan: The power of two samples in generative adversarial networks," *Advances in Neural Information Processing Systems*, vol. 2018-Decem, pp. 1498–1507, 2018.

[84] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 12 2016, pp. 1–9. [Online]. Available: http://arxiv.org/abs/1701.00160https://proceedings.neurips.cc/paper/2016/file/8a3363abe792db2d8761d6403605aeb7-Paper.pdf

[85] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 12 2014. [Online]. Available: https://arxiv.org/abs/1412.6980v9

[86] P. M. Williams, "Using neural networks to model conditional multivariate densities," *Neural Computation*, vol. 8, pp. 843–854, 1996.

[87] R. Neuneier, F. Hergert, W. Finnoff, and D. Ormoneit, "Estimation of conditional densities: A comparison of neural network approaches," *Icann '94*, pp. 689–692, 1994.

[88] H. Schioler and P. Kulczycki, "Neural network for estimating conditional distributions," *IEEE Transactions on Neural Networks*, vol. 8, pp. 1015–1025, 1997.

[89] L. Prechelt, "Early stopping - but when?" *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7700 LECTU, pp. 53–67, 2012.

[90] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2 2015.

[91] C. Szepesvári, "Algorithms for reinforcement learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 9, pp. 1–89, 2010. [Online]. Available: https://doi.org/10.2200/S00268ED1V01Y201005AIM009https://consensus.app/papers/algorithms-reinforcement-learning-szepesvari/c62eb61c9a915e22883e1c9b3b9122e7/https://www.semanticscholar.org/paper/e60f3c1cb857daa3233f2c5b17b6f111ff86698c

[92] C. Yu, J. Liu, S. Nemati, and G. Yin, "Reinforcement learning in healthcare: A survey," *ACM Computing Surveys (CSUR)*, vol. 55, pp. 1–36, 11 2021.

[93] D. Mguni, J. Jennings, and E. M. D. Cote, "Decentralised learning in systems with many, many strategic agents," *ArXiv*, vol. abs/1803.05028, pp. 4686–4693, 2018.

[94] J. Riley, R. Calinescu, C. Paterson, D. Kudenko, and A. Banks, "Reinforcement learning with quantitative verification for assured multi-agent policies," *ICAART 2021 - Proceedings of the 13th International Conference on Agents and Artificial Intelligence*, vol. 2, pp. 237–245, 2021.

[95] Z. Jin, W. Y. Liu, and J. Jin, "Finding shortcuts from episode in multi-agent reinforcement learning," *2009 International Conference on Machine Learning and Cybernetics*, vol. 4, pp. 2306–2311, 2009.

[96] A. Ezrachi and M. Stucke, "Sustainable and unchallenged algorithmic tacit collusion," *Northwestern Journal of Technology and Intellectual Property*, vol. 17, 2020.

[97] T. Klein, "Assessing autonomous algorithmic collusion: Q-learning under sequential pricing," *SSRN Electronic Journal*, 7 2018.

[98] A. Ittoo and N. Petit, "Algorithmic pricing agents and tacit collusion: A technological perspective," *SSRN Electronic Journal*, pp. 1–14, 10 2017.

[99] J. Viehmann, S. Lorenczik, and R. Malischek, "Multi-unit multiple bid auctions in balancing markets: An agent-based q-learning approach," *Energy Economics*, vol. 93, 2021.

[100] Y. Ye, D. Qiu, M. Sun, D. Papadaskalopoulos, and G. Strbac, "Deep reinforcement learning for strategic bidding in electricity markets," *IEEE Transactions on Smart Grid*, pp. 1–1, 8 2019.

[101] N. Rashedi, M. A. Tajeddini, and H. Kebriaei, "Markov game approach for multi-agent competitive bidding strategies in electricity market," *IET Generation, Transmission and Distribution*, vol. 10, pp. 3756–3763, 2016. [Online]. Available: https://www.researchgate.net/publication/305627052

[102] M. Lucic, K. Kurach, M. Michalski, O. Bousquet, and S. Gelly, "Are gans created equal? a large-scale study," *Advances in Neural Information Processing Systems*, vol. 2018-Decem, pp. 700–709, 2018.

[103] B. Chu and S. Qureshi, "Comparing out-of-sample performance of machine learning methods to forecast u.s. gdp growth," *Computational Economics*, pp. 1–43, 9 2022. [Online]. Available: https://link.springer.com/article/10.1007/s10614-022-10312-z

[104] J. Lussange, I. Lazarevich, S. Bourgeois-Gironde, S. Palminteri, and B. Gutkin, "Modelling stock markets by multi-agent reinforcement learning," *Computational Economics*, vol. 57, pp. 113–147, 1 2021. [Online]. Available: https://link.springer.com/article/10.1007/s10614-020-10038-w

[105] X. Zhou and H. Li, "Buying on margin and short selling in an artificial double auction market," *Computational Economics*, vol. 54, pp. 1473–1489, 12 2019. [Online]. Available: https://link.springer.com/article/10.1007/s10614-017-9722-4

[106] M. Shafie-Khah and J. P. Catalão, "A stochastic multi-layer agent-based model to study electricity market participants behavior," *IEEE Transactions on Power Systems*, vol. 30, pp. 867–881, 3 2015.

[107] A. C. Tellidou and A. G. Bakirtzis, "Agent-based analysis of capacity withholding and tacit collusion in electricity markets," *IEEE Transactions on Power Systems*, vol. 22, pp. 1735–1742, 11 2007.

[108] X. Zhao, "The effect of political connections: Model analysis and quantitative simulation," *Emerging Markets Finance and Trade*, vol. 0, pp. 1–13, 2019. [Online]. Available: https://doi.org/10.1080/1540496X.2019.1612362

[109] L. Waltman and U. Kaymak, "Q-learning agents in a cournot oligopoly model," *Journal of Economic Dynamics and Control*, vol. 32, pp. 3275–3293, 10 2008.

[110] J. A. Gerlick and S. M. Liozu, "Ethical and legal considerations of artificial intelligence and algorithmic decision-making in personalized pricing," *Journal of Revenue and Pricing Management*, vol. 19, pp. 85–98, 4 2020. [Online]. Available: https://link.springer.com/article/10.1057/s41272-019-00225-2

[111] L. L. Gormsen, "Algorithmic antitrust and consumer choice," *Economic Analysis of Law in European Legal Scholarship*, vol. 12, pp. 65–86, 2022.

[112] Q. Li, N. Philipsen, C. Cauffman, and C. C. Nl, "Ai-enabled price discrimination as an abuse of dominance: a law and economics analysis," *China-EU Law Journal 2023 9:1*, vol. 9, pp. 51–72, 4 2023. [Online]. Available: https://link.springer.com/article/10.1007/s12689-023-00099-z

[113] E. Calvano, G. Calzolari, V. Denicolò, and S. Pastorello, "Artificial intelligence, algorithmic pricing, and collusion," *American Economic Review*, vol. 110, pp. 3267–97, 10 2020.

[114] S. Assad, R. Clark, D. Ershov, and L. Xu, "Algorithmic pricing and competition: Empirical evidence from the german retail gasoline market," *SSRN Electronic Journal*, 11 2020. [Online]. Available: https://papers.ssrn.com/abstract=3682021

[115] B. Lebrun, "Uniqueness of the equilibrium in first-price auctions," *Games and Economic Behavior*, vol. 55, pp. 131–151, 4 2006.

[116] D. Bergemann, B. Brooks, and S. Morris, "First-price auctions with general information structures: Implications for bidding and revenue," *Econometrica*, vol. 85, pp. 107–143, 1 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.3982/ECTA13958https://onlinelibrary.wiley.com/doi/abs/10.3982/ECTA13958https://onlinelibrary.wiley.com/doi/10.3982/ECTA13958

[117] E. Calvano, G. Calzolari, V. Denicolò, J. E. Harrington, and S. Pastorello, "Protecting consumers from collusive prices due to ai," *Science*, vol. 370, pp. 1040–1042, 11 2020. [Online]. Available: https://www.science.org/doi/10.1126/science.abe3796

[118] R. Axelrod, "The emergence of cooperation among egoists," *American Political Science Review*, vol. 75, pp. 306–318, 6 1981. [Online]. Available: https://www.cambridge.org/core/journals/american-political-science-review/article/abs/emergence-of-cooperation-among-egoists/EEAB3C6460F5BC63A4DE813E1B010B21

[119] L. Buşoniu, R. Babuška, and B. D. Schutter, "A comprehensive survey of multiagent reinforcement learning," pp. 156–172, 2008.

[120] V. Kuleshov and D. Precup, "Algorithms for multi-armed bandit problems," *ArXiv*, vol. abs/1402.6028, 2014.

[121] H. V. Hasselt, "Double q-learning," *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010, NIPS 2010*, pp. 1–9, 2010.

[122] I. Grondman, L. Busoniu, G. A. Lopes, and R. Babuška, "A survey of actor-critic reinforcement learning: Standard and natural policy gradients," pp. 1291–1307, 2012.

[123] C. Graf, V. Zobernig, J. Schmidt, and C. Klöckl, "Computational performance of deep reinforcement learning to find nash equilibria," *Computational Economics*, 2023.

[124] M. S. M. Şeref Ahunbay, M. Bichler, T. Dobos, and J. Knörr, "Solving large-scale electricity market pricing problems in polynomial time," 12 2023. [Online]. Available: https://arxiv.org/abs/2312.07071v1

[125] G. Brero, B. Lubin, and S. Seuken, "Probably approximately efficient combinatorial auctions via machine learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, pp. 397–405, 2 2017. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/10624

[126] M. Beyeler, G. Brero, B. Lubin, and S. Seuken, "imlca: Machine learning-powered iterative combinatorial auctions with interval bidding," pp. 136–136, 7 2021. [Online]. Available: https://dl.acm.org/doi/10.1145/3465456.3467535

[127] P. Hummel and R. P. McAfee, "Machine learning in an auction environment," *WWW 2014 - Proceedings of the 23rd International Conference on World Wide Web*, vol. 17, pp. 7–17, 2014.

[128] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multi-agent systems: A review of challenges, solutions, and applications," *IEEE Transactions on Cybernetics*, pp. 1–14, 2020.

[129] G. Gao, H. Huang, M. Xiao, J. Wu, Y. E. Sun, and S. Zhang, "Auction-based combinatorial multi-armed bandit mechanisms with strategic arms," *Proceedings - IEEE INFOCOM*, vol. 2021-May, 5 2021.

[130] S. Basu and A. Sankararaman, "Double auctions with two-sided bandit feedback," 8 2022. [Online]. Available: https://arxiv.org/abs/2208.06536v1

[131] D. Goktas, S. Zhao, and A. Greenwald, "Zero-sum stochastic stackelberg games," *Advances in Neural Information Processing Systems*, vol. 35, pp. 11 658–11 672, 12 2022.

[132] D. Fudenberg and J. Tirole, *Game Theory*. MIT Press, 1991.

[133] G. Brero, A. Eden, M. Gerstgrasser, D. Parkes, and D. Rheingans-Yoo, "Reinforcement learning of sequential price mechanisms," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 5219–5227, 5 2021. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16659

[134] G. Brero, D. Chakrabarti, A. Eden, M. Gerstgrasser, V. Li, and D. Parkes, "Learning stackelberg equilibria in sequential price mechanisms." Proc. ICML Workshop for Reinforcement Learning Theory, 2021.

[135] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *AAMAS 2017 Workshops, Best Papers, São Paulo, Brazil, May 8-12, 2017, Revised*, 2017, pp. 66–83.

[136] J. Foerster, R. Y. Chen, M. Al-Shedivat, S. Whiteson, P. Abbeel, and I. Mordatch, "Learning with opponent-learning awareness," *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 1, pp. 122–130, 2018.

[137] M. Harper, V. Knight, M. Jones, G. Koutsovoulos, N. E. Glynatsi, and O. Campbell, "Reinforcement learning produces dominant strategies for the iterated prisoner's dilemma," *PLOS ONE*, vol. 12, p. e0188046, 12 2017. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0188046

[138] V. Vassiliades, A. Cleanthous, and C. Christodoulou, "Multiagent reinforcement learning: Spiking and nonspiking agents in the iterated prisoner's dilemma," *IEEE Transactions on Neural Networks*, vol. 22, pp. 639–653, 4 2011.

[139] K. Li and D. Hao, "Cooperation enforcement and collusion resistance in repeated public goods games," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 2085–2092, 7 2019.

[140] R. Chaudhuri, K. Mukherjee, R. Narayanam, and R. D. Vallam, "Collaborative reinforcement learning framework to model evolution of cooperation in sequential

social dilemmas," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12712 LNAI, pp. 15–26, 2021. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-75762-5_2

[141] M. R. Baye and D. Kovenock, "Bertrand competition," *The New Palgrave Dictionary of Economics*, pp. 1–7, 2008.

[142] Y. Zhou, S. Liu, Y. Qing, K. Chen, T. Zheng, Y. Huang, J. Song, and M. Song, "Is centralized training with decentralized execution framework centralized enough for marl?" 5 2023. [Online]. Available: https://arxiv.org/abs/2305.17352v1

[143] S. Sukhbaatar, A. Szlam, and R. Fergus, "Learning multiagent communication with backpropagation," *Advances in Neural Information Processing Systems*, pp. 2252–2260, 2016.

[144] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *Studies in Systems, Decision and Control*, vol. 325, pp. 321–384, 2021.

[145] P. Badjatiya, M. Sarkar, A. Sinha, S. Singh, N. Puri, and B. Krishnamurthy, "Inducing cooperation in multi-agent games through status-quo loss," 1 2020. [Online]. Available: http://arxiv.org/abs/2001.05458

[146] J. Vermorel and M. Mohri, "Multi-armed bandit algorithms and empirical evaluation," pp. 437–448, 2005.

[147] N. Gupta, O.-C. Granmo, and A. Agrawala, "Thompson sampling for dynamic multi-armed bandits," *2011 10th International Conference on Machine Learning and Applications and Workshops*, vol. 1, pp. 484–489, 2011.

[148] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, pp. 235–256, 5 2002. [Online]. Available: https://link.springer.com/article/10.1023/A:1013689704352

[149] H. V. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, 2016, pp. 2094–2100. [Online]. Available: www.aaai.org

[150] B. Lin, D. Bouneffouf, and G. Cecchi, "Online learning in iterated prisoner's dilemma to mimic human behavior," *Lecture Notes in Computer Science*

(including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 13631 LNCS, pp. 134–147, 2022. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-031-20868-3_10

[151] J. Miklós-Thal and C. Tucker, "Collusion by algorithm: Does better demand prediction facilitate coordination between sellers?" *Management Science*, vol. 65, pp. 1552–1561, 4 2019.

[152] A. Ezrachi and M. E. Stucke, "How pricing bots could form cartels and make things more expensive," *Havard Business Review*, 2016. [Online]. Available: https://hbr.org/2016/10/how-pricing-bots-could-form-cartels-and-make-things-more-expensive?autocomplete=true

[153] A. Ezrachi and S. M. E., "Virtual competition: The promise and perils of the algorithm-driven economy," *Havard Press University*, 2016.

[154] M. P. Wellman, K. Tuyls, and A. Greenwald, "Empirical game-theoretic analysis: A survey," 3 2024. [Online]. Available: https://arxiv.org/abs/2403.04018v1

[155] U. Schwalbe, "Algorithms, machine learning, and collusion," *Journal of Competition Law & Economics*, vol. 14, pp. 568–607, 12 2018.

[156] Y. Qiu, Y. Jin, L. Yu, J. Wang, and X. Zhang, "Promoting cooperation in multi-agent reinforcement learning via mutual help," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, vol. 2023-June, 2023.

[157] I. ElSayed-Aly and L. Feng, "Logic-based reward shaping for multi-agent reinforcement learning," 6 2022. [Online]. Available: https://arxiv.org/abs/2206.08881v1

[158] H. Liu, Z. Zhang, and D. Wang, "Wrfmr: A multi-agent reinforcement learning method for cooperative tasks," *IEEE Access*, vol. 8, pp. 216 320–216 331, 2020.

[159] J. G. Kuba, R. Chen, M. Wen, Y. Wen, F. Sun, J. Wang, and Y. Yang, "Trust region policy optimisation in multi-agent reinforcement learning," pp. 1–25, 2021. [Online]. Available: http://arxiv.org/abs/2109.11251

[160] W. Samuelson and R. Zeckhauser, "Status quo bias in decision making," *Journal of Risk and Uncertainty*, vol. 1, pp. 7–59, 1988.

[161] J. Peters and S. Schaal, "Natural actor-critic," *Neurocomputing*, vol. 71, pp. 1180–1190, 3 2008.

[162] V. Mnih, A. P. Badia, L. Mirza, A. Graves, T. Harley, T. P. Lillicrap, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," *33rd International Conference on Machine Learning, ICML 2016*, vol. 4, pp. 2850–2869, 2 2016. [Online]. Available: https://arxiv.org/abs/1602.01783v2

[163] K. Jamieson, M. Malloy, R. Nowak, and S. Bubeck, "lil' ucb : An optimal exploration algorithm for multi-armed bandits," *Journal of Machine Learning Research*, vol. 35, pp. 423–439, 12 2013. [Online]. Available: https://arxiv.org/abs/1312.7308v1

[164] P. Hernandez-Leal, M. Kaisers, T. Baarslag, and E. M. de Cote, "A survey of learning in multiagent environments: Dealing with non-stationarity," 7 2017. [Online]. Available: http://arxiv.org/abs/1707.09183

APPENDIX A IMPLEMENTING A HIERARCHICAL DEEP LEARNING APPROACH FOR SIMULATING MULTILEVEL AUCTION DATA

A.1 SEAO Dataset

Table A.1 outlines the variables used in our study. It is important to note that geographical variables like *countries* or *states* were excluded because, although present in the raw data, only contracts from the province of Quebec are actually recorded in the subset we accessed.

Table A.1 Description of the SEAO dataset.

Variables	**Description**
public contractor	The public entity that creates and publishes the contract for an auction.
municipality	Indicates whether the contract is issued at the municipal level.
sector	The industry sector from which the contract is issued.
subsector	The specific activity sector of the contract.
location	Various locations in Quebec where the contract is executed. This variable also includes combinations of locations for contracts spanning multiple areas.
unspsc	A global classification system for goods and services.
number of bidders	The number of firms participating in an auction.
post-auction expenses	Indicates any additional expenses paid by the public contractor after the contract is awarded.
firms	Firms associated with each bid in an auction.
bids	Standardized log bids, originally in Canadian dollars (CAD) in the raw data.

The cleaning of the raw data involved selecting relevant variables. Many columns in the raw data were excluded as they did not provide informative signals due to their nature (e.g., web links) or redundancy with other variables. The quality of the signals was another selection criterion. Columns like temporal variables or textual entries were omitted due to low quality, evidenced by inconsistencies, excessive missing values, and non-uniform entry formats. Despite centralized dataset management by SEAO, data inconsistencies and missing values are common due to manual updates by various public entities' administrators. These are significant limitations of this dataset.

The raw data, is available here seao data, and the official PDF description file here. Data for each year, and in some cases each month, must be fetched separately. The raw data,

provided in XML format, needs conversion into a workable tabular array. We utilized the Python "xml" library to convert and save the data in pickle (.pkl) format. The code for processing the original XML files is available in the associated GitHub repository for this manuscript. Note that we also provide the cleaned and preprocessed pickle file.

A.2 Methodology Overview

The code for the entire project, as well as instructions for replication, can be found in the open-source GitHub repository. The datasets, including both original and synthetic samples, can be downloaded here

Initially, the SEAO dataset underwent a thorough cleaning process. This involved handling missing values, removing irrelevant columns, and reformatting specific columns to ensure their consistency and reliability.

Following the data cleaning, preprocessing was conducted to transform the dataset and make it suitable for machine learning applications. Discrete variables were transformed using one-hot encoding techniques, while continuous bid values were standardized.

To generate synthetic data, two primary generative models were utilized: CTGAN and TVAE. The CTGAN model was trained using an array of hyperparameters, including distinct embedding dimensions, generator and discriminator dimensions, learning rates, and specific decay rates. Likewise, the TVAE, a variant that incorporates a variational autoencoder structure, was trained with specific parameters, including hidden and latent dimensions.

Once trained, these models were then employed to sample synthetic datasets, replicating the patterns and distributions seen in the original SEAO dataset.

Subsequent to the synthetic data generation, we introduced BidNet neural network model. This model was designed to predict bid values using both discrete and continuous inputs. For training efficiency, the model utilized cross-validation and early stopping methodologies. Several hyperparameters, including learning rate, batch size, and number of epochs, were tuned manually to enhance the model's performance. Alternatively, rigorous automated tuning procedures (e.g., Bayesian optimization) can be used, provided enough computational resources and time are available.

To assess the quality of the synthetic data produced by CTGAN and TVAE, a series of classifiers, including Decision Trees, k-Nearest Neighbors, and Neural Networks, were trained on both the real and synthetic datasets. Performance metrics from these classifiers provided insights into the fidelity and utility of the synthetic data.

Finally, BidNet model's performance was critically evaluated using various metrics. These metrics, namely the Root Mean Square Error (RMSE), Jensen-Shannon distance (JS), and Wasserstein distance (WS), compared the synthetic and real bids, giving a comprehensive understanding of the model's accuracy and effectiveness.

Throughout the entire process, special attention was given to reproducibility. Foundational functionalities ensured consistent random states, allowing for deterministic behavior across runs. Additionally, capabilities were established to save intermediate results, trained models, and to manage computation across various devices, whether CPU or GPU.

A.3 Algorithms

Algorithm 3 Training GANs-based auction features generator

$D_{train} \leftarrow$ Initialize training set
while $C(fake) > threshold$ **do** ▷ The critic can be optimized until $C(fake)$ is near 0.
    Randomly select a discrete variable c with equal probability
    Compute the probability mass function (PMF) of c
    Randomly select a state i^* inherent to c according its PMF
    Create the conditional vector $cond$ so that $\sum_i cond(i) = 1$ and $cond(i^*) = 1$
    for $batch \in \{1, \ldots, N_{batches}\}$ **do** ▷ Gradient descent with mini-batch
        $real \leftarrow d(c_{i^*} = 1) \sim D_{train}$ ▷ Sample batch of real examples respecting the constraint
        $z \sim \mathcal{N}(0, 1)$ ▷ Sample noise
        $fake \leftarrow \tilde{d} \sim G(z)$ ▷ Sample fake examples
        $real \leftarrow [real] \times 10$ ▷ Stack input 10 times for Pac configuration
        $fake \leftarrow [fake] \times 10$
        $L^j \leftarrow \Big(C(fake_j) - C(real_j)\Big) + CE(\tilde{c}, cond)$
        $L^{batch} \leftarrow L^{batch} + \lambda(||\nabla L^{batch}||_2 - 1)^2$ ▷ Apply gradient penalty
        $w_{crit} \leftarrow w_{crit} + Adam(\nabla_{w_{crit}} \frac{1}{m} \sum_i^m L^{batch}(i))$ ▷ Updating C with Adam
        if $batch \bmod k = 0$ **then** ▷ Synchronicity, depends on k
            $w_{gen} \leftarrow w_{gen} + Adam(\nabla_{w_{gen}} \frac{1}{m} \sum_i^m -C(G(z)))$ ▷ Updating G with Adam
        end if
    end for
end while

Algorithm 4 Training tabular VAE for auction features

$D_{train} \leftarrow$ Initialize training set
for $epoch \in N_{steps}$ **do**
  for $batch \in \{1, \ldots, N_{batches}\}$ **do** ▷ Gradient descent with mini-batch
    $real \sim D_{train}$ ▷ Sample batch of real examples
    $(\mu, \sigma^2) \sim Enc(real)$
    $z \sim \mathcal{N}(\mu, \sigma^2)$ ▷ Sample latent input
    $fake \sim Dec(z)$ ▷ Sample fake examples
    $L^j \leftarrow CE(\tilde{c}_j - \arg\max(c)) + (2\sigma^2)^{-1}(x_j - tanh(\tilde{x}_j))^2 + KL(\mathcal{N}(\mu_j, \sigma_j^2), \mathcal{N}(0, 1))$
    $w \leftarrow w + Adam(\nabla_w \frac{1}{m} \sum_i^m L^{batch}(i))$ ▷ Updating parameters with Adam
  end for
end for

Algorithm 5 K-folds cross-validation BidNet training procedure

$D \leftarrow \{D_1, \ldots, D_K\}$ ▷ Initialize K-folds
$loss^* \leftarrow \infty$ ▷ Initialize best model
for $fold \in D$ **do**
  $reset(w_{BidNet})$ ▷ Reset parameters before entering each new fold
  $D_{val} \leftarrow D(fold)$, $D_{train} \leftarrow D(-fold)$
  while has not converged **do**
    for $batch \in \{1, \ldots, N_{batches}\}$ **do** ▷ Gradient descent with mini-batch
      $d \sim D_{train}$ ▷ Sample batch of real examples
      $\hat{\theta} \leftarrow BidNet(d)$
      $L^{train} \leftarrow m^{-1} \sum_i NLL(\hat{\theta})_i$ ▷ compute NLL on training batch
      $w \leftarrow w + Adam(\nabla L^{train})$ ▷ Update BidNet
    end for
    $converged \leftarrow ES(L^{val})$ ▷ Early stopping
    $L^{val} \leftarrow n^{-1} \sum_j NLL(BidNet(D_{val}))_j$ ▷ compute NLL on validation fold
    if $L^{val} < loss^*$ **then**
      $loss^* \leftarrow L^{val}$
      save model
    end if
  end while
end for

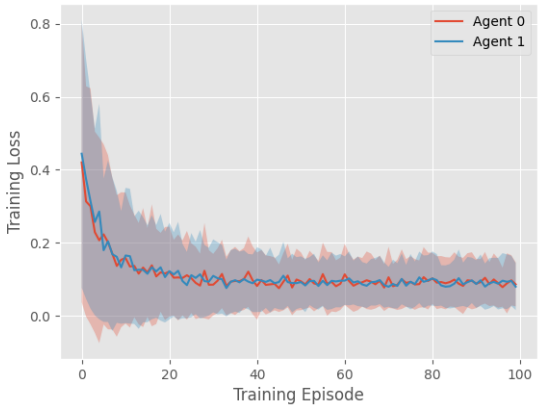
Algorithm 6 Synthetic bid validation / Double validation

1: $\beta \leftarrow \beta^*$ ▷ load best set of parameters for BidNet
2: $\alpha \leftarrow \alpha^*$ ▷ load optimized set of parameters for synthesizer
3: $\tilde{\mathbf{c}} \sim A_{\alpha^*}(\mathbf{z})$ ▷ sample synthetic examples from the trained synthesizer
4: $\mathbf{c} \sim D_{test}$ ▷ sample a test-set of real instances
5: $\hat{b} \sim B_{\beta^*}(\mathbf{c})$ ▷ sample predicted bids from the test-set of real instances using BidNet
6: $\tilde{b} \sim B_{\beta^*}(\tilde{\mathbf{c}})$ ▷ sample fake bids with the synthetic data emanating from the synthesizer
7: $Dist(p(b)||p(\tilde{b}))$ ▷ compute the statistical distance between the fake and real distributions of bids
8: $Dist(p(b)||p(\hat{b}))$ ▷ compute the statistical distance between the predicted and real distributions of bids
9: $Dist(p(\hat{b})||p(\tilde{b}))$ ▷ compute the statistical distance between the predicted and fake distributions of bids

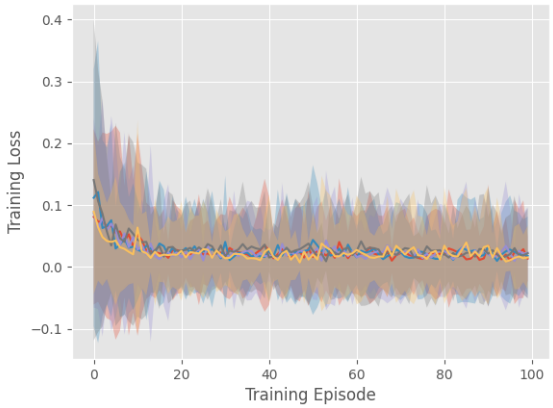
APPENDIX B ALGORITHMIC COLLUSION AND THE MINIMUM PRICE MARKOV GAME

B.1 Additional Figures

Figure B.1 Loss values of the dueling DQN during training of the naive D3QN on the 2-player homogenous MPMG (a) and 5-player heterogeneous MPMG (b).

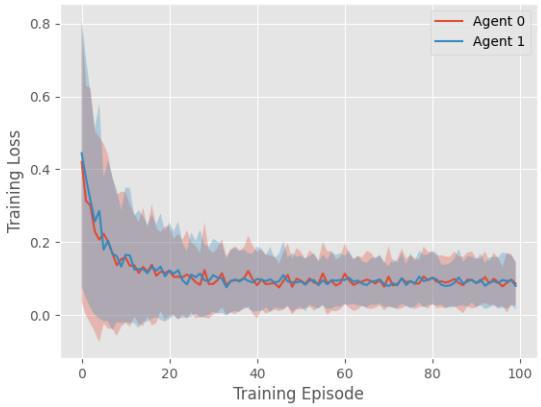


(a) DQN losses on the 2-player homogenous MPMG

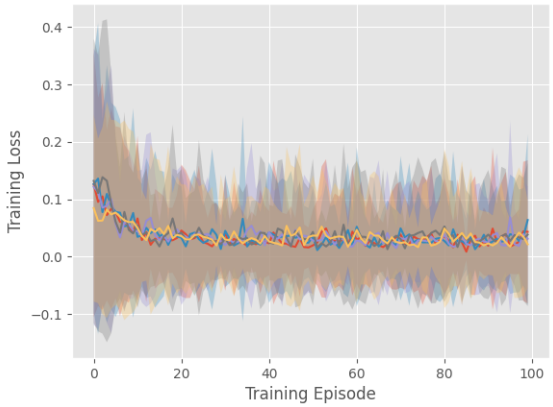


(b) DQN losses on the 5-player heterogeneous MPMG

Figure B.2 Loss values of the dueling DQN during training the D3QN with opponent modeling on the 2-player homogenous MPMG (a) and 5-player heterogeneous MPMG (b).

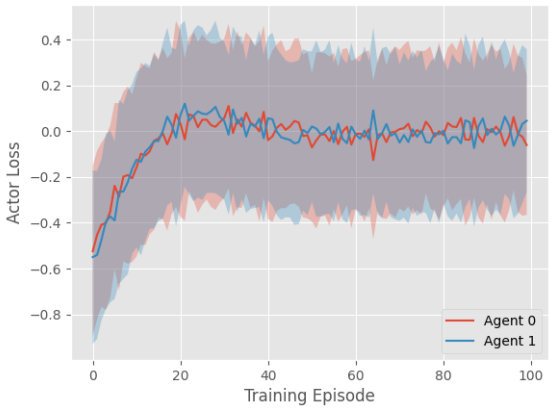


(a) DQN losses on the 2-player homogenous MPMG

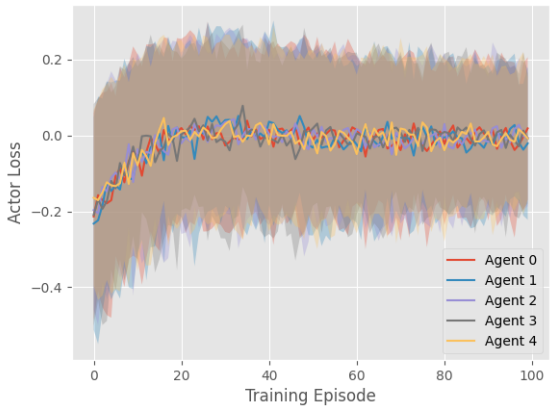


(b) DQN losses on the 5-player heterogeneous MPMG

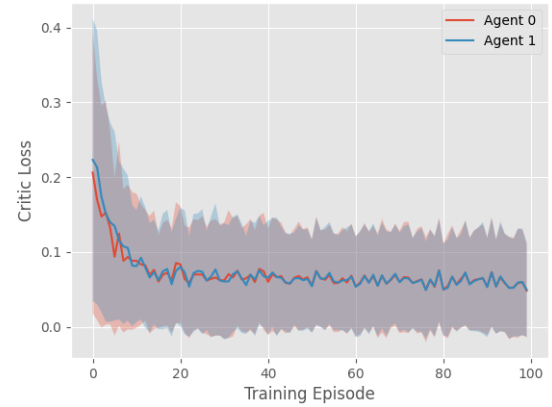
Figure B.3 Actor network loss values during training of MAPPO on the 2-player homogenous MPMG (a) and 5-player heterogeneous MPMG (b). Simalarly, graphs (c) and (d) shows the associated critic network loss values for the 2-player homogeneous and 5-player heterogeneous case.



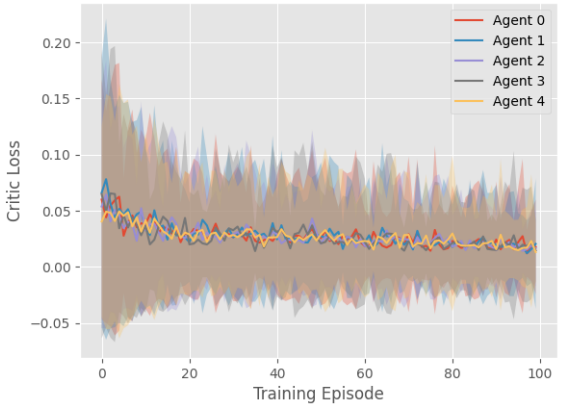
(a) Actor network losses on the 2-player homogenous MPMG



(b) Actor network losses on the 5-player heterogeneous MPMG



(c) Critic network losses on the 2-player homogenous MPMG



(d) Critic network losses on the 5-player heterogeneous MPMG

B.2 Implementation Details

In this section, we expose in detail our implementation of the experiments presented in this manuscript. The code associated to this study is written in Python, and can be found at `https://github.com/IgorSadoune/minimum-price-game`.

E-Greedy Method. Our `EpsilonGreedyMAB` class utilizes an ϵ-greedy strategy with the following hyperparameters:

- `n_arms`: Number of possible actions (arms) for the bandit.
- `epsilon`: Exploration rate, set to 0.3 in the `initialize_agents` function.

The agent selects a random action with probability ϵ and the best known action with probability $1 - \epsilon$. The action values are updated using an incremental average formula

$$Q_{t+1}(a) = Q_t(a) + \frac{1}{N(a)}(R_t - Q_t(a))$$

where $Q_t(a)$ is the estimated value of action a at time t, $N(a)$ is the number of times action a has been selected, and R_t is the reward received at time t.

Upper Confidence Bound. Our `UCBMAB` class implements the UCB algorithm with the following hyperparameters:

- `n_arms`: Number of possible actions (arms) for the bandit.

The agent selects actions based on confidence bounds calculated as:

$$\text{UCB}_t(a) = Q_t(a) + \sqrt{\frac{2 \log t}{N(a)}}$$

where $Q_t(a)$ is the estimated value of action a at time t, $N(a)$ is the number of times action a has been selected, and t is the total number of pulls. Action values are updated using the incremental average formula mentioned above.

Thompson Sampling. Our `ThompsonSamplingMAB` class uses a Bayesian approach to action selection with the following hyperparameters:

- `n_arms`: Number of possible actions (arms) for the bandit.

- α: Beta distribution parameter
- β: Beta distribution parameter

The agent maintains Beta distributions for each action, initialized with $\alpha = 1$ and $\beta = 1$ for each arm. Actions are selected based on samples drawn from these Beta distributions. The Beta distribution parameters are updated based on the received binary rewards as follows:

$$\alpha_{\text{new}} = \alpha_{\text{old}} + \text{reward}$$
$$\beta_{\text{new}} = \beta_{\text{old}} + 1 - \text{reward}$$

This method effectively balances exploration and exploitation by sampling from the probability distributions of each action's expected reward.

Deep Dueling DQN (naive D3QN). The `DuelingDQN` class implements a Dueling Deep Q-Network with the following architecture:

- `feature_layer`: A fully connected layer with 128 units followed by a ReLU activation function.
- `value_stream`: A fully connected layer with 128 units followed by a ReLU activation function, and another fully connected layer with 1 unit representing the state value.
- `advantage_stream`: A fully connected layer with 128 units followed by a ReLU activation function, and another fully connected layer with `action_dim` units representing the advantage values for each action.

The Q-values are computed as:

$$Q(s, a) = V(s) + \left(A(s, a) - \frac{1}{|\mathcal{A}|} \sum_{a'} A(s, a') \right)$$

where $V(s)$ is the state value, $A(s, a)$ is the advantage of action a in state s, and $\mathcal{A}$ is the set of all possible actions.

The `D3QNAgent` class implements a Double Dueling Deep Q-Network agent with the following hyperparameters:

- `state_dim`: Dimensionality of the state space.
- `action_dim`: Dimensionality of the action space.

- `lr`: Learning rate, set to 1×10^{-3}.
- `gamma`: Discount factor, set to 0.99.
- `epsilon`: Initial exploration rate, set to 1.0.
- `epsilon_decay`: Decay rate of epsilon, set to 0.995.
- `epsilon_min`: Minimum exploration rate, set to 0.01.
- `buffer_size`: Size of the replay buffer, set to 10,000.

The agent selects actions using an ϵ-greedy strategy:

- With probability ϵ, select a random action.
- With probability $1 - \epsilon$, select the action with the highest Q-value:

$$a_t = \arg\max_a Q(s_t, a)$$

The agent stores experiences in a replay buffer and updates the Q-network by sampling mini-batches of experiences. The Q-values are updated using the Bellman equation:

$$Q(s_t, a_t) = r_t + \gamma \max_{a'} Q'(s_{t+1}, a')$$

where Q' is the target Q-network. The loss is computed using Mean Squared Error (MSE):

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \left(Q(s_t^i, a_t^i) - y^i \right)^2$$

where $y^i = r_t^i + \gamma \max_{a'} Q'(s_{t+1}^i, a')$. The target network is updated periodically by copying the weights from the Q-network.

Dueling DQN with opponent modeling (D3QN-OM). Our D3QN-OM shares the same `DuelingDQN` class with the naive D3QN as MLP representation. However, the `OpponentModel` class models the behavior of opponents with the following details:

- `state_dim`: Dimensionality of the state space.
- `action_dim`: Dimensionality of the action space.
- `model`: A dictionary that maps states to action distributions.

The opponent model is updated based on observed actions, using the same `D3QNAgent` class, except for an instance of `OpponentModel` for opponent modeling. The opponent model is updated based on observed actions as follows:

1. If the current state is not already in the model, initialize the action distribution for this state as an array of zeros.
2. Increment the count for the observed action in the action distribution array corresponding to the current state.

The agent selects actions using an ϵ-greedy strategy, but when opponent modeling is enabled, Q-values are adjusted based on opponent action distributions

$$Q(s_t, a) \times \text{opponent action distribution}$$

The D3QN-OM agent interacts with stored experiences in a replay buffer as the same way the naive D3QN does.

Multiagent Proximal Policy Optimization (MAPPO). The `MLPNetwork` class implements a multi-layer perceptron network with the following architecture:

- `state_dim`: Dimensionality of the state space.
- `action_dim`: Dimensionality of the action space.
- `layers`: A fully connected network with two hidden layers, each with 128 units followed by a ReLU activation function.

Our `MAPPOAgent` class implements a MAPPO agent with the following hyperparameters:

- `state_dim`: Dimensionality of the state space.
- `action_dim`: Dimensionality of the action space.
- `lr`: Learning rate, set to 1×10^{-3}.
- `gamma`: Discount factor, set to 0.99.
- `clip_param`: Clipping parameter for PPO, set to 0.2.
- `c1`: Coefficient for value function loss, set to 0.5.

- `c2`: Coefficient for entropy bonus, set to 0.01.
- `epsilon`: Initial exploration rate, set to 1.0.
- `epsilon_decay`: Decay rate of epsilon, set to 0.995.
- `epsilon_min`: Minimum exploration rate, set to 0.01.

The agent selects actions using an ϵ-greedy strategy:

- With probability ϵ, select a random action.
- With probability $1 - \epsilon$, select the action with the highest policy probability:

$$a_t = \arg\max_a \pi(a|s_t)$$

- The exploration rate ϵ is decayed over time:

$$\epsilon = \max(\epsilon \times \text{epsilon\_decay}, \text{epsilon\_min})$$

The agent stores experiences in a memory buffer for training:

- Store experiences as tuples of `(state, action, reward, next_state, done)`.

The Generalized Advantage Estimation (GAE) is computed using the following formula:

$$\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t)$$

$$\hat{A}_t = \sum_{l=0}^{\infty} (\gamma\lambda)^l \delta_{t+l}$$

and the policy and value are updated as

- **Policy Update (Actor)**:

$$\text{ratio} = \frac{\pi(a_t|s_t)}{\pi_{\text{old}}(a_t|s_t)}$$

$$\mathcal{L}^{\text{CLIP}}(\theta) = \mathbb{E}_t \left[\min(\text{ratio} \cdot \hat{A}_t, \text{clip}(\text{ratio}, 1-\epsilon, 1+\epsilon) \cdot \hat{A}_t) \right]$$

- **Value Update (Critic)**:

$$\mathcal{L}^{\text{VF}} = \left(\hat{R}_t - V(s_t) \right)^2$$

- **Total Loss**:

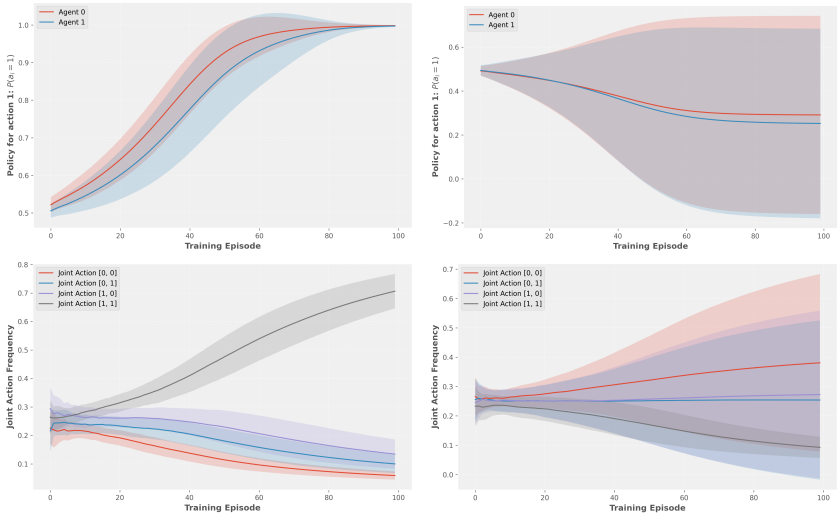
$$\mathcal{L} = \mathcal{L}^{\text{CLIP}} + c_1 \mathcal{L}^{\text{VF}} - c_2 \mathcal{H}$$

where $\mathcal{H}$ is the entropy bonus.

APPENDIX C STRATEGIC EQUILIBRIUM POLICY GRADIENT: ON FOSTERING TACIT COORDINATION IN COORDINATION GAMES

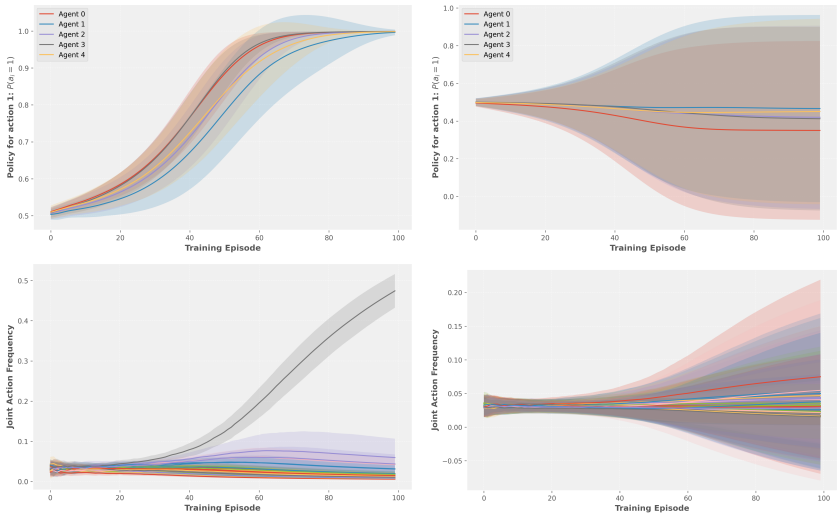
C.1 Additional Figures

Figure C.1 Average training SEPG policies towards action 1 and joint action frequencies over 24 collusive instances (first column), and over 76 non-collusive instances (second column), in the 2-player heterogeneous MPMG. The colored bands show the confidence intervals.



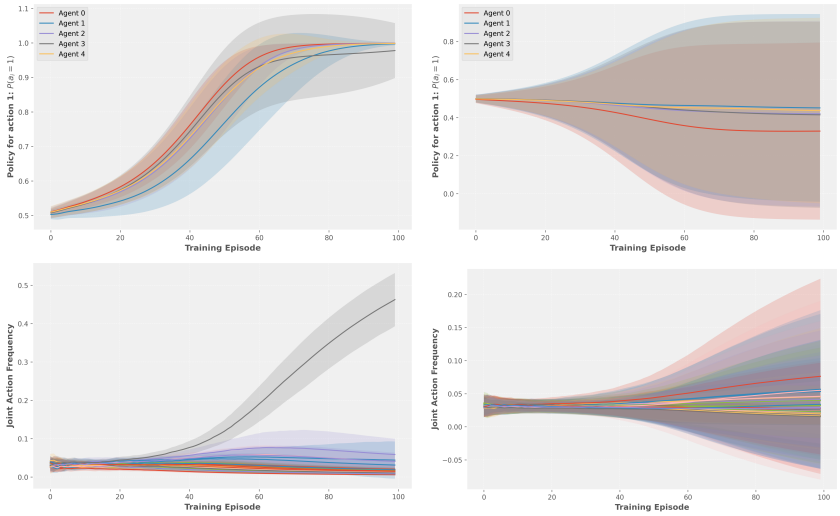
(a) Collusive Instances (b) Non Collusive Instances

Figure C.2 Average training SEPG policies towards action 1 and joint action frequencies over 14 collusive instances (first column), and over 86 non-collusive instances (second column), in the 5-player homogeneous MPMG. The colored bands show the confidence intervals.



(a) Collusive Instances (b) Non Collusive Instances

Figure C.3 Average training SEPG policies towards action 1 and joint action frequencies over 15 collusive instances (first column), and over 85 non-collusive instances (second column), in the 5-player heterogeneous MPMG. The colored bands show the confidence intervals.



(a) Collusive Instances (b) Non Collusive Instances