Neural Stochastic Differential Games for Time-series Analysis

Sungwoo Park [*1] **Byoungwoo Park** [*2] **Moontae Lee** [1 3] **Changhee Lee** [2]

Abstract

Modeling spatiotemporal dynamics with neural differential equations has become a major line of research that opens new ways to handle various real-world scenarios (*e.g.*, missing observations, irregular times, etc.). Despite such progress, most existing methods still face challenges in providing a general framework for analyzing time series. To tackle this, we adopt *stochastic differential games* to suggest a new philosophy of utilizing interacting collective intelligence in time series analysis. For the implementation, we develop the novel gradient descent-based algorithm called *deep neural fictitious play* to approximate the Nash equilibrium. We theoretically analyze the convergence result of the proposed algorithm and discuss the advantage of cooperative games in handling non-informative observation. Throughout the experiments on various datasets, we demonstrate the superiority of our framework over all the tested benchmarks in modeling time-series prediction by capitalizing on the advantages of applying cooperative games. An ablation study shows that neural agents of the proposed framework learn intrinsic temporal relevance to make accurate time-series predictions.

1. Introduction

The key challenge of time-series prediction is understanding how past observations contribute to the future and building a probabilistic model that intrinsically captures the temporal correlation. A model is required to differentiate the relative importance of past observations on the future time-series such that the redundant information is well-suppressed during inference. To address this challenge, a series of previous works followed the philosophy of defining information redundancy by adopting a concept known as *temporal decay* (Che et al., 2018; Mei & Eisner, 2017). Such an inductive bias relies on the belief that the influence of past observations exponentially decreases over time. However, due to this strong assumption, any method built upon the fixed inductive bias may fail to capture different temporal dynamics in various real-world scenarios properly.

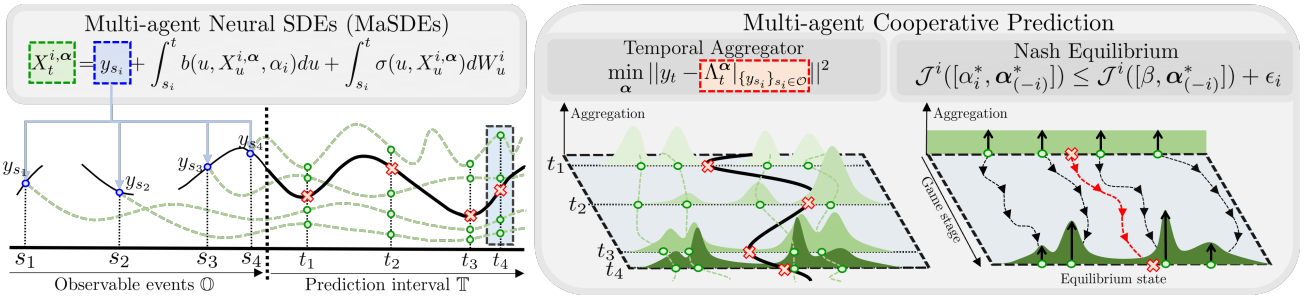
A promising direction to tackle this challenge is to *learn* the temporal correlation structure from data. One candidate for learning the implicit relation is recurrent neural networks (RNNs) which subsequently encode past observations into a latent space with a gating mechanism. Due to the implicitness of latent encoding, the model provides ambiguous explanations of how observations are related to the prediction. Furthermore, conventional RNNs are incapable of capturing the irregularly sampled time-series, which is problematic for many real-world applications.

Recently, remarkable advances have been made to model underlying continuous temporal dynamics utilizing *neural differential equations* (Chen et al., 2018; Li et al., 2020; Kidger et al., 2020) (NDEs). For instance, a stream of research (Rubanova et al., 2019; Deng et al., 2021; Schirmer et al., 2022) has focused on overcoming the initial condition problem of conventional differential equation models by encoding temporal dynamics into the latent space. A set of past observations are incorporated into latent representations and fully utilized for time-series prediction. However, owing to the inexplicable relation between temporal states in the latent representations, these methods are still incapable of showing the explicit rationale that can reveal the impact of past observations on the future.

In this paper, we present a novel framework built upon *game theory* to model temporal dynamics of time-series data. More specifically, we extend the conventional differential equation (DE) to the *multi-agent* counterpart for decomposing the observational time-series. Each agent individually encodes the impact of a partial observation into an underlying stochastic trajectory and interacts with each other to extract meaningful information to predict the future. We formulate the collaboration among agents utilizing *cooperative differential games* (Leitmann, 1974; Staatz, 1983; Sexton, 1986) to integrate the individual information and adaptively balance the importance of each agent. As a result

[*]Equal contribution [1]LG AI Research [2]Artificial Intelligence Graduate School, Chung-Ang University [3]Department of Information and Decision Sciences, University of Illinois Chicago. Correspondence to: Sungwoo Park <sungwoopark.lg@lgresearch.ai>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).



(a) Multi-agent Neural SDEs (b) Temporal Aggregation (c) Differential games

Figure 1: Conceptual illustration of the proposed framework: (a) The proposed MaSDEs encode separate initial conditions and propagate individual decisions over the interval $t \in [s_i, T]$. The bold black line indicates the time-series on prediction interval $\mathbb{T}$, and the dashed green lines are the *decisions* of neural agents. (b) The temporal aggregation integrates separate decisions. The red crosses show the predictor $\Lambda_t^{\boldsymbol{\alpha}}$ representing the cooperative prediction. (c) The optimal action profiles $\boldsymbol{\alpha}^*$ are obtained in the Nash equilibrium.

of the differential game, cooperative agents achieve the *Nash equilibrium* and agree to suppress non-informative observations to highlight the contribution of important observations for accurate prediction.

Main Contribution. To the best of our knowledge, this work is the first attempt to adopt a philosophy of game theory in dealing with multivariate time-series. For tractability of applying differential games, we propose a novel gradient descent-based algorithm called *deep neural fictitious play*, which operates in a tractable and parallel way to obtain the Nash equilibrium. Theoretical results based on the Feynman-Kac formalism guarantee the convergence of the proposed algorithm and unveil the underlying rationale for the temporal relevance of past and future. To demonstrate superiority, we validate our method on various types of multivariate synthetic and real-world datasets. Experimental results show that our method outperforms state-of-the-art DE-based methods and verify the implication of theoretical findings about the advantages of applying cooperative games.

2. Methods

Problem Setup. Consider a general time-series forecasting set-up where each instance of a time-series is defined on the entire time interval $[0, T]$ comprising of observable past (*i.e.*, $\mathbb{O}$) and target future intervals (*i.e.*, $\mathbb{T}$). We assume that the irregularly sampled observations are collected at time stamps $\mathcal{O} = \{s_1, \ldots, s_I\}$ where $s_i \in \mathbb{O}$ for $i \in \mathbb{N} := \{1, \ldots, I\}$. Then, we focus on building a probabilistic model that can predict future values at any time $t \in \mathbb{T}$ based on the set of past observations in $\mathbb{O}$. We define I-simplex as $\Delta^I := \{a \in [0,1]^I | \mathbf{1}^\top a = 1\}$.

Multi-agent Neural SDEs. Suppose we have a set of I number of past observations for a given time-series, *i.e.*, $\{y_{s_i}\}_{s_i \in \mathcal{O}}$. The primary object of our interest is the set of stochastic processes $\mathbf{X}_t^{\boldsymbol{\alpha}} = [X_t^{1,\boldsymbol{\alpha}}, \ldots, X_t^{I,\boldsymbol{\alpha}}]$ parameterized by multiple control agents $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_I]$, which is defined as a collection of solutions to the following multi-agent stochastic differential equations (MaSDEs)[1]:

Definition 2.1. ***(Multi-agent Neural SDEs).*** *Let us define $\mathbb{R}^d$-valued stochastic processes $\mathbf{X}_t^{\boldsymbol{\alpha}}$ controlled by Markovian control agents $\boldsymbol{\alpha}$:*

$$(\mathbf{X}_t^{\boldsymbol{\alpha}}, \underbrace{\{y_{s_i}\}_{s_i \in \mathcal{O}}}_{\textit{past observations}}) : \begin{cases} dX_t^{i,\boldsymbol{\alpha}} = b_t^{\alpha_i} dt + \sigma_t dW_t^i, \\ b_t^{\alpha_i} := b(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i), \\ \sigma_t := \sigma(t, X_t^{i,\boldsymbol{\alpha}}), \ \forall i \in \mathbb{N}, \end{cases} \tag{1}$$

where $\{W_t^i\}_{i \in \mathbb{N}}$ are independent d-dimensional Wiener processes, and $b_t^{\alpha_i} := b : [0,T] \times \mathbb{R}^d \times \mathbb{A} \to \mathbb{R}^d$ and $\sigma_t := \sigma : [0,T] \times \mathbb{R}^d \to \mathbb{R}^{d \times d}$ are drift and diffusion functions, respectively. The control agent $\alpha_i : [0,T] \times \mathbb{R}^d \times \Theta \to \mathbb{R}^d$, which we refer to as a neural agent, is modeled as a neural network $\alpha_i := \alpha_i(t, X_t^{i,\boldsymbol{\alpha}}; \theta_i)$ parameterized by θ_i.

Each neural agent takes spatio-temporal variables $(t, X_t^{i,\boldsymbol{\alpha}})$ and produces infinitesimally successive outputs $(t + dt, X_{t+dt}^{i,\boldsymbol{\alpha}})$. We define a neural agent's *decision* as a continuous stochastic trajectory over the future interval $\mathbb{T}$, *i.e.*, $\{X_t^{i,\boldsymbol{\alpha}}\}_{t \in \mathbb{T}}$. By setting the past observation y_{s_i} as the *initial condition*, the decision of a neural agent can be considered as an information propagator that can explicitly represent the impact of the corresponding past observation via the conditional expression $p(X_{t \in \mathbb{T}} | y_{s \in \mathbb{O}})$; see Figure 1-(a) for a pictorial illustration.

Multi-conditioned Score-based Predictor. Since each neural agent α_i can only represent the individual impact of partial information y_{s_i}, we aggregate the individual decisions made by each neural agent to capitalize on the temporal dynamics available from the entire set of past observations. Inspired by recent work (Song et al., 2021), we introduce the *multi-conditioned predictor* $\Lambda_t^{\boldsymbol{\alpha}}$ that produces the collaborative prediction between neural agents.

[1] From this point forward, we use boldface letters to denote a collection of objects obtained from multi-agents, *e.g.*, $\mathbf{X}_t^{\boldsymbol{\alpha}} = [X_t^{1,\boldsymbol{\alpha}}, \ldots, X_t^{I,\boldsymbol{\alpha}}]$, and we omit the dependence on temporal state when clear in the context.

Definition 2.2. ***(Multi-conditioned Predictor)***[2] *Let us define a simplex-valued function* $\mathbf{A}_t^{\boldsymbol{\alpha}} : \mathbb{T} \times \mathbb{R}^d \times \Theta \to \Delta^I$. *Then, the prediction made by proposed MaSDEs is defined as follows:*

$$d\Lambda_t^{\boldsymbol{\alpha}} := \sum_{i\in\mathbb{N}} A_t^{\alpha_i} \left[b_t^{\alpha_i} - \frac{1}{2}\left(\nabla\cdot\Sigma_t + \Sigma_t \nabla \log p_t^i(\cdot|y_{s_i})\right) dt \right], \tag{2}$$

where we denote $\sigma_t\sigma_t^T := \Sigma_t$, $\nabla \log p_t^i(\cdot|y_{s_i})$ *is the score function of the agent* i *conditioned by past observation* y_{s_i}.

In the definition, we suggest *temporal aggregation* $\mathbf{A}_t^{\boldsymbol{\alpha}} = [A_t^{\alpha_1}, \dots, A_t^{\alpha_I}]$ that provides the attention map (*i.e.*, weighted sum) to each decision regarding the relative importance, *i.e.*, $\mathbf{A}_t^{\boldsymbol{\alpha}} : \mathbb{T} \times \mathbb{R}^d \times \Theta \to \Delta^I$. Given the temporal aggregation, the predictor $\Lambda_t^{\boldsymbol{\alpha}}$ produces an aggregated decision from the decisions made by the individual agents. Since the goal of this work is to capture the future states given past observable events accurately, we hope to find the best combination of neural agents $\boldsymbol{\alpha}$ to minimize the discrepancy at every time $\mathbb{T}$:

$$y_t \overset{[\alpha_1,\dots,\alpha_I]}{\Longleftrightarrow} \Lambda_t^{\boldsymbol{\alpha}}|_{\{y_{s_i}\}_{s_i\in\mathcal{O}}}, \tag{3}$$

where y_t is a **target** to predict, $\Lambda_t^{\boldsymbol{\alpha}}$ is a **prediction** made by MaSDEs conditioned on **past observations** $\{y_{s_i}\}_{s_i\in\mathcal{O}}$.

Stochastic Optimal Control. To systemically search the optimal neural agent, we adopt the *stochastic optimal control* (Yeung & Petrosjan, 2006; Carmona & Delarue, 2018) as our central methodology to formally define an objective functional $\mathcal{J}^i$. Specifically, each neural agent is given a behavioral rule to follow:

Definition 2.3. ***(Objective Functional).*** *Let us consider generic objective functionals* $\mathcal{J}^i$ *of neural agents:*

$$\begin{aligned}&\mathcal{J}^i(t, x, [\alpha_i, \boldsymbol{\alpha}_{(-i)}]) \\ &= \mathbb{E}\left[\int_{\mathbb{T}} h^i\big(s, \mathbf{X}_s^{\boldsymbol{\alpha}}, [\alpha_i, \boldsymbol{\alpha}_{(-i)}]\big) ds + \Psi^i(X_T^{i,\boldsymbol{\alpha}}) \,\middle|\, \mathbf{X}_t = x\right],\end{aligned} \tag{4}$$

where we denote $\mathbf{X}_t^{\boldsymbol{\alpha}} = [X_t^{i,\boldsymbol{\alpha}}, \mathbf{X}_t^{(-i),\boldsymbol{\alpha}}]$ *and other agent's decisions as* $\mathbf{X}_s^{(-i),\boldsymbol{\alpha}} \in \mathbb{R}^{d(I-1)}$.

Among the various designs for functions h^i and Ψ^i, we especially consider the following setting:

$$(\boldsymbol{h}, \boldsymbol{\Psi}) : \begin{cases} h^i\left(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, [\alpha_i, \boldsymbol{\alpha}_{(-i)}]\right) = \|y_t - \Lambda_t^{\boldsymbol{\alpha}}\|^2, \\ \Psi^i(X_T^{i,\boldsymbol{\alpha}}) = \frac{1}{2}||X_T^{i,\boldsymbol{\alpha}}||^2. \end{cases} \tag{5}$$

The *cost* h^i is designed to encourage each neural agent to make contributions in the temporal aggregation under the goal of accurate prediction. The *terminal cost* Ψ^i regularizes decisions at the terminal state and plays a central role in the Feynman-Kac formulas to solve the cooperative game.

Given the environment (*i.e.*, the entire interacting objects), each neural agent changes its action to find a *value function* $\mathcal{V}^i$ that defines the optimal state of actions:

$$\begin{aligned}&\{\mathcal{V}^i = \min_{\alpha_i\in\mathbb{A}^i} \mathcal{J}^i(t, \cdot, [\alpha_i, \boldsymbol{\alpha}_{(-i)}]), \quad i \in \mathbb{N}, \} \\ &\supseteq \{\min_{\theta_i\in\Theta} \mathcal{J}^i(t, x, [\alpha_i(\cdot, \cdot; \theta_i), \boldsymbol{\alpha}_{(-i)}(\cdot, \cdot; \boldsymbol{\theta}_{(-i)})])\},\end{aligned} \tag{6}$$

As an equivalent form, the value function in (6) can be obtained by solving the PDE called *Hamiltonian-Jacobi-Bellman equation* (HJBE) of individual neural agent:

$$\mathcal{V}_t^i + H^i(t, \cdot, F_t^i, [\alpha_i, \boldsymbol{\alpha}_{(-i)}]) + \frac{1}{2}\mathbf{Tr}(\Sigma\nabla^2\mathcal{V}_t^i) = 0, \tag{7}$$

where $\mathcal{V}_t^i := \partial_t \mathcal{V}^i$, H^i is the stochastic Hamiltonian system and F_t^i is an adjoint variable related to σ_t and b_t in MaSDEs. In the following section, we will further elaborate to tackle solving HJBE under cooperative game scenarios.

2.1. Stochastic Differential Games

If the past observation y_s is highly associated with the future event y_t (*i.e.*, $p(y_{t\in\mathbb{T}}|y_{s\in\mathbb{O}}) \uparrow$), it is reasonable to pay high attention (*i.e.*, $\boldsymbol{A}^{\boldsymbol{\alpha}}(t, \cdot) \uparrow$) to capitalize on the observation y_s. Unfortunately, determining the optimal attention is a daunting task as the temporal correlation is intrinsically data-dependent and not given a priori in general. Hence, the method requires a unified structure that (i) can systemically extract information from data to adjust the temporal aggregation $\mathbf{A}^{\boldsymbol{\alpha}}$ based on the relative importance of past observations and (ii) can provide a framework on the coalition of the interacting agents for the proper coordination.

To address the above requirements, we propose a novel framework to formalize the time-series prediction problem as a *cooperative differential games* between multiple agents. Cooperative differential games have been extensively studied in many disciplines (Perelman et al., 2011; Jørgensen et al., 2010; Yeung & Petrosyan, 2012) to investigate the group rationality of agents sharing the same goal. Inspired by these works, we design a cooperative differential game that provides *cooperative prediction* by achieving optimally balanced past information. The critical point is that the neural agent supporting a non-informative observation voluntarily sacrifices its own cost (*i.e.*, $\mathcal{V}^i \uparrow$) by decreasing the influence on temporal aggregation so that the impact of other informative observations (*i.e.*, $\mathcal{V}^{i\neq j} \downarrow$) can be further emphasized. Such cooperative behavior is mutually advantageous to the interacting agents as it ultimately achieves performance improvement. Figure 1-(c) shows the conceptual illustration of the proposed cooperative game where enthusiastic agents gradually balance the homogeneous temporal aggregation to follow the group rationality.

[2]Refer to Appendix A.2 for detailed formulation.

Nash Equilibrium. For a mathematical derivation of the optimal decision states, we introduce an important type of equilibrium state called *Nash Equilibrium* as follows:

Definition 2.4. ***(Markovian ϵ-Nash Equilibrium)*** *Let us consider a set of closed-feedback type Markovian controller parameterized by neural networks $\boldsymbol{\alpha}^* := \{\alpha_i^*\}_{i\in\mathbb{N}}$ that induce value functions. For the arbitrary actions $\boldsymbol{\beta}$ taken in action set $\mathbb{A}$, we say that neural agents are in the ϵ-Nash Equilibrium if the following inequality is satisfied:*

$$\mathcal{J}^i(t,x,[\alpha_i^*,\boldsymbol{\alpha}_{(-i)}^*]) \leq \mathcal{J}^i(t,x,[\beta_i,\boldsymbol{\alpha}_{(-i)}^*]) + \epsilon_i, \quad (8)$$

The inequality in (8) describes an optimal state of interacting agents having no incentive to change their actions since any non-optimal action $\boldsymbol{\beta}$ produces worse cost (with a margin $\boldsymbol{\epsilon} := \{\epsilon_i\}_{i\in\mathbb{N}} > 0$) than taking the optimal actions $\boldsymbol{\alpha}^*$. Here, the marginal constant $\boldsymbol{\epsilon}$ represents a degree of efficiency using neural agents to achieve the perfect equilibrium. While each neural agent mutually finds the best response, the entire cost is comprehensively minimized, and a more accurate prediction can be obtained. Further, the following relation implies that the equilibrium state achieved in (8) is indeed a result of cooperation:

$$\begin{aligned}\mathcal{J}_{co}(t,x,\boldsymbol{\alpha}^*) &= \sum_i^I \mathcal{J}^i(t,x,[\alpha_i^*,\boldsymbol{\alpha}_{(-i)}^*]) \\ &\leq \sum_i^I \mathcal{J}^i(t,x,[\alpha_i^*,\boldsymbol{\beta}_{(-i)}]) + I\sup_i \epsilon_i,\end{aligned} \quad (9)$$

where $\mathcal{J}_{co}$ is the objective functional that describes the coalition cost (Yeung & Petrosjan, 2006) of the cooperative neural agents. The inequality in (9) can be easily shown by the algebraic property of cost functions as the following:

$$\begin{aligned}&\mathbb{E}\left[h^i\left(t,X_t^{i,\boldsymbol{\alpha}},\mathbf{X}_t^{(-i),\boldsymbol{\alpha}},[\alpha_i^*,\boldsymbol{\alpha}_{(-i)}^*]\right)\right] \\ &= \mathbb{E}\left[h^j\left(t,X_t^{j,\boldsymbol{\alpha}},\mathbf{X}_t^{(-j),\boldsymbol{\alpha}},[\alpha_j^*,\boldsymbol{\alpha}_{(-j)}^*]\right)\right], \quad i\neq j.\end{aligned} \quad (10)$$

In order to achieve the Nash equilibrium for multi-agents, the previous interest in considering separate and individual objectives in (6) needs to be shifted to a new type of scenario where interactions are key factors to decide optimality. From this new perspective, we transform the original value function that only considers insensitive feedback into a new one that can consider optimal colleagues' feedback driven by their contribution to the prediction:

$$\begin{cases}\mathcal{V}_t^i = \min_{\alpha_i^*\in\mathbb{A}^i}\mathcal{J}^i(t,\cdot,[\alpha_i^*,\boldsymbol{\alpha}_{(-i)}^*]), \quad i\in\mathbb{N}, \\ \mathcal{V}_t^i + H^i(t,\cdot,F_t^i,[\alpha_i^*,\boldsymbol{\alpha}_{(-i)}^*]) + \frac{1}{2}\mathbf{Tr}(\Sigma\nabla^2\mathcal{V}_t^i) = 0,\end{cases} \quad (11)$$

Since finding an analytic solution to the Nash equilibrium is PPAD hard (Daskalakis et al., 2009; Goldberg, 2011), it is intractable to directly compute the equilibrium owing to a large number of agents considered in time-series prediction problems of our experiments (*i.e.*, $I \approx 50$). Thus, for a tractable computation, we instead rely on an alternative dynamical formulation called *fictitious play* (Cardaliaguet & Hadikhanloo, 2017) in the following section.

3. Deep Neural Fictitious Play

The fictitious play is an iterative procedure that decouples interacting agents and separately searches value functions. The central idea is first to share public information about the entire system and then to decouple value functions by solving individual Hamiltonian-Jacobi-Bellman equations (HJBEs) in (7). To this goal, we start by introducing the set of adjoint equations called the *forward-backward stochastic differential equations* (FBSDEs) (Carmona, 2016) as a probabilistic framework to tackle optimal control problems of obtaining value functions.

Theorem 3.1. ***(Non-linear Feynman Kac theorem (Pham, 2015))****. Let us consider closed-loop feedback type Markovian controls $\boldsymbol{\alpha}_t$. We define the triplet $(\mathbf{X}_t^{\boldsymbol{\alpha}},\mathbf{Y}_t^{\boldsymbol{\alpha}},\mathbf{Z}_t^{\boldsymbol{\alpha}})$ that constitutes the system of SDEs:*

$$\begin{pmatrix}\mathbf{X}_t^{\boldsymbol{\alpha}}\\ \mathbf{Y}_t^{\boldsymbol{\alpha}}\\ \mathbf{Z}_t^{\boldsymbol{\alpha}}\end{pmatrix} : \begin{cases}dX_t^{i,\boldsymbol{\alpha}} = \sigma(t,X_t^{i,\boldsymbol{\alpha}})dB_t^i, \\ dY_t^{i,\boldsymbol{\alpha}} = -H_t^i dt + Z_t^{i,\boldsymbol{\alpha}}\cdot dB_t^i, \\ Y_T^{i,\boldsymbol{\alpha}} = \Psi^i(X_T^{i,\boldsymbol{\alpha}}),\ i\in\mathbb{N},\end{cases} \quad (12)$$

where H_t^i is the stochastic Hamiltonian system. Then, auxiliary adjoint variables $Y_t^{i,\boldsymbol{\alpha}}$ and $Z_t^{i,\boldsymbol{\alpha}}$ can be reformulated as follows:

$$\begin{cases}Y_t^{i,\boldsymbol{\alpha}} = \mathcal{J}^i(t,\mathbf{X}_t^{\boldsymbol{\alpha}},\boldsymbol{\alpha}), \quad Z_t^{i,\boldsymbol{\alpha}} = \nabla_x\mathcal{J}^i(t,\mathbf{X}_t^{\boldsymbol{\alpha}},\boldsymbol{\alpha}), \\ H_t^i = H^i(t,\mathbf{X}_t^{\boldsymbol{\alpha}},F_t^{i,\boldsymbol{\alpha}},[\alpha_i,\boldsymbol{\alpha}_{(-i)}]) \\ \quad = \left[\sigma(t,X_t^{i,\boldsymbol{\alpha}})^{-T}Z_t^{i,\boldsymbol{\alpha}}\right]\cdot b(t,X_t^{i,\boldsymbol{\alpha}},\alpha_i) + h^i(t,\mathbf{X}_t^{\boldsymbol{\alpha}},\boldsymbol{\alpha}),\end{cases} \quad (13)$$

where we denote $F_t^{i,\boldsymbol{\alpha}} := \sigma(t,X_t^{i,\boldsymbol{\alpha}})^{-T}b(t,X_t^{i,\boldsymbol{\alpha}},\alpha_i)$ for simplicity.

To solve the optimal control problem, the existing method (Han & Hu, 2020) considers quadratic linear forms in the forward dynamics and its corresponding global convex Hamiltonian system (*i.e.*, $\nabla_{\alpha^i}H^i = 0$). Despite the numerical advantage in obtaining an optimal control, the restricted form of linear SDEs limits the usage of their method on high-level applications such as time-series prediction. This shortcoming motivates us to utilize neural networks when modeling control agents for enough expressivity. A critical issue, here, is to approximate the solution to non-convex type HJBEs that support the neural networks structure. To tackle this issue, we reformulate the equivalent adjoint problem to find value functions:

$$\theta_i^* = \arg\min_{\theta_i}\int_{\mathbb{T}} dY_t^{i,\boldsymbol{\alpha}}([\theta_i,\boldsymbol{\theta}_{(-i)}])dt. \quad (14)$$

Algorithm 1 Deep Neural Fictitious Play

Require: Time-series data $\boldsymbol{y} = (\{y_{s_i}\}_{s_i \in \mathbb{O}}, \{y_t\}_{t \in \mathbb{T}})$, Neural network Parameters $\boldsymbol{\theta}^{m=0} = [\theta_1^0, \dots, \theta_I^0]$, $\gamma_m := \gamma = 10^{-3}$.
for $m = 1$ to M **do**
  Simulate MaSDEs with action profiles $\boldsymbol{\alpha}^{m-1}$ obtained from the previous stage $m-1$.
  for all $i \in \mathbb{N}$ **in parallel do**
    Fix the environment of the previous stage for the agent i.
    $$\boldsymbol{\theta}^{m-1} = [\theta_i^{m-1}, \bar{\boldsymbol{\theta}}_{(i-1)}^{m-1}], \quad \boldsymbol{\alpha}^{m-1} = [\alpha_i^{m-1}, \bar{\boldsymbol{\alpha}}_{(i-1)}^{m-1}].$$
    Solve the decoupled HJBEs in equation (16) for agent i.
    $$\mathcal{V}_t^{i,m} + \inf_{\alpha_i} H^i([\alpha_i, \bar{\boldsymbol{\alpha}}_{(i-1)}^{m-1}]) + \frac{1}{2}\mathbf{Tr}(\Sigma \nabla^2 \mathcal{V}^{i,m}) = 0,$$
    $$\theta_i^m = \mathbb{E}\left[\theta_i^{m-1} - \gamma \int_{\mathbb{T}} \nabla_{\theta_i} dY_t^{i,\boldsymbol{\alpha}}([\theta_i^{m-1}, \bar{\boldsymbol{\theta}}_{(-i)}^{m-1}])dt\right].$$
    Collect the updated neural parameters $\boldsymbol{\theta}^m = [\theta_1^m, \dots, \theta_I^m]$
  end for
  Update entire action profiles $\boldsymbol{\alpha}^m \leftarrow \boldsymbol{\alpha}^{m-1}(\cdot, \cdot, \boldsymbol{\theta}^m)$.
end for
Return Optimal action profiles $\boldsymbol{\alpha}^* = [\alpha_1^*, \dots, \alpha_I^*]$.

Instead of searching the analytic solution to HJBEs (*i.e.*, $\mathbb{A}^i \approx \mathbf{ker}\nabla_{\alpha^i} H^i$), the newly suggested adjoint equation in (14) enables us to apply the gradient descent scheme to find optimal actions. In this case, our method searches the parameter space to obtain the optimal admissible action (*i.e.*, $\mathbb{A}^i \approx \alpha^i(\cdot, \cdot; \mathbf{ker}\nabla_\theta H^i[\alpha^i(\theta^i)])$) meaning that we focus on the local Nash equilibrium.

Let us define an entire set of action profiles as $\boldsymbol{\alpha}^m$ at stage $m \in \{1, \dots, M\}$. Then, our method iteratively conducts the following two-step optimization:

1) *Information Distribution.* At the beginning of the m-th stage, each neural agent publicly shares the information about the entire system $(\boldsymbol{\alpha}^{m-1}, \mathbf{X}_t^{\boldsymbol{\alpha},m-1})$ from the $(m-1)$-th stage. In this case, we can define the decoupled value function for individual neural agents as follows:

$$\mathcal{V}^{i,m}(t,x) = \inf_{\alpha^i \in \mathbb{A}^i} \mathbb{E}\left[\mathcal{J}^{i,m}(t, \mathbf{X}_t^{\boldsymbol{\alpha}^{m-1}}, [\alpha_i^{m-1}, \bar{\boldsymbol{\alpha}}_{(-i)}^{m-1}]) \middle| \mathbf{X}_t^{\bar{\boldsymbol{\alpha}}^{m-1}}\right], \tag{15}$$

where $\mathbf{X}_t^{\bar{\boldsymbol{\alpha}}^{m-1}}$ indicates the set of decisions driven by action profiles $\bar{\boldsymbol{\alpha}}^{m-1}$ of the $(m-1)$-th stage. In this step, each neural agent recognizes the response of colleagues towards the environment and makes its relative decision. Note that each agent solves a minimization problem according to the fixed environment of the previous stage.

2) *Decoupled Gradient Descent.* After defining the decoupled problems in (15), each neural agent solves its individual adjoint problem in (14) and updates its parameters as

$$\theta_i^{m+1} = \mathbb{E}\left[\theta_i^m - \gamma_m \int_{\mathbb{T}} \nabla_{\theta_i} dY_t^{i,\boldsymbol{\alpha}}([\theta_i^m, \bar{\boldsymbol{\theta}}_{(-i)}^m])dt\right], \tag{16}$$

where γ_m is the learning rate of the optimizer at the m-th stage. In (16), gradient descent is applied to minimize the separate cost $\mathcal{J}^{i,m}$ while fixing the parameters of colleagues $\bar{\boldsymbol{\theta}}_{(-i)}^m$. After solving (16) for I neural agents in a parallel way, we collect action profiles and update the public information for the next stage, *i.e.*, $\boldsymbol{\alpha}_i^{m+1} \leftarrow \boldsymbol{\alpha}_i^m(\cdot, \cdot, \theta + d\theta)$.

3.1. Theoretical Analysis

Convergence of the Fictitious Play. In the previous contents, we suggest an alternative way to approximate the Nash equilibrium via the gradient descent scheme. In what follows, we show the existence of an action set $\mathbb{A}$ related to the marginal constant $\boldsymbol{\epsilon} := \{\epsilon_i\}_{i \in \mathbb{N}}$ in the inequality of (8).

Proposition 3.2. ***(informal)*** *The proposed algorithm 1 assures an existence of action set $\boldsymbol{\alpha}(\boldsymbol{\epsilon}) \in \mathbb{A}$ where neural agents in the set archive the Markovian ϵ-Nash equilibrium. Moreover, there exists a constant m^* such that the obtained optimal actions $\boldsymbol{\alpha}^{m^*} \in \boldsymbol{\alpha}(\boldsymbol{\epsilon})$ preserve the stochastic optimality by solving HJBEs in (7).*

The main finding of Proposition 3.2 is that, under the proposed training scheme of deep neural fictitious play, **proposed MaSDEs with a large capacity assures small marginal values $\boldsymbol{\epsilon}$ of Nash equilibrium**. In other words, the proposed gradient descent-based algorithm in Alg 1 is a well-defined approximation for solving cooperative differential games. Next, we conduct the convergence analysis of multi-conditioned score-based predictors trained by the proposed algorithm.

Corollary 3.3. ***(Convergence of Predictor)*** *There exist a constant C that is proportional to I, $\mathbb{T}$ and Lipschitz constants L_b, L_α for $\{b_i, \alpha_i\}_{i \in \{1,\dots,I\}}$ such that the following relation holds:*

$$\mathbb{E}\left[\int_{\mathbb{T}} \left\|\boldsymbol{\Lambda}_s^{\boldsymbol{\alpha}^m} - \boldsymbol{\Lambda}_s^{\boldsymbol{\alpha}^{m^*}}\right\|^2 ds\right] \propto O(||\boldsymbol{\theta}^m - \boldsymbol{\theta}^{m^*}||^3 C\,[I, \mathbb{T}, L_b, L_\alpha]\,\overline{\Sigma}), \tag{17}$$

where $\overline{\Sigma} := \sup_{t \in \mathbb{T}} \mathbb{E}\left[\|\boldsymbol{\Sigma}(t, \cdot)\|_F\right]$ is the maximal bound with respect to quadratic variations of neural agents.

Corollary 3.3 shows that the convergence of the temporal aggregation is highly dependent on the dynamics of the gradient descent (*i.e.*, $d\boldsymbol{\theta}^m$) during the fictitious play. If one reformulates the proposed cooperative game into a zero-sum type scenario, the opposed direction of gradients induced by conflict agents can cause divergent learning dynamics and unpleasant results. In other words, the proposed gradient descent-based fictitious play fits the cooperative scenario where neural agents pursue a shared goal. Appendix A.5 provides the numerical analysis of the behavior of neural agents under the cooperation/competition to elucidate how the proposed fictitious play operates in different scenarios.

Methods	BAQD MSE ↓ / NLL ↑	Speech Commands MSE ↓ / NLL ↑	Physionet MSE ↓ / NLL ↑
RNN-VAE	0.4493 / −1.878	0.5205 / −2.234	0.6494 / −2.878
GRU-D	0.4299 / −1.781	0.4721 / −1.992	0.4403 / −1.832
Latent ODE (RNN-Enc)	0.4058 / −1.660	0.5098 / −2.180	0.5313 / −2.288
Latent ODE (ODE-Enc)	0.3839 / −1.551	0.4950 / −2.106	0.5046 / −2.154
Latent SDE (RNN-Enc)	0.4049 / −1.655	0.5003 / −2.132	0.5301 / −2.282
Latent SDE (ODE-Enc)	0.3806 / −1.534	0.4980 / −2.121	0.4966 / −2.114
NJ-ODE	0.4803 / −2.033	0.5314 / −2.288	0.5167 / −2.214
Res-Flow	0.4379 / −1.821	0.4883/ − 2.073	0.4785 / −2.025
GRU-Flow	0.4853 / −2.054	0.4979 / −2.120	0.4417 / −1.840
Neural Laplace	0.3633 / −1.447	0.4547 / −1.918	0.5550 / −2.392
CRU	0.3613 / −1.438	0.4464 / −1.863	0.5081 / −2.172
CSDE-TP	0.3481 / −1.371	0.4460 / −1.861	0.3958 / −1.610
Ours	**0.2892** / **−1.077**	**0.4087** / **−1.674**	**0.3829** / **−1.545**

Table 1: Evaluation of time-series prediction on the Air Quality/Speech Commands/Physionet datasets. The best results are highlighted in **bold**. The second best results are colored blue. Evaluations metrics are scaled by 10^{-2} and 10^{-3}, respectively.

Robustness to Non-informative Signals. Finally, we give an in-depth theoretical analysis of how the uncertainty made by non-informative past observations can be dealt with by our method. To model the uninformativeness, we consider a scenario when white noises, *i.e.*, i.i.d d-dimensional Gaussian vectors $\{Z^k_{s_i}\}_{k\in\{1,\cdots,N\}} \sim \mathcal{N}(0,\varsigma)$, are observed during the inference. In this scenario, the temporal aggregation of perturbed signals $\tilde{\Lambda}^{\boldsymbol{\alpha}}_t$ can be rewritten as the following:

$$\tilde{\Lambda}^{\boldsymbol{\alpha}}_t := \Lambda^{\boldsymbol{\alpha}}_t|_{\{y_{s_1},\cdots,y_{s_i}+Z^k_{s_i},\cdots,y_{s_I}\}}. \tag{18}$$

The following result identifies how the rationale of MaSDEs effectively controls the uncertainty:

Proposition 3.4. *Let $\boldsymbol{\lambda}_d$ be a spectrum of noise covariance ς. Then, there exist numerical constants $\boldsymbol{v}, c, C, M_2, D_3, \tilde{L}, \gamma > 0$ such that the following inequality holds with probability at least $1-\varphi$:*

$$\underbrace{\mathcal{W}_2^2(\tilde{\Lambda}^{\boldsymbol{\alpha}}_t, y_t)}_{\textit{Uncertainty}} \leq \underbrace{A^i_t}_{\textit{Aggregation}} \underbrace{\frac{D_3}{4|\mathbb{T}|}\mathbf{1}^T\boldsymbol{\lambda}_d}_{\textit{Variance Effect}} + \underbrace{\frac{1}{2\tilde{L}^2\gamma}\sqrt{\frac{\boldsymbol{v}}{M_2(1+|\mathbb{T}|)}}}_{\textit{Training Errors}} + \underbrace{\left(\frac{\log(2C/\varphi)}{cN}\right)^{-d/2}}_{\textit{Particle Approximation}}. \tag{19}$$

where $\mathcal{W}_2$ is 2-Wasserstein distance, $D_3 = \sup_{t\in\mathbb{T}} \Sigma_t\Sigma_0^{-1}$ is a maximal variation of the model volatility.

The uncertainty induced by perturbed signals on the left-hand side can be bounded by each term, where the first two terms give the following important implications:

- **Canceling Variance Effect.** In the first term, the variance effect (*i.e.*, $\mathbf{1}^T\boldsymbol{\lambda}_d$) is canceled out by learned temporal aggregation A^i_t. As is clear from the inequality, one can presume that MaSDEs robustly capture future dynamics regardless of noisy past events. We will empirically validate this property in Section 5.2.
- **Optimal Error Bound.** Proposition 3.2 and the Nash equilibrium (8) guarantee that the second term is the best trial of cooperative agents to minimize the training error. Since an arbitrary set of actions that fails to satisfy Nash equilibrium induces a larger upper bound of this term, we argue that the proposed training methodology with cooperative games plays a key role in minimizing the uncertainty due to non-informative signals.

4. Related Work

Neural DEs for Time-series Analysis. In the pioneering works (Dupont et al., 2019; Rubanova et al., 2019), the general framework of encoding the complex time-series into the latent space was first introduced to improve the representational power of conventional Neural ODE. Further, latent SDEs (Li et al., 2020) was proposed to enrich conventional deterministic models by considering a stochastic component (*e.g.*, Wiener process). Neural RDE (Morrill et al., 2021) exploited the representation of log-signatures of successive time-series. CLPF (Deng et al., 2021) combined two distinctive ideas, continuous SDE and normalizing flow, to model continuous latent flows. CRU (Schirmer et al., 2022) extended Kalman filters to model-based continuous-discrete filters and showed the relation to neural SDEs. Neural Laplace (Holt et al., 2022) showed a novel interpretation of time-series modeling by introducing the Laplace representation that generalizes conventional DEs in the frequency domain. CSDE-TP (Park et al., 2022) suggested a different perspective by adopting a control-theoretic interpretation of time-series prediction tasks to obtain the optimal paths driven by neural agents.

Modeling Temporal Dynamics. In probabilistic modeling for time-series analysis, existing methods assumed a strong inductive bias on temporal states. The conditional future state $p(\mathbb{T}|\mathbb{O})$ given past observations $\{y_{s_i}\}_{s_i\in\mathbb{O}}$ is related to the temporal difference between $\mathbb{T}$ and $\mathbb{O}$. For

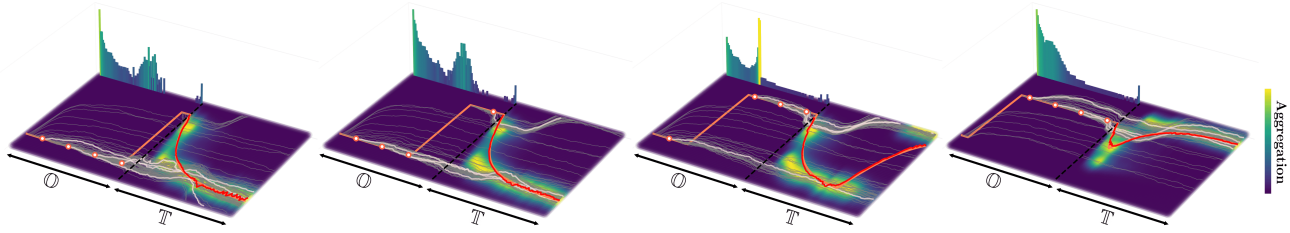


Figure 2: Qualitative result on Mackey-Glass DDE. The highlighted region in the plane depicts the level of temporal aggregation for each decision $X_t^{i,\boldsymbol{\alpha}}$. The vertical histogram in the past interval $\mathbb{O}$ represents the averaged temporal aggregation on the entire future interval $\mathbb{T}$ (*i.e.*, $\mathbb{E}_{t\sim\mathbb{T}}[A_t^{\alpha_i}]$) estimated by agent i.

example, GRU-D (Che et al., 2018) suggested a temporal decay module in the recurrent model that can exponentially decrease the influence of past observations. Another strand of works (Mei & Eisner, 2017; Zuo et al., 2020; Chen et al., 2020) directly parameterized temporal point processes (*e.g.*, Hawkes process) that intrinsically assume the temporal decay with exponential intensity kernels. More recently, Neural CDE (Kidger et al., 2020) applied the cubic spline that combines information of temporally adjacent observations to enhance the representational power of the law data. To the best of our knowledge, we propose the first DE-based framework that fully utilizes the temporal correlation without making any inductive bias on the temporal states.

5. Experiments

Benchmarks. We compared our proposed method against a broad range of DE-based continuous dynamical models. The dynamic models include GRU-D (Che et al., 2018), Latent ODE (Rubanova et al., 2019), Latent SDE (Li et al., 2020), NJ-ODE (Herrera et al., 2021), Res-Flow (Biloš et al., 2021), GRU-Flow (Biloš et al., 2021), CRU (Schirmer et al., 2022), Neural Laplace (Holt et al., 2022), and CSDE-TP (Park et al., 2022). For the multivariate Transformer-based methods, we compare with vanilla Transformer (Vaswani et al., 2017), Reformer (Kitaev et al., 2020), Informer (Zhou et al., 2021) and Autoformer (Wu et al., 2021) in Section A.10. We implemented the benchmarks using open-source codes published by authors except for Latent SDE whose decoder architecture was replaced from Neural ODE to SDE (Li et al., 2020). Further details on hyper-parameters and implementation can be found in Appendix A.7. Our code is available at https://github.com/LGAI-AML/MaSDEs.

5.1. Synthetic Dataset

In the first experiment, we consider synthetic dynamics based on Mackey–Glass DDE (Mackey & Glass, 1977) by simulating deterministic trajectories of *delayed differential equations* (DDEs) defined as follows:

Method	RMSE ↓	MSE ↓	NLL ↑
CSDE-TP	0.759	0.576	−2.513
Latent ODE†	0.385	0.181	−0.537
Latent SDE	0.366	0.134	−0.301
Coupling-Flow†	0.539	0.531	−2.283
Res-Flow†	0.350	0.174	−0.505
Neural Laplace†	0.282	0.128	−0.275
CRU	0.272	0.074	−0.251
Ours	**0.235**	**0.055**	**0.091**

Table 2: Mackey-Glass DDE. † results from (Holt et al., 2022).

$$\textbf{Mackey-Glass DDE:} \quad \frac{dy_t}{dt} = \frac{\beta y_{(t-\tau)}}{1 + y_{(t-\tau)}^n} - \gamma y_t, \quad (20)$$

where we set model design factors as $\tau = 10, n = 10, \beta = 0.25, \gamma = 0.1$. As one might notice in the qualitative result in Figure 2, the simulated trajectory is highly sensitive to the initial condition as it influences the future values in a time-delayed manner. To make an accurate prediction, the prediction models need to capture the long-term dependency of temporal states given past observations.

Result. Table 2 shows comparison results of our method to the benchmarks using the same experimental setting in (Holt et al., 2022). The result shows that our method outperforms all the benchmarks and effectively captures the delayed effect. It is worth highlighting that CSDE-TP failed to capture the temporal relevance and showed the worst performance since it is incapable of information balancing.

5.2. Real-world Dataset

Experimental Settings. We evaluated the time-series prediction performance of ours and the benchmarks on multiple real-world datasets: BAQD (Zhang et al., 2017), Speech (Warden, 2018), and Physionet (Silva et al., 2012). More detailed descriptions of these datasets are provided in Appendix A.7.2. We split each time-series in the interval $[0, T]$ into two sub-intervals: the first 80% as the observation interval, *i.e.*, $\mathbb{O} = [0, 0.8T]$, and the remaining 20%

as the prediction interval, *i.e.*, $\mathbb{T} = [0.8T, T]$. We split time-series samples into two halves as training/evaluation sets in Physionet. For BAQD and Speech datasets, we divided time-series samples into 80/20 training/testing splits for training and evaluation, respectively. For fair comparisons, we normalized data features by utilizing a min-max normalization to ensure all data features lie within a unit cube $[0, 1]^d$. We used averaged mean square errors (MSEs) and negative log-likelihoods (NLLs) as the performance metrics. To evaluate the performance of the benchmarks, we followed an identical protocol suggested by (Rubanova et al., 2019).

Result. Table 1 shows the performance comparison of our method and the benchmarks. As can be seen in the table, our model significantly outperforms all the benchmarks for the evaluated datasets by a large margin. Notably, most of the existing methods based on latent encoding fail to make accurate predictions. In prior works, the past observations are indiscriminately encoded in the latent space without awareness of the temporal uncertainty of irregular time-series. Therefore, the model is vulnerable to the temporally noisy environment during testing, which may cause a large generalization gap. In contrast, neural agents in our method actively make cooperative decisions to calibrate the vicious effect of unseen data. If the agent detects harmful signals, they voluntarily restrain the decision to follow the group rationality (*i.e.*, accurate prediction). Eventually, the cooperative group robustly suppresses the non-informative temporal dynamics during inference.

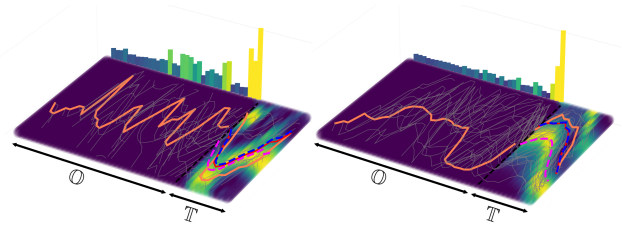


Figure 3: Qualitative result on BAQD. Sampled trajectories excluding agents α^1 (blue) and α^{35} (magenta), respectively. The prediction trajectory with full information is highlighted as (red).

Qualitative Analysis. Figure 3 shows that neural agents under the equilibrium can capture *temporal decay* to optimally extract underlying latent information from the dataset. We can observe this phenomenon at the prediction times proximate to the beginning of the prediction interval where the rational group assigns high energies, whereas this becomes less distinguishable when predicting the distant future. This is remarkable since the cooperative agents learn the temporal decay (*i.e.*, $p(y_{t\in\mathbb{T}}|y_{s\in\mathbb{O}}) \propto e^{-|t-s|}$) *a posteriori* without any prior knowledge. Furthermore, the performance is drastically worsened after excluding the agent α^*_{35} (magenta) in the decision-making whereas no meaningful performance drop can be observed after excluding agent α^*_1 (blue), clearly showing that the decision $X_t^{1,\boldsymbol{\alpha}^*}$ is less informative than others.

Method	$\varsigma = 0.1$	$\varsigma = 0.4$	$\varsigma = 0.7$	$\varsigma = 1.0$
Latent ODE	0.395	0.536	0.969	2.065
Neural Laplace	0.369	0.407	0.458	0.549
CRU	0.370	0.411	0.430	0.448
CSDE-TP	0.359	0.467	0.627	0.849
Ours	**0.314**	**0.330**	**0.336**	**0.340**

Table 3: Robustness to Out-of-distribution signals on BAQD. We evaluate the performance by using MSE ($\times 10^{-2}$).

Robustness to Non-informative Signals. To show the robustness of our method to the non-informative signals, we conduct an out-of-distribution (OOD) experiment on BAQD, which is suggested in (Deng et al., 2021). During the test time, we generate a set of random times from a Poisson process $\{\tau\} \sim \texttt{Pois}(\lambda) \in \mathbb{O}$, and inject white noises at these sampled times with intensity level λ. Table 3 highlights that the existing methods lose the expressivity due to increased noise levels while our method maintains superior performance even under high noise levels ς. The reason for the performance gap can be explicated by the theoretical result in Proposition 3.4, as such the proposed MaSDEs under Nash equilibrium capitalize on the variance-canceling property. When OOD signals are detected by neural agents, the learned temporal aggregation filters out the variance effect in a way that the inequality (19) holds. None of the existing methods are capable of handling non-informative signals since there is no sophisticated mechanism to calibrate the impact of those non-informative observations.

6. Conclusion and Future Work

In this paper, we proposed a novel framework for time-series prediction as an application of cooperative differential games. The new formulation is built upon the multi-agent dynamics called MaSDEs where each individual agent encodes the partial information from past observations. Under the shared goal, neural agents collaborate to achieve the Nash equilibrium and make accurate future predictions. Theoretical analysis shows the convergence of the proposed novel fictitious play highlighting the effectiveness of the proposed differential game.

Future Work. Beyond the basic form of objective functional, the behavioral rule for neural agents can be delicately redesigned for a specified goal of interest. For example, one may extend the proposed framework to the cooperative *mean-field game* (Carmona & Delarue, 2018) where infinitely many agents are considered. In this direction, the system can provide mathematical formalism to understand the individual influences of infinitely many observations.

We hope our work broadens our understanding of the long-term dependency problem for modern recurrent networks via the existing mean-field analysis of neural networks (Perrin et al., 2020; Guo et al., 2019; Min & Hu, 2021).

Acknowledgement

We would like to thank the anonymous reviewers for their constructive suggestions and comments. CL was supported by the IITP grant funded by the Korea government (MSIT) (No. 2021-0-01341, AI Graduate School Program, CAU).

References

Bacry, E., Mastromatteo, I., and Muzy, J.-F. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1 (01):1550005, 2015.

Biloš, M., Sommer, J., Rangapuram, S. S., Januschowski, T., and Günnemann, S. Neural flows: Efficient alternative to neural ODEs. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021.

Cardaliaguet, P. and Hadikhanloo, S. Learning in mean field games: the fictitious play. *ESAIM: Control, Optimisation and Calculus of Variations*, 2017.

Carmona, R. *Lectures on BSDEs, Stochastic Control, and Stochastic Differential Games with Financial Applications*. Society for Industrial and Applied Mathematics, 2016.

Carmona, R. and Delarue, F. *Probabilistic Theory of Mean Field Games with Applications I: Mean Field FBSDEs, Control, and Games*. Probability Theory and Stochastic Modelling. Springer International Publishing, 2018. ISBN 9783319589206.

Che, Z., Purushotham, S., Cho, K., Sontag, D., and Liu, Y. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 2018.

Chen, R. T., Behrmann, J., Duvenaud, D. K., and Jacobsen, J.-H. Residual flows for invertible generative modeling. *NeurIPS*, 2019.

Chen, R. T., Amos, B., and Nickel, M. Neural spatio-temporal point processes. In *International Conference on Learning Representations*, 2020.

Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. Neural ordinary differential equations. In *NeurIPS*, 2018.

Daskalakis, C., Goldberg, P. W., and Papadimitriou, C. H. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009.

Deng, R., Brubaker, M. A., Mori, G., and Lehrmann, A. Continuous latent process flows. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Dupont, E., Doucet, A., and Teh, Y. W. Augmented neural odes. *NeurIPS*, 2019.

Foerster, J., Assael, I. A., De Freitas, N., and Whiteson, S. Learning to communicate with deep multi-agent reinforcement learning. *Advances in neural information processing systems*, 2016.

Fournier, N. and Guillin, A. On the rate of convergence in wasserstein distance of the empirical measure. *Probability Theory and Related Fields*, 162(3):707–738, 2015.

Gardiner, C. *Stochastic methods*, volume 4. Springer Berlin, 2009.

Goldberg, P. W. A survey of ppad-completeness for computing nash equilibria. *Cambridge University Press*, pp. 51, 2011.

Guo, X., Hu, A., Xu, R., and Zhang, J. Learning mean-field games. *Advances in Neural Information Processing Systems*, 32, 2019.

Han, J. and Hu, R. Deep fictitious play for finding Markovian Nash equilibrium in multi-agent games. In *Proceedings of The First Mathematical and Scientific Machine Learning Conference*, 2020.

Herrera, C., Krach, F., and Teichmann, J. Neural jump ordinary differential equations: Consistent continuous-time prediction and filtering. In *International Conference on Learning Representations*, 2021.

Holt, S. I., Qian, Z., and van der Schaar, M. Neural laplace: Learning diverse classes of differential equations in the laplace domain. In *International Conference on Machine Learning*, pp. 8811–8832. PMLR, 2022.

Huang, X. and Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017.

Jacod, J. and Shiryaev, A. *Limit theorems for stochastic processes*, volume 288. Springer Science & Business Media, 2013.

Jørgensen, S., Martín-Herrán, G., and Zaccour, G. Dynamic games in the economics and management of pollution. *Environmental Modeling Assessment*, 15:433–467, 12 2010. doi: 10.1007/s10666-010-9221-7.

Kidger, P., Morrill, J., Foster, J., and Lyons, T. Neural controlled differential equations for irregular time series. In *NeurIPS*, 2020.

Kitaev, N., Kaiser, L., and Levskaya, A. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020.

Leitmann, G. *Cooperative and non-cooperative many players differential games*. Springer, 1974.

Li, X., Wong, T.-K. L., Chen, R. T., and Duvenaud, D. Scalable gradients for stochastic differential equations. In *AISTATS*, 2020.

Mackey, M. C. and Glass, L. Oscillation and chaos in physiological control systems. *Science*, 197(4300):287–289, 1977. doi: 10.1126/science.267326.

Mei, H. and Eisner, J. M. The neural hawkes process: A neurally self-modulating multivariate point process. *Advances in neural information processing systems*, 2017.

Min, M. and Hu, R. Signatured deep fictitious play for mean field games with common noise. In *International Conference on Machine Learning*, pp. 7736–7747. PMLR, 2021.

Morrill, J., Salvi, C., Kidger, P., Foster, J., and Lyons, T. Neural rough differential equations for long time series. *ICML*, 2021.

Nouiehed, M., Sanjabi, M., Huang, T., Lee, J. D., and Razaviyayn, M. Solving a class of non-convex min-max games using iterative first order methods. In *Advances in Neural Information Processing Systems*, 2019.

Ogata, Y. Space-time point-process models for earthquake occurrences. *Annals of the Institute of Statistical Mathematics*, 50(2):379–402, 1998.

Park, S. W., Lee, K., and Kwon, J. Neural markov controlled SDE: Stochastic optimization for continuous-time data. In *International Conference on Learning Representations*, 2022.

Perelman, A., Shima, T., and Rusnak, I. Cooperative differential games strategies for active aircraft protection from a homing missile. *Journal of Guidance, Control, and Dynamics*, 34(3):761–773, 2011.

Perrin, S., Pérolat, J., Laurière, M., Geist, M., Elie, R., and Pietquin, O. Fictitious play for mean field games: Continuous time analysis and applications. *Advances in Neural Information Processing Systems*, 33:13199–13213, 2020.

Pham, H. Feynman-kac representation of fully nonlinear pdes and applications. *Acta Mathematica Vietnamica*, 2015.

Rubanova, Y., Chen, R. T., and Duvenaud, D. K. Latent ordinary differential equations for irregularly-sampled time series. In *NeurIPS*, 2019.

Schirmer, M., Eltayeb, M., Lessmann, S., and Rudolph, M. Modeling irregular time series with continuous recurrent units. In *International Conference on Machine Learning (ICML)*, 2022.

Sexton, R. J. The formation of cooperatives: a game-theoretic approach with implications for cooperative finance, decision making, and stability. *American Journal of Agricultural Economics*, 68(2):214–225, 1986.

Silva, I., Moody, G., Scott, D., Celi, L., and Mark, R. Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012. *Computing in cardiology*, 2012.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Staatz, J. M. The cooperative as a coalition: a game-theoretic approach. *American Journal of Agricultural Economics*, 65(5):1084–1089, 1983.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017.

Warden, P. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*, 2018.

Wu, H., Xu, J., Wang, J., and Long, M. Autoformer: Decomposition transformers with Auto-Correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, 2021.

Ye, M., Hu, G., and Lewis, F. L. Nash equilibrium seeking for n-coalition noncooperative games. *Automatica*, 95: 266–272, 2018.

Yeung, D. W. and Petrosjan, L. A. *Cooperative stochastic differential games*. Springer Science & Business Media, 2006.

Yeung, D. W. and Petrosyan, L. A. *Subgame consistent economic optimization: an advanced cooperative dynamic game analysis*. Springer, 2012.

Zhang, S., Guo, B., Dong, A., He, J., Xu, Z., and Chen, S. Cautionary tales on air-quality improvement in beijing. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 473:20170457, 09 2017.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Virtual Conference*, 2021.

Zuo, S., Jiang, H., Li, Z., Zhao, T., and Zha, H. Transformer hawkes process. In *International conference on machine learning*. PMLR, 2020.

A. Appendix

A.1. Details on Forward-backward Stochastic Differential Equations

In this section, we briefly recall some basic elements of stochastic optimal controls needed in the paper. We start by introducing the *backward stochastic differential equation* used in Section 3. In particular, the adjoint variables $(Y_t^{i,\boldsymbol{\alpha}}, Z_t^{i,\boldsymbol{\alpha}})$ are derived from the classical theory of non-linear Feynman-Kac formula (Carmona & Delarue, 2018):

$$dY_t^{i,\boldsymbol{\alpha}} = -H^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, F_t^{i,\boldsymbol{\alpha}}, [\alpha_i, \boldsymbol{\alpha}_{(-i)}])dt + Z_t^{i,\boldsymbol{\alpha}} \cdot dB_t^i, \tag{A.21}$$

where the stochastic Hamiltonian system with uncontrolled volatility for i-th agent is defined as follows:

$$H^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, F_t^{i,\boldsymbol{\alpha}}, [\alpha_i, \boldsymbol{\alpha}_{(-i)}]) = \left[\sigma(t, X_t^{i,\boldsymbol{\alpha}})^{-T} Z_t^{i,\boldsymbol{\alpha}}\right] \cdot b(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i) + h^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, \boldsymbol{\alpha}). \tag{A.22}$$

For simplicity, we define the auxiliary function F having following form:

$$F^{i,\boldsymbol{\alpha}}(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i) \coloneqq \sigma(t, X_t^{i,\boldsymbol{\alpha}})^{-T} b(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i). \tag{A.23}$$

Next, we introduce the diriftless stochastic differential equation:

$$dX_t^{i,\boldsymbol{\alpha}} = \sigma(t, X_t^{i,\boldsymbol{\alpha}})dB_t^i. \tag{A.24}$$

Then, the Girsanov's theorem gives the following Radon-Nikodym derivative:

$$\frac{d\mathbb{P}}{d\mathbb{Q}} = \varepsilon\left(\int \sigma(t, X_t^{i,\boldsymbol{\alpha}})^{-1} b(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i) \cdot dB_t^i\right)_T, \tag{A.25}$$

where we denote $M^{-T} \coloneqq [M^{-1}]^T$ for any inevitable matrix M. Note that the regularity conditions **(H1)** are essential to ensure the existence of the Doléans-Dade exponential ε which defines the stochastic exponential for the (local) martingale $dM_t^i = F^{i,\boldsymbol{\alpha}}(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i) \cdot dB_t^i$:

$$\varepsilon(M^i)_T \coloneqq e^{M_t^i - M_0^i - \frac{1}{2}[M^i, M^i]_t} = e^{\int_0^t F^{i,\boldsymbol{\alpha}}(u, X_u^{i,\boldsymbol{\alpha}}, \alpha_i) \cdot dB_u^i - \frac{1}{2}\int_0^t |F^{i,\boldsymbol{\alpha}}|^2(u, X_u^{i,\boldsymbol{\alpha}}, \alpha_i)du}. \tag{A.26}$$

The second equality holds since $M_0^i = 0$ and the quadratic variation is calculated as $[M^i, M^i]_t = \int_0^t |F^{i,\boldsymbol{\alpha}}|^2 ds < \infty$. Then, by definition, the transformed representation W_t^i in following equation is also a Wiener process with respect to the probability measure $\mathbb{P}$.

$$W_t^i = B_t^i - \int_0^t \sigma(s, X_s^{i,\boldsymbol{\alpha}})^{-T} b(s, X_s^{i,\boldsymbol{\alpha}}, \alpha_i)ds = B_t^i - \int_0^t F^{i,\boldsymbol{\alpha}}(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i)ds. \tag{A.27}$$

Identically, the above relation can be rewritten as following differential form:

$$dW_t^i = -\left[F^{i,\boldsymbol{\alpha}}(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i)dt - dB_t^i\right]. \tag{A.28}$$

Notably, we can restore the proposed controlled neural SDE from driftless SDE in (A.24) by the relation between W_t^i and B_t^i shown in (A.28).

$$dX_t^{i,\boldsymbol{\alpha}} = b(t, X_t^{i,\boldsymbol{\alpha}}, \alpha_i)dt + \sigma(t, X_t^{i,\boldsymbol{\alpha}})dW_t^i. \tag{A.29}$$

This relation gives the explicit form of adjoint dynamics as follows: $Y_t^{i,\boldsymbol{\alpha}}$:

$$\begin{aligned}
Y_t^{i,\boldsymbol{\alpha}} &= \Psi^i(X_T^{i,\boldsymbol{\alpha}}) + \int_t^T H^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, F_t^{i,\boldsymbol{\alpha}}, [\alpha_i, \boldsymbol{\alpha}_{(-i)}])ds - \int_t^T Z_s^{i,\boldsymbol{\alpha}} \cdot dB_s^i \\
&= \Psi^i(X_T^{i,\boldsymbol{\alpha}}) + \int_t^T h^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, \boldsymbol{\alpha}) + \left(\sigma(s, X_s^{i,\boldsymbol{\alpha}})^{-T} Z_s^{i,\boldsymbol{\alpha}}\right) \cdot b(s, X_s^{i,\boldsymbol{\alpha}}, \alpha_i)\Big) ds - \int_t^T Z_s^{i,\boldsymbol{\alpha}} \cdot dB_s^i \\
&= \Psi^i(X_T^{i,\boldsymbol{\alpha}}) + \int_t^T h^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, \boldsymbol{\alpha})ds + \int_t^T Z_s^{i,\boldsymbol{\alpha}} \cdot \left[F^{i,\boldsymbol{\alpha}}(s, X_s^{i,\boldsymbol{\alpha}}, \alpha_i)ds - dB_s^i\right] \\
&= \Psi^i(X_T^{i,\boldsymbol{\alpha}}) + \int_t^T h^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, \boldsymbol{\alpha})ds - \int_t^T Z_s^{i,\boldsymbol{\alpha}} \cdot dW_s^i.
\end{aligned} \tag{A.30}$$

In the last equality, the Brownian motion is changed from B_s^i to W_s^i, and the generic form of BSDE for tuple $(\mathbb{P}, W_t^i)$ is presented. Since the third term in last line $\int Z_s^{i,\boldsymbol{\alpha}} \cdot dW_s^i$ is a martingale with respect to the measure $\mathbb{P}$, we can identify the conditional expectation with original objective functional J^i in (4) as follows:

$$\mathcal{J}^i = \mathbb{E}[Y_t^{i,\boldsymbol{\alpha}} | \tilde{\mathcal{F}}_t] = \mathbb{E}_{\mathbb{P}}\left[\Psi^i(X_T^{i,\boldsymbol{\alpha}}) + \int_t^T h^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, \boldsymbol{\alpha}) ds | \mathcal{F}_t\right], \tag{A.31}$$

where $\tilde{\mathcal{F}}_t$ is the augmented filtration according to the Brownian motion B_t^i.

A.2. Multi-conditioned Score-based Predictor

The aim of this section is to provide detailed information of the proposed Multi-conditioned Score-based Predictor. We start by introducing a well-known transformation of a generic SDE called *Liouville equation* (Gardiner, 2009):

$$\begin{aligned} dX_t^{i,\boldsymbol{\alpha},k} &= b(t, X_t^{i,\boldsymbol{\alpha},k}, \alpha_i)dt + \sigma(t, X_t^{i,\boldsymbol{\alpha},k})dW_t^i \\ &= \left[b(t, X_t^{i,\boldsymbol{\alpha},k}, \alpha_i) - \frac{1}{2}\nabla \cdot \Sigma(t, X_t^{i,\boldsymbol{\alpha},k}) - \frac{1}{2}\Sigma(t, X_t^{i,\boldsymbol{\alpha},k})\nabla_x \log p_t^i(X_t^{i,\boldsymbol{\alpha},k} | y_{s_i}^k)\right] dt \end{aligned} \tag{A.32}$$

In the second line, one needs to evaluate the score function $\nabla \log p_t^i$, which involves the transformation of the proposed MaSDEs for the deterministic representation. Since the proposed framework applies the time discretization ($i.e.$, $t_{m+1} = t_m + \delta t$) for simulating MaSDEs, the score function started from i.i.d initial state $y_{s_i}^k \sim p_{s_i}$ can be approximated as

$$\begin{aligned} \nabla \log p_t(X_t^{i,\boldsymbol{\alpha},k} | y_{s_i}^k) &= \sum_{m=2}^{M} \nabla \log p_t^i(X_{t_m}^{i,\boldsymbol{\alpha},k} | X_{t_{m-1}}^{i,\boldsymbol{\alpha},k}) + \nabla \log p_t^i(X_{t_1}^{i,\boldsymbol{\alpha},k} | y_{s_i}^k) \\ &= \sum_{m=2}^{M} -(\delta t)^{-1}\Sigma^{-1}(t_{m-1}, X_{t_{m-1}}^{i,\boldsymbol{\alpha},k})\left[X_{t_m}^{i,\boldsymbol{\alpha},k} - X_{t_{m-1}}^{\boldsymbol{\alpha},k} - (\delta t)b(t_{m-1}, X_{t_{m-1}}^{i,\boldsymbol{\alpha},k}, \alpha_i)\right] \\ &\quad - (\delta t)^{-1}\Sigma^{-1}(s_i, y_{s_i}^k)\left[X_{t_1}^{i,\boldsymbol{\alpha},k} - y_{s_i}^k - (\delta t)b(s_i, y_{s_i}^k, \alpha_i)\right]. \end{aligned} \tag{A.33}$$

From the evaluation in (A.33) and the ODE representation in (A.32), we have

$$\begin{aligned} dX_t^{i,\boldsymbol{\alpha},k} = \Bigg[& b(t, X_t^{i,\boldsymbol{\alpha},k}, \alpha_i) - \frac{1}{2}\nabla \cdot \Sigma(t, X_t^{i,\boldsymbol{\alpha},k}) \\ & + \frac{1}{2}\Sigma(t, X_t^{i,\boldsymbol{\alpha},k}) \sum_{m=2}^{M} (\delta t)^{-1}\Sigma^{-1}(t_{m-1}, X_{t_{m-1}}^{i,\boldsymbol{\alpha},k})\left[X_{t_m}^{i,\boldsymbol{\alpha},k} - X_{t_{m-1}}^{i,\boldsymbol{\alpha},k} - (\delta t)b(t_{m-1}, X_{t_{m-1}}^{i,\boldsymbol{\alpha},k}, \alpha_i)\right] \\ & + \frac{1}{2}\Sigma(t, X_t^{i,\boldsymbol{\alpha},k})(\delta t)^{-1}\Sigma^{-1}(s_i, y_{s_i}^k)\left[X_{t_1}^{i,\boldsymbol{\alpha},k} - y_{s_i}^k - (\delta t)b(s_i, y_{s_i}^k, \alpha_i)\right]\Bigg] dt. \end{aligned} \tag{A.34}$$

For convenience, we additionally define the ratio between diffusion covariance at time t_m and t as follows:

$$\hat{\Sigma}_m = \Sigma(t, X_t^{i,\boldsymbol{\alpha}})\Sigma^{-1}(t_m, X_{t_m}^{i,\boldsymbol{\alpha}}) \tag{A.35}$$

Finally, the proposed predictor $\Lambda_t^{\boldsymbol{\alpha}}$ is given as

$$\begin{aligned} d\Lambda_t^{\boldsymbol{\alpha},k} = \sum_{i \in \mathbb{N}} A_i^{\alpha_i}(t, X_t^{i,\boldsymbol{\alpha},k}) \Bigg[& b(t, X_t^{i,\boldsymbol{\alpha},k}, \alpha_i) - \frac{1}{2}\nabla \cdot \Sigma(t, X_t^{i,\boldsymbol{\alpha},k}) \\ & + \frac{1}{2}(\delta t)^{-1}\Bigg\{ \sum_{m=2}^{M} \hat{\Sigma}_{m-1}\left[X_{t_m}^{i,\boldsymbol{\alpha},k} - X_{t_{m-1}}^{i,\boldsymbol{\alpha},k} - (\delta t)b(t_{m-1}, X_{t_{m-1}}^{i,\boldsymbol{\alpha},k}, \alpha_i)\right] \\ & + \hat{\Sigma}_0\left[X_{t_1}^{i,\boldsymbol{\alpha},k} - y_{s_i}^k - (\delta t)b(s_i, y_{s_i}^k, \alpha_i)\right]\Bigg\} dt\Bigg] \end{aligned} \tag{A.36}$$

A.3. Proofs

Assumptions. Throughout the appendix, we make the following assumptions for the proof.

(H1) $\sigma^{-T}(t, x, \alpha(\cdot, \cdot; \theta))b(t, x, \alpha(\cdot, \cdot; \theta))$ is uniformly bounded, twice differentiable, Lipschitz continuous.

(H2) The k-th derivatives of neural agents with respect to both spatial and temporal variables are Lipschitz continuous to the parameter space,

$$||\nabla^{(k)}\alpha_i(\cdot, x; \theta) - \nabla^{(k)}\alpha_i(\cdot, x; \tilde{\theta})|| + ||\partial_t^{(k)}\alpha_i(t, \cdot; \theta) - \partial_t^{(k)}\alpha_i(t, \cdot; \tilde{\theta})|| \leq L_i||\theta - \tilde{\theta}||. \tag{A.37}$$

for all $i \in \{1, \dots, I\}$, $0 \leq k \leq 2$, $\forall \theta, \tilde{\theta} \in \Theta$.

(H3) The expectation of Frobenius norm foe every adjoint variables $Z_t \in \mathbf{Sym}_+^I$ [3] is bounded, $\mathbb{E}\,\|Z_t\|_F < D_2$.

(H4) Let $\{\lambda_i^\varepsilon(t)\}_{\varepsilon \in \{1, \cdots, d\}}$ be the real-valued positive spectrum of diffusion function $\Sigma_t := \sigma\sigma^T \in \mathbf{Sym}_+^I$. Then, we assume that there exists a constant D_3 such that the following inequality holds

$$\left(\frac{\lambda_i^\varepsilon(t)}{\lambda_j^\varepsilon(t)} \vee \frac{\lambda_i^\varepsilon(t)}{\lambda_i^\varepsilon(s)}\right) \leq D_3, \quad a.s., \mathbb{P}, \quad \forall s < t,\ \varepsilon \neq \varepsilon',\ i \neq j \tag{A.38}$$

(H5) An empirical measure of perturbed representation $\hat{\Lambda}_t$ has a finite second moment, and every instance of the dataset lying in the compact subset of $\mathbb{R}^d$. Finally, we assume that the dimensionality of the dataset satisfies *i.e.*, $d/2 \gg 2$.

Lemma A.1. *(Grönwall's Inequality (Jacod & Shiryaev, 2013)) The left inequality induces the inequality in right-hand side:*

$$B(t) \leq A + \int_{\mathbb{T}} B(s)C(s)ds \quad \longrightarrow \quad B(t) \leq Ae^{\int_{\mathbb{T}} C(s)ds}. \tag{A.39}$$

A.3.1. Proof of Proposition 3.1

The first step to showing the convergence of neural agents towards Nash equilibrium is to obtain the deviation of the adjoint variable Y_t during the fictitious play. For this, we first introduce the HJBE of decoupled SDE system. In particular, the proposed system trains neural agents to solve the individual decoupled HJBE at each stage m:

$$\mathcal{V}_t^{i,m+1} + \inf_{\alpha_i^m \in \mathbb{A}} H^i(t, x, F_t^{i,\boldsymbol{\alpha}^m}, [\alpha_i^m, \boldsymbol{\alpha}_{(-i)}^m]) + \frac{1}{2}\mathbf{Tr}(\Sigma\nabla^2\mathcal{V}^{i,m+1}) = 0, \tag{A.40}$$

where the cost functional at the next stage (*i.e.*, $m+1$) is related to the optimal actions in the previous stage that minimizes the decoupled Hamiltonian in (A.40). Then, one of our interests is to investigate the deviation of the Hamiltonian system:

$$\delta H_t^{i,m} = \underbrace{H^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}^m}, F_t^{i,\boldsymbol{\alpha}^m}, [\alpha_i^{m+1}, \boldsymbol{\alpha}_{(-i)}^m)]}_{\text{Fictitious Play}} - \underbrace{H^i(t, \mathbf{X}_t^{\alpha^*}, F_t^{i,\boldsymbol{\alpha}^*}, \boldsymbol{\alpha}^*)}_{\text{Optimal Hamiltonian}}, \tag{A.41}$$

where the deviation shows the difference between Hamiltonian systems that are derived by neural agents lying in optimal and sub-optimal regions, respectively. Similarly, we define the deviations for both adjoint variables:

$$\delta Y_t^{i,m} = Y_t^{i,m} - Y_t^{i,*}, \quad \delta Z_t^{i,m} = Z_t^{i,m} - Z_t^{i,*}, \tag{A.42}$$

Following by the notations in (A.42) and the equality (A.30), the deviation for the adjoint variable (*i.e.*, $\delta Y_t^{i,m}$) at stage m can be represented as the following Itô's differential:

$$d\delta Y_t^{i,m+1} = -\delta H^{i,m}dt + \delta Z_t^{i,m+1} \cdot dB_t^i. \tag{A.43}$$

[3] The matrix manifolds $\mathbf{Sym}_+^I$ is the space of semi-positive definite matrices. In this paper, we regard the Euclidean flat norm (*i.e.*, $\|\cdot\|_F$) is inherited to this space.

Next, we evaluate the squared norm of $\delta \mathbf{Y}_t^{(\cdot)}$ to investigate the convergence of objective functional

$$d\left\|\delta \mathbf{Y}_t^{m+1}\right\|^2 = -2\delta \mathbf{Y}_t^{m+1} \cdot \delta \mathbf{H}_t^m dt + \left\|\delta \mathbf{Z}_t^{m+1}\right\|_F^2 dt + 2(\delta \mathbf{Z}_t^{m+1} \delta \mathbf{Y}_t^{m+1}) \cdot d\mathbf{B}_t. \tag{A.44}$$

Note that the equality in (A.44) follows by Itô's lemma for squared norm of multi-dimensional representation A_t:

$$d\left\|A_t\right\|^2 = 2A_t \cdot \mu_{A_t} dt + \left\|\sigma_{A_t}\right\|_F^2 dt + 2A_t \sigma_{A_t} \cdot dB_t, \tag{A.45}$$

where μ_{A_t} and σ_{A_t} are the terms corresponding to bounded variations and local martingales of the process A_t, respectively. By taking the expectation on both sides of (A.44), we obtain the following result:

$$\begin{aligned}\mathbb{E}\left[\left\|\delta \mathbf{Y}_t^{m+1}\right\|^2\right] &= \mathbb{E}\left[\left\|\delta \mathbf{Y}_T^{m+1}\right\|^2\right] + \mathbb{E}\left[\int_{\mathbb{T}} 2\delta \mathbf{Y}_t^{m+1} \cdot \delta \mathbf{H}_t^m - \left\|\delta \mathbf{Z}_t^{m+1}\right\|_F^2 dt\right] \\ &\leq 4\mathbb{E}\left[\left\|\delta \mathbf{X}_T^{m+1}\right\|^2\right] + 2\mathbb{E}\left[\int_{\mathbb{T}} \left\|\delta \mathbf{Y}_t^{m+1}\right\|^2 \left\|\delta \mathbf{H}_t^m\right\|^2 dt\right] - \mathbb{E}\left[\int_{\mathbb{T}} \left\|\delta \mathbf{Z}_t^{m+1}\right\|_F^2 dt\right].\end{aligned} \tag{A.46}$$

Owing to the characteristic of backward SDE (*i.e.*, $\mathbf{Y}_t^m$), the integral sign is reversed in the second term in (A.46). The expectation with respect to the local martingale term vanishes as set of Wiener processes $\mathbf{B}_t$ are related to $\mathbb{Q}$. Since BSDE imposes the terminal constraint $\mathbf{Y}_T^m = \mathbf{\Psi}^m$, $\mathbf{Y}_T^* = \mathbf{\Psi}^*$, the following result is given by Lipschitzness of $\mathbf{\Psi}$ as:

$$\mathbb{E}\left[\left\|\delta \mathbf{Y}_T^{m+1}\right\|^2\right] = \mathbb{E}\left[\left\|\mathbf{\Psi}^{m+1} - \mathbf{\Psi}^*\right\|^2\right] \leq 4\mathbb{E}\left[\left\|\delta \mathbf{X}_T^{m+1}\right\|^2\right]. \tag{A.47}$$

This shows the inequality in (A.46). By rearranging the relation and applying Hölder's inequality we have

$$\mathbb{E}_{\mathbb{Q}}\left[\left\|\delta \mathbf{Y}_t^{m+1}\right\|^2 + \left\|\delta \mathbf{Z}_t^{m+1}\right\|_F^2\right] \leq 4\mathbb{E}_{\mathbb{Q}}\left[\left\|\delta \mathbf{X}_T^{m+1}\right\|^2\right] + 2\mathbb{E}_{\mathbb{Q}}\left[\int_{\mathbb{T}} \left\|\delta \mathbf{Y}_t^{m+1}\right\|^2 dt\right] \mathbb{E}_{\mathbb{Q}}\left[\int_{\mathbb{T}} \left\|\delta \mathbf{H}_t^m\right\|^2 dt\right]. \tag{A.48}$$

Finally, we apply Grönwall's inequality in Lemma A.1 to above inequality, we obtain

$$\mathbb{E}_{\mathbb{Q}}\left[\left\|\delta \mathbf{Y}_t^{m+1}\right\|^2\right] \leq \mathbb{E}_{\mathbb{Q}}\left[\left\|\delta \mathbf{X}_T^{m+1}\right\|^2\right] e^{|\mathbb{T}| \ln 16 \mathbb{E}\left[\int_{\mathbb{T}} \|\delta \mathbf{H}_t^m\|^2 dt\right]}. \tag{A.49}$$

Now, our next step is to bound the right-hand side of (A.49). We start by estimating an upper bound of the following time averaged mean-squared Hamiltonian (A.21):

$$\mathbb{E}\left[\int_{\mathbb{T}} \left\|\delta \mathbf{H}_t^m\right\|^2 dt\right] \leq \mathbb{E}_{\mathbb{P}}\left[\int_{\mathbb{T}} \left\|\delta \mathbf{F}_t^m \cdot \delta \mathbf{Z}_t^{m+1} + \delta \mathbf{h}_t^m\right\|^2 dt\right]. \tag{A.50}$$

For this, we rearrange the Hamiltonian deviation $\delta \mathbf{H}_t^m$ by inserting two terms $\delta \mathbf{F}_t^m \cdot \delta \mathbf{Z}_t^m$ and $\delta \mathbf{h}_t^m$. In this case, the Hamiltonian deviation is rewritten as

$$\begin{aligned}\delta H_t^{i,m} = &F(t, \mathbf{X}_t^\alpha, [\alpha_i^{m+1}, \boldsymbol{\alpha}_{(-i)}^m]) \cdot (Z_t^{i,m+1} - Z_t^{i,*}) \\ &+ (F(t, \mathbf{X}_t^\alpha, [\alpha_i^{m+1}, \boldsymbol{\alpha}_{(-i)}^m]) - F(t, \mathbf{X}_t^\alpha, \boldsymbol{\alpha}^m)) \cdot Z_t^{i,*} \\ &+ (F(t, \mathbf{X}_t^\alpha, \boldsymbol{\alpha}^m) - F(t, \mathbf{X}_t^\alpha, \boldsymbol{\alpha}^*)) \cdot Z_t^{i,*} \\ &+ h^i(t, \mathbf{X}_t^\alpha, [\alpha_i^{m+1}, \boldsymbol{\alpha}_{(-i)}^m]) - h^i(t, \mathbf{X}_t, \boldsymbol{\alpha}^m) + h^i(t, \mathbf{X}_t^\alpha, \boldsymbol{\alpha}^m) - h^i(t, \mathbf{X}_t^\alpha, \boldsymbol{\alpha}^*).\end{aligned} \tag{A.51}$$

Here, we define the Lipschitz constants of three different objects as follows:

$$\mathbf{Lip}[\cdots] = C_n, \qquad \{\cdots\} \in \{F^{i,m}, \sigma^i, h^i\}, \quad n = 1, 2, 3. \tag{A.52}$$

By the Lipschitz continuity defined above (A.52), one can obtain the squared norm of the Hamiltonian deviation:

$$\begin{aligned}\left\|\delta \mathbf{H}_t^m\right\|^2 \leq &C_1 \left\|F\right\|^2 \left\|\mathbf{Z}_t^{m+1} - \mathbf{Z}_t^*\right\|_F^2 \\ &+ C_1 \left\|\alpha_i^{m+1} - \alpha_i^m\right\|^2 \cdot \left\|\mathbf{Z}_t^*\right\|_F + C_1 \sum_i^I \left\|\boldsymbol{\alpha}^m - \boldsymbol{\alpha}^*\right\|^2 \cdot \left\|Z_t^{i,*}\right\|_F \\ &+ C_3 \left\|\alpha_i^{m+1} - \alpha_i^m\right\|^2 + C_3 \left\|\boldsymbol{\alpha}^m - \boldsymbol{\alpha}^*\right\|^2.\end{aligned} \tag{A.53}$$

Consider we have a proper $M = C_1D_1D_2 \vee 2(C_1D_2 \vee C_3) \vee 2I(C_1D_2 \vee C_3)$ from the definitions of pre-determined constants C_1, C_2, C_3, D_1 and D_2. Then, the above inequality can be rewritten in a compact form:

$$\mathbb{E}\left[\|\delta \mathbf{H}_t^m\|^2\right] \leq M\left[\mathbb{E}\left\|\boldsymbol{\alpha}^{m+1} - \boldsymbol{\alpha}^m\right\|^2 + \mathbb{E}\left\|\boldsymbol{\alpha}^m - \boldsymbol{\alpha}^*\right\|^2\right]. \tag{A.54}$$

Note that the equality $\left\|[\beta_i, \boldsymbol{\alpha}^m_{(-i)}] - \boldsymbol{\alpha}^{m,*}\right\| = \left\|\beta_i - \alpha_i^{m,*}\right\|$ holds for an arbitrary action $\beta \in \mathbb{A}$. Finally, we conclude the result (A.54) by showing following inequalities:

$$\begin{cases} \mathbb{E}\left[\sum_i^I \|\delta \mathbf{Z}_t^*\|_F^2\right] \leq I \sup_i \mathbb{E}\left[\left\|\delta Z_t^{i,*}\right\|_F^2\right] \leq 2ID_2, \\ \\ \sup_{t,(\boldsymbol{\alpha}\in\mathbb{A})} \mathbb{E}\left[\|F(t,x,\boldsymbol{\alpha})\|^2\right] \leq C_1\mathbb{E}\left[\left\|F(0,0,\mathbf{0})^2\right\|\right] \leq D_1. \end{cases} \tag{A.55}$$

In previous contents, the detailed convergent states of adjoint variable Y_t^* are not specified. To continue our discussion from (A.54), we develop a gradient descent-based update rule for training neural agents. For any neural parameters $\theta \in \Theta$, we assume that the i-th agent's action $\theta_i \to \alpha_i(\cdot,\cdot;\theta_i) \triangleq \alpha_i(\theta_i)$ lies in the compact subset $A \in \mathbb{A}(\epsilon) \in \mathcal{L}^2([0,T]\times\mathbb{R}^d)$ (*i.e.*, $\int ||\alpha_i(t,x;\theta_i)||^2 dt dp_t(x) dx < \infty$) for some $\epsilon > 0$. Then, we introduce the following notations:

$$\alpha_i^{m,k} \coloneqq \alpha_i^m(t,x;\theta_i^m(k)), \quad \boldsymbol{\alpha}^{m,k} \coloneqq \{\alpha_i^{m,k}\}_{1\leq i\leq d}, \tag{A.56}$$

where the auxiliary notation $u \in \mathbb{R}^+$ is an indicator for gradient descent steps. Similarly, we define the notations $(Y^{i,m,k}, \mathbf{Y}^{m,k})$ and $(H^{i,m,k}, \mathbf{H}^{m,k})$. Let us define the operator $\mathcal{B} : \mathbb{N}^+ \to \mathcal{L}^2(\mathbb{T}\times\mathbb{R}^d)$ as $\mathcal{B}[\theta_i^m(k)] \coloneqq \alpha_i^{m,k+1}$, and if $k > K$, then we denote $\alpha_i^{m+1} \coloneqq \alpha_i^{m+1,k=0}$. By Lipschitz continuity of neural agents, we have

$$\begin{aligned} \|\mathcal{B}[\theta_i^m(k)] - \alpha_i^m\|_{\mathcal{L}^2}^2 &= \mathbb{E}\left[\|\alpha_i^m(t,x;\theta_i^m(k+1)) - \alpha_i^m(t,x;\theta_i^m(k))\|^2\right] \\ &\leq L^2(i)\mathbb{E}\left[\|\theta_i^m(k) - \theta_i^m(k+1)\|^2\right] \\ &\leq \tilde{L}^2\mathbb{E}\left[\left\|\gamma\nabla_{\theta_i}\mathbb{E}_{\mathbb{Q}}[Y_t^{i,m,k}|\tilde{F}_t]\right\|^2\right] \\ &\leq \tilde{L}^2\gamma^2\mathbb{E}\tilde{\mathbb{E}}_{\mathbb{Q}}\left[\left\|\nabla_{\theta_i}Y_t^{i,m,k}\right\|^2|\tilde{\mathcal{F}}_t\right] \\ &\leq \tilde{L}^2\gamma^2\mathbb{E}\tilde{\mathbb{E}}\left[\left\|\nabla_{\theta_i}h^{i,m,k}\right\|^2|\mathcal{F}_t\right] \\ &= \tilde{L}^2\gamma^2\mathbb{E}\left[\left\|\nabla_{\theta_i}h^{i,m,k}\right\|^2\right], \end{aligned} \tag{A.57}$$

where $\tilde{L} = \max_i L_i$. In the fourth inequality, the probability measure for the integration is switched from $\mathbb{Q}$ to $\mathbb{P}$ with the Radon-Nikodym $d\mathbb{P}/d\mathbb{Q}$. Since the cost function is uniformly bounded with the vanishing derivative of terminal cost (*i.e.*, $\partial_{\theta_i}g^i(X_T) = 0$), the expectation is well-defined. It is worth noting that the filtration $\mathcal{F}_t$ contains the information of past observations $\{y_{s_i}\}$ as we impose constraints to satisfy $X_{s_i}^{i,\alpha} = y_{s_i}$ almost surely for neural agents.

As a next step, we evaluate the upper bound of mean-squared evaluation as follows:

$$\begin{aligned} &\mathbb{E}\left[\left\|\nabla_{\theta_i}h^{i,m,k}\right\|^2\right] \\ &\leq 4\mathbb{E}\left[\left\|\langle\mathbf{A}^{\boldsymbol{\alpha}^m}(t), \mathbf{X}_t^{\boldsymbol{\alpha}^m}\rangle - y_t\right\|^2 \left\|\nabla_{\theta_i}A_i^{\alpha_i^m(\cdot,\cdot;\theta_i^m)}(t)X_t^{i,\alpha_i^m} + A_i^{\alpha_i^m}(t)\nabla_{\theta_i}X_t^{i,\alpha_i^m(\cdot,\cdot;\theta_i)}\right\|^2\right] \\ &\leq 4\mathbb{E}\left[\left\|h^{i,m,k}\right\|^2 \left\|\nabla_{\theta_i}A_i^{\alpha_i^m(\cdot,\cdot;\theta_i^m)}(t)X_t^{i,\alpha_i^m} + A_i^{\alpha_i^m}(t)\nabla_{\theta_i}X_t^{i,\alpha_i^m(\cdot,\cdot;\theta_i^m)}\right\|^2\right]. \end{aligned} \tag{A.58}$$

Let us assume that there exist representations $\hat{b}, \hat{\alpha}$ such that $b(t,X_t,\alpha_i) = \hat{b}(t,X_t)\alpha_i(t,X_t;\theta_i)$ and $A_i(t,X_t;\theta_i) = \hat{A}(t)\alpha_i(t,X_t;\theta_i)$. This brings the evaluations of the following two norm bounds:

$$\begin{aligned} \left\|\mathbb{E}\left[\nabla_{\theta_i}X_t^{i,\alpha_i^m}\right]\right\|^2 &\leq \mathbb{E}\left[\left\|\nabla_{\theta_i}X_t^{i,\alpha_i^m}\right\|^2\right] \leq \int_{s_i}^t \mathbb{E}\left[\left\|\hat{b}(u,X_u)\nabla_{\theta_i}\alpha_i^m(u,X_u;\theta_i^m)\right\|^2\right]du \\ &\leq |\mathbb{T}|\tilde{L}^2\mathbb{E}||\hat{b}||^2. \end{aligned} \tag{A.59}$$

Similarly, we have the gradient norm bound of individual temporal aggregation

$$\mathbb{E}\left[\left\|\nabla_{\theta_i} A_i^{\alpha_i^m}\right\|^2\right] = \mathbb{E}\left[\left\|\hat{A}\nabla_{\theta_i}\alpha_i^m(t, X_t; \theta_i^m)\right\|^2\right] \leq \mathbb{E}\left\|\hat{A}^2\right\| \tilde{L}^2. \tag{A.60}$$

For simplicity, suppose that there exists a constant M_2 such that each function $\hat{b}$, $\hat{A}$, $\partial_t \hat{A}$ has expectation norm bound:

$$\mathbb{E}\left[||\hat{b}||^2 + ||\hat{A}||^2 + \left\|\partial_t \hat{A}\right\|^2\right] < M_2. \tag{A.61}$$

The inequality in (A.58) together with evaluations in (A.59), (A.60) yields:

$$\mathbb{E}\left[\left\|\nabla_{\theta_i} h^{i,m,k}\right\|\right] \leq \mathbb{E}\left[\left\|h^{i,m,k}\right\|^2\right] 4\tilde{L}^2 M_2(1 + |\mathbb{T}|). \tag{A.62}$$

This directly gives the $\mathcal{L}^2$-bound of the operator $\mathbb{B}$ during the fictitious play over stages according to the cost function shown in the right-hand side:

$$\|\mathcal{B}[\theta_i^m(k)] - \alpha_i^m\|_{\mathcal{L}^2}^2 \leq \mathbb{E}\left[\left\|h^{i,m,k}\right\|^2\right] 4\tilde{L}^4\gamma^2 M_2(1 + |\mathbb{T}|). \tag{A.63}$$

Let us say that the gradient descent optimizes the neural parameters to achieve small enough values for the cost function for $k \geq K$.

$$K = \min_{k\in\mathbb{N}^+}\left\{k; \mathbb{E}_{(y_t,\Lambda_t)\sim(\nu_t^N,\mu_t^N)}\left[\left\|h^{i,m,k}\right\|^2\right] \leq \frac{\upsilon_{i,m}}{4\tilde{L}^4\gamma^2 M_2(1 + |\mathbb{T}|)}\right\}. \tag{A.64}$$

Following by the definition of constant K, the $\mathcal{L}_2$ deviation of neural agent can be bounded as follows:

$$\mathbb{E}\left[\left\|\alpha_i^{m+1} - \alpha_i^m\right\|^2\right] := \|\mathcal{B}[\theta_i^m(K)] - \alpha_i^m\|_{\mathcal{L}^2}^2 \leq \upsilon_{i,m}. \tag{A.65}$$

If we denote $\alpha_i^* = \alpha_i^{m_i^*}$, $m_i^* \in \mathbb{N}^+$, the following relation holds by triangle inequalities:

$$\mathbb{E}_{\mathbb{P}}\left[\|\alpha_i^* - \alpha_i^m\|^2\right] \leq (\mathbf{m}_i^* - m)\sup_m \upsilon_{i,m}. \tag{A.66}$$

As $\upsilon_{i,m} \vee \sup_m \upsilon_{i,m} = \sup_m \upsilon_{i,m}$ for all $m \leq m^*$, the expectation of Hamiltonian deviation can be bounded as a summation of two terms in (A.54):

$$\mathbb{E}\left[\|\delta\mathbf{H}_t^m\|^2\right] \leq M(\mathbf{m}^* - m + 1)\boldsymbol{\upsilon}_m, \tag{A.67}$$

where $\boldsymbol{\upsilon}_m = \{\upsilon_{i,m+1}\}$ and $\mathbf{m}^* = \{m_i^*\}$. By inserting above inequality into (A.49) and replacing $m + 1 \to m$, we obtain

$$\mathbb{E}\left[\|\delta\mathbf{Y}_t^m\|^2\right] \leq \mathbb{E}\left[\|\delta\mathbf{X}_T^m\|^2\right] e^{|\mathbb{T}|^2 \ln 16M(\mathbf{m}^* - m + 2)\boldsymbol{\upsilon}_m}. \tag{A.68}$$

Next, we evaluate the upper-bound of mean-squared decision deviations in (A.68):

$$\mathbb{E}_{\mathbb{Q}}\left[\|\delta\mathbf{X}_T^m\|^2\right] \leq \tilde{L}^2|\mathbb{T}|\,\|\boldsymbol{\theta}^m - \boldsymbol{\theta}^*\|^2. \tag{A.69}$$

By denoting the deviation of parameters as $\kappa_m = \|\delta\boldsymbol{\theta}^m\|^2$, we select small enough values $\upsilon_{i,m}$ that is related to the marginal constants $\epsilon_{i,m}$:

$$\boldsymbol{\upsilon}_m := \frac{\ln \boldsymbol{\epsilon}_m - \ln(\kappa_{m+1}\tilde{L}^2|\mathbb{T}|)}{|\mathbb{T}|^2 \ln 16M(m^* - m + 2)}, \quad \boldsymbol{\epsilon} := \sup_{\{m\leq m^*, i\in\{1,\cdots,I\}\}} \boldsymbol{\epsilon}_{i,m}. \tag{A.70}$$

If the neural agent has a large enough capacity to minimize the via gradient descent, these values assure minimal upper bound of the inequality (A.49). Finally, the non-linear Feynman-Kac theorem, which is related to the equation (A.30), directly gives the :

$$\left\|\mathcal{J}^i([\alpha_i^*, \boldsymbol{\alpha}_{(-i)}^*]) - \mathcal{J}^{i,m\to m^*}([\alpha_i^m, \boldsymbol{\alpha}_{(-i)}^m])\right\|^2 \leq \mathbb{E}\left[\left\|\delta\mathbf{Y}_t^{m\to m^*}\right\|^2\right] \leq \boldsymbol{\epsilon}, \tag{A.71}$$

Since $\mathcal{J}^i(\boldsymbol{\alpha}^*) < \mathcal{J}^i(\boldsymbol{\alpha}^m)$, our next step is to show the inequality between cost functionals given the action α_i^m obtained from the fictitious play and the other arbitrary action $\beta^i \neq \alpha_i^m$:

$$\mathcal{J}^{i,m\to m^*}([\alpha_i^m, \boldsymbol{\alpha}_{(-i)}^m]) \leq \mathcal{J}^{i,m\to m^*}([\beta_i, \alpha_i^m]), \quad \beta^i \in \mathbb{A}^i. \tag{A.72}$$

Then, by triangle inequality, we obtain inequality that shows the ϵ-Nash equilibrium:

$$\mathcal{J}^i([\alpha_i^*, \boldsymbol{\alpha}_{(-i)}^*]) \leq \mathcal{J}^i([\beta_i, \boldsymbol{\alpha}_{(-i)}^*]) + \epsilon_i, \quad \forall 1 \leq i \leq I. \tag{A.73}$$

The final step of the proof is to show the stochastic optimality according to the defined marginal constants $\boldsymbol{\epsilon}$. Let $\theta_i^m(k) : \mathbb{N}^+ \to \Theta$ be the trajectory of the neural parameters of the i-th neural agents at learning iteration k. Recall the fact that $\alpha_i^m = \alpha_i^{m-1,K} := \mathcal{B}[\theta_i^m(K-1)]$. Let us denote that $\mathbf{Y}_t^m|_\theta$ is the adjoint variable given neural parameters θ. We define the closed metric balls $\{B_{\delta_i^m}^k\}_{k\in\mathbb{N}^+}$ centered at $\theta_i^m(k)$ with the radius $0 < \delta_i^m$ such that

$$B_{\delta_i^m}^k := \{\vartheta \in \Theta; \|\vartheta - \theta_i^m(k)\| \leq \delta_r^k, \theta_i^m(k) \text{ is a local minimum of } \mathbb{E}\mathbf{Y}_t^m|_{\boldsymbol{\theta}}\}. \tag{A.74}$$

Next, we consider the sub-sequence $\{\theta_i^m(\bar{k})\}_{\bar{k}\in\bar{N}} \subseteq \{\theta_i^m(k)\}_{k\in\mathbb{N}}$, which defines the strictly-decreasing sequence $\{\mathbb{E}\mathbf{Y}_t^m|_{\boldsymbol{\theta}^m(k)}\}_{\bar{k}\in\bar{N}}$ an ordered index set $\bar{N}$. Then, the admissible set $\mathbb{A}^i$ is defined as follows:

$$\mathbb{A}^i := \left\{ \bigcup_{(\bar{k},1)}^{(\bar{K},m^*)} \alpha_i^m(\cdot,\cdot,B_{\delta_i^m}^{\bar{k}}); \bar{K} := \max\{\bar{N}\} \right\} \subset \mathcal{L}^2(T\times\mathbb{R}^d), \quad 1 \leq m \leq m^*, \tag{A.75}$$

where $\bar{K}$ is a maximal element in $\bar{N}$. Intuitively saying, the admissible set $\mathbb{A}^i$ is a collection of local metric balls centered at neural parameters of i-th agent updated by gradient descent.

To define the optimal actions of neural agents, let us consider an arbitrary local convex set $\boldsymbol{\beta} \in \mathcal{C} \subset \mathbb{A}$ where $\mathbb{A}$ is the admissible action set for multi-agents defined as follows:

$$\mathbb{A} := \bigotimes_{i=1}^{I} \mathbb{A}^i \subset [\mathcal{L}^2(T\times\mathbb{R}^d)]^{\otimes I}. \tag{A.76}$$

By the convexity, there exist I pairs $\{(\omega_i, \beta_i)\}_{i\in\{1,\dots,I\}}$ such that the following equality holds:

$$\underbrace{(\mathcal{B}\circ\cdots\circ\mathcal{B}[\theta^m(k=0)])}_{K \text{ times}} = \alpha_i^{m+1} + \omega_i\beta_i \tag{A.77}$$

and $\beta_i(\cdot) \neq \mathbf{0}$. Then, the Gâteaux derivative (Carmona, 2016, Theorem 4.12) of adjoint variable $\mathbf{Y}_t^m$ is derived as follows:

$$\begin{aligned} 0 \leq \frac{d}{d\omega} dY_t^{i,\boldsymbol{\alpha}}(\alpha_i^{m+1} + \omega_i\beta_i) &= \mathbb{E}\left[\int_{\mathbb{T}} \nabla_{\alpha_i} H^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, F_t^{i,\boldsymbol{\alpha}}, [\alpha_i^{m+1}, \boldsymbol{\alpha}_{(-i)}^m]) \cdot \beta_i dt\right] \\ &\leq \mathbb{E}\left[\int_{\mathbb{T}} \left\|\nabla_{\alpha_i} H^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, F_t^{i,\boldsymbol{\alpha}}, [\alpha_i, \boldsymbol{\alpha}_{(-i)}^m])|_{\alpha=\alpha_i^{m+1}+\omega_i\beta_i}\right\|^2 \cdot \|\beta_i\|^2 dt\right]. \end{aligned} \tag{A.78}$$

The first inequality is trivial due to the definition of admissible set $\mathbb{A}^i$. Since $\|\beta\|^2$ is nonzero, one can deduce that $\nabla_{\boldsymbol{\alpha}}\boldsymbol{H}|_{\boldsymbol{\alpha}^m+\boldsymbol{\omega}\boldsymbol{\beta}} \geq 0$. In other words, the action profiles $\boldsymbol{\alpha}^m$ are optimal actions and any arbitrary actions $\beta^i \in \mathbb{C}^i$ are eventually non-optimal. Hence, we can conclude from the above result that the inequality in (A.72) holds in the admissible action set. Finally, we conclude this proof by showing the stochastic optimality is preserved during the fictitious play by the definition of $\mathbb{A}$ that induces the following relation:

$$\boldsymbol{\mathcal{V}}_t + \boldsymbol{H}(t,\cdot,F_t,(\boldsymbol{\alpha}^{m+1}, \boldsymbol{\alpha}_{(\cdot)}^m)) + \frac{1}{2}\mathbf{Tr}(\Sigma\nabla^2\boldsymbol{\mathcal{V}}_t) = 0, \tag{A.79}$$

where $\Sigma = \sigma^T\sigma$. The stochastic optimality is obtained by the Hamiltonian equation (A.79) for every stage.

A.3.2. PROOF OF COROLLARY 3.1

In the proof of Proposition 3.1, we have shown the existence of the local action set that assures the convergence of the fictitious play. In this proof, we show the convergence of predictor $\Lambda_t^{\boldsymbol{\alpha}}$ according to the action set assuring the local Nash equilibrium. We start by calculating the norm deviation for the Itô's differential of the predictor (*i.e.*, $\delta\mathbf{\Lambda}_t^{\boldsymbol{\alpha}}$).

$$\begin{aligned}
\mathbb{E}\left[d\left\|\delta\mathbf{\Lambda}_t^{\boldsymbol{\alpha}}\right\|^2\right] &= \mathbb{E}\left[2\delta\mathbf{\Lambda}_t^{\boldsymbol{\alpha}}\cdot d\left[\delta\mathbf{\Lambda}_t^{\boldsymbol{\alpha}}\right]+\left[d\delta\mathbf{\Lambda}_t^{\boldsymbol{\alpha}}\right]^T d\delta\mathbf{\Lambda}_t^{\boldsymbol{\alpha}}\right] \\
&\leq \sum_j 2\mathbb{E}\Big[\left(\delta\mathbf{A}_j\mathbf{X}_t^j\right)\cdot\big(\delta\mathbf{A}_j\mathbf{b}_j+\delta\mathbf{X}_t^j\partial_t\mathbf{A}_j+\delta\mathbf{X}_t^j[\nabla\mathbf{A}_j]^T\mathbf{b}_j \\
&\qquad\qquad +\frac{1}{2}\mathbf{X}_t^j\mathbf{Tr}[\delta\Sigma_j\nabla^2\mathbf{A}_j]\big)+\delta(\nabla\mathbf{A}_j)^T\Sigma_j dt\Big] \\
&= \sum_j 2\mathbb{E}\Big[\underbrace{\delta(\mathbf{A}_j^2\mathbf{b}_j^T\mathbf{X}_t^j)}_{(a)}+||\delta\mathbf{X}_t^j||^2\big(\underbrace{\delta(\mathbf{A}_j\partial_t\mathbf{A}_j)}_{(b)}+\underbrace{\delta(\mathbf{A}_j\nabla\mathbf{A}_j\mathbf{b}_j^T)}_{(c)} \\
&\qquad\qquad +\underbrace{\frac{1}{2}\mathbf{Tr}[\Sigma_j\nabla^2\delta\mathbf{A}_j]}_{(d)}\big)+\underbrace{(\nabla\delta\mathbf{A}_j)^T\Sigma_j}_{(e)}dt\Big].
\end{aligned} \tag{A.80}$$

Note that we follow the rules for notating the object deviation in the previous proof. The first inequality From the last equality of (A.80), we estimate the upper bounds of each term.

$$(a)\ :\ \mathbb{E}\left[\left|\delta\mathbf{A}_j^2\mathbf{b}_j^T\mathbf{X}_t^j\right|dt\right]\leq\mathbb{E}\left[||\hat{A}||^2||\hat{b}||^2\left\|\delta\alpha_j^m\right\|^2\left\|\delta\mathbf{X}_t^j\right\|^2\right]\leq\mathbb{T}\tilde{L}^2M_2^3\kappa_m^2. \tag{A.81}$$

The result directly follows from By the assumption in (A.61), the following evaluation is trivial:

$$\mathbb{E}\left[\left\|\delta\mathbf{X}_t^j\right\|^2\right]=\mathbb{E}\left[\left\|\int_{\mathbb{T}}\delta\boldsymbol{b}_s^j ds\right\|^2\right]\leq\mathbb{E}\left[\int_{\mathbb{T}}\left\|\delta\boldsymbol{b}_s^j\right\|^2 ds\right]\leq\mathbb{T}M_2\tilde{L}\kappa_m. \tag{A.82}$$

Now, we estimate the two terms $(b),(c)$ that contain spatial and temporal derivatives of aggregation function in the following bracket:

$$\begin{cases}
(b)\ :\ \mathbb{E}\left[|\delta\mathbf{A}_j\partial_t\mathbf{A}_j|\,dt\right]\leq\mathbb{E}\left[\left|\hat{A}_j\cdot\partial_t\hat{A}_j\delta\boldsymbol{\alpha}_j\right|+\left|\hat{A}_j\cdot\delta\partial_t\boldsymbol{\alpha}_j\right|dt\right]\leq 2\tilde{L}\kappa_m M_2 dt, \\
(c)\ :\ \mathbb{E}\left[\left|\delta\mathbf{A}_j\nabla\mathbf{A}_j\mathbf{b}_j^T\right|dt\right]=\mathbb{E}\left[\left\|\hat{A}\right\|^2\left|\alpha_j^m\cdot\nabla\alpha_j^m\right|dt\right]\leq M_2\mathbb{E}\left[\left\|\alpha_j^m\right\|^2\left\|\nabla\alpha_j^m\right\|_F^2 dt\right] \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad \leq M_2\tilde{L}^2\kappa_m^2 dt,
\end{cases} \tag{A.83}$$

where the basic property of vector gradient $\nabla(v\cdot u)=(\nabla v)\cdot u+v\cdot(\nabla u)$ for all $v,u\in\mathbb{R}^d,\nabla v,\nabla u\in\mathbb{R}^{d\times d}$ is used in the estimation of (c). For the last two terms, *i.e.*, $(d),(e)$, we used the fact that the deviation of the matrix Σ is not defined because our MaSDEs assume uncontrollable volatility.

$$\begin{cases}
(d)\ :\ \frac{1}{2}\mathbb{E}\left[|\mathbf{Tr}[\Sigma_j\nabla^2\delta\mathbf{A}_j]|dt\right]=\frac{1}{2}\mathbb{E}\left[\left|\mathbf{Tr}[\Sigma_j\hat{A}_j(\nabla^2\alpha_j^m-\nabla^2\alpha_j^{m^*})]\right|\right] \\
\qquad\qquad\qquad\qquad\qquad\qquad \leq\left\|\hat{A}_j\right\|\frac{1}{2}\mathbb{E}\left[\|\Sigma_j\|_F\left\|\nabla^2\delta\mathbf{A}_j\right\|_F dt\right] \\
\qquad\qquad\qquad\qquad\qquad\qquad \leq\frac{M_2\kappa_m}{2}\mathbb{E}\left[\|\Sigma_j\|_F\right]dt, \\
(e)\ :\ \mathbb{E}\left[\left|\delta(\nabla\mathbf{A}_j)^T\Sigma_j)\right|dt\right]\leq M_2\kappa_m\mathbb{E}[\|\Sigma_j\|_F]dt.
\end{cases} \tag{A.84}$$

Note that the first line in (A.84) can be obtained by the trace inequality (*i.e.*, $\mathbf{Tr}[AB] \leq \|A\|_F \|B\|_F$). By collecting the evaluated terms above, we conclude the proof by showing the following relation:

$$\mathbb{E}\left[\int_{\mathbb{T}} \|\delta \mathbf{\Lambda}_s^{\boldsymbol{\alpha}}\|^2 ds\right] \propto O\left(\left\|\boldsymbol{\theta}^m - \boldsymbol{\theta}^{m^*}\right\|^3 I\mathbb{T}^2 \tilde{L}^3 M_2^3 \sup_{t\in\mathbb{T}} \mathbb{E}\left[\|\mathbf{\Sigma}\|_F\right]\right). \tag{A.85}$$

A.4. Proof of Proposition 3.4

Consider a triplet of empirical measures $(\mu_t^N, \hat{\mu}_t^N, \nu_t^N)$ with N-particles as followings:

$$\nu_t^N = \frac{1}{N}\sum_k^N \delta_{y_t^k}, \quad \mu_t^N = \frac{1}{N}\sum_k^N \delta_{\Lambda_t^{\boldsymbol{\alpha},k}}, \quad \tilde{\mu}_t^N = \frac{1}{N}\sum_k^N \delta_{\tilde{\Lambda}_t^{\boldsymbol{\alpha},k}}, \tag{A.86}$$

where we define the perturbation functional $\mathcal{Z} : \mathcal{P}_2 \to \mathcal{P}_2$ that injects random observations Z that were unseen during the training. Notice that the

$$\tilde{\mu}_t^N = \mathcal{Z}_{\#}\mu_t^N, \quad \tilde{\Lambda}_t^{\boldsymbol{\alpha}}|_{y_{s_1},\cdots,y_{s_i}+Z,\cdots,y_{s_I}} = \Lambda_t^{\boldsymbol{\alpha}}|_{y_{s_1},\cdots,y_{s_i},\cdots,y_{s_I}}, \quad Z \sim q, \tag{A.87}$$

Recall that the original temporal aggregator $\Lambda_t^{\boldsymbol{\alpha}}$ is conditioned by I past observations $(y_{s_1}, \cdots, y_{s_i}, \cdots, y_{s_I})$. Following by the definition of the noisy observation, the perturbed prediction is conditioned by $(y_{s_1}, \cdots, y_{s_i} + Z, \cdots, y_{s_I})$ where $\{Z^k\}_{k\in\{1,\cdots,N\}} \sim q$ for generic Gaussian distributions.

$$\begin{aligned}
\tilde{\mathbf{\Lambda}}_t^{\boldsymbol{\alpha},k} &= A_t^i \int_{s_i}^t \Bigg[b_u^i - \frac{1}{2}\nabla\cdot\Sigma_u + \frac{1}{2}(\delta t)^{-1}\Bigg\{ \sum_{m=2}^M \hat{\Sigma}_{m-1}\left[X_{t_m}^{i,\boldsymbol{\alpha},k} - X_{t_{m-1}}^{i,\boldsymbol{\alpha},k} - (\delta t) b_{t_{m-1}}^i\right] \\
&\quad + \hat{\Sigma}_0\left[X_{t_1}^{i,\boldsymbol{\alpha},k} - (y_{s_i}^k - Z_{s_i}^k) - (\delta t) b_{s_i}^i\right]\Bigg\} du\Bigg] \\
&\quad + \mathbf{A}_t^{(-i)} \int_{\mathbf{s}_{(-i)}}^t \Bigg[\mathbf{b}_u^{(-i)} - \frac{1}{2}\nabla\cdot\Sigma_u + \frac{1}{2}(\delta t)^{-1}\Bigg\{ \sum_{m=2}^M \hat{\Sigma}_{m-1}\left[\mathbf{X}_{t_m}^{(-i),\boldsymbol{\alpha},k} - \mathbf{X}_{t_{m-1}}^{(-i),\boldsymbol{\alpha},k} - (\delta t)\mathbf{b}_{t_{m-1}}^{(-i)}\right] \\
&\quad + \hat{\Sigma}_0\left[\mathbf{X}_{t_1}^{(-i),\boldsymbol{\alpha},k} - \mathbf{y}_{\mathbf{s}_{(-i)}}^k - (\delta t)\mathbf{b}_{\mathbf{s}_{(-i)}}^{(-i)}\right]\Bigg\} du\Bigg] \\
&= \frac{1}{2}(\delta t)^{-1} A_t^i \int_{s_i}^t \hat{\Sigma}_0 Z_{s_i}^k du + \mathbf{\Lambda}_t^{\boldsymbol{\alpha},k}.
\end{aligned} \tag{A.88}$$

In the equation (A.88), we define a ratio of covariance as $\hat{\Sigma}_0 := \hat{\Sigma}_0(t,x) = Q\mathbf{Diag}[\boldsymbol{\lambda}(t)/\boldsymbol{\lambda}(0)]Q^T$.

$$\begin{aligned}
\mathcal{W}_2^2(\nu_t^N, \hat{\mu}_t^N) \leq \mathbb{E}\left[\left\|\tilde{\mathbf{\Lambda}}_t^{\boldsymbol{\alpha},k} - y_t^k\right\|^2\right] &\leq \mathbb{E}\left[\left\|\mathbf{\Lambda}_t^{\boldsymbol{\alpha},k} - \tilde{\mathbf{\Lambda}}_t^{\boldsymbol{\alpha},k}\right\|^2\right] + \mathbb{E}\left[\left\|\mathbf{\Lambda}_t^{\boldsymbol{\alpha},k} - y_t^k\right\|^2\right] \\
&\leq \frac{(A_t^i)^2}{4(\delta t)^2}\mathbb{E}\left[\left\|\int_{s_i}^t \hat{\Sigma}_0 Z_{s_i}^k du\right\|^2\right] + \frac{1}{2\tilde{L}^2\gamma}\sqrt{\frac{\boldsymbol{v}}{M_2(1+|\mathbb{T}|)}} \\
&\leq A_t^i \frac{D_3|\mathbb{T}|}{4}(\delta t)^{-2}\mathbb{E}\left[\left\|Z_{s_i}^k\right\|^2\right] + \frac{1}{2\tilde{L}^2\gamma}\sqrt{\frac{\boldsymbol{v}}{M_2(1+|\mathbb{T}|)}}
\end{aligned} \tag{A.89}$$

The second inequality is a direct consequence of combining (A.88) and (A.64). Following by the assumption **(H4)**, and the fact that $1 \geq A_t^i \geq (A_t^i)^2$, $||\hat{\Sigma}_0||_F^2 \leq \sum_{i\in\mathbb{N}} \lambda_i(t)/\lambda_i(0) \leq D_3$ almost surely, the last equality holds. Consider the following triangle inequality

$$\mathcal{W}_2^2(\nu_t, \hat{\mu}_t) \leq \mathcal{W}_2^2(\nu_t, \nu_t^N) + \mathcal{W}_2^2(\nu_t^N, \hat{\mu}_t^N) + \mathcal{W}_2^2(\hat{\mu}_t, \hat{\mu}_t^N) \tag{A.90}$$

Then, under the assumption **(H5)** with the result of Theorem 2 from (Fournier & Guillin, 2015), we have

$$\textbf{(Empirical Measure Concentration)}: \begin{cases} \mathbb{P}\left[\mathcal{W}_2^2(\Upsilon, \Upsilon^N) \geq \varphi\right] \leq Ce^{-cNx^d} \\ \mathcal{W}_2(\Upsilon, \Upsilon^N) \leq \left(\frac{\log(C/\varphi)}{cN}\right)^{-d/4}, \text{ with probability at least } 1-\varphi \end{cases} \tag{A.91}$$

for two empirical measures $\Upsilon \in \{\hat{\mu}_t, \nu_t\}$. The proof is complete by setting $\delta t = 1/|\mathbb{T}|$ and the fact $\mathbb{E}\left\|Z_{s_i}^k\right\|^2 = \mathbf{Tr}[\Sigma := Q\boldsymbol{\lambda}Q^T = QQ^T\boldsymbol{\lambda}] = \mathbf{1}^T\boldsymbol{\lambda}_d$ in the last term of inequality (A.89).

A.5. Non-cooperative Scenarios

Empirical Study for the Non-cooperation. This section is devoted to answering the following important question: *Why do we consider a cooperative scenario rather than a non-cooperative one?* This question is crucial as the shared goal could have been achieved in a highly competitive and non-cooperative environment (Ye et al., 2018). For an intuitive explanation, we introduce a specific scenario where the neural agent becomes adversarial and determines that interfering with other agent's objectives is the best possible strategy to achieve its own goal. Followed by the definition, the behavioral rules for the adversarial agent can be summarized by the following equations:

$$\textit{(Adversarial Agent)} \quad \begin{cases} \mathcal{V}^j(t,x) = \mathcal{J}^j(t,x,[\alpha_j^*, \boldsymbol{\alpha}_{(-j)}]), \\ \mathcal{J}^i(t,x,[\alpha_i^*, \boldsymbol{\alpha}_{(-i)}]) \leq \mathcal{J}^i(t,x,[\alpha_i^*, \alpha_j^*, \boldsymbol{\alpha}_{(-i,-j)}]). \end{cases} \tag{A.92}$$

The second inequality shows that the adversarial agent (here, the j-th agent) can easily ruin other agent's goal although the victim (here, the i-th agent) produces its best response α_i^* to the adversarial environment. Eventually, one can expect that the competitive group will fail to accomplish the original goal (*i.e.*, the accurate prediction). To formalize the aforementioned scenario by estimating the total amount of inefficiency derived from the non-cooperation, we formulate the non-cooperative game (*e.g.*, min-max) by converting the cooperative behavior into an adversarial one:

$$\hat{h}^j = h^j + \left\| A_j^{\alpha_j} - 1 \right\|^2, \quad \min_{\alpha_j} \left\| A_j^{\alpha_j} - 1 \right\|^2 = \max_{\alpha_j} A_j^{\alpha_j}, \quad j \neq i \in \{1, \ldots, I\}. \tag{A.93}$$

With the adversarial cost $\hat{h}^j$, selfish agents maximize the influence on temporal aggregation by interfering with opponents. As the symmetric relation in (10) fails to be satisfied, the proposed framework with cost function in (A.93) is regarded as a non-cooperative game.

To elucidate the importance of cooperation in our framework of time-series prediction, we conduct an experiment on the Mackey-Glass dataset by setting a single adversarial agent α_1. The red and blue learning curves in Figure A.4 illustrate the coalition costs during the fictitious play over 20 stages produced by competitive and cooperative groups, respectively. As one might expect, the cooperative group shows stable learning dynamics until it converges to an agreement on the accurate prediction. This shows that the proposed fictitious play developed in Section 3 enjoys the desirable characteristic for accurate time-series prediction. Contrarily, the non-cooperative group under the competition fails to achieve a shared goal and shows that the adversarial behavior is fatal to the proposed framework regardless of the number of adversaries. In the optimization perspective, the primal reason for the failure is the instability induced by the naive approach for the gradient-descent based min-max type optimization. As a remedy, one may restrict the admissible action sets by considering an additional constraint on neural agents such as PL condition (Nouiehed et al., 2019) to develop a sophisticated algorithm for a convergent result. Yet, it is an open question whether this theoretical perspective can lead to accurate time-series prediction.

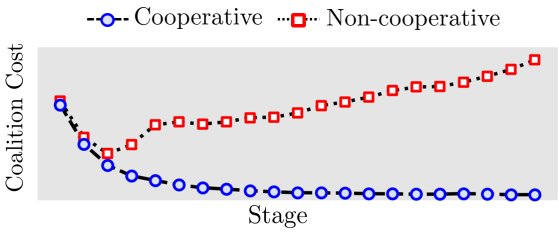


Figure A.4: Coalition cost for two scenarios.

A.6. Summary of Relevant Concepts

The most relevant strand of research about the multi-agent system in the machine learning community is multi-agent reinforcement learning (MARL) (Foerster et al., 2016). However, MARL and the proposed method differ in that MARL usually assumes model-free in discrete time, while we assume model-based in continuous time. The relevant concepts between MARL and ours are summarized in Table A.4.

A.7. Implementation Details

Evaluation. In all experiments with real-world datasets, we train each model for 500 epochs using the Adam optimizer with a learning rate of 10^{-3} and batch size of 128. We reported the MSE and (Gaussian) NLL as suggested in (Rubanova et al., 2019). The performance is evaluated on the predicted parts of the test dataset. For the evaluation of our method, we evaluate MSEs between the predictor $\Lambda_t^{\boldsymbol{\alpha}^*}$ and a *test* data $\hat{y}_t \sim \hat{\nu}_t$ as follows:

$$\textbf{(MSEs)}: \quad \mathbb{E}_{s\sim\mathbb{O}, t\sim\mathbb{T}, \Lambda_t^{\boldsymbol{\alpha}^*}\sim\mathbb{Q}} \left[\left\| \hat{y}_t - \Lambda_t^{\boldsymbol{\alpha}^*}|_{\hat{y}_{s_1},\ldots,\hat{y}_{s_i},\ldots,\hat{y}_{s_I}} \right\|^2 \right], \tag{A.94}$$

Observations		$\{y_{s_i}\}_{s_i \in \mathbb{O}}$
Targets		$\{y_t\}_{t \in \mathbb{T}}$
Agents	Agents	$\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_I], \quad \alpha_i := \alpha_i(\cdot, \cdot; \theta_i)$
Actions	Actions	$\alpha_i(t, X_t^{i,\boldsymbol{\alpha}}; \theta_i)$
States	Decisions	$\mathbf{X}_t^{\boldsymbol{\alpha}} = [X_t^{1,\boldsymbol{\alpha}}, \ldots, X_t^{I,\boldsymbol{\alpha}}] := \textbf{MaSDEs}(\{y_{s_i}\}_{s_i \in \mathbb{O}}, \boldsymbol{\alpha})$
Rewards	Errors	$\mathbb{E}_{t \sim \mathbb{T}}\left[\|y_t - \Lambda_t^{\boldsymbol{\alpha}}\|^2\right]$
Objective functional		$\mathcal{J}^i(t, \mathbf{X}_t^{\boldsymbol{\alpha}}, \boldsymbol{\alpha})$
Value function		$\mathcal{V}^i(t, x) = \min_{\alpha_i^* \in \mathbb{A}^i} \mathcal{J}^i(t, x, [\alpha_i^*, \boldsymbol{\alpha}_{(-i)}^*])$
Nash equilibrium		$\mathcal{J}^i(t, x, [\alpha_i^*, \boldsymbol{\alpha}_{(-i)}^*]) \leq \mathcal{J}^i(t, x, [\beta_i, \boldsymbol{\alpha}_{(-i)}^*]) + \epsilon_i$
Continuous Bellman	HJBEs	$\boldsymbol{\mathcal{V}}_t + \boldsymbol{H}(t, \cdot, F_t, (\boldsymbol{\alpha}^{m+1}, \boldsymbol{\alpha}_{(\cdot)}^m)) + \frac{1}{2}\mathbf{Tr}(\Sigma \nabla^2 \boldsymbol{\mathcal{V}}_t) = 0$

Table A.4: Comparison of relevant concepts. The first column stands for RL terminology; the second column stands for our terminology. In our works, agents and actions are neural networks and infinitesimal outputs given spatio-temporal variables $(t, X_t^{i,\boldsymbol{\alpha}})$. Decisions are continuous stochastic trajectories, and errors are L_2 loss between targets and aggregated decisions.

where the predictor is conditioned by the past observations $\{\hat{y}_{s_i}\}_{s_i \in \mathbb{O}}$ of the testing time-series. As neural agents collaborate to minimize the temporally averaged utilities (*i.e.*, $\mathcal{J}^i \approx \int_{\mathbb{T}} \mathbb{E}||y_s - \Lambda_s^{\boldsymbol{\alpha}}||^2 ds$) via the cooperative differential game, the goal of the proposed cooperative game becomes identical to forecast the time-series $\hat{y}_{t \in \mathbb{T}}$ given observations.

A.7.1. Network Architecture

In this section, we briefly introduce the network architecture of multi-agent neural SDEs as shown in Figure A.5. Given the initial condition $\{y_{(\cdot)}\}_{s_i \in \mathbb{O}}$, each neural agent takes the spatio-temporal variable $(t, X_t^{i,\boldsymbol{\alpha}})$ and produces infinitesimally transformed outputs to propagate its own stochastic trajectory. First, the temporal variable t is embedded into inhomogeneous and non-linear representation as $t' = (t, \sin(t), \cos(t))$. Note that we adopt the temporal privacy function suggested in (Park et al., 2022) for further temporal encoding. After the non-linear time embedding, the intermediate representations $(t', X_t^{i,\boldsymbol{\alpha}})$ are fed into the novel module referred to as *agent identification layer* (AIL), which is defined as follows:

$$\textbf{AIL}(X_t^{i,\boldsymbol{\alpha}}, i) := \zeta_i \left(\frac{X_t^{i,\boldsymbol{\alpha}} - \mathbb{E}(X_t^{i,\boldsymbol{\alpha}})}{\textbf{Std}(X_t^{i,\boldsymbol{\alpha}})} \right) + \zeta_i, \quad \zeta_i = (i+1)/I. \tag{A.95}$$

The AIL is motivated by the adaptive instance normalization (Huang & Belongie, 2017) that adaptively transforms the statistics of intermediate feature representations according to user-guided information (*i.e.*, index for neural agent). This encourages each neural agent to propagate the individual stochastic trajectory and ensures the separated representations. The outputs from AIL are fed into two subsequent linear layers with LipSwish (Chen et al., 2019) and produce values for drift b^i and temporal aggregation functions $A_i^{\alpha_i}$. In the experiments with real-world datasets, we identically set the dimension of hidden layers of each neural agent and the drift network as 128 following the baseline model (Park et al., 2022). We set the number of hidden layers as 2 for neural agents and 1 for the drift network. For the aggregation network, we used a relatively small dimension (*i.e.*, 36) with 2 hidden layers across all the experiments. For the simulation of the stochastic trajectory given the architecture, we discretize sampled times by applying the Euler-Maruyama scheme.

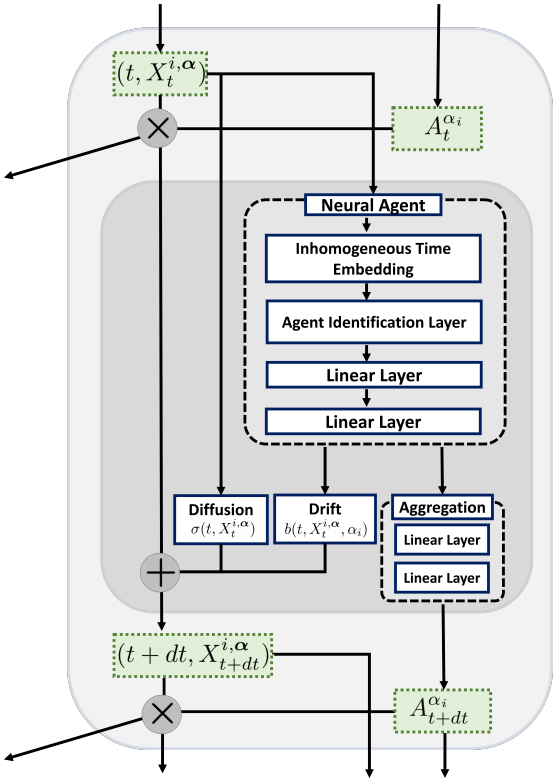


Figure A.5: The architecture of MaSDEs.

A.7.2. Dataset

PhysioNet Challenge 2012, (Silva et al., 2012), contains 8000 multivariate clinical time-series obtained from the intensive care unit (ICU). Each time-series has various clinical features of the patient's first 48 hours after admission to the ICU. We processed the dataset to hourly time-series and eliminated static features so that each time-series has a length of 48 with 35 time-varying features (*e.g.*, Albumin, Heart-rate, etc.,). We used half of the time-series as the training dataset and the remaining parts as the test dataset.

Speech Commands, (Warden, 2018), consists of one-second audio records of various spoken words such as "Yes", "No", "Up", and "Down". Since there are more than 100,000 record samples, we sub-sampled the dataset on two conflicting classes (*i.e.*, "Right" and "Left"). As a result, 6950 time-series records were selected. We pre-processed these time-series by computing Mel-frequency cepstrum coefficients so that each time-series has a length of 54 with 65 channels. We used 80% of selected data as the training dataset and the remaining parts as the test dataset.

Beijing Air-Quality, (Zhang et al., 2017; Dua & Graff, 2017), consists of multi-year air quality recordings across different locations in Beijing. Each sample contains 6-dimensional time-series features of $PM_{2.5}$, PM_{10}, SO_2, NO_2, CO, and O_3, which are recorded per hour. We segmented data per two days so that each sample has a length of 48. We combine recordings of 12 different locations into a single dataset and randomly split 80% of data into the training dataset and 20% of data into the test dataset.

A.8. Ablation Study: Impulse Signals

Modeling physical phenomenon with point processes (*e.g.*, Hawkes process) is common in many domains including finance for a high-frequency market (Bacry et al., 2015) and geology for earthquake (Ogata, 1998). In this experiment, we utilized Gaussian-impulse noises to mimic such discrete noisy events occurring at random times. To generate the Gaussian-impulse noise, we first sample the random time-stamp $\{t_1, t_2, \cdots\}$ from a homogeneous Poisson process with an intensity level of $\lambda = 8.0$. With this sampled time-stamps, we generate the data Y_t over the time interval $[0, 48]$ as follows:

$$Y_t = \begin{cases} 0.6 + U_t, & \text{if} \quad \{t\} \sim \texttt{Pois}(\lambda), \\ Z_t, & \text{otherwise.} \end{cases} \tag{A.96}$$

where $U_t \sim \text{Unif}[0, 1]$, $Z_t \sim \mathcal{N}(0, \sigma^2)$ and the standard deviation of Gaussian noise is set to 0.1. Similar to other datasets, we set the observable and the prediction intervals to contain the first 80% and the last 20% of total points, respectively.

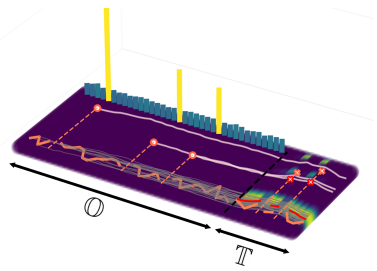


Figure A.6: Gaussian-Impulse Noises.

Figure A.6 illustrates the results of the proposed temporal aggregation of neural agents. Given impulse signals in the observation interval $\mathbb{O}$, neural agents accurately restore random peaks. Interestingly, neural agents $\{\alpha_5, \alpha_{17}, \alpha_{25}\}$ that encode the past impulse signals focus on the times at which impulse noise occurs. In other words, neural agents emulate stochastic peaks by learning intrinsic temporal correlations (*i.e.*, $p(y_{t\in\mathbb{T}}|y_{s\in\mathbb{O}}), t, s \sim \texttt{Pois}$) between the random occurrence times of the past and the future impulse signals.

Based on the additional investigation, we conclude that the proposed neural agents learn various types of temporal dynamics from the data by cooperatively pursuing the group goal. The rational group filters out the influence of observations that are redundant for predicting future states. This encourages neural agents to focus more on the informative signals for accurate predictions by capitalizing on the data statistics.

A.9. Additional Qualitative Results

In this subsection, we provide additional qualitative results on BAQD across overall features of different instances to show the effectiveness of the proposed method on time-series prediction. Figure A.7 shows prediction results for all features and learned aggregations on the BAQD. We can observe that our method exhibits similar (but not strictly restricted) to the temporal decay assumption. As can be seen in Figure A.7, the learned temporal aggregation varies depending on the given observation. We claim that neural agents have a variety of cooperation patterns to generate an accurate prediction. That is, our method flexibly extracts complex and heterogeneous temporal correlations beyond simple temporal decay assumed by conventional methods.

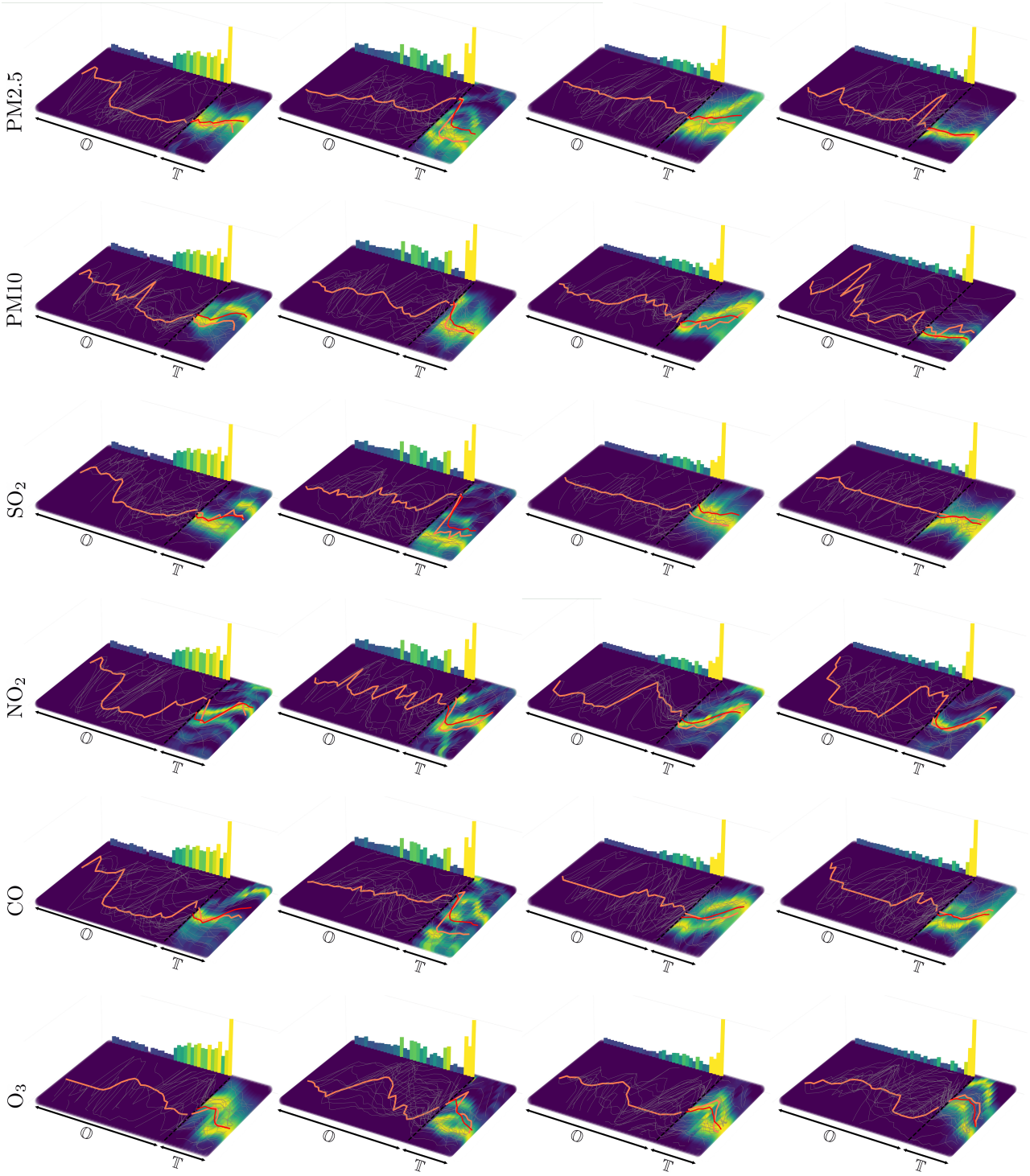


Figure A.7: Additional qualitative results on BAQD dataset.

A.10. Comparisons with Transformer-based Models.

The most significant advantage of using DE-based methods is the capability of handling irregularly sampled or partially observed time-series, which is common in real-world domains such as medicine or finance. Any non-DE-based methods are incapable of dealing with such unstandardized data without careful pre-processing steps (*e.g.*, discretization and

interpolation) to handle irregularity.

Thus, to compare with the state-of-the-art Transformer-based time-series prediction methods, we chose the Speech dataset since it comprises complete time-series observations with regular time intervals. Thus, potential bias from pre-processing steps can be avoided. As can be seen in Table A.5, our method outperforms its non-DE-based counterparts implying that our proposed method can be a promising tool for general time-series modeling under both regular and irregular settings.

Method	Test MSE $\downarrow$ ($\times 10^{-2}$)
Transformer (Vaswani et al., 2017)	0.4476 ± 0.018
Reformer (Kitaev et al., 2020)	0.4378 ± 0.004
Informer (Zhou et al., 2021)	0.4253 ± 0.002
Autoformer (Wu et al., 2021)	0.4904 ± 0.039
Ours	$\mathbf{0.4087 \pm 0.013}$

Table A.5: Test MSE (mean $\pm$ std) on Speech dataset.