Pri-GP: Prior-Aware Distributed Gaussian Process Regression

Zewen Yang [* 1] Xiaobing Dai [* 2] Akshat Dubey [1] Sandra Hirche [2] Georges Hattab [1 3]

Abstract

This paper introduces an innovative approach to enhance distributed cooperative learning using Gaussian process (GP) regression in multi-agent systems (MASs). The key contribution of this work is the development of an elective learning algorithm, namely prior-aware elective distributed GP (Pri-GP), which empowers agents with the capability to selectively request predictions from neighboring agents based on their trustworthiness. The proposed Pri-GP effectively improves individual prediction accuracy, especially in cases where the prior knowledge of an agent is incorrect. Moreover, it eliminates the need for computationally intensive variance calculations for determining aggregation weights in distributed GP. Furthermore, we establish a prediction error bound within the Pri-GP framework, ensuring the reliability of predictions, which is regarded as a crucial property in safety-critical MAS applications.

1. Introduction

In the context of multi-agent systems (MASs), distributed learning entails a collaborative approach where one or more groups of agents join forces to improve their understanding and knowledge of complex tasks, such as robotic swarm navigation (Stirling et al., 2012), underwater vehicle resource exploration missions (Yan et al., 2019; 2020), and air drone search and rescue operations (Alotaibi et al., 2019), etc. To address the inherent challenges posed by uncertain dynamics or environmental conditions in dynamic systems, distributed learning integrates supervised machine learning techniques enabling agents to learn cooperatively. This approach leads to more effective and robust learning capabilities compared to traditional single-agent models (Provost & Hennessy, 1996).

*Equal contribution. [1]Robert Koch Institute, Berlin, Germany [2]Technical University of Munich, Munich, Germany [3]Freie Universität Berlin, Berlin, Germany. Correspondence to: Zewen Yang <yangz@rki.de>.

1.1. Related Work

Specifically, Neural Networks (NNs) emerge as the prevailing methodology approximating complex mappings or functions in MASs (Dai et al., 2019). To learn the unknown patterns jointly, the NN weights are shared among neighboring agents, facilitating the attainment of optimal parameter values (W. Wang & Peng, 2017; Gao et al., 2020). Several research endeavors have been dedicated to system identification within the framework of addressing uncertainties (Jafari & Xu, 2018; Dai et al., 2021). However, as the complexity of NN models increases with additional hidden layers and neurons, the practicality of sharing all NN weights within constrained communication bandwidth becomes unfeasible. This challenge further leads to significant delays in the learning process, rendering it impractical and resulting in the MAS dynamics potentially unstable. Though only exchanging the predictions, the inherent challenge arises from the inadequacy in precisely quantifying prediction uncertainties, thereby impeding its applicability in safety-critical tasks.

An alternative supervised machine learning approach, Gaussian process regression (GPR) (Rasmussen & Williams, 2006), has been widely used in the realm of safety-critical control systems, primarily owing to its distinctive attributes. Under the Bayesian inference framework, GPR not only provides probabilistic predictions, where the prior model can be updated continuously accommodating the incorporation of observations (Schürch et al., 2020), but also offers error bounds endowed with robust guarantees (Lederer et al., 2019). In contrast to NNs-based methods, GP models are only required to share their individual predictions with connected counterparts (Yin et al., 2023). Even if the agent lacks access to the complete training dataset, collaborative improvements in prediction quality can be achieved by aggregating predictions from neighboring agents. The synergy of aggregated predictions from neighboring agents, as detailed in prior research (Yang et al., 2021; Lederer et al., 2023; Yang et al., 2024b), underscores the effectiveness of this approach in achieving improved prediction quality within the multi-agent framework. Moreover, to improve the efficiency of cooperative learning with GPs, the event-triggered mechanism is introduced in (Dai et al., 2024a;b). However, the mentioned literature above based on distributed GP (Tresp, 2000b; Deisenroth & Ng, 2015) im-

poses a constraint in the sense that it mandates the exchange of information with all neighboring agents, offering no flexibility for agents to selectively determine which neighbors to collaborate with. The agent needs to aggregate the predictions from the neighbors, thus each agent is compelled to compute predictions for all of its neighboring counterparts, potentially incurring a substantial increase in computational overhead. For example, the product of GP experts (POE) methods (Cao & Fleet, 2015) and Bayesian committee machine (BCM) methods (Tresp, 2000a; Liu et al., 2018) rely on the posterior variance of GP requiring $\mathcal{O}(N^2)$ calculations. This concern becomes particularly salient in scenarios characterized by limited computational resources at each agent's disposal or when the expeditious generation of predictions is imperative. Furthermore, the rigidity of this collaborative setup raises noteworthy issues, especially in cases where the prior knowledge is erroneous. This is ignored by most distributed GP approaches, for instance, the mixture of GP experts (MOE) approach (Tresp, 2000b), where the aggregation weight is just the reciprocal of the total number of the GP models. In such instances, a uniform collaboration approach may yield suboptimal results, as it does not accommodate the possibility that an individual agent may possess superior predictive capabilities compared to its collaborators.

1.2. Contribution

To address these challenges, we propose a novel approach where agents are empowered with the capability of requesting predictions exclusively, allowing them to actively select their collaborators from among their neighbors. This elective learning method leverages the error between prior knowledge and real observations, called prior-aware elective distributed GP (Pri-GP), which can let the agent smartly choose the neighbors who are worth trusting. Therefore, it not only reduces the heavy computation required by the neighbors but also improves the individual agent's prediction in the distributed cooperative learning framework avoiding aggregating the potentially misleading prediction from the neighbors whose prior knowledge is significantly wrong.

The contribution of this paper is that an elective distributed cooperative learning algorithm for MASs with distributed GPR is proposed. The primary innovation lies in the proposal of an error metric that leverages prior errors, which possesses broader applicability beyond its immediate application and can be seamlessly integrated into various machine learning methodologies. In particular, the proposed Pri-GP approach offers a remarkable degree of flexibility by circumventing the necessity of computing variance for the determination of aggregation weights, a process commonly associated with a computational complexity of $\mathcal{O}(N^2)$ in distributed GP techniques. Additionally, we provide a prediction error bound using the Pri-GP framework, thereby

ensuring the reliability of predictions, a crucial aspect, particularly in the context of safety-critical applications.

The structure of the paper: In Section 2, we present the preliminaries and the problem setting. The proposed Pri-GP algorithm is expounded upon in Section 3. The effectiveness of Pri-GP is demonstrated through comprehensive comparisons with state-of-the-art approaches in Section 4. Lastly, a summary of the paper is provided in Section 5.

In Section 2, we state the preliminaries and problem setting. Then, the proposed Pri-GP is detailed in Section 3. The effectiveness of our methods is demonstrated through comparisons with state-of-the-art approaches in Section 4. Lastly, Section 5 summarizes the paper.

2. Preliminaries and Problem Setting

In this section, we introduce the notation used throughout the paper and elaborate on the graph theory utilized for describing the MAS network in Section 2.1. Subsequently, Section 2.2 outlines the primary objective, followed by Gaussian process regression as the machine learning technique employed for inferring the unknown functions in Section 2.3.

2.1. Notation and Graph Theory

We use the notation $\mathbb{R}_+/\mathbb{R}_{0,+}$ to denote positive real numbers without/with zero and denote natural numbers without/with zero as $\mathbb{N}/\mathbb{N}_0$, respectively. Unless otherwise specified, the identity matrix is $\boldsymbol{I}$, and a matrix or vector consisting of elements 1 is denoted by $\mathbf{1}$, with appropriate sizes as needed. The Euclidean norm of a vector or matrix is represented as $\|\cdot\|$, the cardinality of a set $\mathcal{N}$ as $|\mathcal{N}|$, and the element-wise absolute operator for vector input $\boldsymbol{v}$ as $|\boldsymbol{v}|$.

To characterize the communication network of the distributed MAS, an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is employed among S agents, $S \in \mathbb{N}$. The node set $\mathcal{V} = \{1, \ldots, S\}$ represents the index of the agents, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ signifies the set of edges between nodes, where an edge $(i, j) \in \mathcal{E}$ indicates that agent i and agent j exchange their information between each other. The self-loop included adjacency matrix of $\mathcal{G}$ is denoted by $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{S \times S}$, where all diagonal entries $a_{ii} = 1$, other elements of the matrix $a_{ij} = a_{ji} = 1$ if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. Moreover, we let the set of neighbours of agent i represent as $\mathcal{N}_i = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$ and the set $\bar{\mathcal{N}}_i$ comprises agent i along with its neighbors, meaning it contains agent i itself and all the agents in $\mathcal{N}_i$.

2.2. Problem Description

In this work, we delve into the investigation of a distributed MAS comprising a total of S individual agents. The pri-

mary objective of these homogeneous agents is to identify the identical functional characteristics of their dynamical systems

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}), \tag{1}$$

considering $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\cdot), \cdots, f_m(\cdot)]^{\mathrm{T}}$, where for each dimension, $f_d(\cdot) : \mathbb{X} \to \mathbb{R}, \forall d = 1, \cdots, m$ and $\mathbb{X} \subset \mathbb{R}^m$ is a $m \in \mathbb{N}$ dimensional compact domain. These agents actively engage in communication with one another through a network, herein referred to as $\mathcal{G}$. Concurrently, each agent has the collection of observational data set $\mathbb{D}_i$ with the subscript i indicating the specific agent within the set $\mathcal{V}$, which are subsequently harnessed for the purpose of estimating the unknown functions. It is noteworthy that each agent possesses its own distinct set of prior knowledge pertaining to these unidentified functions, which is represented as $\hat{f}(\cdot)$ considering the same mapping relationship of f. To facilitate the utilization of this prior knowledge and the learned functions for describing the system's dynamics, we introduce the following assumption.

Assumption 2.1. The functions $f(\cdot)$ and $\hat{f}(\cdot)$ exhibit local Lipschitz continuity within the compact domain $\mathbb{X}$, characterized by a Lipschitz constant denoted as $L_f \in \mathbb{R}$, i.e, $\|\nabla f(\boldsymbol{x})\| \le L_f$ for all $\boldsymbol{x} \in \mathbb{X}$.

This assumption is frequently encountered in the context of nonlinear systems (Khalil, 2002), which serves the guarantee of the existence and uniqueness of solutions for nonlinear autonomous systems. In practice, this assumption merely necessitates the system's continuity, with the subsequent establishment of Lipschitz continuity being a derived property within the bounded region denoted as $\mathbb{X}$. Consequently, it can be contended that this assumption imposes no onerous constraints on the system under consideration.

It is important to highlight that in this paper, we have chosen to work with a one-dimensional estimation, i.e., a scalar function, where $d = 1$. This simplification in dimensionality has been adopted for the sake of keeping our notations concise and straightforward. However, it should be noted that the outcomes derived in this work can be readily extended to higher dimensional functions achieved by employing techniques such as the Kronecker product and multi-output learning methods.

To describe the training data set of the agent i comprising streaming data pairs, we denote it as $\mathbb{D}_i$ with $N_i \in \mathbb{N}$ training data pairs. This data set is represented as $\mathbb{D}_i = \{(\boldsymbol{x}_i^{(p)}, y_i^{(p)})\}_{p=1,\ldots,N_i}$, where each pair consists of a training input $\boldsymbol{x}_i \in \mathbb{X}$ and a corresponding training output $y_i \in \mathbb{R}$, and satisfies the following assumption.

Assumption 2.2. The data pair $\{(\boldsymbol{x}_i^{(p)}, y_i^{(p)})\}$ that is obtained by agent $i \in \mathcal{V}$, such that the noise $\varsigma_i^{(p)} \in \mathbb{R}$ of $y_i^{(p)} = f(\boldsymbol{x}_i^{(p)}) + \varsigma_i^{(p)}$ follows a zero-mean, independent

and identical Gaussian distribution, i.e., $\varsigma_i^{(p)} \sim \mathcal{N}(0, \sigma_n^2)$, $\forall p \in \mathbb{N}$ with $\sigma_n > 0$.

As outlined in Assumption 2.2, it is assumed that each agent independently collects their own dataset without sharing it among others. While this assumption necessitates precise and complete measurements of the system states, a requirement commonly encountered in MASs when employing data-driven methodologies, it is possible to effectively address the measurement noise associated with the variable $\boldsymbol{x}$. This can be achieved through various techniques, such as Taylor expansion, as demonstrated in prior works like (Mchutchon & Rasmussen; Kim et al.), or by incorporating noise handling directly into the kernel function, as discussed in (Wang et al., 2022). It is worth noting that there are broader considerations related to noise distribution relaxation, we refer to (Chowdhury & Gopalan; Maddalena et al.). However, these aspects lie beyond the scope of this paper.

In the development of a distributed learning framework for the MAS, the estimation of the unknown function $f(\cdot)$ at time $t_k \in \mathbb{R}_{0,+}$ of the i-th agent is considered as

$$\tilde{f}_i(\boldsymbol{x}(t_k)) = \sum_{j=1}^{S} \omega_{ij} \phi_{ij}\Big(\boldsymbol{x}(t_k), \mathbb{D}_j, \hat{f}_j(\boldsymbol{x}(t_k))\Big), \ i \in \mathcal{V}, \tag{2}$$

where $k \in \mathbb{N}$ is indicating the specific instance of prediction, and the cooperative estimation function $\phi_{ij}(\cdot, \cdot, \cdot)$ corresponds to the prediction mechanism employed by each agent. Specifically, when $i \neq j$, it characterizes the prediction generated through the neighbor agent j with its training data set $\mathbb{D}_j$ and prior knowledge function $\hat{f}_j(\cdot)$. While $i = j$, it represents the prediction of agent i itself using its own prior and training data set.

Therefore, the primary focus of this paper revolves around the development of a collaborative estimation function tailored to augment individual predictions of an unknown function in an elective manner, which can enhance the predictions without necessitating the aggregation of predictions from all neighboring agents, thereby mitigating the computational burden imposed on these neighbors. Importantly, this elective strategy takes into account the varying accuracy of prior knowledge possessed by agents, which is illustrated in Section 3.

2.3. Gaussian Process Regression

In this paper, Gaussian process regression, a supervised machine learning technique, is employed to perform inference on the unknown function $f(\cdot)$. A Gaussian process $\mathcal{GP}(\hat{f}(\cdot), \kappa(\cdot, \cdot))$ is utilized to establish a probabilistic model characterized by two fundamental components: the prior mean function $\hat{f}(\cdot)$, and the kernel function, denoted

as $\kappa(\cdot, \cdot) : \mathbb{X} \times \mathbb{X} \to \mathbb{R}_{0,+}$, which satisfies the following assumption.

Assumption 2.3. The kernel function $\kappa(\|x - x'\|) = \kappa(x, x')$ is chosen as stationary, monotonically decreasing, and Lipschitz continuous with a specified Lipschitz constant denoted as L_κ.

The adoption of a Lipschitz continuous kernel emerges as a judicious selection when dealing with continuous unknown functions within a confined domain. The kernel's monotonic decrease, as reflected in its behavior, implies a diminishing strength of association between the training data and the evaluated point as their Euclidean distance increases. Therefore, a common choice of kernel function is ARD exponential kernel formulated as

$$\kappa(\boldsymbol{x}, \boldsymbol{x}') = \sigma_r^2 \exp\Big(-\frac{1}{2} \sum_{j=1}^m l_j{}^2 (x_j - x_j')^2 \Big), \quad (3)$$

where $\boldsymbol{x} = [x_1, x_2, \ldots, x_m] \in \mathbb{X}$. $\sigma_r \in \mathbb{R}_+$ and $l_j \in \mathbb{R}_+$ are hyper-parameters.

We consider the agent $i \in \mathcal{V}$ within the MAS to be equipped with a GP model characterized by hyperparameters denoted as $\boldsymbol{\theta} = \{\sigma_r, l_s, s = 1, 2, \ldots, m\}$. Additionally, the agent possesses a fixed training dataset denoted as $\mathbb{D}_i$ with $N_i \in \mathbb{N}_0$ training data pairs under Assumption 2.2 and holds different prior of the unknown function $\hat{f}_i(\cdot)$ satisfying Assumption 2.1. Agent i performs predictions at discrete time points denoted as t_k. These predictions are represented as the posterior mean prediction and associated prediction variance (Rasmussen & Williams, 2006) at the query point $\boldsymbol{x}(t_k)$, which are formulated as

$$\begin{aligned} \mu_i\big(\boldsymbol{x}(t_k)|\hat{f}_i(\boldsymbol{x}(t_k)), \mathbb{D}_i\big) = {} & \hat{f}_i(\boldsymbol{x}(t_k)) \quad (4) \\ & + K\big(\boldsymbol{x}(t_k), \boldsymbol{X}_i\big) \boldsymbol{K}(\boldsymbol{X}_i)^{-1}\big(\boldsymbol{Y}_i^{\mathrm{T}} - \hat{\boldsymbol{f}}_i(\boldsymbol{X}_i)^{\mathrm{T}}\big), \end{aligned}$$

$$\begin{aligned} \sigma_i\big(\boldsymbol{x}(t_k)|\hat{f}_i(\boldsymbol{x}(t_k)), \mathbb{D}_i\big) = {} & \kappa(\boldsymbol{x}(t_k), \boldsymbol{x}(t_k)) \quad (5) \\ & - K\big(\boldsymbol{x}(t_k), \boldsymbol{X}_i\big) \boldsymbol{K}(\boldsymbol{X}_i)^{-1} K\big(\boldsymbol{x}(t_k), \boldsymbol{X}_i\big)^{\mathrm{T}}, \end{aligned}$$

respectively, where

$$\boldsymbol{K}(\boldsymbol{X}_i) = \mathcal{K}(\boldsymbol{X}_i, \boldsymbol{X}_i) + \sigma_n^2 \boldsymbol{I}, \quad (6)$$

$$\mathcal{K}(\boldsymbol{X}_i, \boldsymbol{X}_i) = \big[\kappa\big(\boldsymbol{x}_i^{(a)}, \boldsymbol{x}_i^{(b)}\big)\big]_{a,b=1,\ldots,N_i} \quad (7)$$

$$K\big(\boldsymbol{x}(t_k), \boldsymbol{X}_i(t_k)\big) = \big[\kappa\big(\boldsymbol{x}(t_k), \boldsymbol{x}_i^{(1)}\big) \cdots \kappa\big(\boldsymbol{x}_{t_k}, \boldsymbol{x}_i^{(N_i)}\big)\big], \quad (8)$$

$$\boldsymbol{X}_i = \big[\boldsymbol{x}_i^{(1)} \cdots \boldsymbol{x}_i^{(N_i)}\big], \boldsymbol{Y}_i = \big[y_i^{(1)} \cdots y_i^{(N_i)}\big], \quad (9)$$

and the concatenated prior mean value is defined as $\hat{\boldsymbol{f}}(\boldsymbol{X}_i) = \big[\hat{f}\big(\boldsymbol{x}_i^{(1)}\big), \ldots, \hat{f}\big(\boldsymbol{x}_i^{(N_i)}\big)\big]$.

Therefore, each agent i can use the posterior mean to identify the unknown $f(\cdot)$ at $\boldsymbol{x}(t_k)$ by Equation (4), the individual estimation function $\phi_{ii}\big(\boldsymbol{x}(t_k), \mathbb{D}_i, \hat{f}_i(\boldsymbol{x}(t_k))\big) =$

$\mu_i\big(\boldsymbol{x}(t_k)|\hat{f}(\boldsymbol{x}(t_k)), \mathbb{D}_i\big)$. To simplify our notation, we denote $\mu_i(\boldsymbol{x}(t_k)|\hat{f}_i(\boldsymbol{x}(t_k)), \mathbb{D}_i)$ as $\mu_i(\boldsymbol{x}(t_k))$ in the subsequent sections of this paper. Having established the GPR as our foundational tool, the subsequent section is dedicated to the formulation of our elective distributed learning approach.

3. Elective Distributed Learning with Prior-Aware GPR

To assess the reliability of the collaborators, we introduce the prior estimation error in Section 3.1. This metric serves as a quantitative measure for gauging the trustworthiness of neighboring agents in the MAS. Building upon this quantitative foundation, we proceed to formulate an elective distributed learning algorithm, as outlined in Section 3.2. Additionally, to bolster the safety and guarantee of the learning scenario, we establish a prediction error bound within the Pri-GP framework in Section 3.3.

3.1. Prior Error Quantification

In multi-agent systems, each agent possesses a finite training dataset, which naturally leads to a scenario where predictions for points lying beyond the training data domain or in sparsely sampled regions become highly reliant on prior knowledge. Consequently, the accuracy of these predictions is predominantly influenced by the quality of the prior information in the Bayesian framework. This situation underscores the potential challenges arising from incorrect or inadequate prior knowledge. Therefore, there arises a compelling need for a systematic mechanism to assess the degree of inaccuracy associated with prior knowledge, particularly in the presence of observed true values for predictions. To formally quantify the disparity between the prior estimation and observed data, we introduce the concept of prior estimation error denoted by

$$e_i(\boldsymbol{x}(t_k)) = \hat{f}_i(\boldsymbol{x}(t_k)) - y(t_k). \quad (10)$$

Given the availability of system observations, we systematically log the associated errors. To establish the evolving cumulative error over time, we define the average accumulated historical prior estimation error as

$$\varepsilon_i(t_k) = \frac{1}{k} \sum_{l=1}^k |e_i(\boldsymbol{x}(t_{k-l}))|. \quad (11)$$

This metric serves as a pivotal instrument in characterizing the deviation between prior expectations and empirical observations, thereby enhancing our capacity to evaluate and interpret the reliability of the models, which is illustrated in the following lemma.

Lemma 3.1. *The variable $\varepsilon_i(t_k)$ reflects the prediction error on the training data set $\mathbb{D}_i$, and the measurement*

error. In particular, $\varepsilon_i(t_k)$ is written as

$$\varepsilon_i(t_k) = \frac{1}{k}\mathbf{1}^{\mathrm{T}}|\boldsymbol{G}_k(\boldsymbol{\xi}_k + \boldsymbol{\varsigma}_k)|, \qquad (12)$$

where $\boldsymbol{G}_k = (\boldsymbol{I} + \sigma_n^{-2}\mathcal{K}(\boldsymbol{X}_i, \boldsymbol{X}_i))$. The aggregated error denotes $\boldsymbol{\xi}_k = [\xi_0, \cdots, \xi_k]^T$ with $\xi_p = f(\boldsymbol{x}^{(p)}) - \mu_i(\boldsymbol{x}^{(p)}), \forall p = 1, \cdots, k$. The noise vector $\boldsymbol{\varsigma}_k$ collects all measurement from $t = t_0$ to t_k, i.e., $\boldsymbol{\varsigma}_k = [\varsigma^{(0)}, \cdots, \varsigma^{(k)}]^T$, where the individual noise $\varsigma^{(p)}, \forall p = 1, \cdots, k$, follows Assumption 2.2.

Proof. Considering the prediction of each sample in $\mathbb{D}_i$ using (4), the aggregated prediction $\boldsymbol{\mu}_i(\boldsymbol{X}_i) = [\mu_i(\boldsymbol{x}^{(0)}), \cdots, \mu_i(\boldsymbol{x}^{(N_i)})]^T$ is written as

$$
\begin{aligned}
\boldsymbol{\mu}_i(\boldsymbol{X}_i) =& \hat{\boldsymbol{f}}_i(\boldsymbol{X}_k)^{\mathrm{T}} \\
&+ \mathcal{K}(\boldsymbol{X}_i, \boldsymbol{X}_i)\boldsymbol{K}(\boldsymbol{X}_i)^{-1}\big(\boldsymbol{Y}_k^{\mathrm{T}} - \hat{\boldsymbol{f}}_i(\boldsymbol{X}_i)^{\mathrm{T}}\big) \\
=& \big(\boldsymbol{I} - \mathcal{K}(\boldsymbol{X}_i, \boldsymbol{X}_i)\boldsymbol{K}(\boldsymbol{X}_i)^{-1}\big)\hat{\boldsymbol{f}}_i(\boldsymbol{X}_i)^{\mathrm{T}} \\
&+ \mathcal{K}(\boldsymbol{X}_i, \boldsymbol{X}_i)\boldsymbol{K}(\boldsymbol{X}_i)^{-1}\boldsymbol{Y}_k^{\mathrm{T}}. \qquad (13)
\end{aligned}
$$

Then, the aggregated deviation between the measurements $y^{(p)}$ and the prediction $\mu_i(\boldsymbol{x}^{(p)})$ denoted by

$$
\begin{aligned}
\boldsymbol{Y}_k^{\mathrm{T}} - \boldsymbol{\mu}_i(\boldsymbol{X}_k) &= \sigma_n^2 \boldsymbol{K}(\boldsymbol{X}_i)^{-1}\big(\boldsymbol{Y}_k^{\mathrm{T}} - \hat{\boldsymbol{f}}_i(\boldsymbol{X}_k)^{\mathrm{T}}\big) \\
&= \boldsymbol{G}_k^{-1}\big(\boldsymbol{Y}_k^{\mathrm{T}} - \hat{\boldsymbol{f}}_i(\boldsymbol{X}_k)^{\mathrm{T}}\big), \qquad (14)
\end{aligned}
$$

which is also equivalent to

$$\boldsymbol{Y}_k^{\mathrm{T}} - \hat{\boldsymbol{f}}_i(\boldsymbol{X}_k)^{\mathrm{T}} = \boldsymbol{G}_k\big(\boldsymbol{Y}_k^{\mathrm{T}} - \boldsymbol{\mu}_i(\boldsymbol{X}_k)\big)$$

due to the non-singular $\boldsymbol{G}_k$. Note that $\boldsymbol{Y}_k^{\mathrm{T}} - \boldsymbol{\mu}_i(\boldsymbol{X}_k)$ is divided into two parts with prediction error $\boldsymbol{\xi}_k$ and measurement error $\boldsymbol{\varsigma}_k$, i.e., $\boldsymbol{Y}_k^{\mathrm{T}} - \boldsymbol{\mu}_i(\boldsymbol{X}_k) = \boldsymbol{\xi}_k + \boldsymbol{\varsigma}_k$. Moreover, reformulating (11) as $\varepsilon_i(t_k) = k^{-1}\mathbf{1}^{\mathrm{T}}\boldsymbol{e}_i(\boldsymbol{X}_k)$ with $\boldsymbol{e}_i(\boldsymbol{X}_k) = [e_i(\boldsymbol{x}(t_0)), \cdots, e_i(\boldsymbol{x}(t_k))]^T$, the accumulated prior estimation error $\varepsilon_i(t_k)$ is written as

$$\varepsilon_i(t_k) = \frac{1}{k}\mathbf{1}^{\mathrm{T}}|\boldsymbol{Y}_k^{\mathrm{T}} - \boldsymbol{\mu}_i(\boldsymbol{X}_k)| \qquad (15)$$

and then the result in (12) is derived. □

Remark 3.2. Lemma 3.1 shows the accumulated historical prior estimation error encodes the joint effects of the prediction and measurement. Moreover, the coefficient matrix $\boldsymbol{G}_k$ indicates the correlation of the training data by $\mathcal{K}(\boldsymbol{X}_i, \boldsymbol{X}_i)$. As the value of $\boldsymbol{G}_k$ is influenced by the number of training samples, to eliminate the effects from the size of the data set and normalize the prediction performance of the GP model, the mean of the absolute value for the prediction and measurement error is considered by applying $k^{-1}\mathbf{1}^{\mathrm{T}}$. Therefore, $\varepsilon_i(t_k)$ is a reasonable metric to evaluate the performance of the GP models without heavy variance computation.

However, one must consider that the historical estimation errors can exhibit significant disparities, ranging from scenarios where an agent possesses an ideal prior knowledge resulting in zero error, to instances where an agent's prior information is grossly inaccurate, leading to exceedingly substantial errors. Consequently, there arises a necessity to standardize these errors to a suitable range. In this context, it becomes imperative to normalize them within the interval $[0, 1]$, i.e. the min-max normalization for accumulated historical estimation error, which is expressed in

$$\tilde{\varepsilon}_i(t_k) = \frac{\varepsilon_i(t_k) - \varepsilon_{\min,i}(t_k)}{\varepsilon_{\max,i}(t_k) - \varepsilon_{\min,i}(t_k)}, \qquad (16)$$

where $\varepsilon_{\max,i}(t_k) = \max_{i \in \bar{\mathcal{N}}_i} \varepsilon_i(t_k)$ and $\varepsilon_{\min,i}(t_k) = \min_{i \in \bar{\mathcal{N}}_i} \varepsilon_i(t_k)$. The variable $\tilde{\varepsilon}_i(t_k)$ serves the dual purpose of standardizing error magnitudes and facilitating threshold-based decision making. It not only simplifies the comparison and analysis of errors across different scenarios or datasets but also streamlines the establishment of thresholds for acceptable errors, which is used for designing the elective strategy.

3.2. Prior-Aware Elective Cooperative Learning

Through the incorporation of the quantifiable term $\tilde{\varepsilon}$, we introduce an elective learning function denoted as α_{ij} leveraging the average accumulated historical estimation errors (16). Essentially, it informs us about the degree of trust in the GP models and the circumstances in which the agent i requires calculations for prediction from its neighboring agent $j(j \in \bar{\mathcal{N}}_i)$, including itself. This inclusion is particularly relevant when agent i seeks to calculate predictions at query point $\boldsymbol{x}_i(t_k)$ utilizing its local GP model. The elective function for agent i is designed as

$$\alpha_{ij}\left(t_k, \bar{S}_i\right) = \begin{cases} a_{ij} & \tilde{\varepsilon}_j(t_k) < \bar{\varepsilon}_i(t_k) \\ 0 & \text{otherwise} \end{cases}, \; j \in \bar{\mathcal{N}}_i, \quad (17)$$

where $\bar{\varepsilon}_i(t_k)$ is the $\bar{S}_i$-th largest value of the set $\{\tilde{\varepsilon}_i(t_k)\}_{i \in \bar{\mathcal{N}}_i}$ associated with the agent i. This elective function signifies that the agent i exclusively selects cooperative predictions from a subset $\mathcal{S}_i(t_k) = \{j|\alpha_{ij}(t_k, \bar{S}_i) > 0, j \in \bar{\mathcal{N}}_i\}$ of $\bar{\mathcal{N}}_i$. Specifically, the subset $\mathcal{S}_i$ defines the number of $|\mathcal{S}_i| = |\bar{\mathcal{N}}_i| - \bar{S}_i$ trustworthy agents in the set $\bar{\mathcal{N}}_i$ for the aggregation prediction. Therefore, this elective function affords the agent the capability to determine the number of collaborators, including itself, that it wishes to engage in computing joint inferences for the unknown function.

By employing the proposed elective function in conjunction with the normalized accumulated historical error (16), we formulate the elective prior-aware aggregation weight function for the i-th agent designed by

$$\tilde{\omega}_{ij}^{\varepsilon}(t_k) = \varphi_{ij}\big(\cup_{j \in \bar{\mathcal{N}}_i} \alpha_{ij}(t_k, \bar{S}_i)h^{\varepsilon}(\tilde{\varepsilon}_j(t_k))\big), \; j \in \bar{\mathcal{N}}_i, \qquad (18)$$

which can be simplified as

$$\tilde{\omega}_{ij}^{\varepsilon}(t_k) = \varphi_{ij}\big(\cup_{j \in \mathcal{S}_i} h^{\varepsilon}(\tilde{\varepsilon}_j(t_k))\big), \ j \in \mathcal{S}_i, \qquad (19)$$

according to the definition of $\mathcal{S}_i$, where $\varphi_{ij}(\cdot)$ is a proportional function

$$\varphi_{ij}(\cup_{s \in \mathcal{S}_i} w_{is}) = \frac{w_{ij}}{\sum_{s \in \mathcal{S}_i} w_{is}}. \qquad (20)$$

Since smaller estimation error $\tilde{\varepsilon}_j(t_k)$ indicates more reliable performance for GP model j with larger $\tilde{\omega}_{ij}^{\varepsilon}(t_k)$, the positive function $h^{\varepsilon}(\cdot) : \mathbb{R}_+ \to \mathbb{R}_+$ is designed as monotonically decreasing, i.e., $\forall \tilde{\varepsilon}_i, \tilde{\varepsilon}_j \in \mathbb{R}_+$ with $\forall \tilde{\varepsilon}_i \leq \forall \tilde{\varepsilon}_j$ it holds $h^{\varepsilon}(\forall \tilde{\varepsilon}_i) \geq h^{\varepsilon}(\forall \tilde{\varepsilon}_j)$. Moreover, when $h^{\varepsilon}(\cdot)$ is well-defined, i.e., not tend to be infinite, when the input is close to 0. With the above requirements, the function $h^{\varepsilon}(\cdot)$ can be designed as

$$h^{\varepsilon}(\tilde{\varepsilon}_j(t_k)) = \frac{\sigma_{h_i}\sqrt{2\pi}}{\exp\left(-\frac{1}{2}\left(\frac{\tilde{\varepsilon}_j(t_k) - \bar{\varepsilon}_i(t_k)}{\sigma_{h_i}}\right)^2\right)}, \qquad (21)$$

where the scaling factor $\sigma_{h_i} \in \mathbb{R}_+$ is the standard deviation value of the Gaussian distribution in the denominator of Equation (21). The rationale behind utilizing the function (21) instead of Equation (16) as the weighting scheme lies in the fact that the parameter σ_{h_i} is a trainable variable, affording the flexibility to optimize the distribution of aggregation weights. More importantly, an additional crucial consideration is the necessity to prevent singular values from arising. Notably, the elective aggregation weight function Equation (18) presents an advantageous feature wherein the computation of aggregation weights does not impose a significant computational burden, as these weights are determined based on the prior estimation errors of collaborators. However, it is well-established that the posterior variance in GPR serves as an indicator of prediction uncertainties (Deisenroth & Ng, 2015). This metric quantifies the confidence degree of predictions with respect to the training dataset, as employed in the concept presented in (Yang et al., 2021). In light of this, we incorporate this notion with Equation (20) to design the elective weight based on the variance $\tilde{\omega}_{ij}^{\sigma}(t_k)$ of GP as

$$\tilde{\omega}_{ij}^{\sigma}(t_k) = \varphi_{ij}\big(\cup_{j \in \bar{\mathcal{N}}_i} \alpha_{ij}(t_k, \bar{S}_i) h^{\sigma}(\sigma_j(\boldsymbol{x}(t_k)))\big) \quad (22)$$

where $h^{\sigma}(\bullet) : \mathbb{R} \to \mathbb{R}$ is $\bullet^{-2}$. Therefore, considering both the elective weights (18) and (22), we combine them by using the following method

$$\omega_{ij}(t_k) = \rho(\tilde{\omega}_{ij}^{\varepsilon}(t_k), \tilde{\omega}_{ij}^{\sigma}(t_k)), \qquad (23)$$

where $\rho(\cdot, \cdot) : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is designed as a function that can balance the impact between the aggregation weights based on prior estimation error and the weights based on posteriors.

Remark 3.3. The design of the function $\rho(\cdot)$ is restraint under the condition $\sum_{j \in \bar{\mathcal{N}}_i} \tilde{\omega}_{ij}(t_k) = 1$. A valid choice of ρ can be

$$\rho(\tilde{\omega}_{ij}^{e}(t_k), \tilde{\omega}_{ij}^{\sigma}(t_k)) = \varphi_{ij}\left(\cup_{j \in \bar{\mathcal{N}}_i} \big(\tilde{\omega}_{ij}^{\varepsilon}(t_k)\big)^c \big(\tilde{\omega}_{ij}^{\sigma}(t_k)\big)^{1-c}\right), \tag{24}$$

where $0 \leq c \leq 1 \in \mathbb{R}_{0,+}$ serves as a means to modulate the influence of the first and second input variables in a proportional manner. Consequently, the manipulation of the factor c affords us the ability to finely adjust the relative significance of two key metrics. This choice of ρ guarantees $\sum_{j \in \bar{\mathcal{N}}_i} \omega_{ij}(t_k) = 1$ considering the definition of function φ_{ij} in (20).

It is essential to acknowledge that the aggregation weights in Equation (22) entail increased computational demands on the collaborating agents, along with a higher volume of information exchange to convey the posterior variances. Nevertheless, these adjustments yield a richer source of predictive information from the collaborators. This augmentation has the potential to enhance predictions with Equation (24) under fine-tuned hyperparameters. However, Pri-GP provides a valuable avenue for achieving such adaptability, particularly in situations where computational resources are constrained, considering the calculation of variance infeasible due to its inherent complexity $\mathcal{O}(N^2)$ or resulting in substantial processing delays, circumstances under which the POE method may prove ineffective.

Remark 3.4. Owing to the inherent characteristics of Bayesian learning methodologies, the posterior distribution continually refines itself with the assimilation of additional training data, thereby mitigating the influence of the prior distribution. Nevertheless, our approach offers a broader perspective on quantifying the model's confidence, transcending the limitations of localized query points within the training data domain. Moreover, even in scenarios where all predictive regions have been fully explored and observed, it becomes feasible to set the factor $c = 0$ in Equation (24). The Pri-GP transitions into an elective POE, where the determination of aggregation weights relies solely on posterior variance. When $c = 1$, it signifies that the weighting scheme exclusively relies on the prior-aware aggregation weights in Equation (18). Furthermore, it is pertinent to underscore that expeditious acquisition of aggregation weights can be facilitated by bypassing the computation of variance altogether.

As the aggregation weight function defined above, Equation (2) can be written as

$$\tilde{f}_i(\boldsymbol{x}(t_k)) = \sum_{j \in \bar{\mathcal{N}}_i} \omega_{ij}(t_k)\phi_{ij}\big(\boldsymbol{x}(t_k), \mathbb{D}_j, \hat{f}_j(\boldsymbol{x}(t_k))\big)$$
$$= \sum_{j \in \bar{\mathcal{N}}_i} \omega_{ij}(t_k)\mu_j(\boldsymbol{x}(t_k)), \ i \in \mathcal{V}. \qquad (25)$$

Algorithm 1 Pri-GP Algorithm

Require: $S \geq 2$ {number of agents}
1: **for** $i = 1 : S$ **do**
2:   **for** $j \in \bar{\mathcal{N}}_i$ **do**
3:     $\varepsilon_{ij}(t_k)$, $\tilde{\varepsilon}_{ij}(t_k)$, $\alpha_{ij}(t_k, \bar{S}_i) \leftarrow$ Equation (11), Equation (16), Equation (17)
4:     **if** $\alpha_{ij}(t_k, \bar{S}_i) \neq 0$ **then**
5:       **if** $c = 1$ **then**
6:         $\tilde{\omega}_{ij}^{\varepsilon}(t_k) \leftarrow$ Equation (19)
7:         Calculate $\omega_{ij}(t_k) = \varphi_{ij}(\tilde{\omega}_{ij}^{\varepsilon}(t_k), 1)$
8:       **else if** $c \neq 1$ **then**
9:         $\tilde{\omega}_{ij}^{\sigma}(t_k)$, $\omega_{ij}(t_k) \leftarrow$ Equation (22), Equation (23)
10:       **end if**
11:       Calculate $\phi_{ij}\big(\boldsymbol{x}(t_k), \mathbb{D}_j, \hat{f}_j(\boldsymbol{x}(t_k))\big)$
12:     **end if**
13:   **end for**
14:   $\tilde{f}_i(\boldsymbol{x}(t_k)) \leftarrow$ Equation (25)
15:   **if** $y(t_k) \neq \emptyset$ **then**
16:     $e_i(\boldsymbol{x}(t_k)) \leftarrow$ Equation (10)
17:   **end if**
18: **end for**

Therefore, to obtain the aggregated prediction, the exchanged information necessitates the sharing of two critical components: firstly, the posterior mean for prediction; secondly, the cumulative historical estimation error for the computation of the aggregation weights. To facilitate a better understanding of the algorithm's operation, we provide a pseudo-code representation of Pri-GP in Algorithm 1.

3.3. Prediction with Probabilistic Guarantee

Before analyzing the prediction performance for the MAS with the proposed prior-aware elective distributed learning, we first quantify the prediction error bound for a single GP model with prior information, which is shown in the following lemma.

Lemma 3.5. *For an unknown function satisfying Assumption 2.1, a GP model is given with a training data $\mathbb{D} = \{\boldsymbol{X}, \boldsymbol{Y}\}$ set containing $N = |\mathbb{D}| \in \mathbb{N}$ samples under Assumption 2.2. Moreover, choose the kernel function $\kappa(\cdot)$ satisfying Assumption 2.3 and a Lipschitz continuous prior mean function $\hat{f}(\cdot)$ with Lipschitz constant $L_{\hat{f}} \in \mathbb{R}_+$. Pick the grid factor $\tau \in \mathbb{R}_+$ and $\delta \in (0, 1)$, then the prediction error with prior information is uniformly bounded by*

$$|\mu(\boldsymbol{x}) - f(\boldsymbol{x})| \leq \eta(\boldsymbol{x}) = \sqrt{\beta}\sigma(\boldsymbol{x}) + \gamma\tau, \forall \boldsymbol{x} \in \mathbb{X} \quad (26)$$

with a probability of at least $1 - \delta$, where $\gamma = L_f + L_{\hat{f}} + \sqrt{\beta L_{\sigma^2}\tau} + \sqrt{N}L_k\|\boldsymbol{K}(\boldsymbol{X})^{-1}(\boldsymbol{Y}_k^{\mathrm{T}} - \hat{\boldsymbol{f}}(\boldsymbol{X})^{\mathrm{T}})\|$ and the Lipschitz constant of the posterior variance $L_{\sigma^2} = 2L_{\kappa}(1 + N\|\boldsymbol{K}(\boldsymbol{X})^{-1}\|\max_{x,x' \in \mathbb{X}} k(x, x'))$. The constant $\tilde{\beta} = 2\sum_{j=1}^{m}\log(\frac{\sqrt{m}}{2\tau}(\bar{x}_j - \underline{x}_j) + 1) - 2\log\delta$, where

$\bar{x}_j$ *and* $\underline{x}_j$ *denote the maximum and minimum of the j-th dimension of $\boldsymbol{x}$ in the domain $\mathbb{X}$, i.e., $\bar{x}_j = \max_{\boldsymbol{x} \in \mathbb{X}} x_j$ and $\underline{x}_j = \min_{\boldsymbol{x} \in \mathbb{X}} x_j$.*

Proof. To prove the uniform error bound in $\mathbb{X}$, we first define the discrete domain $\mathbb{X}_{\tau}$ based on the grid factor τ, such that for each element $\boldsymbol{x} \in \mathbb{X}$ there exists an element $\boldsymbol{x}' \in \mathbb{X}_{\tau}$ satisfying $\|\boldsymbol{x} - \boldsymbol{x}'\| < \tau$. The domain $\mathbb{X}_{\tau}$ is finite, whose cardinality is bounded according to (Dai et al., 2023a;b) as $|\mathbb{X}_{\tau}| \leq \prod_{j=1}^{m}\left(\frac{\sqrt{m}}{2\tau}(\bar{x}_j - \underline{x}_j) + 1\right)$. Moreover, employing Lemma 5.1 in (Srinivas et al., 2012), the uniform error bound within $\mathbb{X}_{\tau}$ is written as

$$\Pr\{|\mu(\boldsymbol{x}') - f(\boldsymbol{x}')| \leq \sqrt{\beta}\sigma(\boldsymbol{x}'), \forall \boldsymbol{x}' \in \mathbb{X}_{\tau}\} \geq 1 - \delta \quad (27)$$

considering $\beta = 2\log(|\mathbb{X}_{\tau}|/\delta)$. Then, due to fact that $\|\boldsymbol{x} - \boldsymbol{x}'\| \leq \tau$, the prediction error within the domain $\mathbb{X}$ is bounded by

$$\begin{aligned}|\mu(\boldsymbol{x}) - f(\boldsymbol{x})| &\leq |\mu(\boldsymbol{x}) - \mu(\boldsymbol{x}')| + |f(\boldsymbol{x}) - f(\boldsymbol{x}')| \\ &\quad + |\mu(\boldsymbol{x}') - f(\boldsymbol{x}')| \\ &\leq \sqrt{\beta}\sigma(\boldsymbol{x}') + L_f\tau + |\mu(\boldsymbol{x}) - \mu(\boldsymbol{x}')| \\ &\leq \sqrt{\beta}\sigma(\boldsymbol{x}) + \sqrt{\beta L_{\sigma}\tau} + L_f\tau + |\mu(\boldsymbol{x}) - \mu(\boldsymbol{x}')|, \quad (28)\end{aligned}$$

for all $\boldsymbol{x} \in \mathbb{X}$. While the Lipschitz constant for the posterior mean additionally depends on the prior mean, i.e.,

$$\begin{aligned}|\mu(\boldsymbol{x}) - \mu(\boldsymbol{x}')| &\leq |\hat{f}(\boldsymbol{x}) - \hat{f}(\boldsymbol{x}')| \\ &\quad + |(K(\boldsymbol{x}, \boldsymbol{X}) - K(\boldsymbol{x}', \boldsymbol{X}))\boldsymbol{K}(\boldsymbol{X})^{-1}(\boldsymbol{Y}_k^{\mathrm{T}} - \hat{\boldsymbol{f}}(\boldsymbol{X})^{\mathrm{T}})| \\ &\leq \sqrt{N}L_k\|\boldsymbol{K}(\boldsymbol{X})^{-1}(\boldsymbol{Y}^{\mathrm{T}} - \hat{\boldsymbol{f}}(\boldsymbol{X})^{\mathrm{T}})\|\|\boldsymbol{x} - \boldsymbol{x}'\| \\ &\quad + L_{\hat{f}}\|\boldsymbol{x} - \boldsymbol{x}'\|. \quad (29)\end{aligned}$$

Apply (29) into (28), then the uniform error bound in (26) for $\mathbb{X}$ is derived, which completes the proof. $\square$

Based on Lemma 3.5, we derive the overall prediction error bound of the MAS.

Theorem 3.6. *Consider a MAS with S agents using the Pri-GP algorithm to infer the unknown function $f(\cdot)$ under Assumption 2.1. Equip a GP model on each agent i with the kernel function satisfying Assumption 2.3 and a Lipschitz continuous prior $\hat{f}(\cdot)$ with Lipschitz constant $L_{\hat{f},i}$. Moreover, let each agent has its individual data set $\mathbb{D}_i$ satisfying Assumption 2.2. Pick $\tau \in \mathbb{R}_+$ and $\delta \in (0, 1)$, then the overall prediction error denotes*

$$\|\tilde{\boldsymbol{f}}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})\| \leq \|[\tilde{\eta}_1(\boldsymbol{x}_1), \cdots, \tilde{\eta}_S(\boldsymbol{x}_S)]^T\|, \quad (30)$$

with probability of at least $1 - \sum_{i=1}^{S}|\mathcal{S}_i|\delta$, where the aggregated function and prediction denote $\tilde{\boldsymbol{f}}(\boldsymbol{x}) =$

$[\tilde{f}_1(\boldsymbol{x}_1), \cdots, \tilde{f}_S(\boldsymbol{x}_S)]^T$, $\boldsymbol{f}(\boldsymbol{x}) = [f(\boldsymbol{x}_1), \cdots, f(\boldsymbol{x}_S)]^T$, *and*

$$\tilde{\eta}_i(\boldsymbol{x}_i) = \frac{\sum_{j \in \mathcal{S}_i} h_\varepsilon(\tilde{\varepsilon}_j(t_k))^c h_\sigma(\sigma_j(\boldsymbol{x}(t_k))^{1-c} \eta_j(\boldsymbol{x}_i)}{\sum_{j \in \mathcal{S}_i} h_\varepsilon(\tilde{\varepsilon}_j(t_k))^c h_\sigma(\sigma_j(\boldsymbol{x}(t_k))^{1-c}}.$$
(31)

Proof. Due to the property of function $\varphi(\cdot)$, the prediction error for the i-th agent is written as

$$|\tilde{f}_i(\boldsymbol{x}_i(t_k)) - f(\boldsymbol{x}_i(t_k))|$$
$$= \left| \sum_{j \in \bar{\mathcal{N}}_i} \omega_{ij}(t_k) \big(\mu_i(\boldsymbol{x}(t_k)) - f(\boldsymbol{x}_i(t_k)) \big) \right|$$
$$\leq \sum_{j \in \bar{\mathcal{N}}_i} \omega_{ij}(t_k) |\mu_j(\boldsymbol{x}(t_k)) - f(\boldsymbol{x}_i(t_k))|$$
$$\leq \sum_{j \in \bar{\mathcal{N}}_i} \omega_{ij}(t_k) \eta_j(\boldsymbol{x}_i(t_k)),$$
(32)

where the second inequality is derived from Lemma 3.5. Consider the definition of $\omega_{ij}(t_k)$ in (23), the aggregation weight is rewritten as

$$\omega_{ij}(t_k) = \frac{(\tilde{\omega}_{ij}^e(t_k))^c (\tilde{\omega}_{ij}^\sigma(t_k))^{1-c}}{\sum_{s \in \bar{\mathcal{N}}_i} (\tilde{\omega}_{is}^e(t_k))^c (\tilde{\omega}_{is}^\sigma(t_k))^{1-c}}$$
(33)

With (18) and (22), one has

$$(\tilde{\omega}_{ij}^e(t_k))^c (\tilde{\omega}_{ij}^\sigma(t_k))^{1-c}$$
(34)
$$= \frac{\alpha_{ij}(t_k, \bar{S}_i) h_\varepsilon(\tilde{\varepsilon}_j(t_k))^c h_\sigma(\sigma_j(\boldsymbol{x}(t_k)))^{1-c}}{\left(\sum_{s \in \mathcal{S}_i} h_\varepsilon(\tilde{\varepsilon}_j(t_k)) \right)^c \left(\sum_{s \in \mathcal{S}_i} h_\sigma(\sigma_j(\boldsymbol{x}(t_k))) \right)^{1-c}}.$$

Apply (34) into (32), then the result in (31) is derived with the probability of at least $1 - |\mathcal{S}_i|\delta$ using union bound. Moreover, employing union bound again for different agents, the overall prediction error bound in (30) is obtained. □

4. Numerical Evaluation

To effectively elucidate the efficacy of our proposed algorithms, we commence by employing an approximated sine function as a demonstrative vehicle expounding upon the fundamental principles of Pri-GP[1], as explicated in Section 4.1. Furthermore, we showcase the proficiency of our novel algorithms in identifying the dynamics characterizing autonomous systems in Section 4.2.

4.1. Function Approximation

In this subsection, we investigate the MAS comprising 4 agents, each equipped with an identical dataset but possessing distinct prior knowledge represented as $\hat{f}$. The rationale

[1]The code is available at https://github.com/Zewen-Yang/Pri-GP.

behind this experiment is to facilitate an in-depth analysis of the impact of varying prior knowledge on predictions when employing an individual learning strategy. Furthermore, we aim to draw comparisons with different distributed learning methodologies showing the proposed algorithms are superior to others.

The target function for approximation in this scenario is chosen as $sin(2x)$. We set the prior function of agent 1 to $\hat{f}_1(x) = 0$ considering the agent does not have any knowledge of the unknown function, which is a general assumption. Moreover, let agent 3 have the accurate function $\hat{f}_3(x) = sin(2x)$ as the target function and the 2-nd agent and the 4-th agent as $\hat{f}_2(x) = -1$ and $\hat{f}_4(x) = cos(2x)$, respectively. The adjacency matrix of the communication graph of this MAS is

$$\mathcal{A} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}.$$

Let $\bar{S}_1 = \bar{S}_2 = \bar{S}_3 = \bar{S}_4 = 2$, and the 8 identical training input are randomly selected obeying uniform distribution over the range $[0, 2\pi)$. And set the hyperparameters of the kernel function are chosen as $\sigma_r = 1, l_j = 0.2, j = 1, 2, 3$ and the noise variance of the noise is $\sigma_n = 0.1$ for all agents. The curves presented in Figure 1 show the results of function approximation. It becomes apparent that the curve using individual learning with Gaussian Processes (IGP), i.e., the agent predicts the unknown function independently, closely approximates the prior function in the absence of training data. This observation underscores the significant influence of prior knowledge on predictions when no data are available.

To facilitate a more nuanced comparison of our proposed methods with existing approaches, we provide the prediction error, denoted by $\Delta e = |\tilde{f}(x) - f(x)|$, of all agents with violin plots in Figure 2 to analyze the distribution of prediction errors. Furthermore, the average of the 1000 prediction errors regarding each agent is illustrated Table 1. It is evident that Pri-GP methods outperform the other methods, which have the lowest sum of average prediction errors. Notably, while the overall prediction errors for the MAS letting $c \neq 1$ may appear less favorable when compared to $c = 1$, a closer examination reveals that agent 2, in particular, benefits from Pri-GP with $c = 0.5$. The BCM method manifests heightened errors that can be attributed to that BCM tends to disproportionately accentuate the influence of prior variance within the aggregation weights. Moreover, it is crucial to note that the similarity in results of the POE, POE, and GPOE methods arises from the identical training datasets employed by all agents. In order to explore the impact of distinct training datasets, a more intricate scenario is examined in Section 4.2.

Table 1. Average prediction errors ($\times 10^{-2}$).

Methods	Agent 1	Agent 2	Agent 3	Agent 4
Pri-GP ($c=1$)	**0.044**	22.25	0.048	**0.032**
Pri-GP ($c=\frac{1}{2}$)	0.734	**22.13**	0.515	0.958
POE	16.64	24.24	17.83	20.87
GPOE	16.64	24.24	17.83	20.87
BCM	38.48	41.12	37.38	45.09
RBCM	22.26	25.31	21.08	23.11
MOE	16.64	24.24	17.82	20.87
IGP	22.26	42.70	**0.002**	43.65

Figure 2. Violin plots of prediction errors for different methods. The red line is the mean value and the top/bottom horizontal blue bar is the maximal/minimal value.
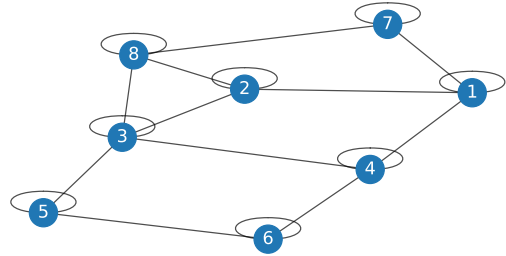

Figure 1. True, prior and posterior curves.

Figure 3. Communication graph.

4.2. Dynamical System Identification

To further demonstrate the capability of Pri-GP in the identification of dynamical systems, this subsection endeavors to exemplify the performance of Pri-GP with a 3 dimensional nonlinear system

$$
\begin{aligned}
\dot{x} &= s(y-x), \\
\dot{y} &= rx - y - xz, \\
\dot{z} &= xy \underbrace{-10\sin(z) - 10x - 0.5(1+\exp(-xy/10))^{-1}}_{f(\chi)},
\end{aligned}
$$

where we assume the unknown component of the system as represented by $f(\chi) : \mathbb{R}^3 \to \mathbb{R}$, where $\chi = [x, y, z]^2$. The MAS comprises 8 agents, each equipped with unique datasets with 100 training data satisfying the conditions specified in Assumption 2.2 randomly distributed in the space $[-10, 20] \times [-25, 30] \times [0, 60]$ (see Figure 4a). All agents begin in the same initial states, which are randomly determined within the range $[0, 1]$. These simulations are conducted 100 times for Monte-Carlo simulations, with a time step of 0.01, and each simulation continues for 150 time steps. The hyperparameters of the kernel function are chosen as $\sigma_r = 1, l_j = 1000, j = 1, 2, 3$ and the noise variance of the noise is $\sigma_n = 0.1$. Furthermore, the selection

---

[2] For additional results of the simulations, refer to Appendix A.

of diverse prior functions in the MAS are as follows

$$
\begin{aligned}
\hat{f}_1(\chi) &= -10\sin(z) - 10x - 0.5(1+\exp(-xy/10))^{-1}, \\
\hat{f}_2(\chi) &= 0, \\
\hat{f}_3(\chi) &= -10\sin(z) - 10x - 0.5(1+\exp(-xy/10))^{-1}, \\
\hat{f}_4(\chi) &= -10\sin(z), \\
\hat{f}_5(\chi) &= -10x, \\
\hat{f}_6(\chi) &= 10y - 0.5(1+\exp(-xy/10))^{-1}, \\
\hat{f}_7(\chi) &= -0.5(1+\exp(-xy/10))^{-1}, \\
\hat{f}_8(\chi) &= -10\cos(z).
\end{aligned}
$$

The communication graph $\mathcal{G}$ of this MAS is shown in Figure 3, and similar to Section 4.1, each agent discharges one neighbor leading $\bar{S}_1 = \bar{S}_3 = 4, \bar{S}_2 = \bar{S}_4 = \bar{S}_5 = \bar{S}_8 = 3$, and $\bar{S}_6 = \bar{S}_7 = 2$. The trajectories of the dynamical system using POE and Pri-GP ($c = 0.5$) for one trail are depicted showcasing the spatial states of agents in Figure 4b, where the initial state is $[0, 1, 1.05]$. In the case of POE, there is no agent that closely follows the true system trajectory, while Pri-GP ($c = 0.5$) enables all agents to accurately identify the system dynamics. Only the trajectory of agent 8 slightly differs from the trajectory of the real system. This difference in performance can be attributed to Pri-GP having an
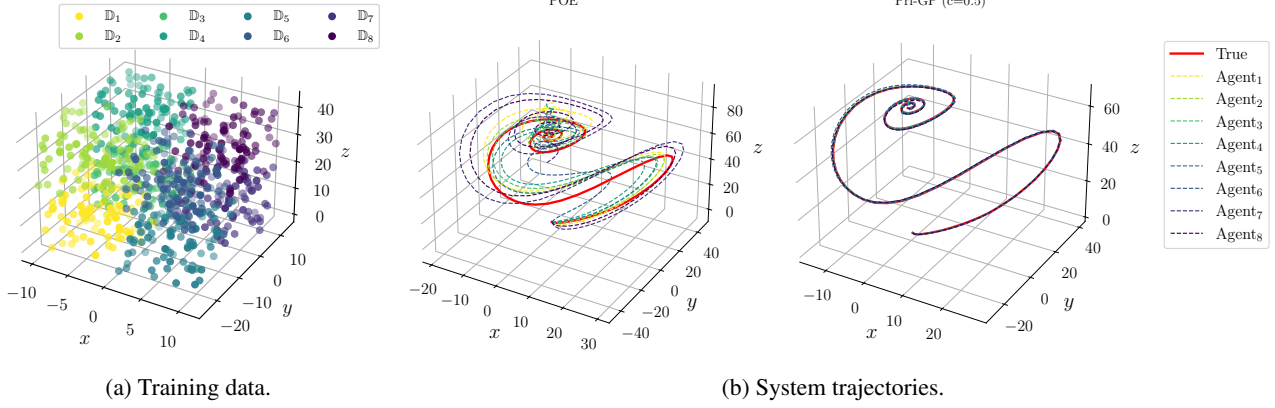
(a) Training data.

(b) System trajectories.

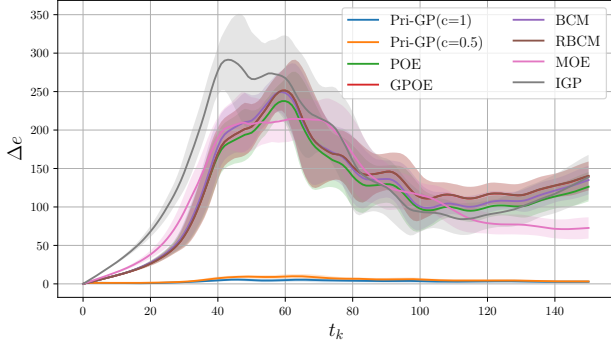*Figure 4.* 3 Dimension Plots.

Figure 5. Mean prediction error with standard deviation.
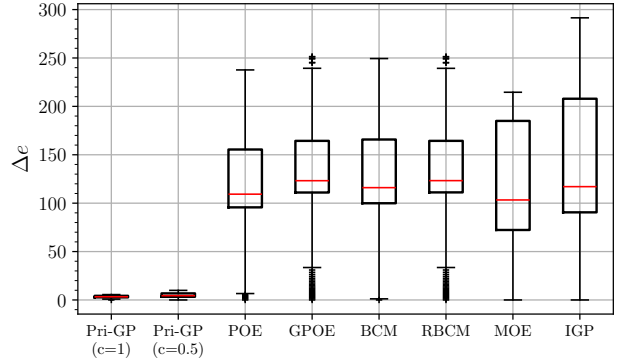

Figure 6. The box plots of mean prediction errors.

inferior prior function and different neighbors. Additionally, it may be influenced by the fact that the training data points of agent 8 are barely aligned with the true trajectory.

Table 2. Average values of minimum, mean, median, and maximum over the 150 iterations.

Methods	Min.	Median	Mean	Max.
Pri-GP ($c=1$)	**0.884**	**2.974**	**3.277**	**8.996**
Pri-GP ($c=\frac{1}{2}$)	2.331	4.882	5.104	10.46
POE	72.47	112.82	114.93	178.74
GPOE	79.91	122.60	124.33	189.42
BCM	77.80	118.75	121.42	190.68
RBCM	79.96	122.66	124.40	189.50
MOE	75.73	114.02	116.31	181.60
IGP	87.96	137.18	141.10	236.12

Figure 5 presents a comparative analysis demonstrating the superior performance of the two Pri-GP methods across the entire experimental process. The solid lines represent the mean predictions obtained from 100 simulations, while the light-shaded areas denote the standard deviation for each

method. It is evident that all alternative approaches exhibit comparable large prediction errors. Furthermore, the box-plots illustrating the mean prediction errors are presented in Figure 6, wherein the red lines represent the mean values, and the $+$ symbols denote outlier points. Table 2 provides a summary of average values, including minimum, mean, median, and maximum, derived from 150 iterations. Notably, Pri-GP demonstrates optimal performance when the parameter c is set to 1, with $c = 0.5$ emerging as the second-best option among the alternatives, which underscores the effectiveness of the proposed algorithm in inferring unknown functions for dynamical systems.

5. Conclusion

In summary, Pri-GP emerges as a robust and promising solution for enhancing distributed cooperative learning within MASs. It introduces a novel approach that not only significantly improves prediction accuracy but also addresses the computational burden is distributed GPR by empowering agents to selectively request predictions from trusted neighbors. It offers several advantages, including improved

prediction accuracy, reduced computational complexity, and the establishment of prediction error bounds, making it a valuable tool for applications where trustworthiness and reliability are paramount. The simulation results support the efficacy of Pri-GP, underscoring its superiority over existing methods in various scenarios, thus validating its potential utility for advancing the capabilities of MASs across a spectrum of domains, from safety-critical systems to resource-efficient distributed networks.

References

Alotaibi, E. T., Alqefari, S. S., and Koubaa, A. Lsar: Multi-uav collaboration for search and rescue missions. *IEEE Access*, 7:55817–55832, 2019. doi: 10.1109/ACCESS. 2019.2912306.

Cao, Y. and Fleet, D. J. Generalized product of experts for automatic and principled fusion of gaussian process predictions, 2015.

Chowdhury, S. R. and Gopalan, A. On Kernelized Multi-armed Bandits. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 844–853. PMLR. URL https://proceedings.mlr.press/v70/chowdhury17a.html.

Dai, H., Xie, J., and Chen, W. Event-triggered distributed cooperative learning algorithms over networks via wavelet approximation. *Neural Process. Lett.*, 50(1): 669–700, aug 2019. ISSN 1370-4621. doi: 10.1007/s11063-019-10031-x. URL https://doi.org/10.1007/s11063-019-10031-x.

Dai, S.-L., He, S., Ma, Y., and Yuan, C. Distributed cooperative learning control of uncertain multiagent systems with prescribed performance and preserved connectivity. *IEEE Transactions on Neural Networks and Learning Systems*, 32(7):3217–3229, 2021. doi: 10.1109/TNNLS. 2020.3010690.

Dai, X., Lederer, A., Yang, Z., and Hirche, S. Can learning deteriorate control? analyzing computational delays in gaussian process-based event-triggered online learning. In Matni, N., Morari, M., and Pappas, G. J. (eds.), *Proceedings of The 5th Annual Learning for Dynamics and Control Conference*, volume 211 of *Proceedings of Machine Learning Research*, pp. 445–457. PMLR, 15–16 Jun 2023a. URL https://proceedings.mlr.press/v211/dai23a.html.

Dai, X., Lederer, A., Yang, Z., and Hirche, S. Gaussian Process-Based Event-Triggered Online Learning with Computational Delays for Control of Unknown Systems: Extended Version. 2023b.

Dai, X., Yang, Z., and Hirche, S. Cooperative online learning for multi-agent system control via gaussian processes with event-triggered mechanism: Extended version, 2024a. URL https://arxiv.org/abs/2304.05138v2.

Dai, X., Yang, Z., Xu, M., Liu, F., Hattab, G., and Hirche, S. Decentralized event-triggered online learning for safe consensus of multi-agent systems with gaussian process regression, 2024b. URL https://arxiv.org/abs/2402.03174.

Deisenroth, M. and Ng, J. W. Distributed gaussian processes. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1481–1490, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/deisenroth15.html.

Gao, F., Chen, W., Li, Z., Li, J., and Xu, B. Neural network-based distributed cooperative learning control for multiagent systems via event-triggered communication. *IEEE Transactions on Neural Networks and Learning Systems*, 31(2):407–419, 2020. doi: 10.1109/TNNLS. 2019.2904253.

Jafari, M. and Xu, H. Intelligent control for unmanned aerial systems with system uncertainties and disturbances using artificial neural network. *Drones*, 2(3), 2018. ISSN 2504-446X. doi: 10.3390/drones2030030. URL https://www.mdpi.com/2504-446X/2/3/30.

Khalil, H. K. *Nonlinear Systems*. Prentice-Hall, 2002.

Kim, H., Chang, H., and Shim, H. Model Reference Gaussian Process Regression: Data-Driven State Feedback Controller. URL http://arxiv.org/abs/2303.09828.

Lederer, A., Umlauft, J., and Hirche, S. Uniform error bounds for gaussian process regression with application to safe control. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/fe73f687e5bc5280214e0486b273a5f9-Paper.pdf.

Lederer, A., Yang, Z., Jiao, J., and Hirche, S. Cooperative control of uncertain multiagent systems via distributed gaussian processes. *IEEE Transactions on Automatic Control*, 68(5):3091–3098, 2023. doi: 10.1109/TAC.2022. 3205424.

Liu, H., Cai, J., Wang, Y., and Ong, Y. S. Generalized robust Bayesian committee machine for large-scale Gaussian process regression. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3131–3140. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/liu18a.html.

Maddalena, E. T., Scharnhorst, P., and Jones, C. N. Deterministic error bounds for kernel-based learning techniques under bounded noise. 134:109896. ISSN 0005-1098. doi: 10.1016/j.automatica.2021.109896. URL https://www.sciencedirect.com/science/article/pii/S0005109821004192.

Mchutchon, A. and Rasmussen, C. Gaussian Process Training with Input Noise. In *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper_files/paper/2011/hash/a8e864d04c95572d1aece099af852d0a-Abstract.html.

Provost, F. J. and Hennessy, D. N. Scaling up: distributed machine learning with cooperation. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 1*, AAAI'96, pp. 74–79. AAAI Press, 1996. ISBN 026251091X.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9.

Schürch, M., Azzimonti, D., Benavoli, A., and Zaffalon, M. Recursive estimation for sparse gaussian process regression. *Automatica*, 120:109127, 2020. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2020.109127. URL https://www.sciencedirect.com/science/article/pii/S0005109820303253.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. W. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012. doi: 10.1109/TIT.2011.2182033.

Stirling, T., Roberts, J., Zufferey, J.-C., and Floreano, D. Indoor navigation with a swarm of flying robots. In *2012 IEEE International Conference on Robotics and Automation*, pp. 4641–4647, 2012. doi: 10.1109/ICRA.2012.6224987.

Tresp, V. A Bayesian Committee Machine. *Neural Computation*, 12(11):2719–2741, 11 2000a. ISSN 0899-7667. doi: 10.1162/089976600300014908. URL https://doi.org/10.1162/089976600300014908.

Tresp, V. Mixtures of gaussian processes. In Leen, T., Dietterich, T., and Tresp, V. (eds.), *Advances in Neural Information Processing Systems*, volume 13. MIT Press, 2000b. URL https://proceedings.neurips.cc/paper_files/paper/2000/file/9fdb62f932adf55af2c0e09e55861964-Paper.pdf.

W. Wang, D. W. and Peng, Z. H. Cooperative learning neural network output feedback control of uncertain nonlinear multi-agent systems under directed topologies. *International Journal of Systems Science*, 48(12):2590–2598, 2017. doi: 10.1080/00207721.2017.1324923. URL https://doi.org/10.1080/00207721.2017.1324923.

Wang, W., Yue, X., Haaland, B., and Jeff Wu, C. Gaussian processes with input location error and applications to the composite parts assembly process. *SIAM/ASA Journal on Uncertainty Quantification*, 10(2):619–650, 2022.

Yan, Z., Yang, Z., Yue, L., Wang, L., Jia, H., and Zhou, J. Discrete-time coordinated control of leader-following multiple auvs under switching topologies and communication delays. *Ocean Engineering*, 172:361–372, 2019. ISSN 0029-8018. doi: https://doi.org/10.1016/j.oceaneng.2018.12.018. URL https://www.sciencedirect.com/science/article/pii/S0029801818308667.

Yan, Z., Yang, Z., Pan, X., Zhou, J., and Wu, D. Virtual leader based path tracking control for multi-uuv considering sampled-data delays and packet losses. *Ocean Engineering*, 216:108065, 2020. ISSN 0029-8018. doi: https://doi.org/10.1016/j.oceaneng.2020.108065. URL https://www.sciencedirect.com/science/article/pii/S0029801820310076.

Yang, Z., Sosnowski, S., Liu, Q., Jiao, J., Lederer, A., and Hirche, S. Distributed learning consensus control for unknown nonlinear multi-agent systems based on gaussian processes. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pp. 4406–4411, 2021. doi: 10.1109/CDC45484.2021.9683522.

Yang, Z., Dai, X., Dubey, A., Hirche, S., and Hattab, G. Whom to Trust? Elective Learning for Distributed Gaussian Process Regression, 2024a. URL https://arxiv.org/abs/2402.03014.

Yang, Z., Dong, S., Lederer, A., Dai, X., Chen, S., Sosnowski, S., Hattab, G., and Hirche, S. Cooperative Learning with Gaussian Processes for Euler-Lagrange Systems Tracking Control under Switching Topologies, 2024b. URL https://arxiv.org/abs/2402.03048.

Yin, Z., Dai, X., Yang, Z., Shen, Y., Hattab, G., and Zhao, H. Learning-based control for pmsm using distributed gaussian processes with optimal aggregation strategy. In *IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society*, pp. 1–7, 2023. doi: 10.1109/IECON51785.2023.10312503.
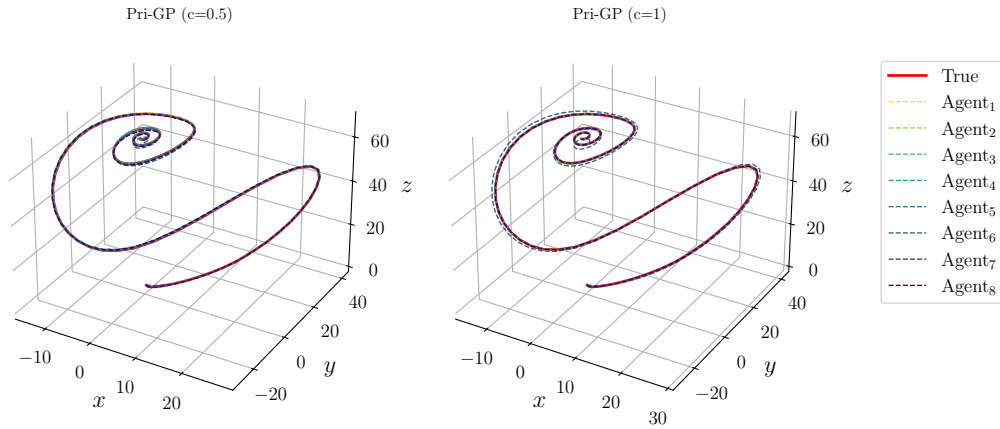
A. Additional Results

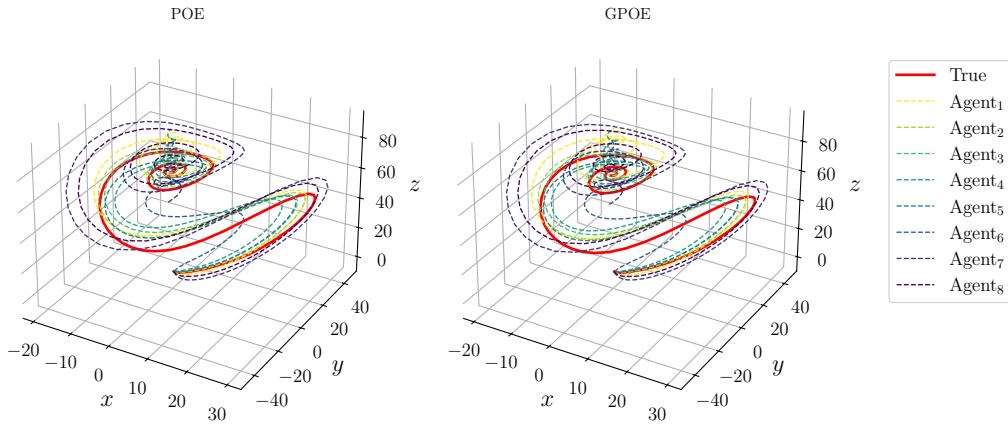*Figure 7.* System trajectories of Pri-GP ($c = 0.5$) and Pri-GP ($c = 1.0$).
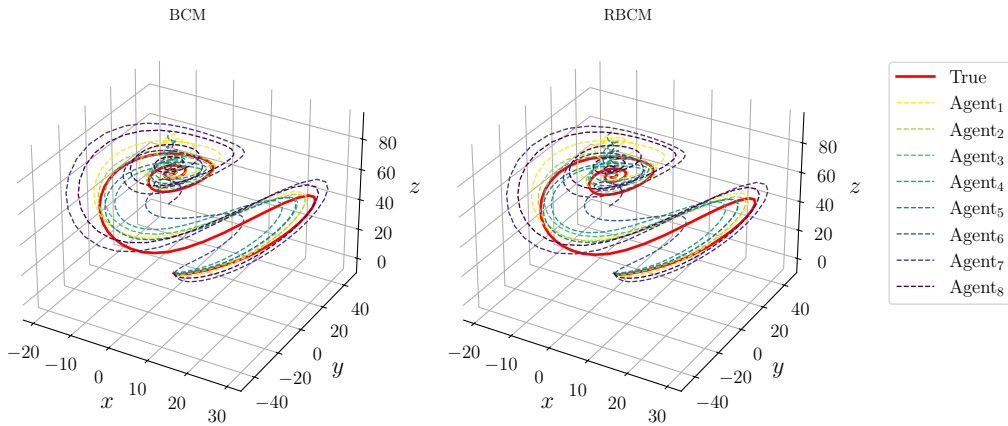

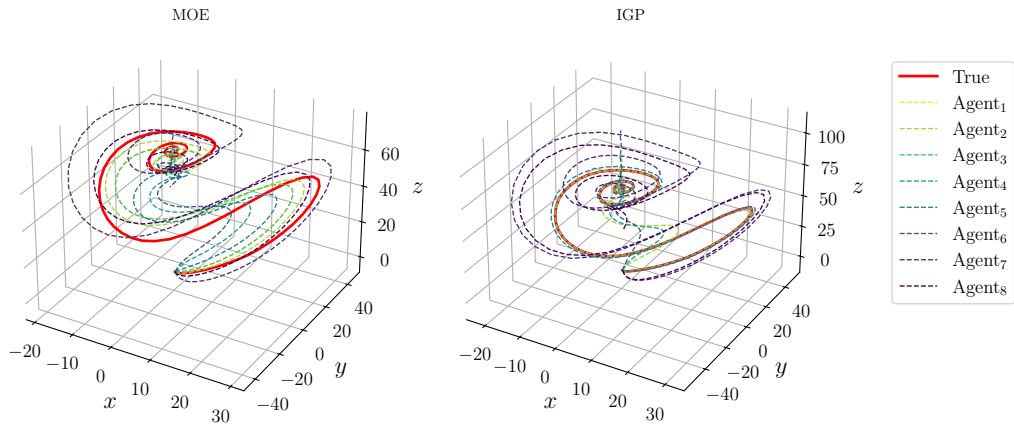*Figure 8.* System trajectories of POE and GPOE.

Figure 9. System trajectories of BCM and RBCM.

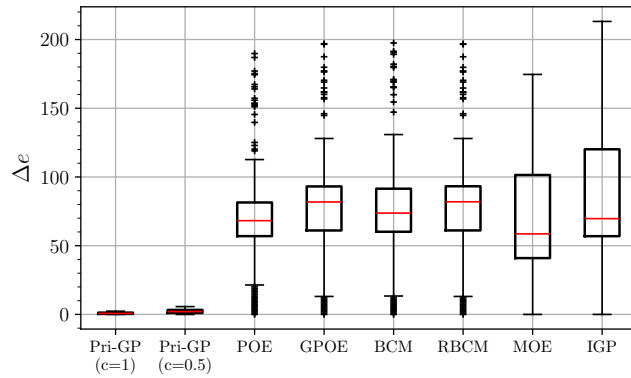*Figure 10.* System trajectories of MOE and IGP.

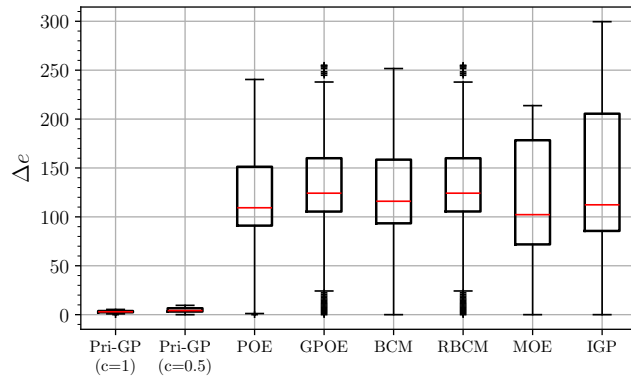*Figure 11.* The box plots of minimum prediction errors.

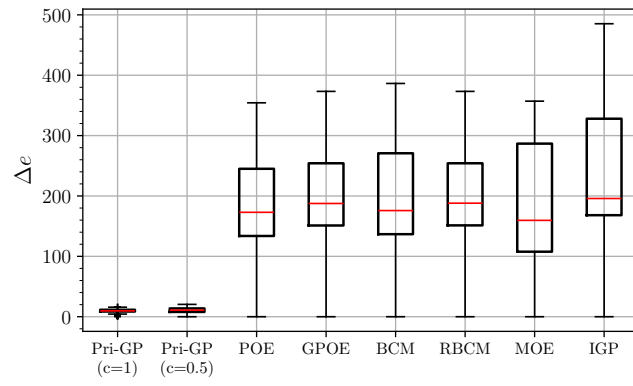
Figure 12. The box plots of median prediction errors

Figure 13. The box plots of maximum prediction errors.