IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 23, 2022, accepted February 21, 2022, date of publication February 24, 2022, date of current version March 8, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3154095

Actively Learning Gaussian Process Dynamical Systems Through Global and Local Explorations

SHENGBING TANG, KENJI FUJIMOTO, (Member, IEEE), AND ICHIRO MARUTA, (Member, IEEE)

Department of Aeronautics and Astronautics, Kyoto University, Kyoto 615-8540, Japan

Corresponding author: Kenji Fujimoto (k.fujimoto@ieee.org)

This work was supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant 21H01351.

ABSTRACT Usually learning dynamical systems by data-driven methods requires large amount of training data, which may be time consuming and expensive. Active learning, which aims at choosing the most informative samples to make learning more efficient is a promising way to solve this issue. However, actively learning dynamical systems is difficult since it is not possible to arbitrarily sample the state-action space under the constraint of system dynamics. The state-of-the-art methods for actively learning dynamical systems iteratively search for an informative state-action pair by maximizing the differential entropy of the predictive distribution, or iteratively search for a long informative trajectory by maximizing the sum of predictive variances along the trajectory. These methods suffer from low efficiency or high computational complexity and memory demand. To solve these problems, this paper proposes novel and more sample-efficient methods which combine global and local explorations. As the global exploration, the agent searches for a relatively short informative trajectory in the whole state-action space of the dynamical system. Then, as the local exploration, an action sequence is optimized to drive the system's state towards the initial state of the local informative trajectory found by the global exploration and the agent explores this local informative trajectory. Compared to the state-of-the-art methods, the proposed methods are capable of exploring the state-action space more efficiently, and have much lower computational complexity and memory demand. With the state-of-the-art methods as baselines, the advantages of the proposed methods are verified via various numerical examples.

INDEX TERMS Active learning, dynamical system, Gaussian process, global and local explorations.

I. INTRODUCTION AND RELATED WORK

The acquisition of accurate models of dynamical systems [1], [2] is essential for many applications, such as controller design and model-based reinforcement learning. If the accurate analytic models are hard to be derived from the first principles due to the high complexity of the dynamical systems, the data-driven learning method will be a useful alternative. The Gaussian process (GP) has been commonly used to learn dynamical systems from training data because of its advantageous properties [3]–[5], such as working well with little training data, providing a measure of the uncertainty about the estimated model, being able to incorporate prior knowledge by the mean or kernel function.

Typically the data-driven learning of dynamical systems requires large amount of training data, which may be time consuming and cause system wear. Active learning [6] can be seen as a strategy for optimal data selection to make learning more efficient. At first the literature related to actively learning the unknown function is introduced. These literature assume that it is possible to query any point in the input space of the unknown function. The works in [7] and [8] actively learn the spatial phenomena using GPs which can be formulated as the problem of selecting the optimal sensor locations from a finite set of possible locations. The optimal sensor locations are obtained by maximizing the differential entropy or the mutual information, which has been shown to be NP-hard. Instead of simultaneously optimizing all the sensor locations, an approximate algorithm which sequentially selects the sensor location is proposed, and a theoretical bound which quantifies the advantage of the active learning strategy over a priori design strategy is provided. The work in [9] researches the problem of actively learning complex physical systems like buildings using the fully Bayesian GP.

The associate editor coordinating the review of this manuscript and approving it for publication was Seifedine Kadry.

At each iteration, the next sampling point is chosen by maximizing the information gain, namely reducing the maximum uncertainty in the hyperparameters of the GP.

Actively learning dynamical systems is fundamentally different from actively learning unknown functions due to the fact that it is not possible to arbitrarily sample the state-action space under the constraint of system dynamics. Taking the inverted pendulum system as an example, initially the pendulum hangs down, thus the state corresponding to the swing-up is very informative. To visit the swing-up state, an action sequence which could drive the system's state from the initial state to the swing-up state should be obtained at first.

An algorithm for actively exploring the state-action space of the dynamical system is proposed in [10]. The dynamical system is modeled by the Bayesian linear regression. At each iteration, at first an informative action sequence is optimized by maximizing the sum of predictive variances along the trajectory or minimizing the differential entropy of the posterior distribution of the model parameters. Then the obtained informative action sequence is applied to the true system to collect samples. If the corresponding optimization problem can be solved successfully, this method will work well because it takes the sample efficiency along the trajectory into account. However, when the horizon is relatively large, solving the optimization problem is quite difficult and the computational complexity and memory demand are very high because the objective function (sum of predictive variances or differential entropy of the posterior distribution of the model parameters) is too complicated. Furthermore, the Bayesian linear regression is not expressive enough to model highly nonlinear dynamical systems. The work in [11] proposes two methods for actively learning dynamical systems. The first method searches for an informative action sequence by maximizing the sum of differential entropies of predictive distributions along the trajectory at each iteration, which is similar to the method in [10] and suffers from the same problems. The second method extends the ideas of actively learning unknown functions proposed in [7] and [8] to dynamical systems which are modeled by GPs. At each iteration, at first an informative state-action pair is selected by maximizing the differential entropy of the predictive distribution. Then based on the currently estimated model an action sequence is optimized to steer the system's state from the initial state to an informative state. This method is able to visit informative states which are far away from the initial state and difficult to be visited using random actions. But it is very inefficient since it focuses on exploring a single informative state-action pair each time.

Active exploration is also an important issue in reinforcement learning. The work in [12] augments the objective (expected sum of rewards) with the expected entropy of the policy to encourage exploration. The work in [13] realizes active exploration through augmenting the reward function with the information gain which is defined over the posterior distribution of the parameters of the environment model.

As a global optimization method, Bayesian optimization (BO) [14] can be used to search for the optimal controller parameter by maximizing a performance objective. To reduce the number of interactions with the true system, BO actively explores the space of the controller parameter and iteratively selects the controller parameter to be evaluated. The work in [15] uses BO to learn the gait parameter of a planar biped robot. The map from the controller parameter to the performance objective is model by a GP. At each iteration, the next querying controller parameter is chosen by maximizing the acquisition function.

This paper proposes novel methods to actively learn the dynamical systems which are modeled by GPs. The proposed methods combine global and local explorations. As the global exploration, the agent searches for a relatively short informative trajectory in the whole state-action space of the dynamical system. Then, as the local exploration, an action sequence is optimized to drive the system's state towards the initial state of the local informative trajectory found by the global exploration and the agent explores this local informative trajectory. By focusing on exploring an informative trajectory instead of a single informative point [11] at each iteration, the proposed methods are capable of exploring the state-action space more efficiently and have faster convergence rates. Instead of optimizing a long informative trajectory starting from the initial state [10], [11], the proposed methods search for a relatively short informative trajectory. As a result, the proposed methods have much lower computational complexity and memory demand compared with the methods in [10] and [11] because the corresponding optimization problems are much easier to solve. With the existing methods in the literature as baselines, the advantage of the proposed methods is verified on three dynamical systems.

II. PROBLEM FORMULATION

Consider a discrete-time dynamical system:

$$x_{t+1} = f(x_t, u_t), \tag{1}$$

with a continuous-valued state $x \in \mathbb{R}^{d_x}$, a continuous-valued action $u \in \mathbb{R}^{d_u}$ and the unknown transition dynamics f to be learned.

A. GAUSSIAN PROCESS

The system function f is modeled by a GP, with the current state-action pair $z_t = (x_t, u_t) \in \mathbb{R}^{d_x+d_u}$ as the training input, the consecutive state x_{t+1} as the training output. A GP model is completely specified by the mean function $m(\cdot)$ and the kernel function $k(\cdot, \cdot)$ [16]. In this paper, the zero prior mean function $m \equiv 0$ and the squared exponential kernel function k are used:

$$k(z_i, z_j) = \sigma_f^2 \exp\left(-\frac{1}{2}(z_i - z_j)^{\mathrm{T}} \Lambda^{-1}(z_i - z_j)\right) + \sigma_n^2 \delta_{ij}, \tag{2}$$

with $\Lambda = \mathrm{diag}([l_1^2, l_2^2, \ldots, l_{d_x+d_u}^2])$ being the characteristic length-scales, σ_f^2 being signal variance, σ_n^2 being noise variance. The item δ_{ij} is a Kronecker delta which is one iff $i = j$ and zero otherwise. Given a trajectory $\{x_0, u_0, x_1, u_1, \ldots, x_{N-1}, u_{N-1}, x_N\}$, the training inputs and outputs for the GP are collected as $X = \{(x_0, u_0), \ldots, (x_{N-1}, u_{N-1})\}$, $Y = \{x_1, \ldots, x_N\}$ respectively. With the training data (X, Y), the hyperparamters $\{\sigma_f^2, \Lambda, \sigma_n^2\}$ are optimized by maximizing the log marginal likelihood [16], [17]:

$$\begin{aligned}
&\log p(Y|X, \sigma_f^2, \Lambda, \sigma_n^2) \\
&= \int p(Y|\mathbf{f}, X)p(\mathbf{f}|X)\mathrm{d}\mathbf{f} \\
&= -\frac{1}{2}Y^{\mathrm{T}}\left[K(X,X) + \sigma_n^2\mathrm{I}\right]^{-1} Y \\
&\quad -\frac{1}{2}\log|K(X,X) + \sigma_n^2\mathrm{I}| - \frac{N}{2}\log 2\pi, \qquad (3)
\end{aligned}$$

with $\mathbf{f} = [f(z_0), f(z_1), \ldots, f(z_{N-1})]^{\mathrm{T}}$. For multiple outputs, an independent GP is used for each output dimension. At the testing input z, the predictive distribution of the corresponding function value $f(z)$ is Gaussian,

$$p(f(z)|X, Y, z) = \mathcal{N}(\mu(z), \sigma^2(z)), \qquad (4)$$

where the mean and variance are given by:

$$\mu(z) = K(z, X)\left[K(X,X) + \sigma_n^2\mathrm{I}\right]^{-1} Y, \qquad (5)$$

$$\sigma^2(z) = k(z, z) - K(z, X)\left[K(X,X) + \sigma_n^2\mathrm{I}\right]^{-1} K(X, z). \qquad (6)$$

The GPs are trained using the package GPflow [18].

B. INFORMATION CRITERION

Active learning aims at selecting samples which are maximally informative about the dynamical system. As a result, accurate models of the dynamical systems can be obtained with fewer samples. To select informative samples, it is necessary to define some criteria which could quantify how informative the samples are. There are two common information-based criteria. The first one is the differential entropy of the predictive distribution of the function value $f(z)$ at the input z. Since each dimension of the system function f is modeled by an independent GP, the predictive distribution of $f(z)$ at the input z is multivariate Gaussian distribution:

$$p(f(z)|X, Y, z) = \mathcal{N}(\mu, \Sigma), \qquad (7)$$

where $\mu = [\mu_1, \ldots, \mu_{d_x}]^{\mathrm{T}}$ and $\Sigma = \mathrm{diag}([\sigma_1^2, \ldots, \sigma_{d_x}^2])$ are predictive means (5) and variances (6) of all dimensions respectively. The differential entropy of the multivariate Gaussian distribution is calculated by:

$$\begin{aligned}
H[f(z)] &= -\int p(f(z)) \ln p(f(z))\mathrm{d}f(z) \\
&= \frac{d_x}{2}\ln(2\pi e) + \frac{1}{2}\sum_{i=1}^{d_x}\ln\sigma_i^2. \qquad (8)
\end{aligned}$$

Obviously, maximizing the differential entropy is equivalent to maximizing the predictive variances. With this criterion, the active learner queries the input z whose function value $f(z)$ is most uncertain.

The second criterion is the information gain (mutual information). The parameters of the GP model are collected as $\theta = (\sigma_f^2, \Lambda, \sigma_n^2)$. The uncertainty in the model of the dynamical system can be represented through a distribution over the model parameter θ. Denote the current data as $D = (X, Y)$, the data to be selected by the active learner as D_{new}. The information gain I between the two distributions $p(\theta|D)$ and $p(\theta|D \cup D_{\mathrm{new}})$ is:

$$I = H[\theta|D] - \mathbb{E}_{D_{\mathrm{new}}}\left[H[\theta|D \cup D_{\mathrm{new}}]\right], \qquad (9)$$

where H is the differential entropy. Maximizing the information gain I is equivalent to maximizing the reduction of the uncertainty in θ. Thus the information gain criterion selects samples D_{new} which maximally reduce the uncertainty in the model. It is quite expensive to compute and optimize the information gain since there are no closed forms for the posterior distributions $p(\theta|D)$ and $p(\theta|D \cup D_{\mathrm{new}})$. This paper adopts the differential entropy defined in (8) as the information criterion for active learning.

The difficulty for actively learning dynamical systems is that it is not possible to arbitrarily sample the state-action space. For example, an informative state-action pair $z^* = (x^*, u^*)$ could be found by maximizing the differential entropy in (8). However, to obtain the sample $(z_*, f(z_*))$, the system's state should be steered to x_* at first.

III. ACTIVE LEARNING THROUGH GLOBAL AND LOCAL EXPLORATIONS

The goal of active learning is to iteratively choose the most informative samples so that an accurate model can be learned using as few training samples as possible [19]–[21]. Common strategies for active learning include: (1) iteratively querying the input for which the model output is least certain; (2) iteratively selecting a sample which could maximally reduce the uncertainty in the model. Most of active learning algorithms do not have any theoretical guarantee on the consistency or the sample efficiency [20].

Actively learning dynamical systems is fundamentally different because it is not possible to arbitrarily sample the state-action space under the constraint of system dynamics. The state-of-the-art methods for actively learning dynamical systems follow the idea of active learning, at the same time take the constraint of system dynamics into account. These methods iteratively search for an informative trajectory by maximizing the sum of predictive variances along the trajectory with the system dynamics satisfied. These methods also do not have theoretical guarantee on the sample efficiency. The proposed methods in this paper follow the similar framework, and try to solve the problems of the state-of-the-art methods, such as inefficiency, high computational complexity and memory demand. The advantages of the proposed methods over the state-of-the-art methods are verified via various numerical examples.

In this paper, it is assumed that the system is controllable, the initial state of the system is always at x_0, and the action is so bounded that the state space of the system can not be efficiently explored using random actions. At first two existing algorithms for actively learning dynamical systems in the literature are introduced. Then more sample-efficient methods will be proposed.

A. PRELIMINARY METHODS

1) SEPARATED SEARCH AND CONTROL

The work in [11] extends the ideas of actively learning unknown functions proposed in [7] and [8] to dynamical systems which are modeled by GPs. At each iteration, at first an informative state-action pair $z^* = (x^*, u^*)$ is selected by maximizing the differential entropy of the predictive distribution:

$$\begin{aligned} z^* = \;& \arg\max_z H[f(z)] = \arg\max_z \frac{1}{2}\left\{1 + \ln(2\pi\sigma^2(z))\right\} \\ & \text{s.t.} \quad u_{\min} \le u \le u_{\max} \end{aligned} \tag{10}$$

where $u_{\min}$ and $u_{\max}$ are lower and upper bounds of the action u. Then an action sequence $(u_0, \ldots, u_{M-1})$ is optimized to drive the system's state from x_0 to x^*. After arriving at x^*, the action u^* is applied, and the system's state reaches $f(z^*)$. By this way, the informative sample $(z^*, f(z^*))$ is obtained. Finally the action sequence $(u_0, \ldots, u_{M-1}, u^*)$ is applied to the true system to collect samples. Repeat the three steps above until the learned model is accurate enough. This method is called Separated Search and Control because it separates the search for an informative state-action pair (x^*, u^*) from designing an action sequence which steers the system's state from x_0 to x^*. This method is able to visit the informative states which are far away from the initial state and difficult to be visited under random actions. But it is very inefficient since it focuses on exploring a single informative sample $(z^*, f(z^*))$ at each iteration.

2) INFORMATIVE CONTROL TRAJECTORY

The method Informative Control Trajectory proposed in [10] and [11] tries to find an action sequence which is expected to provide the most informative sequence of observations when executed on the real system. The informative action sequence is optimized by maximizing the sum of predictive variances along the trajectory with system dynamics satisfied:

$$\begin{aligned} & u_0, \ldots, u_{T-1} \\ & = \arg\max_{u_0,\ldots,u_{T-1}} \sum_{t=0}^{T-1} \sigma^2(x_t, u_t) \\ & \quad \text{s.t.} \quad x_{t+1} = \mu(x_t, u_t), \quad \text{for} \quad t = 0, \ldots, T-2 \\ & \qquad\quad u_{\min} \le u \le u_{\max} \end{aligned} \tag{11}$$

where μ is the predictive mean in (5) and σ^2 is the predictive variance in (6). Compared with the method Separated Search and Control, this method takes the sample efficiency along the trajectory into account. If the optimization problem in (11) can be solved successfully, this method will work well. However, with relatively large horizon T, solving the optimization problem in (11) is quite difficult, and the computational complexity and memory demand are very high because the objective function (sum of predictive variances) is too complicated. The work in [11] chooses the horizon T as 10 or 15. With larger T, the optimization problem in (11) can not be solved successfully using the shooting method. In the case of very bounded actions, in order to explore the unknown areas of the state space which are far away from the initial state, the horizon T must be chosen relatively large, as a result the shooting method may obtain a bad local optimum and the algorithm will fail to efficiently explore the whole state space.

B. GLOBAL AND LOCAL EXPLORATIONS

This paper proposes more sample-efficient methods which combine global and local explorations. As the global exploration, the agent searches for an informative area in the whole state-action space of the dynamical system. Then, as the local exploration, the agent efficiently explores the local informative area found by the global exploration. Assume that the area around the most informative state-action pair is also very informative. So the informative area could be found by searching for the most informative state-action pair $z^* = (x^*, u^*)$:

$$\begin{aligned} z^* = \;& \arg\max_z H[f(z)] \\ & \text{s.t.} \quad u_{\min} \le u \le u_{\max}. \end{aligned} \tag{12}$$

In order to explore the informative area around (x^*, u^*), the system's state should be driven to x^* from x_0 at first. Making use of the currently estimated GP model, an action sequence which could steer the system's state from x_0 to x^* can be obtained by solving the following optimal control problem with the quadratic cost:

$$\begin{aligned} & u_0, \ldots, u_{M-1} \\ & = \arg\min_{u_0,\ldots,u_{M-1}} \sum_{t=0}^{M-1} \mathbb{E}_{x_t}\left[(x_t - x^*)^{\mathrm{T}} Q (x_t - x^*)\right. \\ & \qquad \left. + u_t^{\mathrm{T}} R u_t\right] + \mathbb{E}_{x_M}\left[(x_M - x^*)^{\mathrm{T}} Q_T (x_M - x^*)\right] \\ & \quad \text{s.t.} \quad u_{\min} \le u \le u_{\max}. \end{aligned} \tag{13}$$

The expectation is taken with respect to the marginal distribution of the state $p(x_t|u_0, \ldots, u_{t-1}), t = 1, \ldots, M$. The propagation of the marginal state distribution in probabilistic models can be approximated by moment matching [22] or linearization of the predictive mean of the GP [23]. This paper adopts the maximum likelihood observations assumption [24], which propagates only the mean of the marginal state distribution. The optimal control problem in (13) is

Algorithm 1 Global and Local Explorations

```
init: Choose horizons $M$ and $T$. Generate initial samples $D_{\text{init}}$ by applying random actions.
Initially the total samples are $D = D_{\text{init}}$.
for $i = 1, 2, \dots, N_{\text{iter}}$ do
    Train the GP model using total samples $D$.
    Search for the most informative state-action pair $z^* = (x^*, u^*)$ by solving (12).
    Based on the currently estimated model, optimize an action sequence $\{u_0, \dots, u_{M-1}\}$ which could drive the system's state from $x_0$ to $x^*$, by solving (14). Denote $u^*$ as $u_M$.
    Starting from $x_0$, taking actions $\{u_0, \dots, u_{M-1}, u_M\}$, the system's state arrives at $x_{M+1}$ which can be derived using (15).
    Optimize an action sequence $\{u_{M+1}, \dots, u_{M+T}\}$ which could efficiently explore the local informative area around $x_{M+1}$, by solving (16).
    Apply the actions $\{u_0, \dots, u_M, u_{M+1}, \dots, u_{M+T}\}$ to the real system and collect new samples $D_{\text{new}}$.
    Add newly collected samples to total samples $D = D \cup D_{\text{new}}$.
end for
```

simplified as:

$$\begin{aligned} &u_0, \dots, u_{M-1} \\ &\quad = \arg\min_{u_0,\dots,u_{M-1}} \sum_{t=0}^{M-1} \Big[(x_t - x^*)^{\mathrm{T}} Q (x_t - x^*) \\ &\qquad + u_t^{\mathrm{T}} R u_t\Big] + (x_M - x^*)^{\mathrm{T}} Q_T (x_M - x^*) \\ &\qquad \text{s.t.} \quad x_{t+1} = \mu(x_t, u_t), \quad \text{for} \quad t = 0, \dots, M-1 \\ &\qquad\qquad u_{\min} \le u \le u_{\max}. \end{aligned} \tag{14}$$

Starting from the initial state x_0, taking the action sequence $\{u_0, \dots, u_{M-1}\}$, the system's state is expected to reach x^*. Then the action u^* is applied, and the system's state reaches x_{M+1}. To simplify the notation, denote u^* as u_M. The state x_{M+1} could be derived by applying the actions $\{u_0, \dots, u_{M-1}, u_M\}$ to the estimated model:

$$x_{t+1} = \mu(x_t, u_t), \quad \text{for} \quad t = 0, \dots, M. \tag{15}$$

The remaining job is to optimize an action sequence which could efficiently explore the local informative area around x_{M+1}:

$$\begin{aligned} &u_{M+1}, \dots, u_{M+T} \\ &\quad = \arg\max_{u_{M+1},\dots,u_{M+T}} \sum_{t=0}^{T-1} H[f(x_{M+t+1}, u_{M+t+1})] \\ &\qquad \text{s.t.} \quad x_{M+t+2} = \mu(x_{M+t+1}, u_{M+t+1}), \\ &\qquad\qquad \text{for} \ t = 0, \dots, T-2 \\ &\qquad\qquad u_{\min} \le u \le u_{\max}. \end{aligned} \tag{16}$$

Algorithm 1 summarizes the framework of the proposed method Global and Local Explorations.

Algorithm 2 Improved Global and Local Explorations

```
init: Choose horizons $M$ and $T$. Generate initial samples $D_{\text{init}}$ by applying random actions.
Initially the total samples are $D = D_{\text{init}}$.
for $i = 1, 2, \dots, N_{\text{iter}}$ do
    Train the GP model using total samples $D$.
    Search for an informative trajectory $\{\bar{x}_0, \bar{u}_0, \bar{u}_1, \dots, \bar{u}_{T-1}\}$ by solving (17).
    Based on the currently estimated model, optimize an action sequence $\{u_0, \dots, u_{M-1}\}$ which could drives the system's state from $x_0$ to $\bar{x}_0$, by solving (18).
    Apply the actions $\{u_0, \dots, u_{M-1}, \bar{u}_0, , \dots, \bar{u}_{T-1}\}$ to the real system and collect new samples $D_{\text{new}}$.
    Add newly collected samples to total samples $D = D \cup D_{\text{new}}$.
end for
```

C. IMPROVED GLOBAL AND LOCAL EXPLORATIONS

The proposed method Global and Local Explorations finds a local informative area from the state-action space by globally searching for the most informative state-action pair (x^*, u^*) and assuming that the area around (x^*, u^*) should be also very informative. This assumption is a bit heuristic. In this subsection, a more reasonable way to search for a local informative area is proposed, and the corresponding algorithm is called Improved Global and Local Explorations.

A local informative trajectory $\{\bar{x}_0, \bar{u}_0, \bar{u}_1, \dots, \bar{u}_{T-1}\}$ can be found by maximizing the sum of differential entropies along the trajectory:

$$\begin{aligned} &\bar{x}_0, \bar{u}_0, \dots, \bar{u}_{T-1} \\ &\quad = \arg\max_{\bar{x}_0,\bar{u}_0,\dots,\bar{u}_{T-1}} \sum_{t=0}^{T-1} H[f(\bar{x}_t, \bar{u}_t)] \\ &\qquad \text{s.t.} \quad \bar{x}_{t+1} = \mu(\bar{x}_t, \bar{u}_t), \quad \text{for} \quad t = 0, \dots, T-2 \\ &\qquad\qquad u_{\min} \le u \le u_{\max}, \end{aligned} \tag{17}$$

where $\bar{x}_0$ is the initial state of the local informative trajectory. This trajectory is very informative since all the information along the trajectory is taken into account. To explore this local informative trajectory, the system's state should be driven to $\bar{x}_0$ from x_0 at first. Based on the currently estimated GP model, an action sequence which could steer the system's state from x_0 to $\bar{x}_0$ can be obtained by solving the following optimal control problem:

$$\begin{aligned} &u_0, \dots, u_{M-1} \\ &\quad = \arg\min_{u_0,\dots,u_{M-1}} \sum_{t=0}^{M-1} \Big[(x_t - \bar{x}_0)^{\mathrm{T}} Q (x_t - \bar{x}_0) \\ &\qquad + u_t^{\mathrm{T}} R u_t\Big] + (x_M - \bar{x}_0)^{\mathrm{T}} Q_T (x_M - \bar{x}_0) \\ &\qquad \text{s.t.} \quad x_{t+1} = \mu(x_t, u_t), \quad \text{for} \quad t = 0, \dots, M-1 \\ &\qquad\qquad u_{\min} \le u \le u_{\max}. \end{aligned} \tag{18}$$

Algorithm 2 summarizes the framework of the method Improved Global and Local Explorations. The optimization problems in (14) and (16)-(18) are solved by the multiple shooting method using the package CasADi [25].

For the proposed methods Global and Local Explorations and Improved Global and Local Explorations, to explore the local informative trajectory found by the global exploration, an action sequence $\{u_0, \ldots, u_{M-1}\}$ with horizon M which could drive the system's state from the initial state x_0 to the desired state (x^* or $\bar{x}_0$) should be obtained at first, as shown in (14) or (18). If M is too small, the system's state can not reach the desired state even if the action sequence is optimized using the true dynamics. If M is too large, the sample efficiency along the trajectory from x_0 to the desired state will be very low. Thus M should be large enough so that the system's state could reach any desired state starting from the initial state x_0, but at the same time M should not be too large so that the sample efficiency along the trajectory from x_0 to the desired state will not be too low. When learning dynamical systems by data-driven methods, it is common to use the analytical physical model (may be inaccurate due to the difficulty of modeling complex friction and damping) as the prior information [26], [27]. The analytical physical model such as ordinary differential equations (ODEs) can help the choice of M. Denote ODEs derived using the Lagrange method as:

$$\dot{x} = g(x, u; w), \tag{19}$$

with unknown physical parameter w identified using training data. Regarding ODEs as the true dynamics, choosing a set of desired states $\{x_1^*, x_2^*, \ldots, x_l^*\}$ which are far away from the initial state x_0, determine through trial and error how many time steps the system needs to reach these desired states starting from x_0. The horizon M should be large enough so that the system's state could reach any desired state starting from x_0. For systems whose analytical physical models are hard to derive, it is better to choose a relatively large M. The horizon T is the sample size of the local informative trajectory found by the global exploration. With larger T, the algorithm could explore the state-action space more efficiently, but solving the optimization problems (16) and (17) becomes much more difficult, as well as the computational complexity and memory demand are much higher, and vice versa. On the premise of being able to solve the optimization problems (16) and (17), larger values of T are preferred.

In (14) and (18), since the currently estimated GP model is used for planning, it is not guaranteed that the optimized action sequence $\{u_0, \ldots, u_{M-1}\}$ is actually going to drive the system's state towards the desired state (x^* or $\bar{x}_0$). Whether the system's state reaches the desired state under $\{u_0, \ldots, u_{M-1}\}$ is not important. If the actually reached state deviates seriously from the desired state, it means that the estimated GP model is very inaccurate along the trajectory, thus the collected samples are very informative and will greatly improve the accuracy of the GP model.

Compared with the method Separated Search and Control, the proposed methods are capable of exploring the state-action space much more efficiently because the agent focuses on exploring an informative area instead of a single informative point at each iteration. The method Informative Control Trajectory optimizes an informative action sequence $\{u_0, \ldots, u_{T-1}\}$ with the initial state x_0 at each iteration. In the case of very bounded actions, in order to explore the unknown areas of the state space which are far away from x_0, the horizon T must be chosen relatively large. Because the objective function is too complicated, the optimization problem in (11) can not be solved successfully, and the computational complexity and memory demand are very high. As a result, the method Informative Control Trajectory is not able to efficiently explore the whole state space. Instead of optimizing a long informative trajectory with the initial state x_0, the proposed methods search for a relatively short informative trajectory with the initial state x_{M+1} or $\bar{x}_0$ by solving (16) or (17). Then to explore this informative trajectory, an action sequence is optimized to drive the system's state towards x_{M+1} or $\bar{x}_0$ from x_0. By this way, on the one hand the algorithm can efficiently explore the whole state space. On the other hand, the optimization problems are much easier to solve, and the computational complexity and memory demand are much lower compared with the method Informative Control Trajectory. These conclusions will be verified via the following empirical evaluations.

IV. SIMULATIONS

A. SIMULATED DYNAMICAL SYSTEMS

The proposed methods and the existing methods are compared on three simulated dynamical systems as shown in Fig. 1.

1) CART-POLE

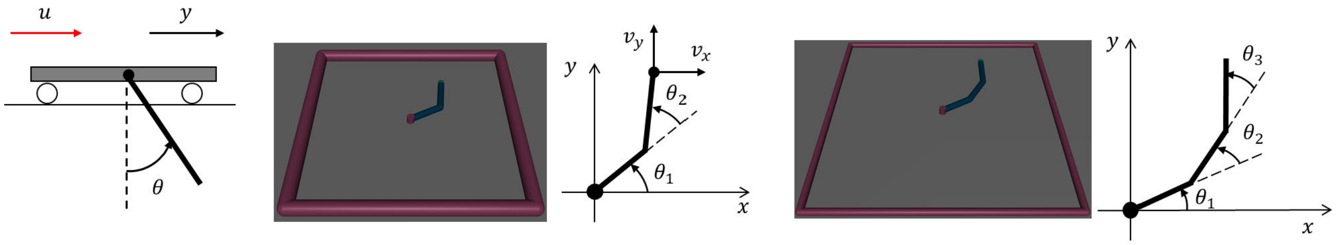
The physical parameters are: cart mass $m_1 = 0.5$kg, pole mass $m_2 = 0.5$kg, pole length $l = 0.6$m, friction coefficient between the cart and its rail $\mu = 0.2$Ns/m, bounded horizontal force $u \in [-3.5, 3.5]$N. The time discretization is $\Delta t = 0.1$s. The action u is piecewise constant and can be modified every $\Delta t = 0.1$s. The cart-pole system has four states: the position of the cart y, its velocity $\dot{y}$, the angle of the pole θ, and the angular velocity $\dot{\theta}$, which are collected as $x = [y, \dot{y}, \theta, \dot{\theta}]^{\mathrm{T}}$. The pole angle θ is measured anticlockwise from hanging down. The initial state is $x_0 = [0, 0, 0, 0]^{\mathrm{T}}$.

2) MODIFIED REACHER-V2 IN MUJOCO [28]

The modified Reacher-v2 is a two-link planar manipulator with two actuated joints. The bounded torques in the original Reacher-v2 are $u_1, u_2 \in [-1, 1]$N. To increase the difficulty of exploration, the bounded torques in the modified Reacher-v2 are set as $u_1, u_2 \in [-0.2, 0.2]$N. The time discretization is $\Delta t = 0.01$s. The state is $x = [\cos\theta_1, \cos\theta_2, \sin\theta_1, \sin\theta_2, v_x, v_y]^{\mathrm{T}}$, with θ_1, θ_2 joint angles, v_x, v_y velocities of the fingertip in the x, y directions. The initial state is $x_0 = [1, 1, 0, 0, 0, 0]^{\mathrm{T}}$.

3) 3-LINK REACHER

This is a three-link planar manipulator with three actuated joints. The bounded torques are $u_1, u_2, u_3 \in [-0.2, 0.2]$N.



(1) cart-pole (2) modified Reacher-v2 (3) 3-link Reacher

FIGURE 1. Simulated dynamical systems.

TABLE 1. Horizons.

	cart-pole	modified Reacher-v2	3-link Reacher
Sepa	$M = 40$	$M = 60$	$M = 60$
Info	$T = 41$	$T = 61$	$T = 61$
Glob	$M = 30, T = 10$	$M = 50, T = 10$	$M = 50, T = 10$
iGlob	$M = 31, T = 10$	$M = 51, T = 10$	$M = 51, T = 10$

The time discretization is $\Delta t = 0.01$s. The state is $x = [\cos\theta_1, \cos\theta_2, \cos\theta_3, \sin\theta_1, \sin\theta_2, \sin\theta_3, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3]^{\mathrm{T}}$, with $\theta_1, \theta_2, \theta_3$ joint angles, $\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3$ angular velocities. The initial state is $x_0 = [1, 1, 1, 0, 0, 0, 0, 0, 0]^{\mathrm{T}}$.

The angle $\theta \in [-\pi, \pi]$ is a discontinuous function of the GP input (x, u) which is hard to be modeled by the GP with the squared exponential kernel. To avoid the difficulty of modeling discontinuity, the angle θ in the state x of the cart-pole is extended to $(\sin\theta, \cos\theta)$. Then the state of the cart-pole is augmented to $x = [y, \dot{y}, \sin\theta, \cos\theta, \dot{\theta}]^{\mathrm{T}}$. An independent GP is used for each dimension of the state. Assume that the initial state of the system is always at x_0. For all systems, the action u is set so limited that it is difficult to explore the areas of the state space far away from the initial state x_0 when applying random actions.

The initial training data is generated by applying random actions. All methods share the same initial training data. To fairly compare the models learned by various methods over iterations, all methods obtain the same number of samples at each iteration. For different methods and different systems, corresponding horizons are summarized in Table 1. The meanings of M and T are explained in Section III. In the remaining of the paper, Sepa represents the method Separated Search and Control; Info represents the method Informative Control Trajectory; Glob represents the method Global and Local Explorations; iGlob represents the method Improved Global and Local Explorations.

B. EVALUATION CRITERIA

The quality of the learned model is evaluated on four criteria: (1) accuracy of the one-step prediction; (2) accuracy of the long-term prediction; (3) exploration ratio of the state space; (4) informativeness of the explored trajectory.

1) ACCURACY OF THE ONE-STEP PREDICTION

Randomly generate a set of samples of the state-action pair (x, u) from the state-action space $\mathcal{X} \times \mathcal{U}$, denoted as $(x_i, u_i)_{i=1}^{N_1}$. For each (x_i, u_i), the predictive distribution of the next state is Gaussian with mean $\mu(x_i, u_i)$ and variance $\sigma^2(x_i, u_i)$, (5) and (6). The true value of the next state is $f(x_i, u_i)$. The root mean square error (RMSE) for the one-step prediction is calculated by:

$$\mathrm{RMSE}_{\text{one-step}} = \sqrt{\frac{1}{N_1}\sum_{i=1}^{N_1}(\mu(x_i, u_i) - f(x_i, u_i))^2}. \quad (20)$$

The mean absolute percentage error (MAPE) for the one-step prediction is defined as:

$$\mathrm{MAPE}_{\text{one-step}} = \frac{1}{N_1}\sum_{i=1}^{N_1}\left|\frac{\mu(x_i, u_i) - f(x_i, u_i)}{f(x_i, u_i)}\right|. \quad (21)$$

Since each dimension of the system function f is modeled by an independent GP, there is a RMSE and a MAPE for each dimension of the state. For the cart-pole, randomly generate 1250 samples of (x, u) from the state-action space $\{y \in [-20, 20], \dot{y} \in [-10, 10], \theta \in [-\pi, \pi], \dot{\theta} \in [-10, 10], u \in [-3.5, 3.5]\}$. For the modified Reacher-v2, randomly generate 1250 samples of (x, u) from the state-action space $\{\theta_1 \in [-\pi, \pi], \theta_2 \in [-\pi, \pi], v_x \in [-15, 15], v_y \in [-15, 15], u_1 \in [-0.2, 0.2], u_2 \in [-0.2, 0.2]\}$. For the 3-link Reacher, randomly generate 5000 samples of (x, u) from the state-action space $\{\theta_1 \in [-\pi, \pi], \theta_2 \in [-\pi, \pi], \theta_3 \in [-\pi, \pi], \dot{\theta}_1 \in [-10, 10], \dot{\theta}_2 \in [-10, 10], \dot{\theta}_3 \in [-10, 10], u_1 \in [-0.2, 0.2], u_2 \in [-0.2, 0.2], u_3 \in [-0.2, 0.2]\}$.

2) ACCURACY OF THE LONG-TERM PREDICTION

Generate a set of testing trajectories $\{x_0^{(i)}, u_0^{(i)}, x_1^{(i)}, u_1^{(i)}, \ldots\}$ $(i = 1, 2, \ldots, N_2)$. From each testing trajectory $\{x_0^{(i)}, u_0^{(i)}, x_1^{(i)}, u_1^{(i)}, \ldots\}$, randomly choose N_3 states as the initial states, and predict the states K steps into the future (cart-pole: $K = 10$, modified Reacher-v2: $K = 20$, 3-link Reacher: $K = 20$). For example, if the state $x_j^{(i)}$ is chosen as the initial state, the predicted state can be derived from:

$$\begin{aligned} \bar{x}_{j+k+1}^{(i)} &= \mu(\bar{x}_{j+k}^{(i)}, u_{j+k}^{(i)}), \quad \text{for} \quad k = 0, \ldots, K-1 \\ \bar{x}_j^{(i)} &= x_j^{(i)}, \end{aligned} \quad (22)$$

where μ is the predictive mean in (5). The predicted state is $\bar{x}_{j+K}^{(i)}$ and the true state is $x_{j+K}^{(i)}$. Summarize all the predicted

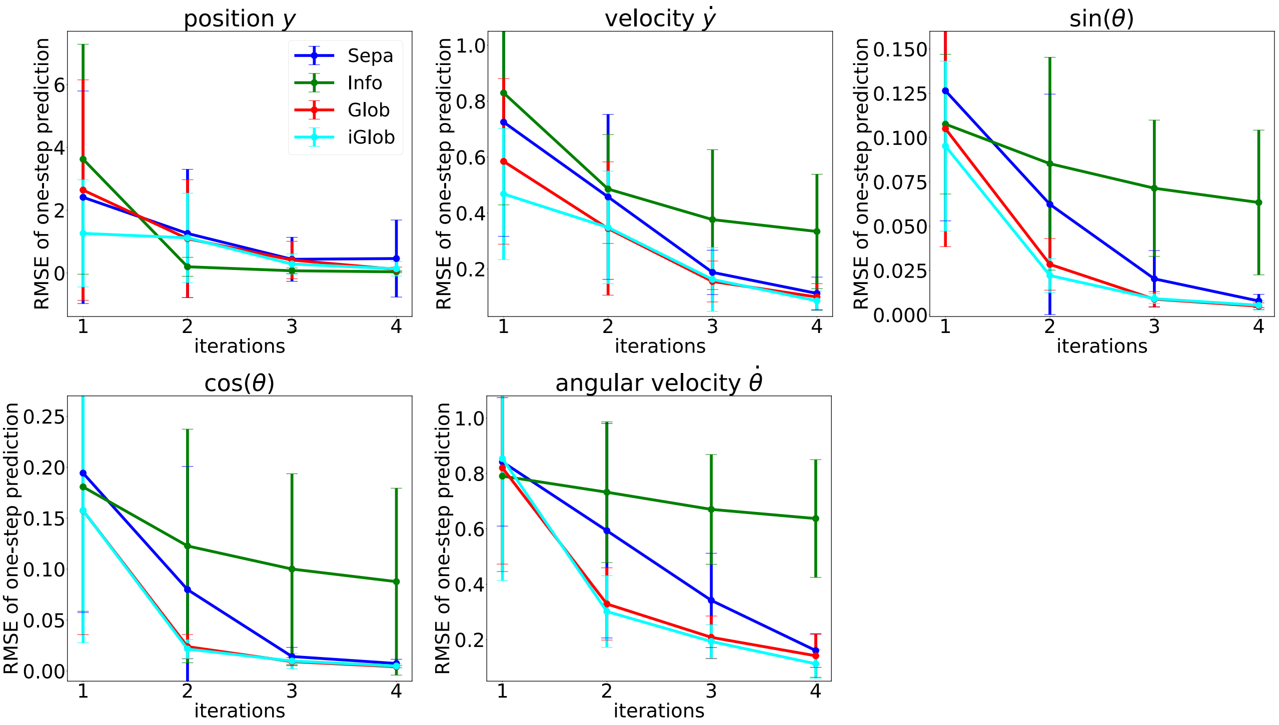


FIGURE 2. RMSEs of one-step prediction over iterations for the cart-pole.

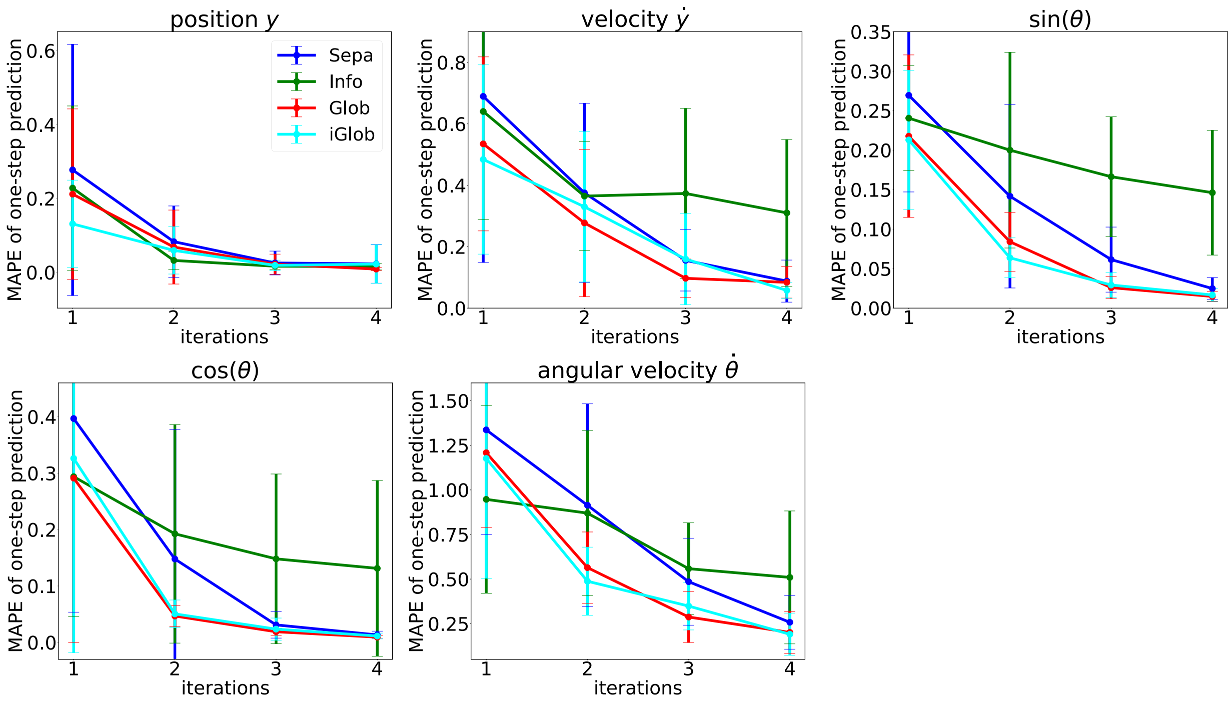


FIGURE 3. MAPEs of one-step prediction over iterations for the cart-pole.

states as $\{\bar{x}_l\}_{l=1}^{N_2N_3}$, all the true states as $\{x_l\}_{l=1}^{N_2N_3}$, the RMSE for the long-term prediction is calculated by:

$$\mathrm{RMSE}_{\text{long-term}} = \sqrt{\frac{1}{N_2N_3}\sum_{l=1}^{N_2N_3}(\bar{x}_l - x_l)^2}. \tag{23}$$

The MAPE for the long-term prediction is calculated by:

$$\mathrm{MAPE}_{\text{long-term}} = \frac{1}{N_2N_3}\sum_{l=1}^{N_2N_3}\left|\frac{\bar{x}_l - x_l}{x_l}\right|. \tag{24}$$

Similarly for each dimension of the state, there is a RMSE and a MAPE of the long-term prediction.

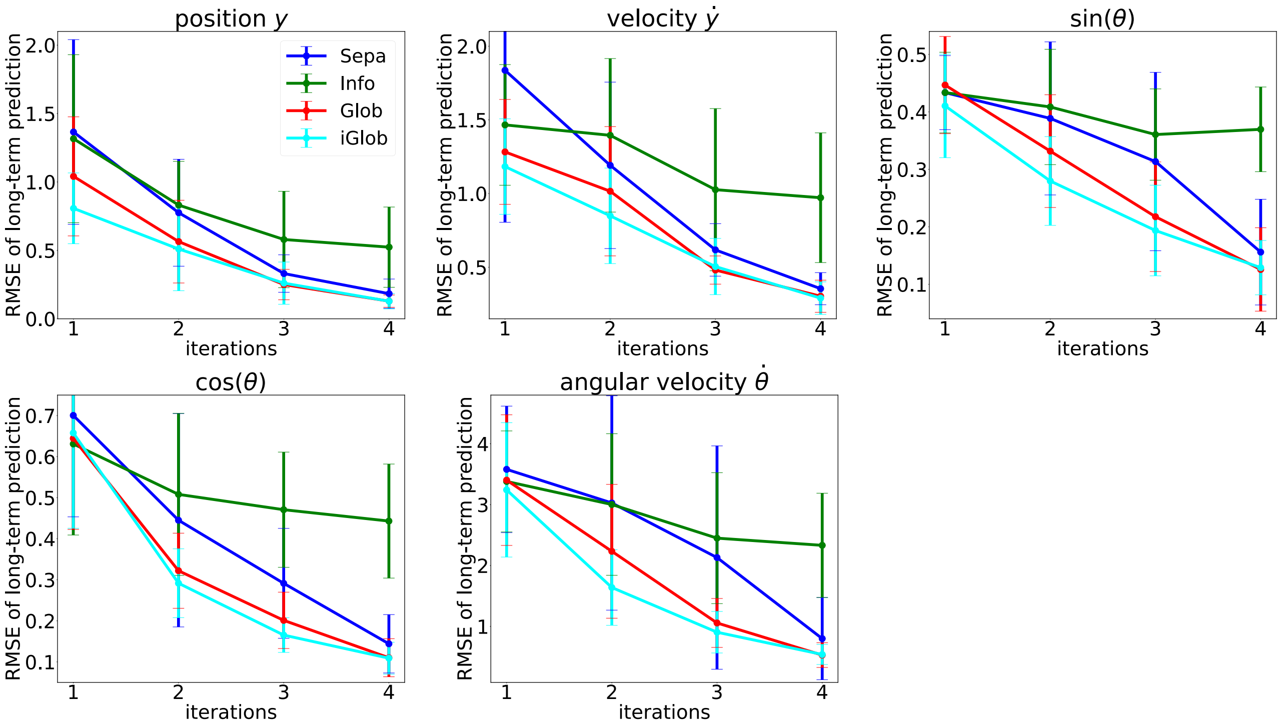


FIGURE 4. RMSEs of long-term prediction over iterations for the cart-pole.

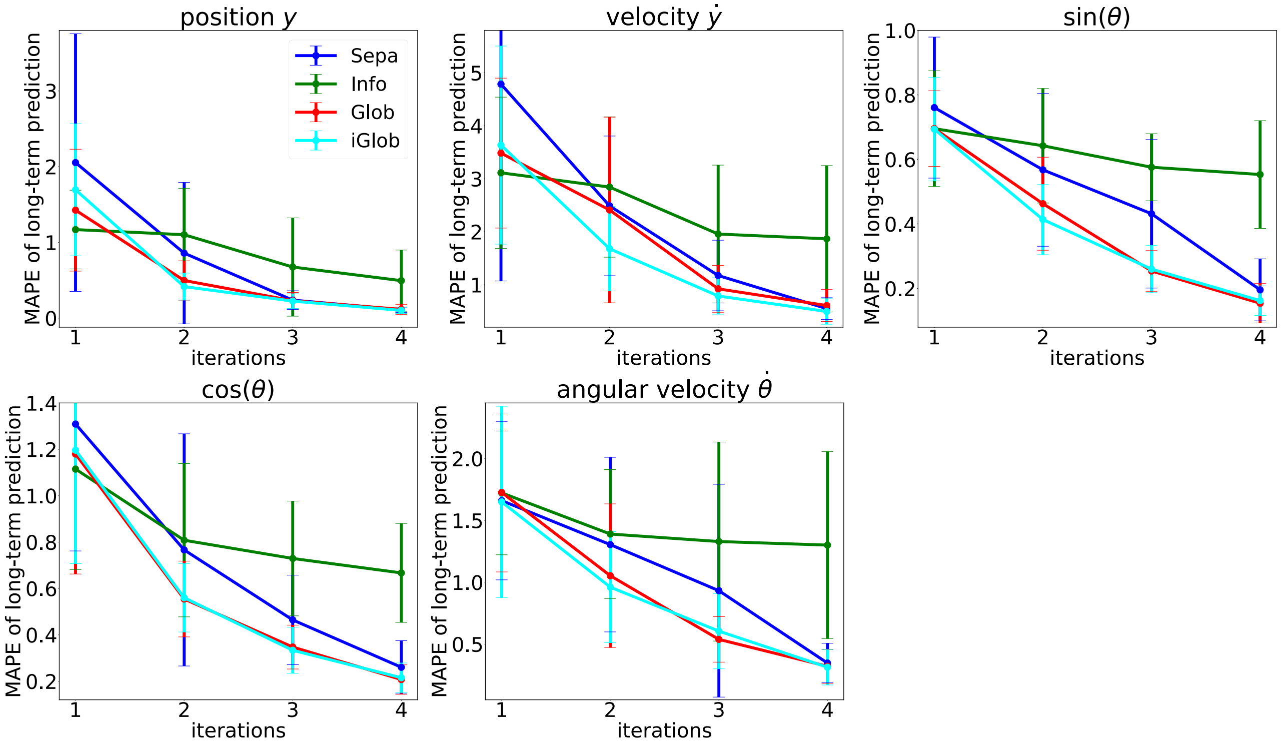


FIGURE 5. MAPEs of long-term prediction over iterations for the cart-pole.

Generate 6, 6 and 8 testing trajectories for the cart-pole, the modified Reacher-v2 and the 3-link Reacher respectively.

3) EXPLORATION RATIO OF THE STATE SPACE

The exploration efficiency of the active learning algorithm can be quantified by the exploration ratio. The state space

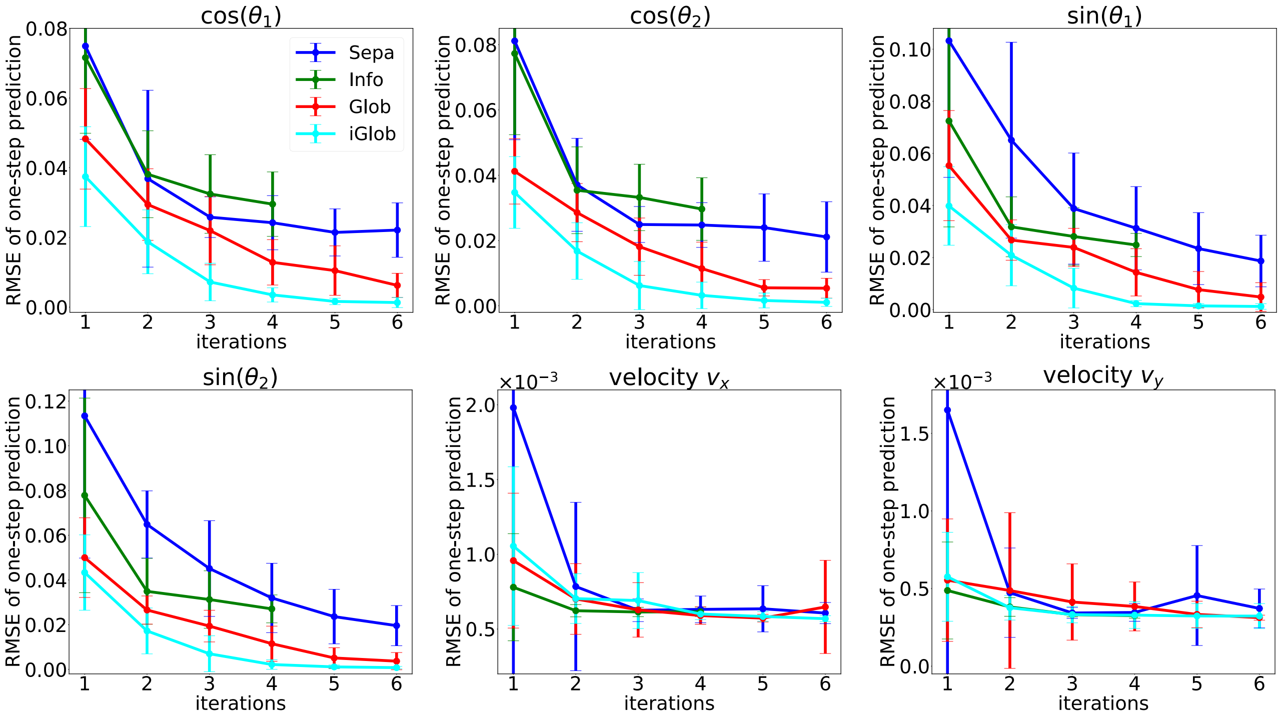


FIGURE 6. RMSEs of one-step prediction over iterations for the modified Reacher-v2.

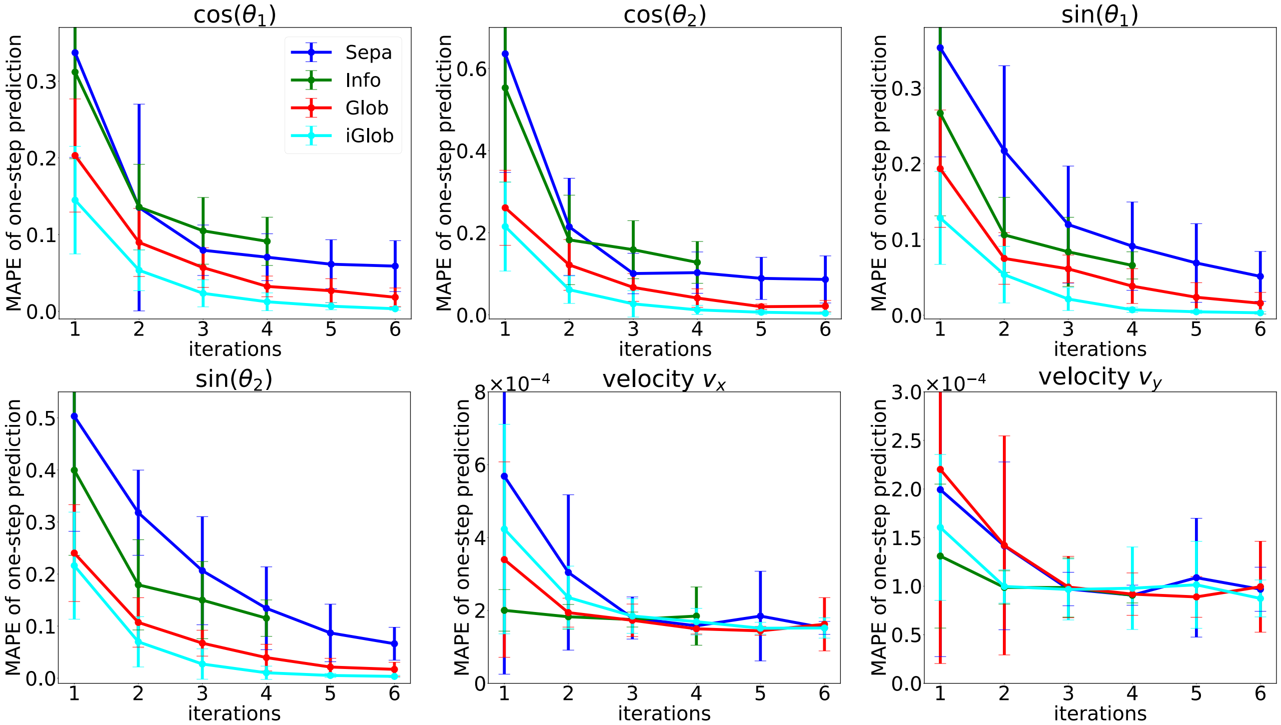


FIGURE 7. MAPEs of one-step prediction over iterations for the modified Reacher-v2.

$\mathcal{X}$ for the cart-pole is $\{\dot{y} \in [-10, 10], \theta \in [-\pi, \pi], \dot{\theta} \in [-10, 10]\}$. The position y is not included because it is less important for the cart-pole's dynamics. Discretize the state space $\mathcal{X}$ with bins [10, 20, 10]. The state space $\mathcal{X}$ for the modified Reacher-v2 is $\{\theta_1 \in [-\pi, \pi], \theta_2 \in [-\pi, \pi], v_x \in [-15, 15], v_y \in [-15, 15]\}$. Discretize the state space $\mathcal{X}$

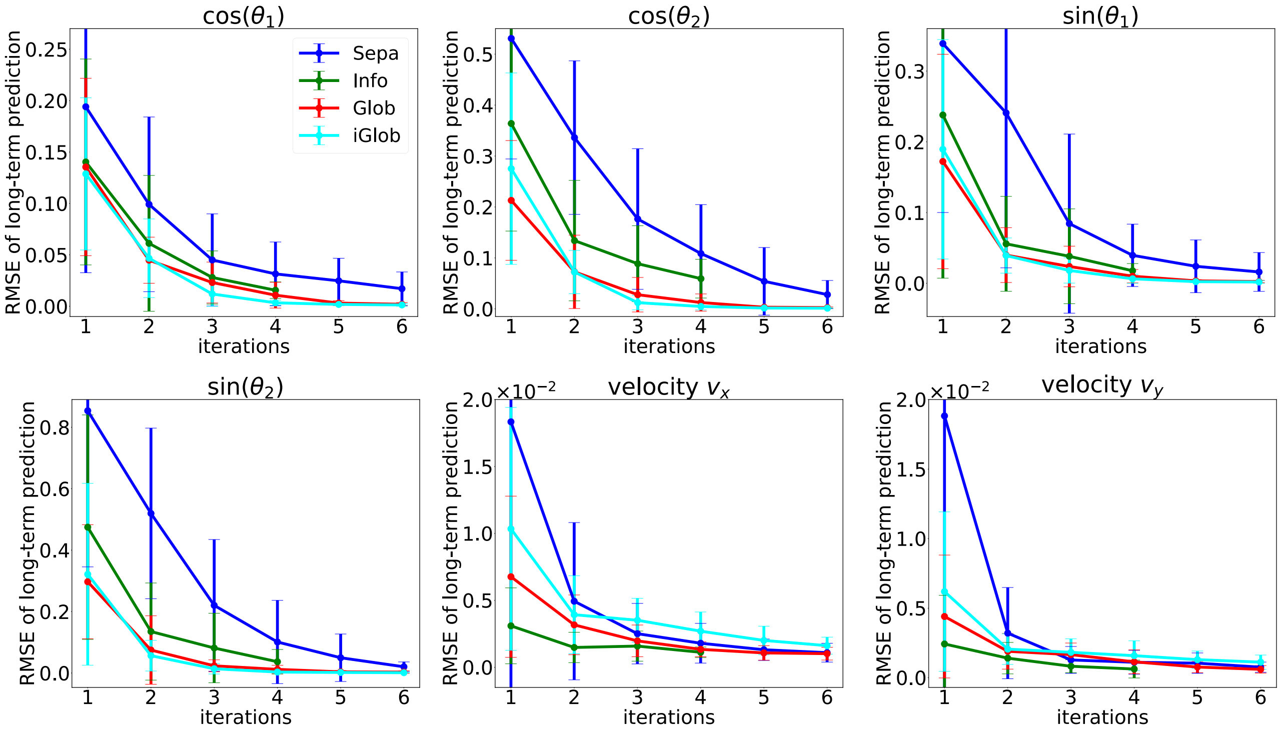


FIGURE 8. RMSEs of long-term prediction over iterations for the modified Reacher-v2.

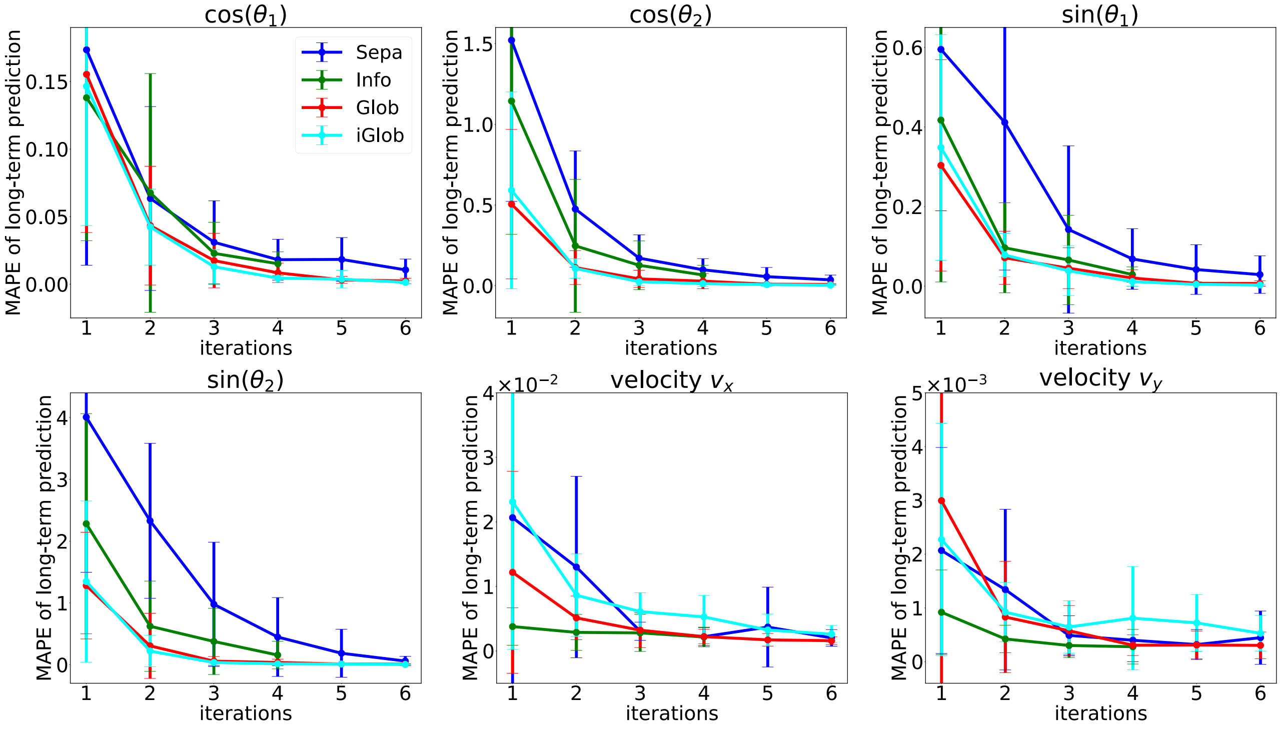


FIGURE 9. MAPEs of long-term prediction over iterations for the modified Reacher-v2.

with bins [20, 20, 10, 10]. The state space $\mathcal{X}$ for the 3-link Reacher is $\{\theta_1 \in [-\pi, \pi], \theta_2 \in [-\pi, \pi], \theta_3 \in [-\pi, \pi], \dot{\theta}_1 \in [-10, 10], \dot{\theta}_2 \in [-10, 10], \dot{\theta}_3 \in [-10, 10]\}$. Discretize the state space $\mathcal{X}$ with bins [20, 20, 20, 10, 10, 10]. Suppose that the number of visited cells of the state space $\mathcal{X}$ is N_{visited} and the number of total cells is N_{total}. The exploration ratio

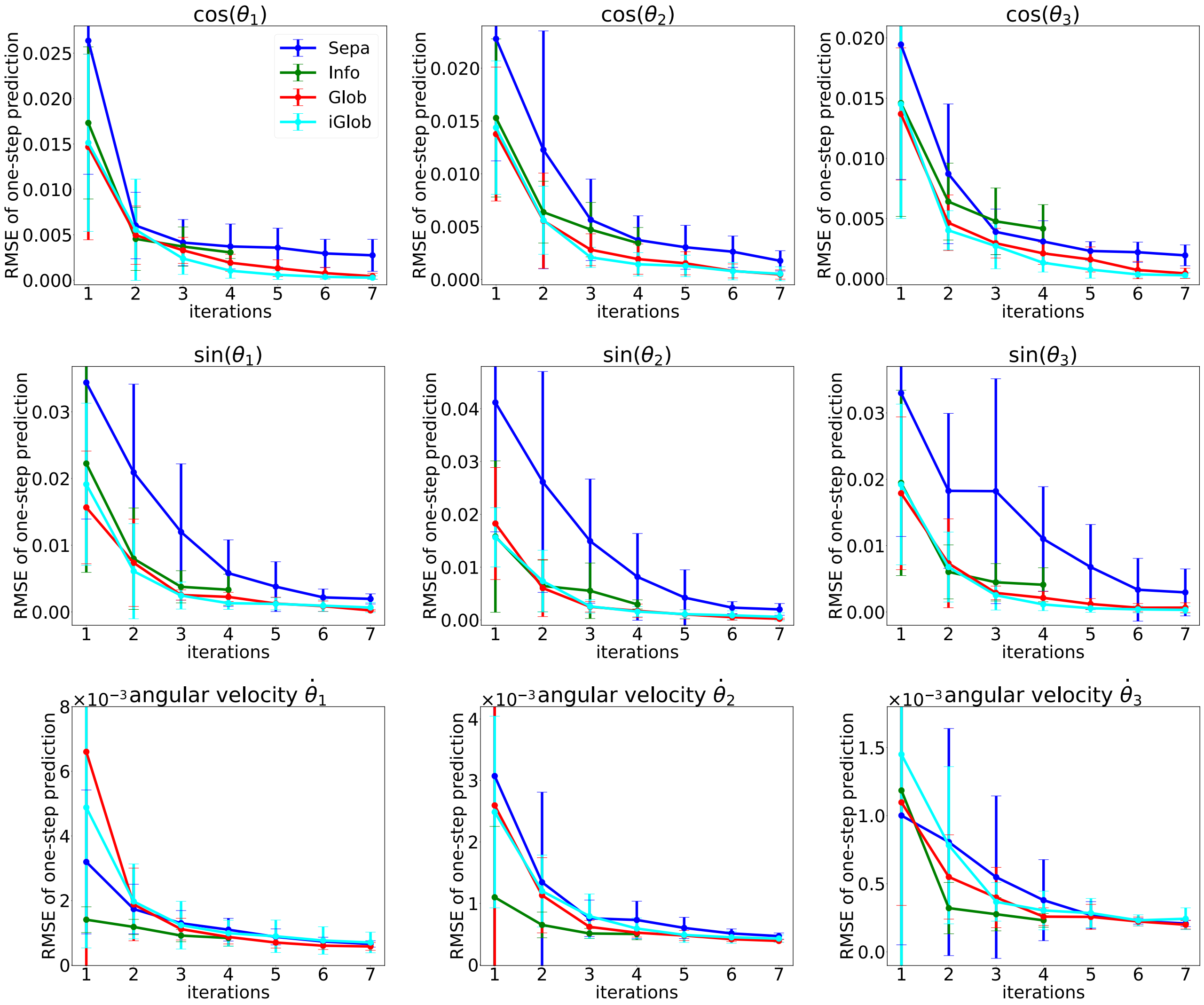


FIGURE 10. RMSEs of one-step prediction over iterations for the 3-link Reacher.

TABLE 2. Average running time for each algorithm (seconds).

	cart-pole	modified Reacher-v2	3-link Reacher
Sepa	101	1221 (345)	5488 (625)
Info	786	– (1499)	– (3870)
Glob	179	1933 (547)	7288 (1342)
iGlob	140	1799 (518)	7342 (1797)

is calculated by:

$$\text{exploration ratio} = \frac{N_{\text{visited}}}{N_{\text{total}}}. \tag{25}$$

4) INFORMATIVENESS OF THE EXPLORED TRAJECTORY

At each iteration, the active learning algorithm optimizes an action sequence $\{u_0, u_1, \ldots, u_{T'-1}\}$. Then a trajectory is explored by applying this action sequence to the true system, $\{x_0, u_0, x_1, u_1, \ldots, x_{T'-1}, u_{T'-1}, x_{T'}\}$. The informativeness of this explored trajectory can be measured by the sum of differential entropies along the trajectory:

$$\text{sum of differential entropies} = \sum_{t=0}^{T'-1} H[f(x_t, u_t)]. \tag{26}$$

V. RESULTS AND DISCUSSION

The number of iterations N_{iter} is 4, 6, 7 for the cart-pole, the modified Reacher-v2 and the 3-link Reacher respectively. For the modified Reacher-v2 and the 3-link Reacher, the evaluation of Info is not possible when the number of iterations N_{iter} exceeds 4 due to its high memory demand, so Info only goes through 4 iterations. Independently run each algorithm 20 times on each system. Table 2 shows the average time per run of the algorithm. The time unit is seconds. The values in parentheses mean the average running times when $N_{\text{iter}} = 4$. Info takes much more time than other methods due to the high computational complexity of solving (11). Glob and iGlob take longer time than Sepa.

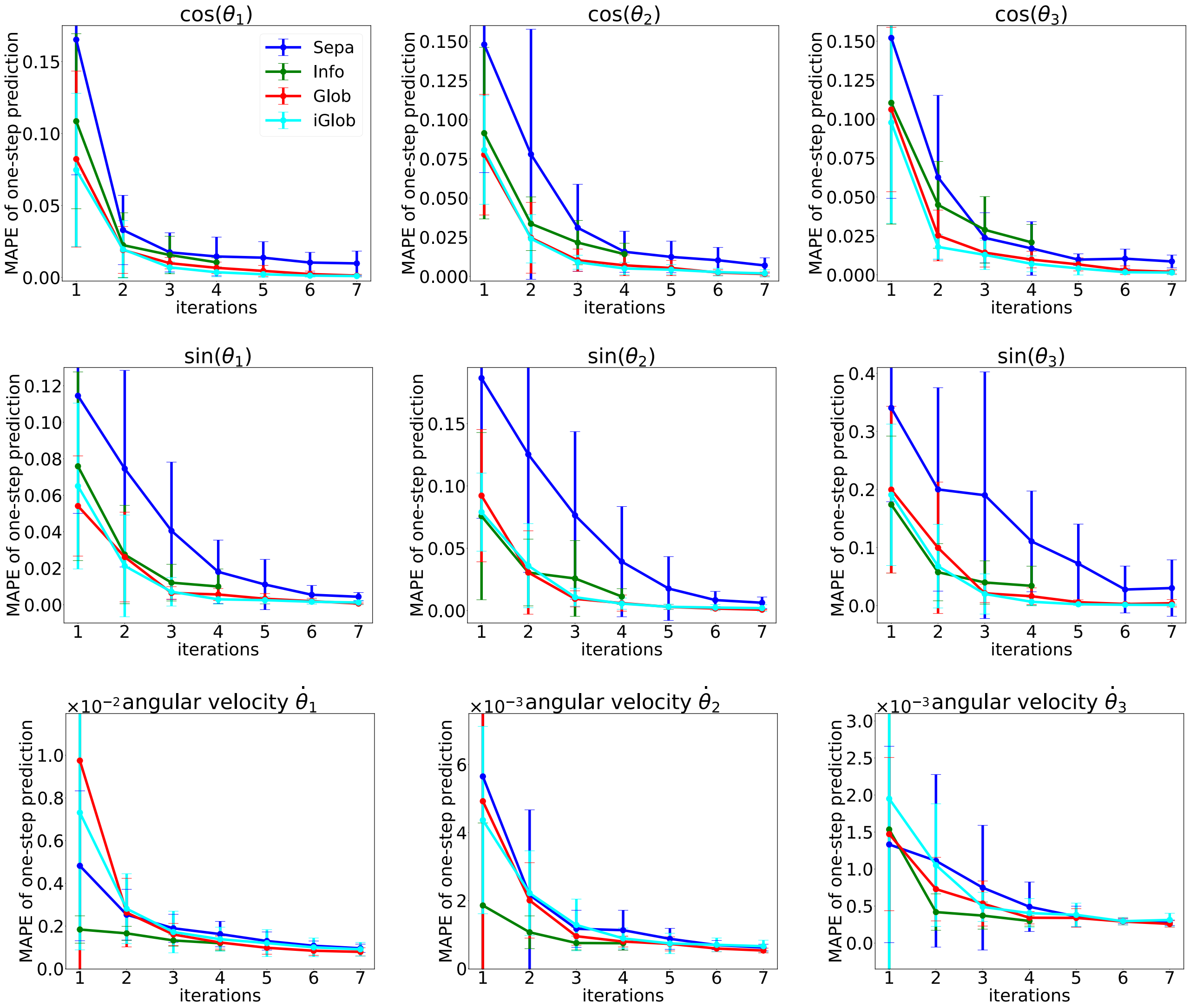


FIGURE 11. MAPEs of one-step prediction over iterations for the 3-link Reacher.

The RMSEs and MAPEs of the one-step prediction and the long-term prediction over iterations are shown in Figs. 2 to 13. Each plot in a figure represents one dimension of the state. The means (curves) and standard deviations (error bars) are obtained over 20 independent runs of the algorithms. For the cart-pole (Figs. 2 to 5), Glob and iGlob have the best performances because the RMSEs and MAPEs of the one-step prediction and the long-term prediction decrease the fastest; Sepa follows; and Info performs the worst. The reasons are as follows. With very bounded actions, in order to explore the informative areas of the state space which are far away from the initial state, the horizon T of Info must be chosen relatively large. However, solving the optimization problem in (11) is very difficult when the horizon T is large, and the solution may be a very bad local optimum. For the cart-pole, the state around the swing-up is very informative. With Info, the agent can not successfully visit the swing-up state in most cases. That is why the RMSEs and MAPEs obtained from Info decrease the slowest. With Sepa, the agent is able to explore the swing-up state gradually during iterations. However, Sepa is very inefficient since it focuses on visiting a single informative sample at each iteration. As a result, the RMSEs and MAPEs obtained from Sepa, Glob and iGlob converge to almost the same values in the end, but the RMSEs and MAPEs obtained from Sepa decrease slower than those obtained from Glob and iGlob. The proposed methods Glob and iGlob combine global and local explorations. At each iteration, the agent globally searches for an informative trajectory in the whole state-action space instead of a single informative point, and then explores the found local informative trajectory. By this way, the agent is able to explore the state-action space much more efficiently. Finally iGlob searches for the informative trajectories in a better way than Glob as discussed in Section III-C. That is why iGlob performs slightly better than Glob.

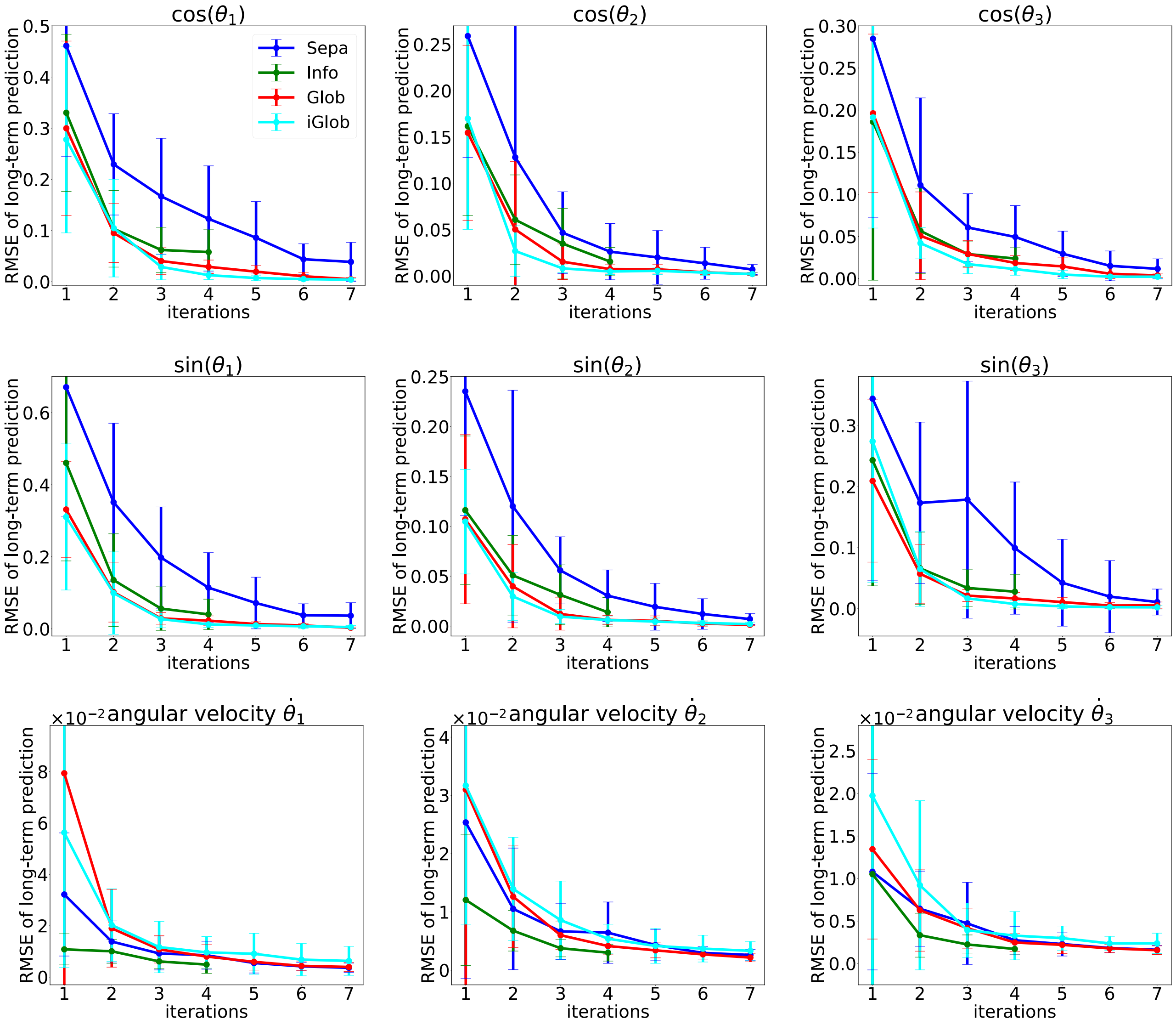


FIGURE 12. RMSEs of long-term prediction over iterations for the 3-link Reacher.

For the modified Reacher-v2 (Figs. 6 to 9), Glob and iGlob have the best performances because the RMSEs and MAPEs of the one-step prediction and the long-term prediction decrease the fastest; Info follows; and Sepa performs the worst. For v_x, v_y, the RMSEs and MAPEs obtained from Glob, iGlob and Info are very small and very close. The RMSEs and MAPEs obtained from Sepa decease the slowest because Sepa focuses on exploring a single informative sample at each iteration and is very inefficient. Compared with Sepa, Info has the potential to explore the state-action space more efficiently. However, solving the optimization problem in (11) with relatively large horizon T is very difficult and a bad local optimum may be obtained. As a result, the RMSEs and MAPEs obtained from Info decrease faster than those obtained from Sepa, but slower than those obtained from Glob and iGlob. Finally iGlob performs better than Glob because iGlob searches for the informative areas in a better way than Glob.

The results of the 3-link Reacher (Figs. 10 to 13) are similar to those of the modified Reacher-v2. Glob and iGlob have the best performances because the RMSEs and MAPEs of the one-step prediction and the long-term prediction decrease the fastest; Info follows; and Sepa performs the worst. For angular velocities $\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3$, the RMSEs and MAPEs obtained from Glob, iGlob, Info and Sepa are very small and very close.

In general, the accuracy of the learned model is correlated with the exploration ratio of the learning algorithm. The higher exploration ratio is more likely to lead to a learned model with smaller RMSEs and MAPEs. The exploration ratios over iterations are shown in Fig. 14. For the cart-pole, the exploration ratios of Glob and iGlob increase fastest,

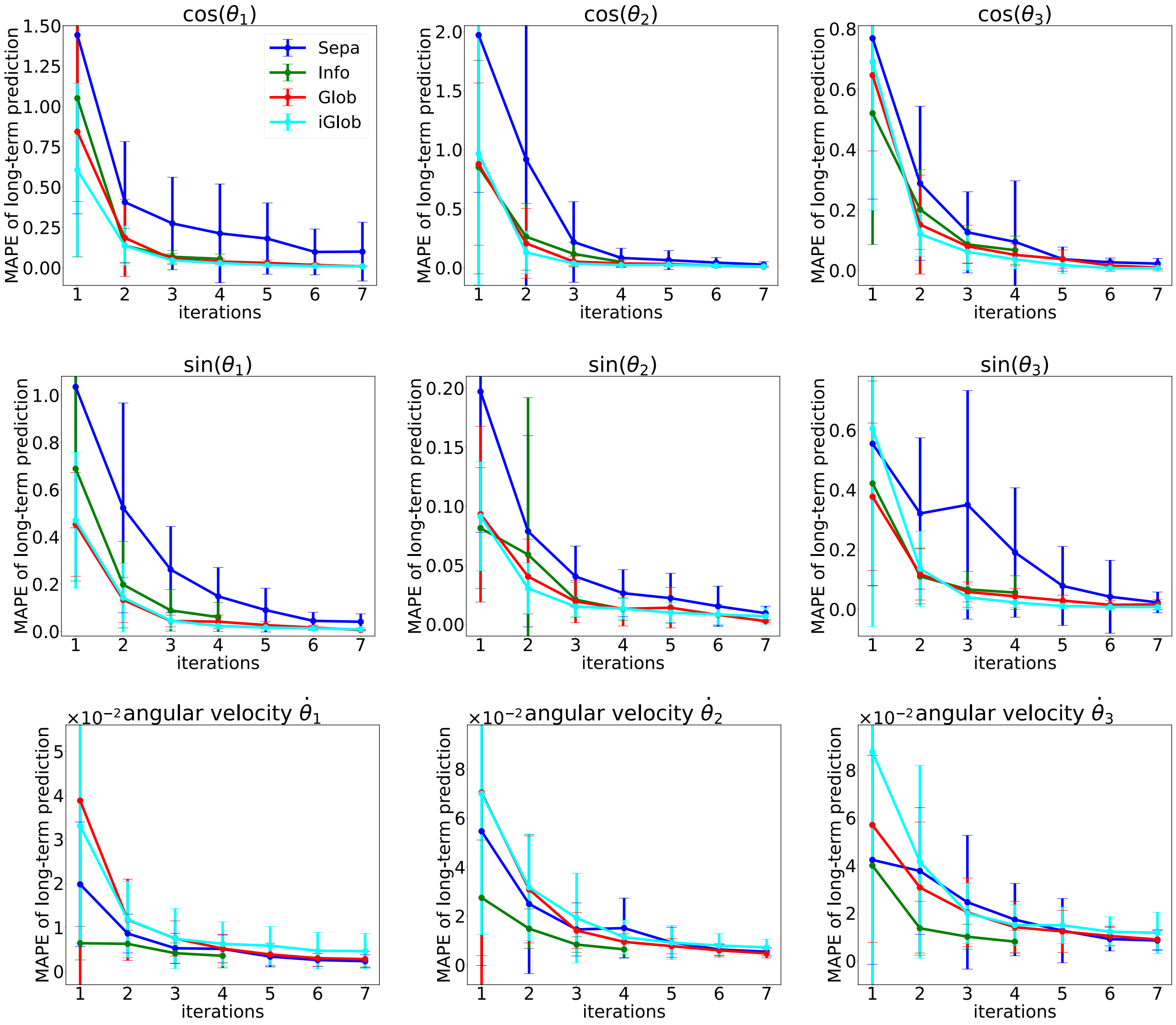


FIGURE 13. MAPEs of long-term prediction over iterations for the 3-link Reacher.

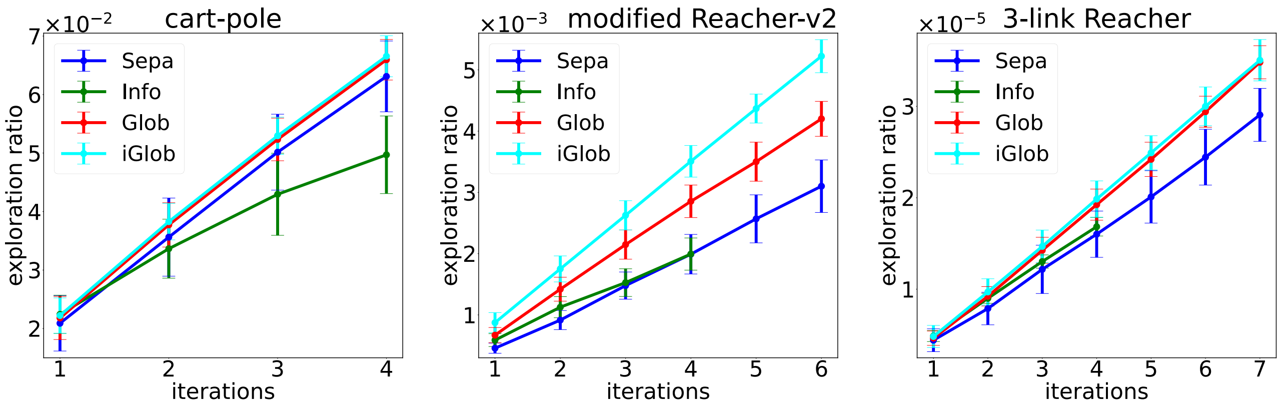


FIGURE 14. Exploration ratios over iterations.

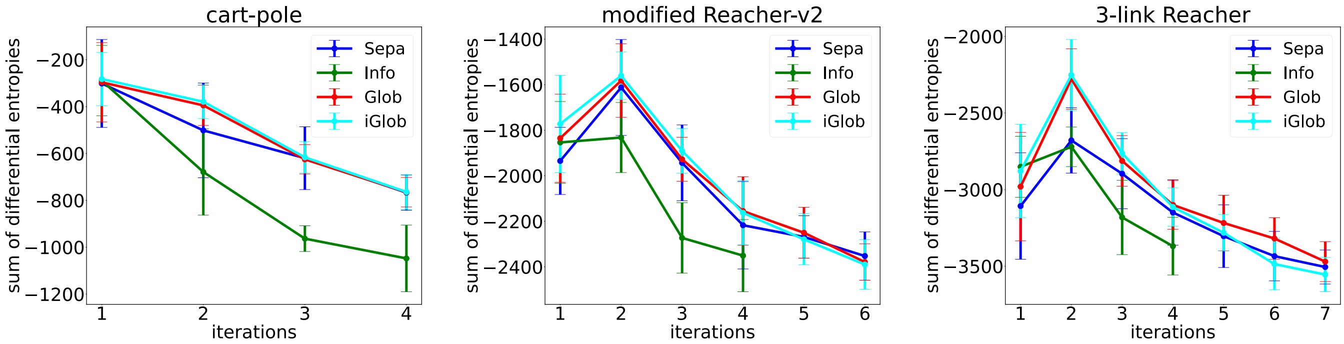


FIGURE 15. Sum of differential entropies along the explored trajectory over iterations.

followed by Sepa, and subsequently Info. And the exploration ratios of iGlob increase slightly faster than those of Glob. For the modified Reacher-v2 and the 3-link Reacher, the exploration ratios of Glob and iGlob increase fastest, followed by Info, and subsequently Sepa. And the exploration ratios of iGlob increase faster than those of Glob. For all systems, the results of exploration ratios correspond exactly to the results of RMSEs and MAPEs of the one-step prediction and the long-term prediction.

The sum of differential entropies along the explored trajectory over iterations is shown in Fig. 15. For all systems, Glob and iGlob have the largest sum of differential entropies, and iGlob has slightly larger sum of differential entropies than Glob. Larger sum of differential entropies represents the more informative trajectory. Thus the proposed methods could explore more informative trajectories at each iteration.

The above simulations demonstrate the advantages of the proposed active learning algorithms. The computational complexity and memory demand of Glob and iGlob are slightly higher than those of Sepa, but much lower than those of Info. Glob and iGlob can explore the state spaces more efficiently and yield more accurate models of the dynamical systems.

VI. CONCLUSION

This paper proposes more sample-efficient methods to actively learn the dynamical systems which are modeled by Gaussian processes. By combining the global and local explorations, the proposed methods could explore the state-action space of the dynamical system more efficiently, generate more informative samples, and learn a more accurate model. The proposed methods are compared to the existing methods in the literature on simulated dynamical systems in terms of the one-step and long-term predictive accuracies, the exploration ratio and the informativeness of the explored trajectory. The simulation results demonstrate that the proposed methods perform better.

In future work, the proposed methods will be scaled to dynamical systems with high-dimensional states by replacing Gaussian processes with Bayesian neural networks, which scale much better with the number of samples. Furthermore, as another information-based criterion, the information gain will be incorporated into the proposed methods.

REFERENCES

[1] T. B. Schön, A. Wills, and B. Ninness, "System identification of nonlinear state-space models," *Automatica*, vol. 47, no. 1, pp. 39–49, 2011.

[2] K. Fujimoto, A. Taniguchi, and Y. Nishida, "System identification of nonlinear state-space models with linearly dependent unknown parameters based on variational Bayes," *SICE J. Control, Meas., Syst. Integr.*, vol. 11, no. 6, pp. 456–462, Nov. 2018.

[3] M. Deisenroth, D. Fox, and C. E. Rasmussen, "Gaussian processes for data-efficient learning in robotics and control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 2, pp. 408–423, Feb. 2015.

[4] K. Chatzilygeroudis, V. Vassiliades, F. Stulp, S. Calinon, and J.-B. Mouret, "A survey on policy search algorithms for learning robot controllers in a handful of trials," *IEEE Trans. Robot.*, vol. 36, no. 2, pp. 328–347, Apr. 2020.

[5] K. Fujimoto, H. Beppu, and Y. Takaki, "Numerical solutions of Hamilton-Jacobi inequalities by constrained Gaussian process regression," *SICE J. Control, Meas., Syst. Integr.*, vol. 11, no. 5, pp. 419–428, Sep. 2018.

[6] R. Martinez-Cantin, N. de Freitas, A. Doucet, and J. A. Castellanos, "Active policy learning for robot planning and exploration under uncertainty," in *Robotics: Science and Systems*, vol. 3. MIT Press, 2007, pp. 321–328.

[7] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learn. Res.*, vol. 9, no. 2, pp. 235–284, 2008.

[8] A. Krause and C. Guestrin, "Nonmyopic active learning of Gaussian processes: An exploration-exploitation approach," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 449–456.

[9] A. Jain, T. Nghiem, M. Morari, and R. Mangharam, "Learning and control using Gaussian processes," in *Proc. ACM/IEEE 9th Int. Conf. Cyber-Phys. Syst. (ICCPS)*, Apr. 2018, pp. 140–149.

[10] M. Schultheis, B. Belousov, H. Abdulsamad, and J. Peters, "Receding horizon curiosity," in *Proc. Conf. Robot Learn.*, 2020, pp. 1278–1288.

[11] M. Buisson-Fenet, F. Solowjow, and S. Trimpe, "Actively learning Gaussian process dynamics," in *Proc. 2nd Conf. Learn. Dyn. Control*, 2020, pp. 5–15.

[12] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. 35th Int. Conf. Mach. Learn.*, 2018, pp. 1861–1870.

[13] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. De Turck, and P. Abbeel, "VIME: Variational information maximizing exploration," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1109–1117.

[14] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 2951–2959.

[15] R. Calandra, A. Seyfarth, J. Peters, and M. P. Deisenroth, "Bayesian optimization for learning gaits under uncertainty," *Ann. Math. Artif. Intell.*, vol. 76, nos. 1–2, pp. 5–23, Feb. 2016.

[16] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2005.

[17] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[18] A. G. D. G. Matthews, M. van der Wilk, T. Nickson, K. Fujii, A. Boukouvalas, P. León-Villagrá, Z. Ghahramani, and J. Hensman, "GPflow: A Gaussian process library using tensorflow," *J. Mach. Learn. Res.*, vol. 18, no. 40, pp. 1–6, 2017.

[19] B. Settles, "Active learning literature survey," Dept. Comput. Sci., University of Wisconsin-Madison, Madison, WI, USA, Tech. Rep. 1648, 2009.

[20] C. C. Aggarwal, X. Kong, Q. Gu, J. Han, and P. S. Yu, "Active learning: A survey," in *Data Classification: Algorithms and Applications*, C. C. Aggarwal, Ed. Boca Raton, FL, USA: CRC Press, 2014, pp. 571–605, doi: 10.1201/b17320.

[21] Y. Gal, R. Islam, and Z. Ghahramani, "Deep Bayesian active learning with image data," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 1183–1192.

[22] J. Q. Candela, A. Girard, J. Larsen, and C. E. Rasmussen, "Propagation of uncertainty in Bayesian kernel models-application to multiple-step ahead forecasting," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 2, Apr. 2003, p. 701.

[23] J. Ko and D. Fox, "GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models," *Auto. Robots*, vol. 27, no. 1, pp. 75–90, 2009.

[24] R. Platt, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, "Belief space planning assuming maximum likelihood observations," in *Proc. Robot., Sci. Syst. Conf.*, 2010.

[25] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: A software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, no. 1, pp. 1–36, Mar. 2018.

[26] J. Ko, D. J. Klein, D. Fox, and D. Haehnel, "Gaussian processes and reinforcement learning for identification and control of an autonomous blimp," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 742–747.

[27] S. Tang, K. Fujimoto, and I. Maruta, "Learning dynamic systems using Gaussian process regression with analytic ordinary differential equations as prior information," *IEICE Trans. Inf. Syst.*, vol. 104, no. 9, pp. 1440–1449, 2021.

[28] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A physics engine for model-based control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 5026–5033.

SHENGBING TANG is pursuing the Ph.D. degree with the Department of Aeronautics and Astronautics, Kyoto University. His research interests include Bayesian inference, machine learning, and reinforcement learning.

KENJI FUJIMOTO (Member, IEEE) received the B.Sc. and M.Sc. degrees in engineering and the Ph.D. degree in informatics from Kyoto University, Japan, in 1994, 1996, and 2001, respectively. From 1997 to 2004, he was a Research Associate at the Graduate School of Engineering and Graduate School of Informatics, Kyoto University. From 2004 to 2012, he was an Associate Professor at the Graduate School of Engineering, Nagoya University, Japan. He is currently a Professor with the Graduate School of Engineering, Kyoto University. His research interests include nonlinear control and stochastic systems theory.

ICHIRO MARUTA (Member, IEEE) received the B.E., master's, and Ph.D. degrees in informatics from Kyoto University, in 2006, 2008, and 2011, respectively. He was a Research Fellow of the Japan Society for the Promotion of Science, from 2008 to 2011. From 2012 to 2017, he was an Assistant Professor at the Graduate School of Informatics, Kyoto University. In 2017, he joined as a Lecturer at the Department of Aeronautics and Astronautics, Graduate School of Engineering, Kyoto University, where he has been an Associate Professor, since 2019.