Proceedings of Machine Learning Research vol 242:1400–1414, 2024 6th Annual Conference on Learning for Dynamics and Control

Increasing Information for Model Predictive Control with Semi-Markov Decision Processes

Rémy Hosseinkhan-Boucher* REMY.HOSSEINKHAN@UNIVERSITE-PARIS-SACLAY.FR
Stella Douka* DOUKA.STYLIANI@UNIVERSITE-PARIS-SACLAY.FR
Onofrio Semeraro ONOFRIO.SEMERARO@UNIVERSITE-PARIS-SACLAY.FR
Lionel Mathelin LIONEL.MATHELIN@LISN.UPSACLAY.FR

Université Paris-Saclay, Orsay, France
CNRS, Laboratoire Interdisciplinaire des Sciences du Numérique, Orsay, France

Editors: A. Abate, M. Cannon K. Margellos, A. Papachristodoulou

Abstract

Recent works in Learning-Based Model Predictive Control of dynamical systems show impressive sample complexity performances using criteria from Information Theory to accelerate the learning procedure. However, the sequential exploration opportunities are limited by the system local state, restraining the amount of information of the observations from the current exploration trajectory. This article resolves this limitation by introducing temporal abstraction through the framework of Semi-Markov Decision Processes. The framework increases the total information of the gathered data for a fixed sampling budget, thus reducing the sample complexity.

Keywords: Expected Information Gain; Temporal Abstraction; Sample Complexity

1. Introduction

Machine Learning Control (MLC) is an interdisciplinary area of statistical learning and control theory which solves model-free optimal control problems (Duriez et al., 2016). Among the multiple approaches of the vast field of data-driven control, two classes have received notable attention by the machine learning community: *Learning-Based Model Predictive Control (LB-MPC)* (Hewing et al., 2020) and *Model-Based Reinforcement Learning (MB-RL)* (Abbeel et al., 2006; Recht, 2018; Moerland et al., 2022). The former refers to the combination of *Model Predictive Control (MPC)*, an optimisation method based on a sufficiently descriptive model of the system dynamics (Grüne and Pannek, 2011), and learning methods which enable the improvement of the prediction model from recorded data while possibly modeling uncertainty (Aswani et al., 2013; Koller et al., 2019). The latter combines general function approximators such as linear models (Tsitsiklis and Van Roy, 1997), or more generally neural networks (Sutton et al., 1999a), with *Dynamic Programming (DP)* (Bellman, 1957) principles to solve the underlying optimisation problem.

Despite the recent impressive results in learning complex dynamical models (Ha and Schmidhuber, 2018), the *sample complexity* of the learning process remains a major issue in the field of data-driven control (Kakade, 2003; Li et al., 2021, and see the references therein), in which the sample complexity is defined as the sample size required to learn a good approximation of the target concept (Mohri et al., 2018). For this reason, recent works (Mehta et al., 2022b,a) in LB-MPC have

*Both authors contributed equally to this work.

focused on the design of exploration strategies based on the Information Theory concept of *Expected Information Gain (EIG)* or negative *Conditional Mutual Information (CMI)* (Lindley, 1956). The resulting criterion allows for quantifying the gain of information given by a new state-control observation on the estimated optimal system trajectory. Hence, this tool can be used as an acquisition function to guide the exploration of the state-control space. The concept of acquisition function is borrowed from the field of *Bayesian Optimisation (BO)*. In particular, the work of Mehta et al. (2022b) relies on the broader black-box BO framework of Neiswanger et al. (2021).

In a setting where the data is collected along the trajectory of the dynamical system of interest, the diversity of the resulting dataset (which may be characterised by the quantity of information) is conditioned on the subsequent states of the system. Informally, the setting in which the sampling procedure is constrained by the current system state may introduce information redundancy if the system exhibits high auto-correlation or if the current state is in a slowly evolving region of the state space. Indeed, as shown in Figure 1 (auto-correlation from a perturbated fixed point of a controlled Lorenz 63' system), the auto-correlation from an initial state can be high in average for a long period of time while the control intensity allows to reduce the correlation of the sequence of states.

However, for dynamical systems characterised by a broad range of time scales, the notion of *temporal abstraction*, described in the below paragraphs, (Precup, 2000; Machado et al., 2023) may play a key role in overcoming the issue mentioned here.

Abstraction in Artificial Intelligence refers to a broad range of techniques in order to provide a more compact representation of the problem at hand (Boutilier and Dearden, 1994; Banse et al., 2023). In the framework of *Markov Decision Process (MDP)*, the work of Sutton et al. (1999b) sheds light on the limitation induced by standard MDP modeling: "There is no notion of a course of action persisting over a variable period of time. [...] As a consequence, conventional MDP methods are unable to take advantage of the simplicities and efficiencies sometimes available at higher levels of temporal abstraction."

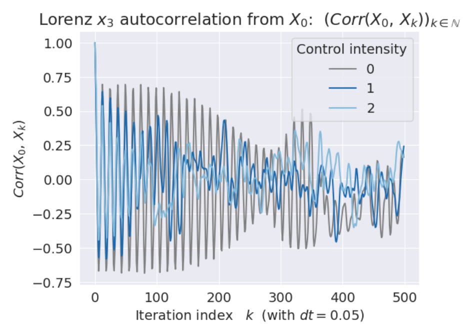


Figure 1: $(Cov(X_0, X_k))_{k\in\mathbb{N}}$ for the controlled Lorenz system x_3 component under multiple control intensities.

Temporal abstraction can refer to the concept of selecting the right level of time granularity to facilitate the description of the world model to achieve a given task. In simpler words, in the present case, temporal abstraction is the idea of representing and reasoning about actions and states at different time-scales and duration.

In the present article, temporal abstraction through *Semi-Markov Decision Processes (SMDP)* modeling is introduced to improve the informativeness of the sequential exploration of the state-control space. SMDP modeling is shown to obtain a better sample complexity of the dynamics model estimator. This article thus extends the previous work of Mehta et al. (2022a) by introducing temporal abstraction to the acquisition function. The paper is organised as follows. Section 2 reviews the related works. Section 3 introduces the problem setting. Section 4 presents the hypothesis and the experimental setup while Section 5 presents the results and Section 6 concludes the paper.

2. Related Works

Information Driven Model-Based Control The foundations of the *Bayesian Experimental Design* have been laid by the seminal work of Lindley (1956) where the author presents a measure of the information provided by an experiment. More recently, MacKay (1992) termed *Expected Information Gain (EIG)* a measure of the information provided by an observation allowing, in his own terms, to *actively select particularly salient data points*. In the field of LB-MPC, such a criterion has been used to cherry-pick the most informative state-control pair to learn the dynamics of the system (Mehta et al., 2022b,a). Their work is based on the broader Bayesian Optimisation method of Neiswanger et al. (2021) designed to optimise "blackbox" functions. An extensive review of Bayesian Optimisation and its applications is available in this latter paper.

Learning-Based Model Predictive Control The history of learning-based models may be traced back to the seminal work by Stratonovich (1960) in probability theory which stimulated several contributions, notably the work of Kalman and Bucy (1961), that were to compose a body of work generally referred to as filtering theory. More recently, Kamthe and Deisenroth (2018) model the system dynamics with Gaussian Processes (GP) and use MPC for data efficiency. GPs are also used in the PILCO model (Deisenroth and Rasmussen, 2011) which has a high influence in MB-RL. Koller et al. (2019) model the uncertainty of the system dynamics for safe-RL. The work of Bonzanini and Mesbah (2020) presents a stochastic LB-MPC strategy to handle this uncertainty.

Semi-Markov Decision Processes Temporal abstraction in reinforcement learning was pioneered in Sutton (1995) and Precup and Sutton (1997); Precup (2000). Specifically, Sutton (1995) proposed learning a model and value function at different levels of temporal abstraction. The actions in SMDPs take variable amounts of time and are intended to model temporally-extended courses of action. Recent works for continuous-time control use variants of Neural Ordinary Differential Equations to model dynamics delays (Du et al., 2020; Holt et al., 2023). A classical use of SMDP is for queueing control and equipment maintenance (Puterman, 2014) where time-delays are prominent.

3. Problem Setting

3.1. Control Model

This work considers a control model given by the following d-dimensional discrete-time dynamical system X (Duflo, 1997) on a probability space $(\Omega, \mathcal{F}, P)$ defined by

$$\begin{aligned} X_{k+1} &= F\left(X_k,\, U_k,\, \eta_k\right) \\ X_0 &\sim \mathcal{N}(x_e,\, \sigma_e^2 I_d) \end{aligned} \tag{1}$$

with $X_k \in \mathcal{X}$, $U_k \in \mathcal{U}$ and $\eta_k \in \mathcal{Z}$ for any $k \in \mathbb{N}$, where $\mathcal{X}$, $\mathcal{U}$ and $\mathcal{Z}$ are respectively the corresponding state, control and disturbance spaces. The initial state starts from a reference state x_e (a system equilibrium or *fixed point*[1]) on which centered Gaussian noise with diagonal covariance is additively applied, $X_0 \sim \mathcal{N}(x_e,\, \sigma_e^2 I_d)$. The *i.i.d.* random process $(\eta_k)_{k\in\mathbb{N}}$ is such that η_k is independent of all previous states and controls for any $k \in \mathbb{N}$. The distribution of η_k for any $k \in \mathbb{N}$ is denoted by P_η. Coupled with the dynamics, an instantaneous cost function $c : X \times U \to \mathbb{R}_+$ is also given to define the control model.

[1] In this work a fixed point is considered as a point of the state space $x_e \in \mathcal{X}$ such that $F(x_e, 0) = x_e$.

In the sequel, it will be convenient to define the control model as a *Markov Control Model (MCM)* (Hernández-Lerma, 1989) defined by the following transition probability $\mathcal{P}$ on $\mathcal{X} \times \mathcal{U}$:

$$\mathcal{P}(B_{\mathcal{X}},\, (x,u)) = \int_{\mathcal{Z}} 1_{B_{\mathcal{X}}}\left(F(x,u,z)\right) P_\eta(dz) = P_\eta(\{z \in \mathcal{Z} \mid F(x,u,z) \in B_{\mathcal{X}}\}) \tag{2}$$

for any $B_{\mathcal{X}} \in \mathcal{B}(\mathcal{X})$ (Borel σ-algebra) and $(x,u) \in \mathcal{X} \times \mathcal{U}$. The function 1 is the indicator function.

Hence, the conditional distribution of X_{k+1} given X_k and U_k is given by

$$\mathcal{P}(B_{\mathcal{X}}, (x,u)) = P(X_{k+1} \in B_{\mathcal{X}} \mid X_k = x, U_k = u) \tag{3}$$

Additionally, in this context, a *policy* π is a transition probability on $\mathcal{U}$ given $\mathcal{X}$, *i.e.*, a distribution on controls conditioned on states. In the rest of the paper, $\pi\,(x, du) = \delta_{\{u\}}$ is the Dirac measure at u. Hence the notation is simplified to $\pi\,(x) = u$.

Together, a control model, a policy π and an initial distribution on $\mathcal{X}$ define a stochastic process with distribution P^π on the space of trajectories $(\mathcal{X} \times \mathcal{U})^T$. The distribution of the process is given by $P(dx_0 du_0 dx_1 \dots) = P_{X_0}(dx_0)\pi(du_0 \mid dx_0)\mathcal{P}(dx_1 \mid dx_0, du_0)\dots$ More details on the stochastic process are given in Hernández-Lerma and Lasserre (1996); Puterman (2014). Lastly, the history process $(H_k)_{k\in\mathbb{N}}$ is defined as $H_k = (X_0,\, U_0, \dots, X_k)$ for any $k \in \mathbb{N}$. When $k = T$, H_T is called the *trajectory* of the process. The process $(X_k, U_k, X_{k+1})_{k\in\mathbb{N}}$ is called the *transition process* and the marginal process $(X_k)_{k\in\mathbb{N}}$ is called a *Markov Decision Process (MDP)*.

3.2. Control Problem

The studied control problem is to find a policy π^* which minimises the following performance criterion

$$J^\pi = \mathbb{E}^\pi\left[\sum_{k=0}^{T} c\left(X_k, U_k\right)\right] \tag{4}$$

where $T \in \mathbb{N}$ is a given time-horizon and $\mathbb{E}^\pi$ denotes the expectation under the probability measure P^π. The quantity J^π is called the value function or objective function. The history process under π^* is called the optimal history process and is denoted by $(H_k^*)_{k\in\mathbb{N}}$ and the random variable H_T^* is called the *optimal trajectory*.

In this work, the optimal policy π^* is estimated with *Model Predictive Control (MPC)* applied on a model of the dynamics. The MPC approach consists of solving a finite-horizon optimal control problem at each time step, formally it defines the following policy

$$\pi^{\text{MPC}}(x) = u_0^* \tag{5}$$

$$s.t. \quad (u_0^*,\, \dots,\, u_{T^{\text{MPC}}}^*) = \underset{(u_0,\, \dots,\, u_{T^{\text{MPC}}})}{\arg\min}\; \mathbb{E}^{(u_0,\, \dots,\, u_{T^{\text{MPC}}})}\left[\sum_{k=0}^{T^{\text{MPC}}} c\left(X_k, u_k\right) \mid X_0 = x\right] \tag{6}$$

where $T^{\text{MPC}} \leq T$ is the MPC planning horizon, $x \in \mathcal{X}$ is the current state and the expectation is taken with the underlying probability measure $P^{(u_0,\, \dots,\, u_{T^{\text{MPC}}})}$ which characterised by a deterministic policy (Dirac measures) concentrated at u_k for all $0 \leq k \leq T^{\text{MPC}}$. The policy obtained with MPC on $\mathcal{P}$ is denoted by π^{MPC}. The history process under π^{MPC} is denoted by $H^{\text{MPC}} = (H_k^{\text{MPC}})_{k\in\mathbb{N}}$, it is an approximation of the optimal history process $(H_k^*)_{k\in\mathbb{N}}$ and the random variable H_T^{MPC} is an approximation of the optimal trajectory H_T^*. The objective function under π^{MPC} is denoted by J^{MPC}. The MPC procedure is here performed with the *iCEM* algorithm, an improved version of the *Cross Entropy Method (CEM)* (Rubinstein and Kroese, 2004; Pinneri et al., 2021), a zeroth-order optimisation algorithm based on Monte-Carlo estimation.

3.3. Gaussian Process Modeling

The use of *Gaussian Process (GP)* regression to model relevant quantities of controlled dynamical systems has long been proposed (Kuss and Rasmussen, 2003; Deisenroth and Rasmussen, 2011; Kamthe and Deisenroth, 2018) notably for its distributional nature, thus its ability to model uncertainty. By definition, a GP is a stochastic process (here indexed by $\mathcal{X} \times \mathcal{U}$) such that any finite collection of random variables has a joint Gaussian distribution.

Continuing from the aforementioned papers, GP regression is used to model the transition probability $\mathcal{P}$ with a model estimator $\hat{\mathcal{P}}_{\mathcal{D}}$ such that

$$\hat{\mathcal{P}}_{\mathcal{D}}(\;\cdot\;, (x,u)) \sim \mathcal{N}\left(\mu(x,u),\, \Sigma\left((x,u),\, (x,u)\right) \mid \mathcal{D}\right) \tag{7}$$

where μ and Σ are respectively the mean and covariance functions of the GP and $\mathcal{D}$ is a dataset of observations from the transition process $(X_k, U_k, X_{k+1})_{k\in\mathbb{N}}$. The distribution $\hat{\mathcal{P}}_{\mathcal{D}}$ of Equation (7) is the predictive posterior distribution of the GP conditioned on the dataset $\mathcal{D}$ (the reader is referred to Rasmussen and Williams (2006) for more details on GP regression). The processes $\hat{X}$, $\hat{U}$ and $\hat{H}$ are respectively the state, control and history processes under the approximate model and the same rules of notation apply as for the original processes. The MPC policy obtained with the approximate model $\hat{\mathcal{P}}_{\mathcal{D}}$ is denoted by $\widehat{\pi}^{\text{MPC}}$. The history process under $\widehat{\pi}^{\text{MPC}}$ is denoted by $\widehat{H}^{\text{MPC}} = (\widehat{H}_k^{\text{MPC}})_{k\in\mathbb{N}}$ and the objective function under $\widehat{\pi}^{\text{MPC}}$ is denoted by $\widehat{J}^{\text{MPC}}$.

Notably, *this work focuses on the sample complexity required to estimate a model $\hat{\mathcal{P}}_{\mathcal{D}}$ of the true dynamics $\mathcal{P}$ accurate enough to obtain a MPC policy $\widehat{\pi}^{MPC}$ that is close to the optimal policy* π^*.

Hence, two time units are considered: the sampling iteration n which represents the number of observations gathered from the system so far, and the time index k of the current state X_k of the underlying dynamical system X. It is supposed in the following that $n \leq k$: it is not possible to gather more observations than the number of time steps of the system.

3.4. Expected Information Gain

For a fixed sampling budget n and a fixed configuration (*e.g.* the horizon T^{MPC}, the number of samples for the Monte-Carlo estimation of the cost or the other hyper-parameters of the iCEM algorithm) to perform the MPC procedure π^{MPC}, the control performance mainly lies in the quality of the model estimator $\hat{\mathcal{P}}_{\mathcal{D}_n}$. It depends on two main elements: the choice a priori of the mean and kernel functions μ and Σ and the collection $\mathcal{D}_n$ of n observations. From the work of Mehta et al. (2022b,a), the selection of the observations can be guided by the maximisation of the *Expected Information Gain (EIG)* on the optimal trajectory.

Let suppose the time iteration k of the underlying observed process X is equal to the number of samples gathered, *i.e.*, $k = n$ and the dataset is already collected[2] at the sampling iteration n such that $\mathcal{D}_n = ((x_i, u_i, x'_i))_{i=0}^{n-1}$ and denote by (X_n, U_n) a new random state-control pair to draw

[2]In this specific case of $k = n$, the dataset $\mathcal{D}_n$ simply contains the whole past trajectory of X, it is a realisation of H_n, in other words $\mathcal{D}_n = H_n(\omega)$ for some random outcome $\omega \in \Omega$.

from the system. The goal is to select the state-control pair (x, u) that maximises the *Expected Information Gain* EIG on the optimal trajectory which is defined by

$$\mathrm{EIG}_n(x, u) = \mathcal{H}\left[\hat{H}_T^* \mid \mathcal{D}_n\right] - \mathbb{E}_{P_{X_{n+1}|\mathcal{D}_n,\, X_n=x,\, U_n=u}}\left[\mathcal{H}\left[\hat{H}_T^* \mid \mathcal{D}_n, X_n = x, U_n = u, X_{n+1}\right]\right] \tag{8}$$

where $\mathcal{H}$ denotes the differential entropy of a random variable. In other words, given a level of uncertainty $\mathcal{H}\left[\hat{H}_T^* \mid \mathcal{D}_n\right]$ on the optimal trajectory $\hat{H}_T^*$, the EIG measures the reduction of this uncertainty when the dataset of the model estimator is augmented with the transition tuple (x, u, X_{n+1}).

An intriguing interpretation can be made by noticing that (8) is also equal to the negative *Conditional Mutual Information (CMI)* (Pinsker, 1964; Cover and Thomas, 2006) of the optimal trajectory $\hat{H}_T^*$ and the new state X_{n+1} given the dataset $\mathcal{D}_n$ and the state-control pair (X_n, U_n) [3]. Thus, maximising the EIG is equivalent to minimising the CMI between the optimal trajectory and the new transition tuple, hence tending to draw new states sharing less information with the optimal trajectory conditioned on the dataset $\mathcal{D}_n$ and the event $(X_n = x, U_n = u)$. Indeed, by definition, the CMI quantifies the independence between the distribution of the optimal trajectory and the distribution of the new state given both the dataset and the current state-action pair.

By symmetry of the EIG, a more tractable formulation is given by

$$\mathrm{EIG}_n(x, u) = \mathcal{H}\left[X_{n+1} \mid \mathcal{D}_n,\, X_n = x,\, U_n = u\right] - \mathop{\mathbb{E}}_{P_{\hat{H}_T^*|\mathcal{D}_n}}\left[\mathcal{H}\left[X_{n+1} \mid \mathcal{D}_n, X_n = x,\, U_n = u,\, \hat{H}_T^*\right]\right] \tag{9}$$

It is in practice estimated by Monte-Carlo sampling as detailed in Section 4.

In the original work of Mehta et al. (2022b), the EIG is maximised with a greedy Monte-Carlo algorithm (uniform sampling) that selects the next state-control pair (x, u) to interact with the true system and subsequently update the dataset $\mathcal{D}_n$ with the new transition tuple $(x,\, u,\, x')$ where x' is sampled from the true transition probability $\mathcal{P}(\;\cdot\;, (x, u))$. It assumes any state-control pair (x, u) can be evaluated and queried at any time step. The authors' algorithm is called *Bayesian Active Reinforcement Learning (BARL)*; the dataset and EIG obtained with this algorithm are denoted by $\mathcal{D}_n^{\mathrm{BARL}}$ and $\mathrm{EIG}^{\mathrm{BARL}}$ respectively. In this setting, the dataset support is the whole state-control space, $\mathrm{Supp}(\mathcal{D}_n^{\mathrm{BARL}}) = (\mathcal{X} \times \mathcal{U} \times \mathcal{X})^n$.

However, in many real-world applications, the system is not always controllable and the state-control pairs that can be queried are limited to a subset induced by the system trajectory. This constraint has been considered in the work following the original paper (Mehta et al., 2022a) where the authors proposed to restrict the dataset support to the trajectory of the system. This second algorithm is called[4] *Trajectory Information Planning (TIP)* and similarly the dataset and EIG obtained with this algorithm are denoted by $\mathcal{D}_n^{\mathrm{TIP}}$ and $\mathrm{EIG}^{\mathrm{TIP}}$ respectively.

In this case, the dataset support is limited to the trajectory of the system, $\mathrm{Supp}(\mathcal{D}_n^{\mathrm{TIP}}) \subseteq \{((x_k,\, u_k,\, x_{k+1}))_{k=1}^n \in (\mathcal{X} \times \mathcal{U} \times \mathcal{X})^n \mid \exists (z_k)_{k=1}^n \in \mathcal{Z}^n,\, x_{k+1} = F(x_k,\, u_k,\, z_k),\, 0 \leq i \leq n\} \subseteq (\mathcal{X} \times \mathcal{U} \times \mathcal{X})^n = \mathrm{Supp}(\mathcal{D}_n^{\mathrm{BARL}})$. This set inclusion implies that the optimal EIG obtained with TIP is lower than the one obtained with BARL provided the transition probability estimator

[3] Here and after, a slight abuse of notation is made as the dataset $\mathcal{D}_n$ should be written $\mathcal{D}_n = ((x_i, u_i, {x'}_i))_{i=0}^{n-1}$ since the sole random quantities are X_n and $\hat{H}_T^*$ but it is omitted for the sake of readability.

[4] It is important to mention that the main asset of TIP is to provide a whole trajectory as input to the EIG, which is not used in this work. Thus, only the property of querying observations by following the trajectory of the system is used here.

$\hat{\mathcal{P}}_{\mathcal{D}_n}$ are the same for both algorithms for a fixed current state $x \in \mathcal{X}$, $\max_{\{(x,u),\, u' \in \mathcal{U}\}} \mathrm{EIG}(x, u') \leq \max_{\{(x',u') \in \mathcal{X} \times \mathcal{U}\}} \mathrm{EIG}(x', u')$.

Besides, the latter algorithm (TIP) do not take into account the potential benefits of including dynamics time scales in the sampling process. In the next section, an extension of the TIP algorithm is proposed to increase the EIG for each of the sampling iteration through the *introduction of delayed state-control pairs in the setting of Semi-Markov Decision Processes (SMDP)*. The new algorithm builds upon TIP by considering the inclusion of temporally-extended actions in the data-collection procedure to reach more distant system states that are not reachable with the original TIP algorithm, hence increasing the amount of information gathered from the system. A similar use of action repetition improves learning in Deep-RL (Sharma et al., 2017; Lakshminarayanan et al., 2017).

3.5. Semi-Markov Decision Processes Extension

A formal definition of *temporal abstraction* is given through the concept of *options* defined by Sutton et al. (1999b) where it refers to *temporally extended courses of action*. This concept has been shown by Parr (1998) to be equivalent to the construction of *Semi-Markov Decision Processes (SMDP)* which are defined below.

Let call *decision epoch* the time index k of the underlying dynamics $(X_k)_{k \in \mathbb{N}}$ defined by equation (1). *Semi-Markov Control Models (SMCM)* generalise the concept of MCM by letting the decisions be random variables. Indeed, consider a strictly increasing random sequence $(\kappa_j)_{j \in \mathbb{N}}$ of integers. The random quantities $\tau_j = \kappa_j - \kappa_{j-1}$ with support in some finite space $\mathcal{T} \subsetneq \mathbb{N} \setminus \{0\}$ are called *inter-decision times* and the random index κ_j are called *random decision epochs*. The resulting stochastic process $(X_{\kappa_j})_{j \in \mathbb{N}}$ is called a *semi-Markov Decision Process*. For a more detailed probabilistic construction, see (Puterman, 2014, p. 534) and (Hernández-Lerma, 1989, p. 15).

In the scope of this paper, *SMDP are used to model the temporal extension of the control process*. The corresponding SMCM is introduced by first extending the control space from $\mathcal{U}$ to $\mathcal{U} \times \mathcal{T}$ such that the temporal extension of the control is encoded in the last coordinate of the control tuple, and the new dynamics is given by $\mathcal{P}^{\mathrm{SMDP}}(dx' \mid (x, (u, t))) = P(X_{k+t} \mid X_k = x, U_{k:k+t-1} = u)$ where $U_{k:k+t-1} = u$ means that *the control process is constant* between k and $k + t - 1$. The latter definition illustrates the fact that during the inter-decision time $\tau = t$, the control process is constant and equal to u.

From now on, this construction allows to enlarge the support of the dataset $\mathcal{D}_n$, for a fixed number of observations n while maintaining a rollout, trajectory-based sampling procedure. Indeed, the dataset support is now $\mathrm{Supp}(\mathcal{D}_n^{\text{SM-TIP}}) \subseteq \{((x_{k_j}, u_{k_j}, x_{k_j+1}))_{j=1}^n \in (\mathcal{X} \times \mathcal{U} \times \mathcal{X})^n \mid \exists (z_k)_{k=1}^{n \sup(\mathcal{T})} \in \mathcal{Z}^{n \sup(\mathcal{T})},\, x_{k+1} = F(x_k, u_k, z_k),\, 0 \leq k \leq n \sup(\mathcal{T}),\, (k_j)_{j=1}^n \in \mathcal{T}^n,\, k_j < k_{j+1}\}$, the transitions tuples extracted from the set of all possible subsequences of the trajectory up to the maximal reachable time value.

Therefore, $\mathrm{Supp}(\mathcal{D}_n^{\text{TIP}}) \subseteq \mathrm{Supp}(\mathcal{D}_n^{\text{SM-TIP}})$. Consequently, this suggests an extension of the EIG to the SMDP setting. Let $t \in \mathcal{T}$ be an inter-decision time and $\mathcal{D}_n^{\text{SM-TIP}}$ be the dataset under the SMDP setting at the sampling iteration n, the resulting $\mathrm{EIG}_n^{\text{SM-TIP}}(x, (u, t))$ is defined as

$$\mathcal{H}[X_{\kappa_n+t+1} | \mathcal{D}_n,\, X_{\kappa_n} = x, U_{\kappa_n:\kappa_n+t} = u, \kappa_n] - \mathbb{E}_{P_{\hat{H}_T^* | \mathcal{D}_n}}\Big[\mathcal{H}\Big[X_{\kappa_n+t+1} \,|\, \mathcal{D}_n, X_{\kappa_n} = x, U_{\kappa_n:\kappa_n+t} = u, \hat{H}_T^*, \kappa_n\Big]\Big] \tag{10}$$

Hence, this measure allows the introduction of temporal abstraction in the sampling procedure by considering the inter-decision delay to increase the potential information gain. However, despite being tractable in trajectory rollout settings, the metric defined by (10) needs to look ahead in the future to be computed (non-causal). Last, note that $\text{EIG}^{\text{SM-TIP}}(x, u, 1) = \text{EIG}^{\text{TIP}}(x, u)$.

4. Method and Experiments

The main objective of this work is to demonstrate the increase in the total information gathered from a system with the introduction of temporal abstraction via the $\text{EIG}^{\text{SM-TIP}}$ measure. To this end, a comparison between the original TIP algorithm and the proposed SMDP extension is performed on two controlled dynamical systems, the Inverted Pendulum (Trélat, 2005) and the Lorenz Attractor (Vincent and Yu, 1991).

The algorithm controls the path of the dynamical system $(X_k)_{k\in\mathbb{N}}$ and collects observations $(X_i,\, U_i,\, X_{i+1})_{i=0}^{n-1}$ to populate the dataset $\mathcal{D}_n$ and improve the GP transition probability estimator $\hat{\mathcal{P}}_{\mathcal{D}_n}$. The indices n and k are respectively the sampling iteration and the time index of the underlying dynamical system $(X_k)_{k\in\mathbb{N}}$. The TIP algorithm supposes $n = k$ (data collected at each time step) while $n \leq k$ (there are time steps where no data is collected) for the SMDP extension. In the SMDP case, the inter-decision time τ_n rules the optional sampling procedure which defines the random decision epochs $\kappa_n = \kappa_{n-1} + \tau_n$. The random decision epochs κ_n define when the algorithm can query the system $(X_k)_{k\in\mathbb{N}}$.

To estimate $\text{EIG}_n^{\text{SM-TIP}}$, a collection of bootstrapped future states, candidate control points and inter-decision times are sampled. The bootstrapped future states $X_{\kappa_n+t} = x_t$ are estimated with the GP model. This may lead to a bias in the estimation of the EIG due to the bootstrapping error. The candidate control points and inter-decision times $(u,\, t)$ are sampled from a uniform distribution $U(\mathcal{U} \times \mathcal{T})$ at time κ_n to solve $\arg\max_{(u,\,t)\in\mathcal{U}\times\mathcal{T}} \text{EIG}_n^{\text{SM-TIP}}(x_t, (u, t))$. In this work, $\mathcal{T} = \{1, \ldots, t_{\max}\}$ for some $t_{\max} \in \mathbb{N}$. The $\text{EIG}_n^{\text{SM-TIP}}$ is estimated by the Monte-Carlo estimator $\widehat{\text{EIG}}_n^{\text{SM-TIP}}(x, (u, t))$ given by

$$\mathcal{H}\left[X_{\kappa_n+t+1} \mid \mathcal{D}_n, X_{\kappa_n} = x, U_{\kappa_n+t} = u, \kappa_n\right] - \frac{1}{m}\sum_{i=1}^{m} \mathcal{H}\Big[X_{\kappa_n+t+1} \mid \mathcal{D}_n, X_{\kappa_n} = x, U_{\kappa_n+t} = u, \hat{H}_{T_i}^{\text{MPC}}, \kappa_n\Big] \tag{11}$$

where m is the number of Monte-Carlo samples of the optimal trajectory $\hat{H}_{T_i}^{\text{MPC}}$ under $\hat{\mathcal{P}}_{\mathcal{D}_n}$. The entropy values are easily computed since the conditional distribution of the new state given the dataset and the current state-control pair is a Gaussian distribution with mean and covariance given by the GP posterior. More details on this procedure and the settings used are available in the paper of Mehta et al. (2022a).

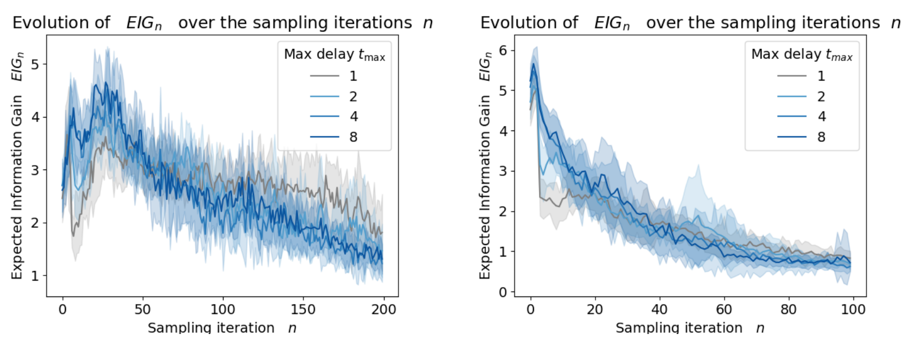
Every two sampling iterations n, the MPC policy $\widehat{\pi}^{\text{MPC}}$ is evaluated on the true system and the objective function is computed. Four independent experiments with different maximal inter-decision time $t_{\max} \in \{1, 2, 4, 8\}$ are performed. For each of the experiments, the algorithm is run for 10 independent trials (seeds) to alleviate the variability proper to data-driven control methods (Henderson et al., 2018). The cost function is defined as $c(x, u) = \|x\|^2$ in the case of the Lorenz attractor while the classic Gym (Brockman et al., 2016) cost function (also norm-based) is used for the Inverted Pendulum. The sampling budget is set to $n_{\max} = 100$ for the Lorenz system and $n_{\max} = 200$ for the Inverted Pendulum. To implement the SMDP, the system is stepped forward in time with the action kept constant during inter-decision times. Details on the implementation and experimental settings are available on https://github.com/ReHoss/lbmpc_semimarkov.

5. Results

Among the relevant quantities to be reported, the evolution of the EIG, the interdecision times and the evaluation of the objective function are of interest to question the hypothesis raised in Section 4.

First, the evolution of the amount of information gathered during sampling through a comparison of $(\text{EIG}_n^{\text{TIP}})_{n=1}^{n_{\max}}$, and $(\text{EIG}_n^{\text{SM-TIP}})_{n=1}^{n_{\max}}$ presented in Figure 2 to assess the impact of the SMDP extension. Second, the corresponding inter-decision times $(\tau_n)_{n=1}^{n_{\max}}$ are shown in Figure 3 to evaluate the necessity of temporal abstraction. Lastly, the evolution of the objective function $J^{\widehat{\pi}^{\text{MPC}}}$ from 5 fixed initial conditions X_0 is shown as a function of the sampling iteration n in Figure 4 to analyse the effective results of the proposed method. For all the figures, the shaded area represents the standard error over the 10 independent trials.

About the first point, one can observe that in all cases, the EIG is larger for SM-TIP than for TIP ($t_{\max} = 1$) until one-fourth of the sampling budget is reached. This suggests that the SMDP extension is beneficial to the information gathering process at the beginning of the sampling procedure. This may be explained by the fact that the inter-decision times allow to de-correlate the collected states via the same mechanism illustrated in Figure 1. Note also that, in the case of Lorenz (Figure 2(*a*)), the EIG after approximately half of the sampling procedure is superior for TIP than SM-TIP since more information (state-actions pairs minimising the mutual information) remain to be gathered.

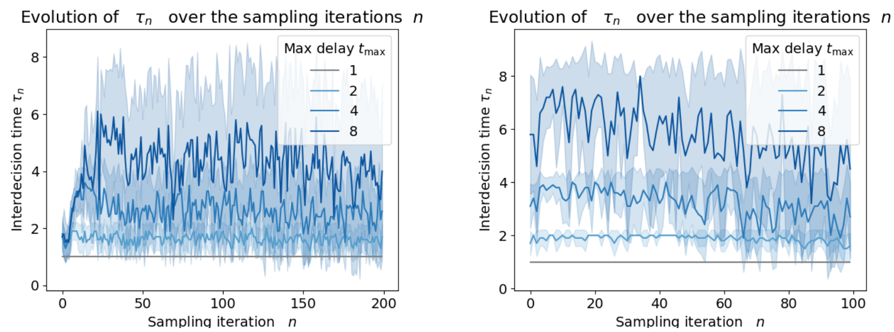


(*a*) Lorenz (*b*) Pendulum

Figure 2: Evolution of the Expected Information Gain $\text{EIG}^{\text{SM-TIP}}$ over the number of sampling iterations.

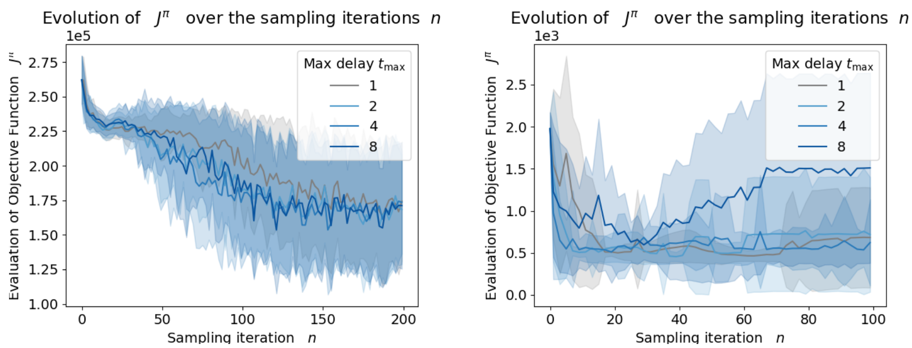
Examining the inter-decision times $(\tau_n)_{n=1}^{n_{\max}}$ on Figure 3, it can first be observed that globally $\tau_n > 1$ for the SMDP algorithms (where $t_{\max} > 1$). This shows that the sequential maximal EIG is approximately reached for inter-decision times that are larger than the original MDP decision times. This confirms the relevance of temporal abstraction to increase the information gathering process. However, the inter-decision times are not necessarily always equal to $t_{\max}$, suggesting the more informative observations are not always the temporally most distant ones.

Moving on to the objective function, in the case of the Lorenz system (Figure 4(*a*)), the evaluation performances show the learning speed is greater for the SM-TIP settings ($t_{\max} > 1$) than for the TIP setting ($t_{\max} = 1$). For the Pendulum case (Figure 4(*b*)), except for the SM-TIP setting where $t_{\max} = 8$, the proposed approach shows better sample complexity since very few iterations are required to reach optimality (light blue curves ($t_{\max} \in \{2, 4\}$) are below the grey curve ($t_{\max} = 1$) for the first (up to $n = 20$) sampling iterations. Furthermore, one of the reasons the $t_{\max} = 8$ fails to achieve optimal performances is likely the *bootstrapping prediction error* (not shown in this document) which increases with $t_{\max}$. Indeed, as mentioned in Section 4 due to the



(*a*) Lorenz (*b*) Pendulum

Figure 3: Inter-decision time τ chosen by the SMDP during training.



(*a*) Lorenz (*b*) Pendulum

Figure 4: Evolution of the objective function $J^{\widehat{\pi}^{\text{MPC}}}$ to evaluate the system during training.

non-causal property of $\text{EIG}^{\text{SM-TIP}}$, there exists a trade-off between the temporal extension of the dynamics to reach the new region of the state space and the bootstrapping error which increases with the temporal extension.

6. Conclusion

This study demonstrates that, when restricted to the trajectory of the system, the total information gathered for a given sampling budget can be increased by introducing temporal abstraction through the usage of SMDPs. Results show that learning the dynamics of the Inverted Pendulum and the Lorenz system is more data-efficient with the use of temporally-extended actions.

Future work may extend this methodology to more complex systems, leveraging the flexibility of SMDPs. These systems have the potential to reach highly informative regions and efficiently capture rapid changes in system dynamics, as the information content can be increased when considering the time resolution as a decision variable.

In summary, this work offers a concise yet comprehensive glimpse into the potential of SMDPs in Model Predictive Control. The results on known systems establish a robust foundation for broader applications and unveil potential future advancements in control strategies.

Acknowledgments

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR), under grant ANR-REASON (ANR-21-CE46-0008). This work was performed using HPC resources from GENCI-IDRIS (Grant 2023-[AD011014278]).

References

Pieter Abbeel, Morgan Quigley, and Andrew Y. Ng. Using inaccurate models in reinforcement learning. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 1–8, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi: 10.1145/1143844.1143845.

Anil Aswani, Humberto Gonzalez, S. Shankar Sastry, and Claire Tomlin. Provably safe and robust learning-based model predictive control. *Automatica*, 49(5):1216–1226, 2013. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2013.02.003.

Adrien Banse, Licio Romao, Alessandro Abate, and Raphael Jungers. Data-driven memory-dependent abstractions of dynamical systems. In Nikolai Matni, Manfred Morari, and George J. Pappas, editors, *Proceedings of The 5th Annual Learning for Dynamics and Control Conference*, volume 211 of *Proceedings of Machine Learning Research*, pages 891–902. PMLR, 15–16 Jun 2023.

Richard E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, 1957. ISBN 9781400835386. doi: doi:10.1515/9781400835386.

Angelo Domenico Bonzanini and Ali Mesbah. Learning-based stochastic model predictive control with state-dependent uncertainty. In Alexandre M. Bayen, Ali Jadbabaie, George Pappas, Pablo A. Parrilo, Benjamin Recht, Claire Tomlin, and Melanie Zeilinger, editors, *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120 of *Proceedings of Machine Learning Research*, pages 571–580. PMLR, 10–11 Jun 2020.

Craig Boutilier and Richard Dearden. Using abstractions for decision-theoretic planning with time constraints. In *AAAI Conference on Artificial Intelligence*, 1994.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym, 2016.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, July 2006. ISBN 0471241954.

Marc Deisenroth and Carl Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on Machine Learning*, pages 465–472, 01 2011.

Jianzhun Du, Joseph Futoma, and Finale Doshi-Velez. Model-based reinforcement learning for semi-markov decision processes with neural odes. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19805–19816. Curran Associates, Inc., 2020.

M. Duflo. *Random Iterative Models*. Applications of Mathematics : Stochastic Modelling and Applied Probability. Springer, 1997. ISBN 9783540571001.

Thomas Duriez, Steven Brunton, and Bernd Noack. *Machine Learning Control – Taming Nonlinear Dynamics and Turbulence*, volume 116. Springer International Publishing, 11 2016. ISBN 978-3-319-40623-7. doi: 10.1007/978-3-319-40624-4.

Lars Grüne and Jürgen Pannek. *Nonlinear Model Predictive Control: Theory and Algorithms*. Springer London, 01 2011. ISBN 978-0-85729-500-2. doi: 10.1007/978-0-85729-501-9.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

O. Hernández-Lerma. *Adaptive Markov Control Processes*. Applied mathematical sciences. Springer-Verlag, 1989. ISBN 9780387969664.

Onésimo Hernández-Lerma and Jean B. Lasserre. *Discrete-Time Markov Control Processes: Basic Optimality Criteria*. Springer New York, 1 edition, 1996. doi: 10.1007/978-1-4612-0729-0.

Lukas Hewing, Kim P. Wabersich, Marcel Menner, and Melanie N. Zeilinger. Learning-based model predictive control: Toward safe learning in control. *Annual Review of Control, Robotics, and Autonomous Systems*, 3(1):269–296, 2020. doi: 10.1146/annurev-control-090419-075625.

Samuel Holt, Alihan Hüyük, Zhaozhi Qian, Hao Sun, and Mihaela van der Schaar. Neural laplace control for continuous-time delayed systems. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 1747–1778. PMLR, 25–27 Apr 2023.

Sham M. Kakade. *On the Sample Complexity of Reinforcement Learning*. Phd thesis, University College London, 2003.

R. E. Kalman and R. S. Bucy. New Results in Linear Filtering and Prediction Theory. *Journal of Basic Engineering*, 83(1):95–108, 03 1961. ISSN 0021-9223. doi: 10.1115/1.3658902.

Sanket Kamthe and Marc Deisenroth. Data-efficient reinforcement learning with probabilistic model predictive control. In Amos Storkey and Fernando Perez-Cruz, editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 1701–1710. PMLR, 09–11 Apr 2018.

Torsten Koller, Felix Berkenkamp, Matteo Turchetta, Joschka Boedecker, and Andreas Krause. Learning-based model predictive control for safe exploration and reinforcement learning, 2019.

Malte Kuss and Carl Rasmussen. Gaussian processes in reinforcement learning. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems*, volume 16. MIT Press, 2003.

Aravind Lakshminarayanan, Sahil Sharma, and Balaraman Ravindran. Dynamic action repetition for deep reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), 02 2017. doi: 10.1609/aaai.v31i1.10918.

Gen Li, Laixi Shi, Yuxin Chen, Yuantao Gu, and Yuejie Chi. Breaking the sample complexity barrier to regret-optimal model-free reinforcement learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 17762–17776. Curran Associates, Inc., 2021.

D. V. Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, 1956. ISSN 00034851.

Marlos C. Machado, Andre Barreto, Doina Precup, and Michael Bowling. Temporal abstraction in reinforcement learning with the successor representation. *Journal of Machine Learning Research*, 24(80):1–69, 2023.

David J. C. MacKay. Information-Based Objective Functions for Active Data Selection. *Neural Computation*, 4(4):590–604, 07 1992. ISSN 0899-7667. doi: 10.1162/neco.1992.4.4.590.

Viraj Mehta, Ian Char, Joseph Abbate, Rory Conlin, Mark Boyer, Stefano Ermon, Jeff Schneider, and Willie Neiswanger. Exploration via planning for information about the optimal trajectory. In *Advances in Neural Information Processing Systems*, 2022a.

Viraj Mehta, Biswajit Paria, Jeff Schneider, Stefano Ermon, and Willie Neiswanger. An experimental design perspective on model-based reinforcement learning. In *International Conference on Learning Representations*, 2022b.

Thomas M. Moerland, Joost Broekens, Aske Plaat, and Catholijn M. Jonker. Model-based reinforcement learning: A survey, 2022.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 2 edition, 2018. ISBN 978-0-262-03940-6.

Willie Neiswanger, Ke Alexander Wang, and Stefano Ermon. Bayesian algorithm execution: Estimating computable properties of black-box functions using mutual information. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8005–8015. PMLR, 18–24 Jul 2021.

R. E. Parr. *Hierarchical Control and Learning for Decision Processes*. PhD thesis, University of California Berkeley, 1998.

Cristina Pinneri, Shambhuraj Sawant, Sebastian Blaes, Jan Achterhold, Joerg Stueckler, Michal Rolinek, and Georg Martius. Sample-efficient cross-entropy method for real-time planning. In

Jens Kober, Fabio Ramos, and Claire Tomlin, editors, *Proceedings of the 2020 Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pages 1049–1065. PMLR, 16–18 Nov 2021.

M. S. Pinsker. *Information and information stability of random variables and processes / by M.S. Pinsker. Translated and edited by Amiel Feinstein.* Holden-Day series in time series analysis. Holden-Day, San Francisco, 1964.

Doina Precup. *Temporal Abstraction in Reinforcement Learning*. PhD thesis, University of Massachusetts Amherst, 2000.

Doina Precup and Richard S Sutton. Multi-time models for temporally abstract planning. In M. Jordan, M. Kearns, and S. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10. MIT Press, 1997.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning.* Adaptive computation and machine learning. MIT Press, 2006. ISBN 026218253X.

Benjamin Recht. A tour of reinforcement learning: The view from continuous control, 2018.

Reuven Y. Rubinstein and Dirk P. Kroese. *The Cross Entropy Method: A Unified Approach To Combinatorial Optimization, Monte-Carlo Simulation (Information Science and Statistics).* Springer-Verlag, Berlin, Heidelberg, 2004. ISBN 038721240X.

Sahil Sharma, Aravind S. Lakshminarayanan, and Balaraman Ravindran. Learning to repeat: Fine grained action repetition for deep reinforcement learning. In *International Conference on Learning Representations*, 2017.

R. L. Stratonovich. Conditional markov processes. *Theory of Probability & Its Applications*, 5(2): 156–178, 1960. doi: 10.1137/1105015.

Richard S. Sutton. Td models: Modeling the world at a mixture of time scales. In *International Conference on Machine Learning*, 1995.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In S. Solla, T. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999a.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1-2):181–211, August 1999b. ISSN 0004-3702. doi: 10.1016/s0004-3702(99)00052-1.

Emmanuel Trélat. Contrôle optimal : théorie et applications, 01 2005.

J.N. Tsitsiklis and B. Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5):674–690, 1997. doi: 10.1109/9.580874.

Thomas L. Vincent and Jianzu Yu. Control of a chaotic system. *Dynamics and Control*, 1(1):35–52, Mar 1991. ISSN 1573-8450. doi: 10.1007/BF02169423.