A Fundamental Trade-off in Continuous Value Estimation: Managing Temporal Resolution

Zichen Zhang* VINCENT.ZHANG@UALBERTA.CA
Francesco Zanini* FZANINI@UALBERTA.CA
Junxi Zhang* JUNXI3@UALBERTA.CA
Alex Ayoub* AAYOUB@UALBERTA.CA
Johannes Kirschner* JKIRSCHN@UALBERTA.CA
Masood Dehghan MASOOD1@UALBERTA.CA
Dale Schuurmans* DAES@UALBERTA.CA
University of Alberta, Canada

Abstract

A default assumption in reinforcement learning and optimal control is that experience arrives at discrete time points on a fixed clock cycle. Many applications, however, involve continuous systems where the time discretization is not fixed but instead can be managed by a learning algorithm. By analyzing Monte-Carlo value estimation for LQR systems we uncover a fundamental trade-off between approximation and statistical error in value estimation. Importantly, these two errors behave differently with respect to time discretization, which implies that there is an optimal choice for the temporal resolution that depends on the data budget. These findings show how adapting the temporal resolution can significantly improve value estimation quality in LQR systems given finite experience. Empirically, we demonstrate the trade-off in numerical experiments.

Keywords: Temporal Discretization, Langevin System, LQR, Policy Evaluation, Continuous Time, Experimental Design

1. Introduction

Many real-world applications of control and reinforcement learning involve systems with continuous state spaces and that evolve continuously in time. For instance, a physical system such as a robot naturally involves continuous control variables. On the other hand, sensor measurements typically arrive at a preset sampling frequency. A common belief is that a finer time discretization always leads to better estimation of the system properties such as the control cost. However, we show that this is only true with an unlimited data budget. With finite data, a higher temporal resolution means that *more* data is collected within *fewer* episodes. This inevitably leads to the question on how to *optimally* choose the time discretization for the task at hand.

In practice, there are always limitations on how much data can be collected, stored and processed. The practitioner hence faces a fundamental trade-off: a high temporal resolution leads to a better approximation of the continuous-time system from discrete measurements, whereas collecting data along a larger number of trajectories leads to lower variance in the

∗. equal contribution

estimation with respect to stochasticity in the system. This is indeed true for any system with stochastic dynamics, even if the learner has access to *exact* (noiseless) measurements of the system's state. In this paper, we show that data efficiency can be improved by leveraging a precise understanding of the trade-off between approximation error and statistical estimation error in long term value estimation — two factors that react differently to the level of temporal discretization.

Contributions We consider a fully continuous and stochastic scenario that permits a tight analysis of the approximation and estimation errors incurred in value estimation. In particular, we consider the canonical case of Monte Carlo value estimation in a Langevin dynamical system (linear dynamics perturbed by a Wiener process) with quadratic instantaneous costs. Although the setup is specialized, it allows the fundamental approximation-estimation trade-off to be clearly identified and exactly characterized. In this scenario, we are able to quantify the effects of temporal discretization with sufficient precision to align with experimental verification.

1.1 Related Work

There is a sizable literature on reinforcement learning in continuous-time systems (e.g. Doya, 2000; Lee and Sutton, 2021; Lewis et al., 2012; Bahl et al., 2020; Kim et al., 2021; Yildiz et al., 2021). But these previous works have largely focused on deterministic dynamics, and do not investigate trade-offs in temporal discretization that allow for its optimization. A smaller body of work has considered learning continuous-time control under stochastic (Baird, 1994; Bradtke and Duff, 1994; Munos and Bourgine, 1997; Munos, 2006), or bounded (Lutter et al., 2021) perturbations, but with a focus on making standard learning methods more robust to small time scales (Tallec et al., 2019), again without explicitly managing the temporal discretization level. There have also been works that characterize the effects of temporal truncation in infinite horizon problems (Jiang et al., 2016; Droge and Egerstedt, 2011). Despite these prevailing topics in the literature, we find that managing temporal discretization offers substantial improvements not captured by these previous studies.

The LQR setting is a standard framework in control theory and it gives rise to a fundamental optimal control problem (Lindquist, 1990), which has proven itself to be a challenging scenario for Reinforcement Learning algorithms (Tu and Recht, 2019; Krauth et al., 2019). The stochastic LQR considers linear systems driven by additive Gaussian noise with a quadratic form for the cost, which is sought to be minimised by means of a feedback controller. Although it is a well-understood scenario and a closed form of the optimal controller is known thanks to the separation principle (Georgiou and Lindquist, 2013), only recently the statistical properties of the long-term cost have been investigated (Bijl et al., 2016). The work in our paper also closely related to the now sizable literature on reinforcement learning in LQR systems (Bradtke, 1992; Krauth et al., 2019; Tu and Recht, 2018; Dean et al., 2020; Tu and Recht, 2019; Dean et al., 2018; Fazel et al., 2018; Gu et al., 2016). These existing works uniformly focused on the discrete time setting, although the benefits of managing spatial rather than temporal discretization has been considered (Sinclair et al., 2019; Cao and Krishnamurthy, 2020). Wang et al. (2020) studied the continuous-time LQR setting but it focused on the exploration problem rather than the temporal discretization.

There is compelling empirical evidence that managing temporal resolution, typically via action persistence (Lakshminarayanan et al., 2017; Sharma et al., 2017; Huang et al., 2019; Huang and Zhu, 2020; Dabney et al., 2021; Park et al., 2021), can greatly improve learning performance. Even grid worlds (Sutton and Barto, 2018) can be seen as leveraging a form of action persistence, where a coarse spatial discretization is imposed on an otherwise continuous two dimensional navigation problem to improve learning efficiency. These empirical findings have recently been supported by an initial theoretical analysis (Metelli et al., 2020) that shows temporal discretization plays a role in determining the effectiveness of fitted Q-iteration. The analysis by Metelli et al. (2020) does not consider fully continuous systems, but rather remains anchored in a base level discretization and only provides worst case upper bounds that do not necessarily capture the detailed tradeoffs one faces in practice.

The task of choosing the temporal resolution can also be understood as an *experimental design* problem (Chaloner and Verdinelli, 1995). By choosing the time-discretization, the experimenter determines how to allocate measurements for a given data budget. More specifically, when considering the mean-square error of the Monte-Carlo estimator, the design objective becomes non-linear (Ford et al., 1989). What is peculiar to our objective is that, for any fixed design, there is a constant approximation error (bias) that persists even when the number of data points becomes infinite. At the same time, the bias can also be managed by scarifying estimation error (variance). Optimal designs that consider the bias-variance trade-off jointly have been studied previously (e.g. Bardow, 2008; Mutny et al., 2020; Mutnỳ and Krause, 2022). For the use of experimental design in reinforcement learning, see, e.g. (Lattimore et al., 2020).

2. One-Dimensional Langevin Systems

We formally study one-dimensional systems that evolves following the Langevin equation:

$$dx(t) = ax(t)dt + \sigma dw(t). \tag{1}$$

Here $x(t) \in \mathbb{R}$ is the state variable, $a \in \mathbb{R}$ is the drift coefficient and $w(t)$ is a Wiener process with scale parameter $\sigma > 0$. This is the prototypical case for evaluating a fixed deterministic policy in the linear quadratic regulator (LQR) framework. We may assume that $a \leq 0$, i.e. the system is stable (or marginally stable).

Let $x_i(t)$ be the sample path (i.e. a realisation of the stochastic dynamics) from a fixed starting state $x(0) = x_0$ for episodes $i = 1, \ldots, M$ and $t \in [0, T]$. We define the finite-time realisation of the cost in episode i as

$$J_i = \int_0^T r_i^2(t)dt = \int_0^T q x_i^2(t)dt, \tag{2}$$

where $r_i(t) = q x_i(t)^2$ is the quadratic cost function for a fixed $q > 0$. Importantly, J_i is a random variable with respect to the stochasticity of the system evolution. The expected cost V is the expectation $V = \mathbb{E}[J_1]$.

Assume that for a total data budget B, we collect samples along M trajectories $\mathcal{D} = \{x_i(t_0), x_i(t_1), ..., x_i(t_{N-1})\}_{i=1}^M$ at discrete time steps $t_k = kh$, where the step-size is $h = T/N$ and $B = NM$. Naturally, we can use a Monte-Carlo estimator (Riemann sum) to estimate

the trajectory cost J_i from discrete-time observations:

$$\hat{J}_i(h) = \sum_{k=0}^{N-1} hqx_i^2(kh). \tag{3}$$

Given data from M episodes, we estimate the expected cost using the empirical average:

$$\hat{V}_M(h) = \frac{1}{M}\sum_{i=1}^{M} \hat{J}_i(h). \tag{4}$$

As our main objective, we are interested in controlling the expected mean-squared error:

$$\mathrm{MSE}(h, B, T, a, \sigma, q) = \mathbb{E}[(\hat{V}_M(h) - V)^2] \tag{5}$$

The mean-squared error is a function of the total data budget B and the step size h, and of the system variables T, a and σ. Since the square of the cost parameter q^2 factors out of the expression, we set $q = 1$ and remove the dependence on q in what follows, in favour of simpler notation. Note that the number of episodes is not a free variable and can be expressed as $M = \frac{Bh}{T}$.

2.1 Main Results

Perhaps surprisingly, the mean-squared error of the Riemann estimator for the Langevin system (1) can be computed in closed form. We do so by obtaining closed-form expressions for the second and forth moments of the random trajectories $x_i(t)$. This result is summarized in the next theorem.

Theorem 1 (Mean-squared error) *The mean-squared error for the expected cost V under Monte-Carlo estimator, $MSE(h, B, T, a, \sigma) = \mathbb{E}[(\hat{V}_M(h) - V)^2]$, is*

$$MSE(h, B, T, a, \sigma) = E_1(h, T, a) + \frac{E_2(h, T, a)}{B},$$

where

$$E_1(h, T, a) = \frac{\sigma^4 \left(-2ah + e^{2ah} - 1\right)^2 \left(e^{2aT} - 1\right)^2}{16a^4 \left(e^{2ah} - 1\right)^2},$$

$$E_2(h, T, a) = \frac{\sigma^4 T \left[h\left(e^{2aT} - 1\right)\left(4e^{2ah} + e^{2aT} + 1\right) - \left(e^{2ah} - 1\right)\left(e^{2ah} + 4e^{2aT} + 1\right)T\right]}{2a^2 \left(e^{2ah} - 1\right)^2}.$$

The proof is provided in Appendix A. While daunting at first sight, the result completely characterizes the error surface as a function of the step size h and the budget B. For instance, given any fixed B, we can optimize h to minimize the mean-squared error *exactly* by searching over possible step-sizes $h_m = T/m$ for $m = 1, \ldots, B$ (assuming knowledge of the system parameters a, σ and T).

In the case of marginal stability, a cleaner form of the MSE emerges from the analysis, which is easier to interpret. Therefore we consider the case of $a = 0$ separately. Taking the limit $a \to 0$ of the previous expression gives the following result:

Corollary 2 (MSE for marginally stable system) *Assume a marginally stable system, $a = 0$. Then the mean-squared error of the Monte-Carlo estimator is*

$$MSE(h, B, T, \sigma) = \frac{\sigma^4 T^2 h^2}{4} + \frac{\sigma^4 T^2}{3B} \cdot \left(\frac{T^3}{h} - 2T^2 + 2hT - h^2 \right).$$

The error can be understood as an *approximation error* (controlled only by h^2), a *variance term* that decreases with the number of episodes as $\frac{1}{M} = \frac{T}{Bh}$, and lower order terms. For a fixed data budget B, h has to be chosen to balance these two terms. Specifically, we get the following:

$$h^*(B, T) := \arg\min_{h>0} \mathrm{MSE}(h, B, T, \sigma) = T \left(\frac{2}{3B} \right)^{1/3} + o(B^{-1/3}). \tag{6}$$

From this, we can compute the optimal number of episodes $M^* \approx \frac{Bh}{T} = \left(\frac{2}{3}\right)^{1/3} B^{2/3}$. We remark that under the assumption $B \gg 1$, we also obtain that $M^* \gg 1$. This is in agreement with the implicit requirement that h is big enough to consider at least one whole trajectory, i.e. $h > T/B$.
Consequently, the mean-squared error for the optimal choice of h is

$$\mathrm{MSE}(h^*, B, T, \sigma) = 3\,(3/2)^{1/3}\,\sigma^4 T^4 B^{-2/3} + \mathcal{O}(B^{-1}).$$

In other words, the optimal error rate as a function of the data budget is $\mathcal{O}(B^{-2/3})$.

We can further obtain a similar form for h^* for the general system, when $a \leq 0$.

Corollary 3 (Optimal step size) *For $B \gg 1$, the optimal step-size is*

$$h^*(B, T, a, \sigma) = \left(-\frac{T\left(4aT - e^{4aT} + e^{2aT}(8aT - 4) + 5\right)}{2a^2(e^{2aT} - 1)^2} \right)^{1/3} B^{-1/3} + o(B^{-1/3}).$$

Moreover, $MSE(h^*, B) = \mathcal{O}(B^{-2/3})$.

Proof Note that the leading terms in h for the error terms are

$$\begin{aligned} E_1(h, T, a) &= \frac{\sigma^4 (e^{2aT} - 1)^2}{8a^2} h^2 + \mathcal{O}(h^3), \\ \frac{E_2(h, T, a)}{B} &= -\frac{\sigma^4 T\left(4aT - e^{4aT} + e^{2aT}(8aT - 4) + 5\right)}{8a^4 B} \cdot h^{-1} + \frac{\sigma^4 T(1 - e^{2aT} + 4aTe^{2aT})}{4a^3 B} \\ &\quad + \mathcal{O}\left(\frac{h}{B}\right). \end{aligned}$$

Solving for the optimal h^* yields the result. ■

3. Experiments

We run numerical experiments on the Langevin dynamical systems to demonstrate the trade-off analyzed in the previous section: results are shown in Fig. 1. We varied the dynamics parameter a in the list $[-16, -8, -4, -2, -1, -0.5, -0.25, 0]$, corresponding to eight different systems. Note that $a = 0$ is a special case, which recovers a scaled Wiener Process (the marginally stable system). The other hyperparameters used in the figure are: $T = 8,\ \sigma = 1,\ q = 1$. To approximate the outer expectation in the objective Eq.(5), we ran the systems for 50 trials and the expected cost V was computed in closed form (in Appendix A). We observed the same trade-off in all systems, shown on the left figure. The higher $|a|$, the lower the variability of the cost for one full trajectory. Therefore the trade-off favours a smaller h since the estimation error is easier to handle. On the right, we showed how the error changes as we change the data budget $B = \{2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}\}$, and the improvement that can be obtained by enlarging it. As illustrated in the plot, as we increase the data budget, the error reduces but the trade-off remains.

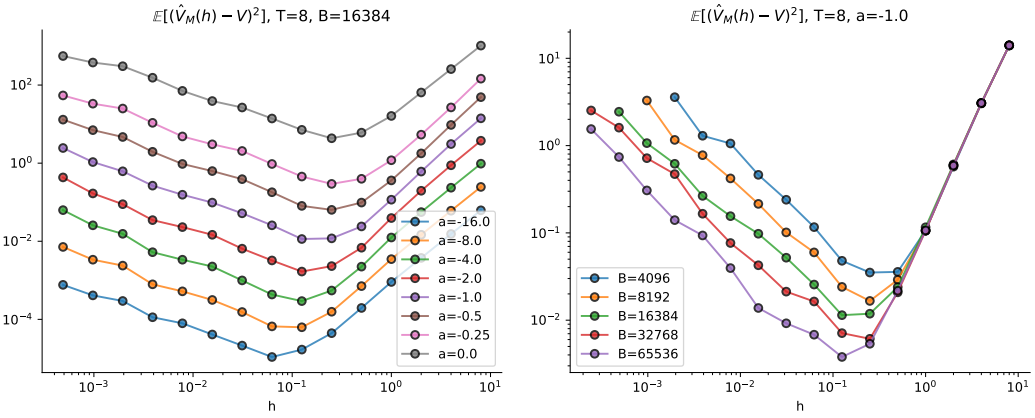

Figure 1: Mean-Squared Error Trade-Off in Langevin dynamical systems

4. Conclusion

We provided a precise characterization of the approximation and estimation errors incurred by Monte-Carlo value estimation in a Langevin dynamical system with quadratic cost. The analysis reveals a fundamental bias-variance trade-off, modulated by the level of temporal discretization h. Simulation experiments confirm that the analysis accurately captures the tradeoff in a precise, quantitative manner. These findings show that managing the temporal discretization level h can greatly improve the quality of value estimation under a fixed data budget B. There are several directions for future work, including considering other value estimation techniques (temporal differencing, system identification), policy optimization, and more general systems, such as non-linear dynamics or non-Gaussian perturbations.

References

Srikhar Bahl, Mustafa Mukadam, Abhinav Gupta, and Deepak Pathak. Neural dynamic policies for end-to-end sensory motor learning. *Advances in Neural Information Processing Systems*, 34, 2020.

Leemon C Baird. Reinforcement learning in continuous time: Advantage updating. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, volume 4, pages 2448–2453. IEEE, 1994.

André Bardow. Optimal experimental design of ill-posed problems: The meter approach. *Computers & Chemical Engineering*, 32(1-2):115–124, 2008.

Hildo Bijl, Jan-Willem van Wingerden, Thomas B Schön, and Michel Verhaegen. Mean and variance of the lqg cost function. *Automatica*, 67:216–223, 2016.

Steven Bradtke. Reinforcement learning applied to linear quadratic regulation. In S. Hanson, J. Cowan, and C. Giles, editors, *Advances in Neural Information Processing Systems*, volume 5. Morgan-Kaufmann, 1992. URL `https://proceedings.neurips.cc/paper/1992/file/19bc916108fc6938f52cb96f7e087941-Paper.pdf`.

Steven J. Bradtke and Michael O. Duff. Reinforcement learning methods for continuous-time Markov decision problems. In *Advances in Neural Information Processing Systems*, 1994.

Tongyi Cao and Akshay Krishnamurthy. Provably adaptive reinforcement learning in metric spaces. *Advances in Neural Information Processing Systems*, 34, 2020.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.

Will Dabney, Georg Ostrovski, and Andre Barreto. Temporally extended ϵ-greedy exploration. In *Proceedings of the International Conference on Learning Representations*, 2021.

Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. *Advances in Neural Information Processing Systems*, 31, 2018.

Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. On the sample complexity of the linear quadratic regulator. *Foundations of Computational Mathematics*, 20(4):633–679, 2020.

Kenji Doya. Reinforcement learning in continuous time and space. *Neural computation*, 12 (1):219–245, 2000.

Greg Droge and Magnus Egerstedt. Adaptive time horizon optimization in model predictive control. In *Proceedings of the American Control Conference*, pages 1843–1848, 2011.

Maryam Fazel, Rong Ge, Sham Kakade, and Mehran Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *International Conference on Machine Learning*, pages 1467–1476. PMLR, 2018.

Ian Ford, DM Titterington, and Christos P Kitsos. Recent advances in nonlinear experimental design. *Technometrics*, 31(1):49–60x, 1989.

Tryphon T. Georgiou and Anders Lindquist. The separation principle in stochastic control, redux. *IEEE Transactions on Automatic Control*, 58(10):2481–2494, 2013. doi: 10.1109/TAC.2013.2259207.

Shixiang Gu, Timothy Lillicrap, Ilya Sutskever, and Sergey Levine. Continuous deep q-learning with model-based acceleration. In *International conference on machine learning*, pages 2829–2838. PMLR, 2016.

Yunhan Huang and Quanyan Zhu. Infinite-horizon linear-quadratic-gaussian control with costly measurements. *arXiv preprint arXiv:2012.14925*, 2020.

Yunhan Huang, Veeraruna Kavitha, and Quanyan Zhu. Continuous-time markov decision processes with controlled observations. In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 32–39. IEEE, 2019.

Nan Jiang, Alex Kulesza, Satinder Singh, and Richard Lewis. The dependence of effective planning horizon on model accuracy. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2016.

Jeongho Kim, Jaeuk Shin, and Insoon Yang. Hamilton-Jacobi deep Q-learning for deterministic continuous-time systems with Lipschitz continuous controls. *Journal of Machine Learning Research*, 22, 2021.

Karl Krauth, Stephen Tu, and Benjamin Recht. Finite-time analysis of approximate policy iteration for the linear quadratic regulator. In *Advances in Neural Information Processing Systems 32*, pages 8514–8524. 2019.

Aravind S. Lakshminarayanan, Sahil Sharma, and Balaraman Ravindran. Dynamic action repetition for deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

Tor Lattimore, Csaba Szepesvari, and Gellert Weisz. Learning with good feature representations in bandits and in rl with a generative model. In *International Conference on Machine Learning*, pages 5662–5670. PMLR, 2020.

Jaeyoung Lee and Richard S. Sutton. Policy iterations for reinforcement learning problems in continuous time and space — fundamental theory and methods. *Automatica*, 126, 2021.

Frank L. Lewis, Draguna Vrabie, and Kyriakos G. Vamvoudakis. Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers. *IEEE Control Systems Magazine*, 32(6):76–105, 2012. doi: 10.1109/MCS.2012.2214134.

Anders Lindquist. Linear stochastic systems (peter e. caines). *SIAM Review*, 32(2):325–328, 1990. doi: 10.1137/1032067. URL `https://doi.org/10.1137/1032067`.

Michael Lutter, Shie Mannor, Jan Peters, Dieter Fox, and Animesh Garg. Value iteration in continuous actions, states and time. In *Proceedings of the International Conference on Machine Learning*, 2021.

Alberto Maria Metelli, Flavio Mazzolini, Lorenzo Bisi, Luca Sabbioni, and Marcello Restelli. Control frequency adaptation via action persistence in batch reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2020.

Rémi Munos. Policy gradient in continuous time. *Journal of Machine Learning Research*, 7, 2006.

Rémi Munos and Paul Bourgine. Reinforcement learning for continuous stochastic control problems. *Advances in neural information processing systems*, 10, 1997.

Mojmír Mutnỳ and Andreas Krause. Experimental design for linear functionals in reproducing kernel hilbert spaces. *arXiv preprint arXiv:2205.13627*, 2022.

Mojmir Mutny, Johannes Kirschner, and Andreas Krause. Experimental design for optimization of orthogonal projection pursuit models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10235–10242, 2020.

Seohong Park, Jaekyeom Kim, and Gunhee Kim. Time discretization-invariant safe action repetition for policy gradient methods. *Advances in Neural Information Processing Systems*, 34, 2021.

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

Sean R. Sinclair, Siddhartha Banerjee, and Christina Lee Yu. Adaptive discretization for episodic reinforcement learning in metric spaces. In *Proceedings of the ACM Conference on Measurement and Analysis of Computing Systems*, 2019.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Corentin Tallec, Léonard Blier, and Yann Ollivier. Making deep q-learning methods robust to time discretization. In *International Conference on Machine Learning (ICML)*, number 97, pages 6096–6104, 2019.

Stephen Tu and Benjamin Recht. Least-squares temporal difference learning for the linear quadratic regulator. In *International Conference on Machine Learning*, pages 5005–5014. PMLR, 2018.

Stephen Tu and Benjamin Recht. The gap between model-based and model-free methods on the linear quadratic regulator: An asymptotic viewpoint. In *Conference on Learning Theory*, pages 3036–3083. PMLR, 2019.

Haoran Wang, Thaleia Zariphopoulou, and Xun Yu Zhou. Reinforcement learning in continuous time and space: A stochastic control approach. *Journal of Machine Learning Research*, 21(198):1–34, 2020. URL `http://jmlr.org/papers/v21/19-144.html`.

Cagatay Yildiz, Markus Heinonen, and Harri Lähdesmäki. Continuous-time model-based reinforcement learning. In *International Conference on Machine Learning*, pages 12009–12018. PMLR, 2021.

Appendix A. Proof of Theorem 1

Proof We first note that

$$\mathbb{E}[\hat{V}_M(h)] = \frac{h}{M}\sum_{i=1}^{M}\sum_{k=0}^{N-1}\mathbb{E}[x_i^2(kh)] = h\sum_{k=0}^{N-1}\mathbb{E}[x^2(kh)]$$

where we denote $x(t) = x_1(t)$ for simplicity. Next we expand the mean-squared error

$$\begin{aligned}
\mathbb{E}[(\hat{V}_M(h) - V)^2] &= \mathbb{E}[\hat{V}_M^2(h)] - 2V\mathbb{E}[\hat{V}_M(h)] + V^2 \\
&= \frac{h^2}{M^2}\mathbb{E}\left[\left(\sum_{i=1}^{M}\sum_{k=0}^{N-1}x_i^2(kh)\right)^2\right] - 2V\mathbb{E}[\hat{V}_M(h)] + V^2 \\
&= \frac{h^2}{M^2}\sum_{i,j=1}^{M}\sum_{k,l=0}^{N-1}\mathbb{E}[x_i^2(kh)x_j^2(lh)] - 2V\mathbb{E}[\hat{V}_M(h)] + V^2 \\
&= \frac{h^2}{M}\sum_{k,l=0}^{N-1}\mathbb{E}[x^2(kh)x^2(lh)] + \frac{M^2-M}{M^2}\mathbb{E}[\hat{V}_M(h)]^2 - 2V\mathbb{E}[\hat{V}_M(h)] + V^2
\end{aligned}$$

For the last equality, note that $\mathbb{E}[\hat{V}_M(h)]^2 = h^2\sum_{k,l=0}^{N-1}\mathbb{E}[x^2(kh)]\mathbb{E}[x^2(lh)]$. It remains to compute the expressions. For the second moment of the state variable, we have

$$\mathbb{E}[x^2(t)] = \frac{\sigma^2}{2a}\left(e^{2at} - 1\right) \tag{7}$$

Assuming that $s \le t$, we get the following for the forth moments:

$$\mathbb{E}[x^2(s)x^2(t)] = \frac{\sigma^4}{4a^2}(e^{2as}-1)e^{2at}\left\{(e^{-2as} - e^{-2at}) + 3(1 - e^{-2as})\right\} \tag{8}$$

Note that by symmetry, a similar expression follows for $s \ge t$.

Using these expressions, for the expected cost we get

$$V = \int_0^T \mathbb{E}[x^2(t)]dt = \frac{\sigma^2}{2a}\int_0^T \left(e^{2at} - 1\right)dt = \frac{\sigma^2}{2a}\left(\frac{e^{2aT}-1}{2a} - T\right)$$

A similar expression was previously obtained in (Bijl et al., 2016, Theorem 3). Next, the expected estimated cost is

$$\mathbb{E}[\hat{V}_M(h)] = h\sum_{k=0}^{N-1}\mathbb{E}[x^2(kh)] = \frac{\sigma^2 h}{2a}\sum_{k=0}^{N-1}\left(e^{2akh} - 1\right) = \frac{\sigma^2 h}{2a}\left[\frac{1-e^{2aT}}{1-e^{2ah}} - N\right]$$

Lastly, it remains to compute the sum

$$\frac{h^2}{M}\sum_{k,l=0}^{N-1}\mathbb{E}[x^2(kh)x^2(lh)] = \frac{2h^2}{M}\sum_{k<l}^{N-1}\mathbb{E}[x^2(kh)x^2(lh)] + \frac{h^2}{M}\sum_{k=0}^{N-1}\mathbb{E}[x^4(kh)]$$

This calculation can be done on paper, but the result is more easily obtained using symbolic computation. We provide the Wolfram Language commands in Appendix A.1 (including calculations of the corollaries). It remains to collect all terms to get the final result. ■

A.1 Wolfram Language Commands

```
Ex2[k_, h_, a_, o_] := o^2 / (2 a) (E^(2 a k h) - 1)

Exs2xt2[s_, t_, a_, o_] := o^4 / (4 a^2) (E^(2 a s) - 1) E^(2 a t) (E^(-2 a
↪  s) - E^(-2 a t) + 3 (1 - E^(-2 a s)))

EJ[a_, T_, o_] := o^2 / (2 a) (1 / (2a) (E^(2 a T) - 1) - T)

EVM[N_ ,h_, a_, o_] := h Sum[Ex2[k, h, a, o], {k,0, N-1}]

EVM2[M_, N_, h_, a_, o_] := (M^2 - M) / M^2  EVM[N, h, a, o]^2  + h^2 / M
↪  Sum[Exs2xt2[k h, k h, a, o], {k, 0, N-1}] + 2 h^2 / M  Sum[Exs2xt2[j h,
↪  k h, a, o], {k, 1, N-1}, {j, 0, k-1}]

Final[h_, B_, T_, a_, o_] := Simplify[EVM2[h B / T, T/h, h, a, o] +  EJ[a, T,
↪  o]^2 - 2 EJ[a, T, o] EVM[T/h, h, a, o]]

FinalSimple[h_, B_, T_, a_, o_] := FullSimplify[Final[h, B, T, a, o],{ a < 0,
↪  h > 0, T > 0, B > 0, o > 0}]

Limit[Final[h, B, T, a, o], a->0]

FinalSeries[h_, B_, T_, a_, o_] := Simplify[Series[Final[h, B, T, a, o], {h,
↪  0, 2}], { a <0, h > 0, T>0, B>0, o>0}]
```