FEYNMAN-KAC NUMERICAL TECHNIQUES FOR STOCHASTIC OPTIMAL CONTROL

A Dissertation
Presented to
The Academic Faculty

By

Kelsey P. Hawkins

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Interactive Computing
Institute for Robotics and Intelligent Machines

Georgia Institute of Technology

December 2021

FEYNMAN-KAC NUMERICAL TECHNIQUES FOR STOCHASTIC OPTIMAL CONTROL

Thesis committee:

Dr. Panagiotis Tsiotras
Aerospace Engineering
Georgia Institute of Technology

Dr. Samuel Coogan
Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Dmitry Berenson
Electrical Engineering and
Computer Science Dept.
University of Michigan

Dr. Evangelos Theodorou
Aerospace Engineering
Georgia Institute of Technology

Dr. Kyriakos Vamvoudakis
Aerospace Engineering
Georgia Institute of Technology

Date approved: August 25th, 2021

In that simple statement is the key to science — it doesn't make a difference how beautiful your guess is, it doesn't make a difference how smart you are, who made the guess, or what his name is — if it disagrees with experiment, it's wrong.

Richard Feynman

For my father and mother, Dr. Rody and Vina Hawkins,

and my partner, Samiksha Kaul.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Panagiotis Tsiotras, for his support and guidance in producing this work, and specifically for suggesting a challenging problem which I was proud to work on and make my own. I am also indebted to Dr. Ali Pakniyat, whose collaboration, discussions, and moral support was essential for keeping me from doubting myself and getting me to completion. Further, I must thank Dr. Evangelos Theodorou's advisement and encouragement. His expertise in this field helped me move in the right direction. In addition, I would like to thank Aaron Bobick, Henrik Christensen, and School of Interactive Computing for helping fund and support my research while I searched for a thesis topic. Also, I would like to thank the other members of my Dissertation Committee, Dr. Dmitry Berenson, Dr. Sam Coogan, and Dr. Kyriakos Vamvoudakis, for their time and interest in evaluating this work.

Special thanks to my friends and colleagues in Robograds and the Dynamics and Control Systems Laboratory, which made graduate school a very meaningful part of my life. I would also like to thank Charlie Kemp and Mike Stilman for spurring my interest in robotics in my first year at Georgia Tech.

Most importantly, I would like to thank my parents, Dr. Rody Hawkins and Vina Hawkins; my siblings, Kenan Hawkins, Kena Newkirk, and William Newkirk; and especially my partner, Samiksha Kaul. They were there for me during the hardest times of this process, and it is only because these people believed in me that I could believe in myself.

The author gratefully acknowledges the support for this work offered by NSF awards CMMI-1662523 and IIS-2008686 and ONR award N00014-18-1-2828. Any views and conclusions contained herein are those of the author, and do not necessarily represent the official positions, express or implied, of the sponsoring agencies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Three significant advancements are proposed for improving numerical methods in the solution of forward-backward stochastic differential equations (FBSDEs) appearing in the Feynman-Kac representation of the value function in stochastic optimal control (SOC) problems. First, borrowing from the nomenclature of the reinforcement learning (RL) community, we propose a novel characterization of FBSDE estimators as either on-policy or off-policy, highlighting the intuition for these techniques that the distribution over which value functions are approximated should, to some extent, match the distribution the policies generate. This insight leads to an FBSDE SOC problem, a specialization of the SOC problem which posits that obtaining a good probability measure over which to approximate the value function is as important as finding the approximation itself.

Second, two novel numerical estimators are proposed for improving the accuracy of single-timestep updates. In contrast to the current numerical approaches that are based on the discretization of the continuous-time FBSDEs, we propose a converse approach, namely, we obtain a discrete-time approximation of the on-policy value function, and then we derive a discrete-time estimator that resembles the continuous-time counterpart. The proposed approach allows for the construction of higher accuracy estimators along with error analysis. The proposed estimators show significant improvement in terms of accuracy over Euler-Maruyama-based estimators used in competing approaches. In the case of LQR problems, we demonstrate both in theory and in numerical simulation that our estimators result in near machine-precision level accuracy, in contrast to previously proposed methods that can potentially diverge on the same problems.

Third, we propose a new method for accelerating the global convergence of FBSDE methods. By the repeated use of the Girsanov change of probability measures, it is demonstrated how a McKean-Markov branched sampling method can be utilized for the forward integration pass, as long as the controlled drift term is appropriately compensated in the

backward integration pass. Subsequently, a numerical approximation of the value function is proposed by solving a series of function approximation problems backwards in time along the edges of a space-filling tree. Moreover, a local entropy-weighted least squares Monte Carlo (LSMC) method is developed to concentrate function approximation accuracy in regions most likely to be visited by optimally controlled trajectories. The proposed methodology is numerically demonstrated on linear and nonlinear stochastic optimal control problems with non-quadratic running costs of dimension up to $n = 5$, which reveals significant convergence improvements over previous FBSDE-based numerical solution methods.

CHAPTER 1
INTRODUCTION AND BACKGROUND

The Feynman-Kac representation theorem establishes the intrinsic relationship between the solution of a broad class of second-order parabolic and elliptic partial differential equations (PDEs) and the solution of forward-backward stochastic differential equations (FBSDEs) (see, e.g., [1, Chapter 7]), investigations of which were brought to prominence in [2, 3, 4]. Among this class of PDEs are the Hamilton-Jacobi (HJ) PDEs, including the Hamilton-Jacobi-Bellman (HJB) PDE which is associated with the solution of stochastic optimal control (SOC) problems [1, 5, 6, 7]; the Hamilton-Jacobi-Isaacs PDE which is associated with the solution of stochastic differential games [8, 9, 10]; and other applications relating to level set propagation [11], stochastic exit time problems [12], and pursuit evasion reachability [13, 14]. For deterministic control problems associated with first-order PDEs, especially those without classical solutions (problems for which the necessary derivatives exist everywhere), the *vanishing viscosity* method can be used to approximate the PDE with a non-degenerate second order PDE by effectively adding a small amount of noise to the dynamics [15, 6]. Given its potential for application to a wide variety of problems, Feynman-Kac FBSDE-based numerical methods have been gaining traction as a framework to solve nonlinear SOC problems in robotics and controls, including problems with quadratic cost [16], minimum-fuel (L_1-running cost) problems [17], differential games [18, 19], and reachability problems [16].

While initial investigations of Feynman-Kac FBSDE applications in robotics domains demonstrate promise in terms of flexibility and theoretical validity, numerical algorithms that leverage this theory have not yet matured. For even modest problems, state-of-the-art algorithms can be unstable, producing value function approximations which quickly diverge. The primary issue originates from the fact that the forward sampling distribution of

interest (e.g., the near-optimal trajectory distribution for the SOC problem) is not initially available, leading to the iterative methods discussed below. Producing more robust numerical methods is critical for the broader adoption of FBSDE methods for real-world tasks. For the purposes of this work, we choose to focus on improving Feynman-Kac FBSDE techniques for nonlinear SOC applications; we note that since the theory across different problems follows from the same Feynman-Kac representation methodology, any advancements in the SOC domain are likely to be easily adaptable to FBSDE techniques for solving problems like differential games or random stopping time problems.

Numerical methods for solving SOC problems remain an active area of research because state-of-the-art methods satisfy some, but not all, of the following desirable properties: (a) are computable in *high-dimensional* state spaces ($n \geq 4$), (b) they admit *general* nonlinear dynamics and nonquadratic costs, (c) they search *globally* for an optimal policy, and (d) *converge rapidly* to the solution. The focus of this research is to develop a method that satisfies most of these properties, though the trade-off between finding the global optimum and rapid convergence may depend on the complexity and dimensionality of the problem being solved.

1.1 Iterative Feynman-Kac FBSDE Systems

Feynman-Kac FBSDE systems are a pair of stochastic differential equations (SDEs): a forward SDE (FSDE) whose solution X_s takes values in an n-dimensional state space and has an initial value constraint, and a backward SDE (BSDE) Y_s, whose solution is a 1-dimensional value process that has a terminal value constraint. In this work, we investigate what we call here *iterative FBSDE* (iFBSDE) numerical methods of the Feynman-Kac-type, first explored in [18, 16, 19, 17]. These methods are distinct from the wide swath of research into non-iterative FBSDE methods [20, 21, 22, 23, 24] in that iFBSDE methods modify the characterization of the FBSDE system being solved in every iteration. Although the theoretical justification of non-iFBSDE and iFBSDE methods are similar, the numerical

challenges and application domains differ: non-iFBSDE methods are frequently applied to numerical finance problems dominated by high diffusion where the state space distribution of interest can be sampled in a straightforward way because its dynamics are already available, whereas iFBSDE methods are more suited to SOC problems where diffusion is relatively low but the state space distribution of interest (e.g., the optimally controlled trajectory distribution) is not initially known.

Non-iFBSDE methods typically begin by sampling a large number of forward SDE trajectories, often $M \geq 100,000$, and then solve the backward SDE over this distribution. Since in iFBSDE methods the forward SDE distribution changes after each iteration, these methods must be more efficient in sampling, e.g., $M \leq 5,000$ trajectory samples, because local accuracy is not important if the local FBSDE distribution is not of particular interest. The challenge at each iteration is to find a good enough approximation of the value function along the current distribution of forward trajectories, so that we may improve subsequent trajectory sampling distributions. Asymptotic convergence and accuracy guarantees for the solutions of BSDEs are largely unhelpful because we will not densely sample until, at least, our forward distribution is close to the optimal distribution. Further, methods like Picard-type iteration, e.g., the estimation of the Y_s and Z_s processes in [21, p. 1795], break down with smaller number of samples because the estimators have relatively high conditional variance. A more detailed characterization of the iFBSDE problem is a contribution of this work and is the topic of Chapter 3.

Algorithm 1 Iterative FBSDE

1: $\Pi^1 \leftarrow (f^1, \sigma, h^1, g, \text{etc.})$ ▷ Initial FBSDE representation
2: **for** $k = 1, \cdots, N_{\text{iter}}$ **do** ▷ Iteration loop
3: $X_s \leftarrow \textsc{ForwardPass}(\Pi^k)$ ▷ Generate FSDE distribution
4: $Y_s \leftarrow \textsc{BackwardPass}(\Pi^k, X_s)$ ▷ Solve BSDE
5: $\Pi^{k+1} \leftarrow \textsc{ImproveFBSDE}(\Pi^k, X_s, Y_s)$ ▷ Adjust FBSDE problem
6: **end for**

Algorithm 2 BackwardPass

1: $Y_T \leftarrow g(X_T)$ ▷ Terminal condition (time $t = T$)

2: **for** $t_i = T - \Delta t, T - 2\Delta t, \cdots, 0$ **do** ▷ Backward step loop

3: $\quad \widehat{Y}_{t_i} \leftarrow \text{ESTIMATOR}(\Pi^k, X_s, Y_{t_{i+1}})$ ▷ Backwards integration

4: $\quad Y_{t_i} \leftarrow \text{LEASTSQMONTECARLO}(\Pi^k, \widehat{Y}_{t_i})$ ▷ Function regression

5: **end for**

Iterative FBSDE numerical methods, as illustrated in Algorithm 1, broadly consist of three steps per iteration: a forward pass which generates Monte Carlo samples of the forward stochastic process X_s, a backward pass which iteratively approximates the value function backwards in time using the Feynamn-Kac representation equality $Y_t = V(t, X_t)$, and finally an improvement of the FBSDE characterization to be utilized in the next iteration. The backward pass, as illustrated in Algorithm 2 consists of a series of steps backward in time where first an estimator $\widehat{Y}_{t_i}$ for the value function is computed via backward integration of the backward SDE, and next, a least-squares Monte Carlo (LSMC) scheme is used to implicitly solve for Y_{t_i} using parametric function approximation [25]. Before discussing this method in more detail, next we discuss how iFBSDE methods fit into the broader field of SOC numerical methods.

1.2 Related Works

Feynman-Kac FBSDE methods solve SOC problems that do not fit neatly under any of the classes of techniques typically used to solve problems in the robotics and controls communities. In this section we will discuss how FBSDE methods compare to these other classes, then briefly discuss the state-of-the-art in FBSDE methods.

1.2.1 Finite Difference-Type Methods

Finite-difference (FD) and finite-element schemes represent a set of methods which obtain a solution over a bounded domain by directly solving the HJB equation. Grid-based-FD

algorithms (see, e.g., [26, 27]) discretize the entire state space and hence find a global solution, but perform poorly when the space dimension is greater than, say, 4, an issue commonly known as the "curse of dimensionality". There is also ample research into the development of meshless methods for solving PDEs, such as radial basis function (RBF) collocation and RBF-finite difference (RBF-FD) formulations [28]. FBSDE methods share significant similarities with these RBF-FD methods, in the sense that the value function is approximated by solving the PDE at an unstructured set of collocation points. The primary drawback of RBF methods is that they do not offer an efficient method for choosing the collocation points, and since it is difficult to know a priori what the best points are, point selection might regress into a grid-based method. Specifically, a sufficiently broad and dense sampling of a high-dimensional state space might require roughly the same number of collocation points as a grid-based method in order to be well-conditioned for value function regression [29]. FBSDE methods, on the other hand, provide a framework by which to choose the collocation points, the forward SDE, and thus ground the solution in paths reachable from the initial state of interest.

Another drawback of RBF methods, particularly for use as a value function model, is that obtaining the parameterized model requires the solution of a linear regression problem of size equal to the number of collocation points. Although there are methods to induce sparsity in the problem, regression of this size is still time consuming, especially if a different model is used for every time step and iteration.

1.2.2 Linear Quadratic Regulator

The linear quadratic regulator (LQR) problem is a special type of SOC problem where the dynamics are linear and costs are quadratic, the solution of which can be found via the Riccati equations, and whose optimal value function is proven to be quadratic (see, e.g., [1, Chapter 6]). Since the Riccati equations are ordinary differential equations (or recurrence relations in the discrete-time case), the solution to LQR problems is obtained very rapidly

and with very high accuracy. The fact that optimal control problems are well behaved for LQR systems forms the basis, in part, for many of the following methods.

1.2.3 Differential Dynamic Programming

Differential dynamic programming (DDP) methods such as classic DDP [30], iterative linear quadratic regulator (iLQR) [31], and Gauss-Newton shooting methods [32], sample a trajectory following a nominal policy in a forward pass, followed by a subsequent backward pass which locally approximates the value function around the trajectory. The policy is updated and the forward-backward passes repeat until convergence. The algorithm scales well to high dimensions because the value function parameterization uses a quadratic local approximation.

Although FBSDE methods seem similar to differential dynamic programming (DDP) techniques [30, 31, 33], the approach is significantly different. DDP methods require first and second order derivatives of the dynamics, and directly compute a quadratic approximation of the value function using constraints on the derivatives of the value function. Comparatively, FBSDE LSMC only uses estimates of the value function at a distribution of states, using derivatives of the value function to improve the accuracy of the estimator. FBSDE methods are more flexible in that they do not require evaluating or approximating derivatives of the dynamics terms and can be used with models of the value function which are not necessarily quadratic. Furthermore, for most DDP applications, a quadratic running cost with respect to the control is required for appropriate regularization [34, Section 2.2.3], whereas the FBSDE method more easily accommodates non-quadratic running costs (e.g., of the class L_1 or zero-valued), lending to a variety of control applications [17].

A key feature of FBSDE methods is their ability to generate a parametric model for the value function over the entire time horizon which, in turn, can be used for the evaluation and assessment of the stochastic performance of closed-loop control policies. This feature differentiates both FBSDE and DDP methods from model predictive control (MPC)

methods [35], which, in general, only produce the current-best optimal control signal, re-evaluated at every time step [36].

1.2.4 Stochastic Maximum Principle

Similarly to DDP, methods based on Pontryagin's maximum principle such as adjoint-process shooting methods involve forward and backward passes. However, the stochastic maximum principle (SMP) formulation for nonlinear problems [37] is challenging to apply without reducing it to an approximate LQR method with Riccati-equation backward passes. The challenge with using generalized SMP is that far more variables are needed for approximation: in addition to the value function, the adjoint vector must be approximated, as well as a second order adjoint matrix (similar to the Riccati matrix), adding $n + n^2$ degrees of freedom to the problem, instead of just approximating the one-dimensional scalar value function. Since all function approximations introduce an error, increasing the number of functions to be approximated is likely to significantly increase the numerical error for high-dimensional problems.

1.2.5 Model Predictive Control

Multi-parametric optimization model predictive control (MPC) approaches like those used in [38, 39] cast optimal control problems as linear or nonlinear optimization problems. Such approaches typically require linear dynamics and specific cost forms, and are thus less general for application. Further, complicated optimization problems in high-dimensions might not reliably produce a solution within a reasonable amount of time.

Path-integral (PI) approaches (introduced in [40]) like PI-relative entropy policy search [41], and model predictive path integral (MPPI) [31, 42] typically rely on rolling out trajectories using randomly sampled controls, computing their path-integrated costs, and using inference schemes to inform a control policy. In MPPI's case, a weighted average of the random control signals is used to produce a new nominal control signal, giving more weight

to paths with low costs. The advantage of path-integral schemes is that value function approximation is largely sidestepped, eliminating the potential error and instability it brings along with it. Further, trajectory sampling is highly parallelizable and fast. The disadvantage is that, like DDP-based methods, exploration is largely local, and thus highly susceptible to local optima. Further, the construction of the problem makes quadratic assumptions about control costs, reducing slightly the generality of its application.

1.2.6 Reinforcement Learning

The reinforcement learning (RL) community has given increasing attention to solving SOCs [43, 44, 45, 46, 47], especially as deep neural network (DNN) function models and policy gradient methods [48] have grown increasingly popular. DNNs are particularly attractive for representing value functions because they can: (a) easily scale to high dimensions, (b) represent increasingly complex functions by adding more layers and units to layers, and (c) theoretically approximate any function with enough parameters [49]. The general trade-off for these methods is that parameters must be slowly trained using stochastic gradient descent (SGD), a method which converges slowly. Recently "actor-critic" methods have received increasing interest. They maintain a separate representation of both the "actor" (the policy) and the "critic" (the value function associated with the current policy). Actor-critic methods combine the stability advantages of learning the policy directly with the *sample efficiency*[1] of Q-value learning methods. Though the RL community has plenty of well-studied approaches including dynamic programming methods such as value iteration and policy iteration, temporal-difference (TD) methods such as Q-learning or SARSA, and Monte Carlo n-step extensions to TD methods, most of the early work was constrained to state spaces that could be fully enumerated, like grids [50]. With the development of policy gradient methods [48] and eventually deep RL algorithms within the past several years, it has become clear that RL methods could compete with, and in several ways sur-

[1] The number of training samples needed to converge.

pass, optimal control methods for computing and representing optimal control policies in high dimensions [43].

A large difference between most SOC approaches in previous sections and TD RL methods is that in previous methods the value function is computed in a systematic way, a full timestep at a time, either by analytic back-propagation with DDP methods, finite difference grid updates, or by function approximation in FBSDE methods. TD methods, on the other hand, typically treat collected state-action-reward-state-action (SARSA) tuples as unstructured data, called a *replay buffer*, which informs a generalized machine learning algorithm. The slow and incremental updates of SGD allows learning on these unstructured datasets to converge even when the policy is changing. Least-squares TD methods, which attempt to solve a least-squares regression problem and thus result in large steps in the parameter space, are known to have issues in RL applications because they fail to forget old data [50, Section 9.8, p. 229].

1.2.7 Path/Motion Planning

In path/motion planning problems, rapidly-exploring random tree (RRT) methods efficiently explore a state space for a path from the initial state to some goal set [51]. Path planning methods typically focus on navigating around a set of obstacles, for which paths cannot cross, in a high-dimensional state space. RRT* methods consistently "rewire" the tree to guarantee that the best path in the tree approaches global optimal in the limit [52]. Various RRT* derivatives have been proposed to improve convergence speed and to expand the types of metrics which can be considered [53]. However, many RRT*-based implementations are not general in terms of the costs because they typically consider path length as the running cost.

1.2.8 Iterative FBSDEs

Feynman-Kac FBSDE methods exploit variations on the following theorem, by first casting the control problem as a PDE (the HJB equations in the case of SOC problems), and then solving the FBSDEs which represent the solution of the PDE.

General Nonlinear Feynman-Kac Representation Theorem

Theorem 1.1. *[1, Chapter 7, Theorem 4.5] For Lipschitz-continuous functions*[2] $f(t,x)$, $\sigma(t,x)$, $h(t,x,y,z)$, $g(x)$, *the PDE*

$$\begin{aligned} \partial_t V + \frac{1}{2}\operatorname{tr}[\sigma\sigma^\top \partial_{xx} V] + (\partial_x V)^\top f + h(t,x,V,\sigma^\top \partial_x V)\big|_{t,x} &= 0, \\ V(T,x) &= g(x), \end{aligned} \tag{FK-PDE}$$

has the representation

$$Y_s = V(s, X_s), \tag{1.1}$$

$$Z_s = \sigma(s, X_s)^\top \partial_x V(s, X_s), \tag{1.2}$$

Q*-almost surely (a.s.) where* (X_s, Y_s, Z_s) *is the solution to the FBSDEs*

$$\mathrm{d}X_s = f(s, X_s)\,\mathrm{d}s + \sigma(s, X_s)\,\mathrm{d}W_s^{\mathsf{Q}}, \qquad X_0 = x_0, \tag{1.3}$$

$$\mathrm{d}Y_s = -h(s, X_s, Y_s, Z_s)\,\mathrm{d}s + Z_s^\top\,\mathrm{d}W_s^{\mathsf{Q}}, \qquad Y_T = g(X_T), \tag{1.4}$$

where W_s^{Q} *is Brownian in* Q.

There is a lot of flexibility in how f and h are chosen, in the sense that the pair can be chosen differently, yet still represent the equivalent (FK-PDE). This suggests that for a given problem associated with a particular (FK-PDE), the f term in (1.3) can be modified

[2] Refer to the citation for the precise assumptions.

at will, and as long as the h term in (1.4) is compensated appropriately, the pair will, by virtue of the theory, solve the same PDE.

In fact, we can arrive at a stronger result through application of Girsanov's theorem (see, e.g., [7, Chapter 5, Theorem 10.1]) to both (1.3) and (1.4).

General Nonlinear Feynman-Kac-Girsanov Representation Theorem

Theorem 1.2. *The Feynman-Kac PDE* (FK-PDE) *has the representation*

$$Y_s = V(s, X_s), \tag{1.5}$$

$$Z_s = \sigma(s, X_s)^\top \partial_x V(s, X_s), \tag{1.6}$$

P*-a.s. where* (X_s, Y_s, Z_s) *is the solution to the FBSDEs*

$$\mathrm{d}X_s = (f(s, X_s) - \sigma(s, X_s)D_s)\,\mathrm{d}s + \sigma(s, X_s)\,\mathrm{d}W_s^{\mathsf{P}}, \qquad X_0 = x_0, \tag{1.7}$$

$$\mathrm{d}Y_s = -(h(s, X_s, Y_s, Z_s) + Z_s^\top D_s)\,\mathrm{d}s + Z_s^\top\,\mathrm{d}W_s^{\mathsf{P}}, \qquad Y_T = g(X_T), \tag{1.8}$$

where W_s^{P} *is Brownian in* P *and* D_s *is any adapted process, bounded* P*-a.s..*

This is much stronger than the method discussed previously, since D_s can be an arbitrarily selected process and not just a deterministic function of x.

This theory is used in [18, 16, 17, 19], to create an iFBSDE technique, described as an importance sampling algorithm. In each iteration, a different D_s process is selected to change the forward distribution, and the compensated BSDE is solved accordingly.

One of the problems with this formulation is that it is easy to misinterpret what the numerical application of the theory can realistically produce, especially in the context of an SOC problem. Specifically, it suggests that we can choose nearly any drift in (1.7) through modification of D_s, including cancelling out the drift entirely, and still arrive at the solution to the optimal value function. In numerical applications of this theory, the

trajectory distribution produced by (1.7) is critical to our understanding of what it means to "solve" the SOC problem. Among the other contributions of this work, we re-characterize this generalization in the context of iterative SOC methods, presented in Chapter 3.

While the generalized Feynman-Kac-Girsanov theorem is interesting from a theoretical perspective, it is unwieldy for practical numerical applications. Since the result is grounded in theorems unknown outside of stochastic systems theory, it is difficult for the uninitiated to gain an intuition for how and why error is introduced into a numerical method. Further, though the introduction of the D_s term offers significant flexibility in the theoretical approach to solving SOC problems, it is unclear how this flexibility can be utilized without producing methods with poor performance in numerical approximations.

1.3 Thesis Contributions

The goal of the investigations presented in this work is to prepare Feynman-Kac iFBSDE methods for broader numerical application and research. To this effect, we offer three major contributions:

- We characterize FBSDE systems as *on-* or *off-policy* and propose the FBSDE SOC problem, a specialization of the SOC problem for the FBSDE methodology, for the purposes of improving problem intuition (Chapter 3).
- We re-derive FBSDE theory using discrete-time methods, resulting in estimators with significantly improved accuracy over small timesteps (Chapter 4).
- We propose a framework by which global convergence of FBSDE methods is improved by interpreting path/motion planning methods as the forward pass of the FBSDE method (Chapter 5).

We briefly summarize these contributions in more detail.

1.3.1 On/Off-Policy FBSDE and The FBSDE SOC Problem (Chapter 3)

The contribution of Chapter 3 is to provide a more interpretable formulation of the FBSDEs, borrowing partly from the RL nomenclature of denoting methods either as *on-policy* or *off-policy*. Instead of solving directly for an approximation of the optimal value function V^*, we solve for the on-policy value function V^μ, which returns the expected cost-to-go under the policy μ. We designate as on-policy FBSDE estimators those that are produced by forward sampling the dynamics governed by μ, then solving for V^μ, the same value function associated with the sampled dynamics. We designate off-policy estimators as those where the forward sampling is not strictly governed by μ. On-policy estimators have the benefit of high accuracy especially along the distribution of trajectories associated with the policy. Off-policy estimators offer a significant amount of flexibility in how the forward distribution is sampled, but in numerical approximations tend to add bias to the estimator. The framing of the estimators in this way naturally cues the interpretation of how one should choose the drift in the off-policy method, that it should be kept small to reduce bias.

The on/off-policy designation also highlights the importance of the trajectory distribution in arriving at a numerical solution for FBSDE methods. It is not enough to find the optimal value function/policy for some arbitrary region of the state space, we need to find a good approximation which widely covers the distribution of optimal trajectories. The other primary contribution of this section is a formalization of this nuance in a proposed FBSDE SOC problem.

1.3.2 Improving FBSDE Estimators With Discrete-Time Analysis (Chapter 4)

In the currently available algorithms in the literature, Euler-Maruyama approximations are employed for discretizing the continuous-time FBSDEs to solve for an approximation of the continuous-time value function [16]. In this chapter, instead of the direct application of the Euler-Maruyama approximation on the Feynman-Kac FBSDEs, we begin by formulating a discrete time problem with the Euler-Maruyama approximation of the dynamics,

costs, and value function. Next, we derive discrete-time relationships which resemble their continuous-time counterparts using Taylor expansions and the discrete-time Bellman equation. By doing so, we arrive at a set of alternative estimators for the value function which are far more accurate, especially on the LQR problem, for which it results in near-machine-precision accuracy.

The primary contributions of Chapter 4 are as follows:

- Proposing a pair of alternative estimators for the value function used in the backward pass of a Girsanov-drifted Feynman-Kac FBSDE numerical method.
- Characterizing the theoretical bias and variance of these estimators and showing their theoretic superiority to previously proposed estimators.
- Numerically confirming the theoretical results on representative SOC problems.

1.3.3 Solution of FBSDEs Using McKean-Markov Branched Sampling (Chapter 5)

We expand upon the above ideas and invoke Girsanov's theorem for Feynman-Kac FBSDEs in a broader setting than that of [16, 17, 19], showing that the forward sampling measure can be modified at will; this enables us to incorporate methods from other domains, namely, rapidly-exploring random trees (RRTs) (see, e.g., [51] and the recent survey in [53]) in order to more efficiently explore the state space in the forward pass. Using RRTs in the forward sampling allows us to spread samples evenly over the reachable state space, increasing the likelihood that near-optimal samples are well-represented in the forward pass sample distribution. By sampling more efficiently and relying less on incremental approximations of the value function to guide our search, we can achieve faster and more robust convergence than previous FBSDE methods. In the backward pass, we take advantage of the path-integrated running costs and estimates of the value function to produce a heuristic which weighs paths in the function approximation according to a local-entropy measure-theoretic optimization. Although local-entropy path integral theory and RRTs have been

used together in [54], the method of this article is more closely related to the path-integral approach to control [31]. Our method similarly performs forward passes to broadly sample the state space, but, to the contrary, follows them with backward passes to obtain approximations for the value functions, and consequently to obtain closed loop policies over the full horizon.

The primary contributions of this chapter are as follows:

- Providing the theoretical basis for the use of McKean-Markov branched sampling in the forward pass of FBSDE techniques.
- Introducing an RRT-inspired algorithm for sampling the forward SDE.
- Presenting a technique for concentrating value function approximation accuracy in regions containing optimal trajectories.
- Proposing an iterative numerical method for the purpose of approximating the optimal value function and its policy.

A subjective review of the performance of the reviewed SOC methods is provided in Table 1.1, alongside the two contributed improvements detailed in Chapter 4 and Chapter 5. Given the contributions discussed in the previous sections, we now present our thesis statement:

Thesis Statement

The contributed methodology significantly improves the attractiveness of Feynman-Kac FBSDE-based numerical methods by providing a more intuitive presentation, estimators with provable accuracy guarantees, and a framework which accelerates global convergence.

The proposed methodology is very general, flexible, and relatively simple to apply, and

Table 1.1: Comparison of SOC methods: finite-differences (FD), differential dynamic programming (DDP) (iLQR in particular [31]), model predictive control (MPC) (MPPI in particular [35]), deep reinforcement learning (RL), rapidly-exploring random trees (RRT), forward-backward SDEs (FBSDEs) [16, 55], the contributed discrete-time FBSDEs (DT-FBSDE) of Chapter 4, and the contributed forward-backward RRTs (FBRRT) of Chapter 5. The subjective performance ratings are interpreted as "- -" for very poor performance, "-" for poor, "-" for good, and "+ +" for very good. The closed-loop policy row refers to whether the representation naturally produces a representation for the policy (and value function), as opposed to a nominal trajectory. Model flexibility refers to how accommodating the algorithm is to problems which are not linear-quadratic, and whether linearization is required. LQR-accurate refers to whether the algorithm immediately and accurately converges to the optimal solution when presented with the LQR problem.

	Method							
	Related Methods						**Contributions**	
	FD	DDP	MPC	RL	RRT	FBSDE	**DT-**	**FBRRT**
High dimension	- -	++	+	++	+	+	**+**	**+**
Closed-loop policy	++	+	-	++	-	+	**+**	**+**
Computational speed	- -	++	++	- -	-	-	**-**	**-**
Model flexibility	++	-	+	++	+	++	**++**	**++**
LQR-accurate	+	++	-	-	- -	-	**++**	**++**[3]
Broad exploration	++	-	-	+	++	-	**-**	**++**

thus offers a wide range of applications and opportunities for further research. The results of numerical experiments demonstrate that our contributions have significantly improved the state-of-the-art in Feynman-Kac FBSDE methods, and thus show that this area of research has plenty of room for further enhancements.

The following chapter introduces the background for the continuous-time SOC problem. The next three chapters comprise the contributions of this work, discussed previously. The final chapter presents a conclusion and discusses some potential directions for future work.

[3]This work only evaluates the improvements in DT-FBSDE and FBRRT independently, but the estimator in FBRRT can easily be substituted with the estimators developed in the DT-FBSDE work.

CHAPTER 2
STOCHASTIC OPTIMAL CONTROL THEORY

In this chapter we introduce the continuous-time stochastic optimal control (C-SOC) problem, beginning with a detailed presentation of the theory used in this work. If the reader is mostly familiar with stochastic systems theory, it is suggested that they skip to Section 2.2 and reference the following section as needed.

2.1 Stochastic Systems Theory

We begin with an overview of important topics from probability theory which will be used in this work.

2.1.1 Probability Spaces and Random Elements

We begin with an overview of basic probability theory which is contextually relevant to the discussion in later chapters. Letting Ω denote a **sample space**, a σ**-field** (-algebra) $\mathcal{F}$ is a collection of events $A \in \mathcal{F}$, $A \subseteq \Omega$, such that $\Omega \in \mathcal{F}$, $A \in \mathcal{F}$ implies $A^{\mathsf{C}} \in F$, and $A, B \in \mathcal{F}$ implies $A \bigcup B \in \mathcal{F}$. A set and a σ-field $(\Omega, \mathcal{F})$ is called a **measurable space**. A **probability space** is a triple $(\Omega, \mathcal{F}, \mathsf{P})$ consisting of some measurable space, and a **probability measure** P whose domain is the events in $\mathcal{F}$ and satisfies the basic postulates of probability:

- $\mathsf{P}(\Omega) = 1$,
- $\mathsf{P}(A) \geq 0, \quad \forall A \in \mathcal{F}$,
- If $\{A_i\}$ are disjoint events, then $\mathsf{P}(\bigcup_i A_i) = \sum_i \mathsf{P}(A_i)$.

A **random element** is a map

$$X : \Omega \mapsto \Omega',$$

from one measurable space $(\Omega, \mathcal{F})$ to another $(\Omega', \mathcal{F}')$ which has the property that it is **measurable**, that is,

$$X^{-1}(\mathcal{F}') \subseteq \mathcal{F},$$

where,

$$X^{-1}(\mathcal{F}') := \{X^{-1}(A) : A \in \mathcal{F}'\},$$
$$X^{-1}(A) := \{\omega \in \Omega : X(\omega) \in A\}.$$

Random elements induce the set function $\mathsf{P} \circ X^{-1}$, defined as

$$\mathsf{P} \circ X^{-1}(A') = \mathsf{P}(X^{-1}(A')), \quad \forall A' \in \mathcal{F}',$$

and denoted P_X. The triple $(\Omega', \mathcal{F}', \mathsf{P}_X)$ is a probability space where P_X is called the **distribution** of X. For the metric space over n-dimensional vectors (and 1-dimensional variables) equipped with the L^2-norm, $(\mathbb{R}^n, \|\cdot\|_2)$, the Borel σ-field $\mathcal{B}(\mathbb{R}^n)$ is the smallest σ-field generated by the open sets of the metric space. Unless otherwise noted, $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$ is considered to be the measurable space associated with **random vectors** or 1-dimensional **random variables**. The other random elements considered in this work include **joint random variables** over pairs or sets of random vectors $\omega \mapsto (X(\omega), Y(\omega))$ generating the **joint distribution** $\mathsf{P}_{(X,Y)}$, **discrete-time processes** $\omega \mapsto \{X_i(\omega)\}$, and **continuous-time processes** $\omega \mapsto X_s(\omega)$. If we would like to talk only about the possible events some random element may have, we denote this σ-field $\sigma(X) := X^{-1}(\mathcal{F}')$. We note that for continuous

functions f, $f(X)$ (including X) is always $\sigma(X)$-measurable.

2.1.2 Expectations and Radon-Nikodym Derivative

The **expectation value** of some expression is an integral over the sample space Ω, denoted

$$\mathbf{E}_{\mathsf{P}}[X] \equiv \int_{\Omega} X \mathrm{d}\mathsf{P} \equiv \int_{\Omega} X(\omega) \mathrm{d}\mathsf{P}(\omega).$$

For **simple random variables** defined as

$$X(\omega) = \sum_{i=1}^{k} a_i \mathbf{1}_{A_i}(\omega), \quad \text{such that} \quad \sum_{i=1}^{k} A_i = \Omega, \quad |a_i| < \infty,$$

where $\mathbf{1}_{A_i}$ is the **indicator function**

$$\mathbf{1}_{A_i}(\omega) := \begin{cases} 1 & \text{if } \omega \in A_i \\ 0 & \text{o.w.} \end{cases},$$

the expectation is defined as

$$\begin{aligned} \mathbf{E}_{\mathsf{P}}[X] &\equiv \int_{\Omega} \sum_{i=1}^{k} a_i \mathbf{1}_{A_i}(\omega) \mathrm{d}\mathsf{P} \\ &= \sum_{i=1}^{k} a_i \int_{\Omega} \mathbf{1}_{A_i}(\omega) \mathrm{d}\mathsf{P} \\ &= \sum_{i=1}^{k} a_i \mathsf{P}(A_i). \end{aligned}$$

We say that a random variable is **integrable** if the expectation of its absolute value is finite.

When we use the expression P-**almost surely** (P-a.s.) to refer to some statement ϕ, involving random variables on a probability space, we mean that for the event $E := \{\omega \in \Omega : \phi(\omega) \text{ is true}\}$, there exists a null-event $N \in \mathcal{F}$ in P, that is, $\mathsf{P}(N) = 0$, such that $E^{\mathsf{C}} \subseteq N$. For two measures Q and P on the same measurable space, we say Q is **absolutely**

continuous with respect to P if for all $A \in \mathcal{F}$, $\mathsf{P}(A) = 0$ implies $\mathsf{Q}(A) = 0$, and denote it $\mathsf{Q} \ll \mathsf{P}$. We say Q and P are **equivalent** if both $\mathsf{Q} \ll \mathsf{P}$ and $\mathsf{P} \ll \mathsf{Q}$. The **Radon-Nikodym theorem** suggests that if $\mathsf{Q} \ll \mathsf{P}$, then there exists a measurable random variable Θ such that

$$\mathsf{Q}(A) = \int_\Omega \mathbf{1}_A \, \Theta \, \mathrm{d}\mathsf{P},$$

for all $A \in \mathcal{F}$, which we often denote

$$\mathrm{d}\mathsf{Q} = \Theta \, \mathrm{d}\mathsf{P}.$$

We can construct a probability measure Q using this theorem, given P and a non-negative measurable random variable $\Theta \geq 0$, P-a.s. such that $\mathbf{E}_\mathsf{P}[\Theta] = 1$. If $\Theta > 0$, P-a.s. then Q and P are equivalent probability measures.

The **density** of a random vector X in P is the Radon-Nikodym derivative $p_X := \frac{\mathrm{d}\mathsf{P}_X}{\mathrm{d}x}$ between the distribution P_X of X and the Lebesgue measure $\mathrm{d}x$, the standard measure of volume in $\mathbb{R}^n$.

2.1.3 Stochastic Processes

Letting the time interval be $\mathcal{T} = \{0, \ldots, N\}$, $N \in \mathbb{N}$, in the discrete-time case or $\mathcal{T} = [0, T]$, $T \in \mathbb{R}_+$, in the continuous-time case, a **stochastic process** (or just, process) is a random element $\{X_i\}_{i=0}^N$ / X_s such that for each time in $i/t \in \mathcal{T}$, X_i / X_t is a random vector in $\mathbb{R}^n$. A **sample path / trajectory** is the realization of a particular sample ω as $\{X_i(\omega)\}_{i=0}^N$ / $X_s(\omega)$. When discussing conditional expectations and probabilities, we sometimes discuss conditioning on the full set of events possible up to that time. The **filtration** $\{\mathcal{F}_i\}_{i=0}^T$ / $\{\mathcal{F}_s\}_{s\in[0,T]}$ is an increasing set of σ-fields representing the history of all events on all processes up to and including that time. By increasing, we mean $\mathcal{F}_i \subseteq \mathcal{F}_{i+j}$ / $\mathcal{F}_s \subseteq \mathcal{F}_{s+\Delta t}$ for $j > 0$ / $\Delta t > 0$, the idea that we only learn more about events as time pro-

gresses. A complete, filtered probability space is a triple $(\Omega, \{\mathcal{F}_i\}, \mathsf{P})$ / $(\Omega, \{\mathcal{F}_s\}, \mathsf{P})$ where $(\Omega, \mathcal{F}_N =: \mathcal{F}, \mathsf{P})$ / $(\Omega, \mathcal{F}_T =: \mathcal{F}, \mathsf{P})$ is a probability space which contains all the P-null sets, events with probability zero. We say a process X_s is **adapted** to the filtration if X_t is $\mathcal{F}_t$-measurable. This expresses the concept that if we know the distribution of everything at time t, then we know the distribution of X_t as well. This concept is easier to understand with the definition of conditional expectation.

Let $\mathcal{G} \subseteq \mathcal{F}$ be a sub-σ-field of σ-field $\mathcal{F}$. The **conditional expectation** of an integrable random variable X with respect to $\mathcal{G}$ in the measure P is the integrable and $\mathcal{G}$-measurable random variable $\mathbf{E}_{\mathsf{P}}[X|\mathcal{G}]$ such that

$$\int_{\Omega} \mathbf{1}_G \, X \, \mathrm{d}\mathsf{P} = \int_{\Omega} \mathbf{1}_G \, \mathbf{E}_{\mathsf{P}}[X|\mathcal{G}] \mathrm{d}\, \mathsf{P}, \quad \forall G \in \mathcal{G}.$$

The primary settings of $\mathcal{G}$ we use in this work are either conditioning on random vectors,

$$\mathcal{G} = \sigma(X), \qquad \rightarrow \qquad \mathbf{E}_{\mathsf{P}}[Y|X] := \mathbf{E}_{\mathsf{P}}[Y|\sigma(X)],$$

or on the filtration

$$\mathcal{G} = \mathcal{F}_t, \qquad \rightarrow \qquad \mathbf{E}_{\mathsf{P}}[X_s|\mathcal{F}_t] \quad \text{similar to} \quad \mathbf{E}_{\mathsf{P}}[X_s|X_0, \ldots, X_t, Y_0, \ldots, Y_t, \ldots].$$

In reality, $\sigma(X_0, \ldots, X_t, Y_0, \ldots, Y_t, \ldots) \subseteq \mathcal{F}_t$. Letting $\mathcal{G} \subseteq \mathcal{F}$ be a σ-field, X, Y be two $\mathcal{F}$-measurable random vectors, and f be a continuous function, the primary properties of conditional expectation are

- **Stability:** If X is $\mathcal{G}$-measurable then $\mathbf{E}_{\mathsf{P}}[X|\mathcal{G}] = X$.
- **Pulling out known factors:** If X is $\mathcal{G}$-measurable then $\mathbf{E}_{\mathsf{P}}[XY|\mathcal{G}] = X \, \mathbf{E}_{\mathsf{P}}[Y|\mathcal{G}]$ and $\mathbf{E}_{\mathsf{P}}[f(X)Y|\mathcal{G}] = f(X) \, \mathbf{E}_{\mathsf{P}}[Y|\mathcal{G}]$.
- **Law of total expectation:** $\mathbf{E}_{\mathsf{P}}[\, X \,] = \mathbf{E}_{\mathsf{P}}[\, \mathbf{E}_{\mathsf{P}}[X|\mathcal{G}] \,]$.

- **Tower property (smoothing):** For sub-σ-fields $\mathcal{G}_1 \subseteq \mathcal{G}_2 \subseteq \mathcal{F}$,

$$\mathbf{E}_{\mathsf{P}}[\,\mathbf{E}_{\mathsf{P}}[X|\mathcal{G}_2]\,|\mathcal{G}_1] = \mathbf{E}_{\mathsf{P}}[\,X\,|\mathcal{G}_1].$$

- **Conditional variance:** $\mathrm{Var}_{\mathsf{P}}[X|\mathcal{G}] := \mathbf{E}_{\mathsf{P}}[\,(X - \mathbf{E}_{\mathsf{P}}[X|\mathcal{G}])^2\,|\mathcal{G}]$.

For the process X_s adapted to the filtration, this means that for $0 \leq t \leq \tau \leq T$, $\mathbf{E}_{\mathsf{P}}[f(X_t)Y|\mathcal{F}_\tau] = f(X_t)\mathbf{E}_{\mathsf{P}}[Y|\mathcal{F}_\tau]$ and $\mathbf{E}_{\mathsf{P}}[\,\mathbf{E}_{\mathsf{P}}[X_\tau|\mathcal{F}_t]\,] = \mathbf{E}_{\mathsf{P}}[\,X_\tau\,]$.

The **conditional expectation of** Y **given** $X = x \in \mathbb{R}^n$ is the Radon-Nikodym derivative

$$\mathbf{E}_{\mathsf{P}}[Y|X = x] := \frac{\mathrm{d}\nu}{\mathrm{d}\mathsf{P}_X},$$

where P_X is the distribution of X and

$$\nu(B) = \int_{Y^{-1}(B)} X \mathrm{d}\mathsf{P},$$

for $B \in \mathcal{F}'$, the σ-field X maps to. It has the property that

$$\mathbf{E}_{\mathsf{P}}[Y|X = x](X) = \mathbf{E}_{\mathsf{P}}[Y|X],$$

$\mathsf{P}|_{X^{-1}}(\mathcal{F}')$-a.s.. Essentially, this special type of conditional expectation is only defined on the parts of the space where X has non-trivial density.

A P-**martingale** is a special type of adapted process X_s which P-a.s. satisfies

$$\mathbf{E}_{\mathsf{P}}[X_\tau|\mathcal{F}_t] = X_t.$$

A standard n-dimensional P-**Brownian process** (Wiener process) W_s^{P} is a special type of

martingale where $W_0^P = 0_n$ and

$$\mathbf{E}_{\mathsf{P}}[(W_\tau^{\mathsf{P}} - W_t^{\mathsf{P}})(W_\tau^{\mathsf{P}} - W_t^{\mathsf{P}})^\top | \mathcal{F}_t] = (\tau - t) I_n,$$

P-a.s., i.e., the distribution of differences is a multivariate normal distribution

$$W_\tau^{\mathsf{P}} - W_t^{\mathsf{P}} \sim \mathcal{N}(0_n, (\tau - t) I_n).$$

2.1.4 Stochastic Differential Equations

An **Itô integral** (for a Brownian differential) is the asymptotic limit of integrating with respect to successive differences of a Brownian process W_s^{P}, denoted

$$Y_{t,\tau} := \int_t^\tau \sigma_s \mathrm{d}W_s^{\mathsf{P}}.$$

We have two important properties: that the conditional mean is zero with respect to the beginning of the integral,

$$\mathbf{E}_{\mathsf{P}}\left[\int_t^\tau f_s \mathrm{d}W_s^{\mathsf{P}} \middle| \mathcal{F}_t \right] = 0,$$

for vector process f_s, and **Itô isometry**, the following result for matrix process σ_s,

$$\mathbf{E}_{\mathsf{P}}\left[\left(\int_t^\tau \sigma_s \mathrm{d}W_s^{\mathsf{P}} \right) \left(\int_t^\tau \sigma_s \mathrm{d}W_s^{\mathsf{P}} \right)^\top \middle| \mathcal{F}_t \right] = \mathbf{E}_{\mathsf{P}}\left[\int_t^\tau \mathrm{tr}(\sigma_s \sigma_s^\top) \mathrm{d}s \middle| \mathcal{F}_t \right],$$

where tr is the trace operator, the sum of the diagonal elements of the matrix.

We say that a continuous process X_s solves a Markovian **Itô stochastic differential equation (SDE)** if it satisfies a particular Itô integral equation defined by a given starting random variable ξ_0, a drift function $f : [0, T] \times \mathbb{R}^n \to \mathbb{R}^n$, and a diffusion function matrix

$\sigma : [0, T] \times \mathbb{R}^n \to \mathbb{R}^{n \times k}$, and denoted as

$$\mathrm{d}X_s = f(s, X_s)\,\mathrm{d}s + \sigma(s, X_s)\,\mathrm{d}W_s^{\mathsf{P}}, \qquad X_0 = \xi_0.$$

If the condition is given at the end of the interval, i.e. $X_T = \xi_T$, we call this, instead, backward SDE (BSDE), the former called a forward SDE (FSDE) in this context. We say the solution X_s is **strong** if for any given Brownian process, W_s, $X_0 = \xi_0$ P-a.s. and for $t \in [0, T]$

$$X_t = X_0 + \int_0^t f(s, X_s)\,\mathrm{d}s + \int_0^t \sigma(s, X_s)\,\mathrm{d}W_s,$$

P-a.s.. We say the solution is **weak** if the particular Brownian process W_s^{P} and measure P used is part of the solution. We are now mostly equipped to state the problem we are interested in.

2.2 Stochastic Optimal and On-Policy Value Functions

In this section we introduce the continuous-time stochastic optimal control (C-SOC) problem, the assumptions and corresponding guarantees which can be made about the existence of solutions, and introduce the on-policy value function which will be used to approximate solutions. We are specifically interested in the finite-horizon C-SOC problem on the time interval $[0, T]$. Based upon the Dynamic Programming methodology [1, Chapter 4], we first consider the generalized sub-problem on the interval $[t, T]$ where $t \in [0, T]$. Let $(\Omega, \mathcal{F}, \{\mathcal{F}_s\}_{s\in[t,T]}, \mathsf{Q})$ be a complete, filtered probability space, on which W_s^{Q} is a k-dimensional standard Brownian (Wiener) process with respect to the probability measure Q and adapted to the filtration $\{\mathcal{F}_s\}_{s\in[t,T]}$. Consider a stochastic nonlinear system governed

by the Itô differential equation

$$\mathrm{d}X_s = f(s, X_s, u_s)\,\mathrm{d}s + \sigma(s, X_s)\,\mathrm{d}W_s^{\mathsf{Q}}, \tag{2.1}$$

where X_s is a state process taking values in $\mathbb{R}^n$, $u_{[t,T]}$ is a progressively measurable input process taking values in the compact set $\mathcal{U} \subseteq \mathbb{R}^m$, and $f : [0,T] \times \mathbb{R}^n \times \mathcal{U} \to \mathbb{R}^n$, $\sigma : [0,T] \times \mathbb{R}^n \times \mathcal{U} \to \mathbb{R}^{n\times k}$ are the Markovian drift and diffusion functions, respectively. The cost associated with a given starting time $t \in [0,T]$, starting state $X_t = x_t \in \mathbb{R}^n$, and control process $u_{[t,T]}$ is

$$J(t, x_t; u_{[t,T]}) := \mathbf{E}_{\mathsf{Q}}\bigg[\int_t^T \ell(s, X_s, u_s)\,\mathrm{d}s + g(X_T)\bigg], \tag{2.2}$$

where $\ell : [0,T] \times \mathbb{R}^n \times \mathcal{U} \to \mathbb{R}_+$ is the running cost $g : \mathbb{R}^n \to \mathbb{R}_+$ is the terminal cost, and $\mathbf{E}_{\mathsf{Q}}$ refers to the expectation over the probability measure Q. The optimal expected cost-to-go for any start time and state (t, x_t) is encoded in the following function.

Continuous-Time Stochastic Optimal Value Function

The continuous-time stochastic optimal value function $V^* : [0,T] \times \mathbb{R}^n \to \mathbb{R}_+$ is defined as the optimization

$$V^*(t, x_t) = \inf_{u_{[t,T]}} J(t, x_t; u_{[t,T]}), \tag{C-V^*}$$

over the space of admissible control processes $u_{[t,T]}$ (see [1, Chapter 4, Section 3.1]).

Determining this function is important for finding optimal policies and solving the optimal control problems, in general.

We now briefly discuss what assumptions may be made about the parameters f, σ, ℓ, g of (C-V^*) and what guarantees they provide. Henceforth, for any function $\phi(t,x)$ or $\phi(x)$, we denote $\partial_t\phi$, $\partial_x\phi$, and $\partial_{xx}\phi$ the partial derivative of ϕ with respect to t, the gradient with

respect to x, and Hessian with respect to x, respectively. Further, we denote $C^{1,2}([0,T],\mathbb{R}^n)$ ($C_b^{1,2}([0,T],\mathbb{R}^n)$) as the set of functions for which $\phi, \partial_t\phi, \partial_x\phi, \partial_{xx}\phi$ are all continuous (and bounded) on $(t,x) \in [0,T] \times \mathbb{R}^n$.

We always make the following assumption of (C-V^*):

Assumption (A1) The functions f, σ, ℓ, g are uniformly Lipschitz continuous in x, that is, there exists a constant $L > 0$ such that for each $\phi = f, \sigma, \ell, g$,

$$\begin{aligned} \|\phi(t,x,u) - \phi(t,\widehat{x},u)\| &\leq L\|x - \widehat{x}\|, \quad \forall t \in [0,T], x, \widehat{x} \in \mathbb{R}^n, u \in \mathcal{U}, \\ \|\phi(t,0,u)\| &\leq L, \quad \forall t \in [0,T], u \in \mathcal{U}. \end{aligned} \tag{A1}$$

Given assumption (A1), (C-V^*) is a unique viscosity solution of the Hamilton-Jacobi-Bellman (HJB) equations

$$\begin{aligned} \partial_t V^* + \frac{1}{2}\operatorname{tr}[\sigma(t,x)\sigma(t,x)^\top \partial_{xx} V^*] + \inf_{u \in \mathcal{U}} \left\{ (\partial_x V^*)^\top f(t,x,u) + \ell(t,x,u) \right\} = 0, \\ V^*(T,x) = g(x), \end{aligned} \tag{C-HJB}$$

[1, Chapter 4, Theorem 5.2, Theorem 6.1]. The optimal value function is guaranteed to be Lipschitz in x_t for fixed t, but only Hölder continuous of order $1/2$ in t for fixed x_t. [1, Chapter 4, Proposition 3.1].

In order to guarantee a smoother value function we must make stronger assumptions on the parameters.

Assumption (A2) (Implies A1) For each $\phi = f, \sigma, \ell$ and $u \in \mathcal{U}$,

$$\phi \in C_b^{1,2}([0,T],\mathbb{R}^n). \tag{A2}$$

We further assume g is thrice continuously differentiable and bounded in x.

Assumption (A3) The diffusion term σ is square, nonsingular on $[0, T] \times \mathbb{R}^n$, and its inverse is uniformly bounded, that is,

$$\sup_{t,x} \|\sigma(t, x)^{-1}\| < \infty. \tag{A3}$$

If we make the stronger assumptions (A2) and (A3), we can guarantee that (C-HJB) has a unique solution $V^* \in C_b^{1,2}([0, T], \mathbb{R}^n)$ [6, Chapter 4, Theorem 4.2]. In this case, we say that (C-HJB) has a classical solution since $\partial_t V^*, \partial_x V^*, \partial_{xx} V^*$ are well defined on $[0, T] \times \mathbb{R}^n$.

When we have a problem which satisfies (A1) but not (A2) and (A3), it might be useful to approximate V^* with an auxiliary problem guaranteed to be smooth. Consider the following assumption:

Assumption (A4) (Implies A3) The nominal diffusion term $\widehat{\sigma}$ has been modified to add a small $\varepsilon > 0$ amount of noise in all n dimensions such that

$$\sigma = \sigma^\varepsilon \approx \mathrm{square}_n(\widehat{\sigma}), \tag{A4}$$

satisfies assumption (A3), where square_n fits the matrix into an $n \times n$ matrix padded with zeros.

For example, the following choice always exists for any $\widehat{\sigma} \in \mathbb{R}^{n \times m}$. The matrix $S = \widehat{\sigma}\widehat{\sigma}^\top + \varepsilon I_n$ is uniformly strictly positive definite and thus it can always be decomposed as $S = \sigma^\varepsilon \sigma^{\varepsilon\top}$ for some invertible σ^ε, obtained, e.g., with the Cholesky decomposition or

matrix square root. Taking assumption (A1) and the regularization (A4), we have both that the augmented problem converges to the nominal problem as $\varepsilon \to 0$ [1, Chapter 4, Proposition 4.1], and that (C-HJB) admits a classical solution [1, p. 197]. This method for regularizing PDEs is called the *vanishing viscosity* method. We can use this method for any nominal $\widehat{\sigma}$, including deterministic problems where $\widehat{\sigma} \equiv 0$.

The iterative approach to solve the optimal control problem is to successively improve approximations of the optimal policy and optimal value function (π^*, V^*), refining an arbitrary policy μ and its associated on-policy value function V^μ which characterizes the cost-to-go under this policy. Consider the space of admissible feedback policies, that is, measurable functions $\mu : [0,T] \times \mathbb{R}^n \to \mathcal{U}$ for which there exists a weak SDE solution for

$$\mathrm{d}X_s = f_s^\mu \,\mathrm{d}s + \sigma_s \,\mathrm{d}W_s^{\mathsf{Q}}, \tag{2.3}$$

where $f_s^\mu := f^\mu(s, X_s)$, $f^\mu := f(t, x, \mu(t,x))$, and henceforth abbreviate ℓ, and σ similarly. Given a target policy μ, the on-policy value function $V^\mu : [0,T] \times \mathbb{R}^n \to \mathbb{R}_+$ is defined as

$$V^\mu(t, x_t) = \mathbf{E}_{\mathsf{Q}}[\int_t^T \ell_s^\mu \,\mathrm{d}s + g(X_T)\,], \tag{2.4}$$

with the process X_s satisfying the SDE (2.3), and starting at $X_t = x_t \in \mathbb{R}^n$. The associated continuous-time on-policy Hamilton-Jacobi PDE is

$$\begin{aligned} \partial_t V^\mu + \frac{1}{2}\operatorname{tr}[\sigma\sigma^\top \partial_{xx} V^\mu] + (\partial_x V^\mu)^\top f^\mu + \ell^\mu\big|_{t,x} = 0, \\ V^\mu(T,x) = g(x), \end{aligned} \tag{C-HJ}$$

for $(t,x) \in [0,T) \times \mathbb{R}^n$. The guarantees on the solutions of (C-HJ) are similar to the C-HJB case, with f^μ, ℓ^μ replacing f, ℓ. Note, however, that the smoothness properties of f^μ, ℓ^μ are not necessarily guaranteed by smoothness properties of f, ℓ.

Since we are searching over a space of feedback policies for an optimal policy, it is

useful to know when we can guarantee the existence of an optimal policy in this space. When (C-HJB) has a classical solution, a feedback policy π^* satisfying the inclusion

$$\pi^*(t,x) \in \arg\min_{u \in U} \{\ell(t,x,u) + f(t,x,u)^\top \partial_x V^*(t,x)\}, \tag{2.5}$$

is optimal, that is, $V^{\pi^*} \equiv V^*$, according to the classical stochastic verification theorem [1, Chapter 5, Theorem 5.1]. When (C-HJB) has a viscosity solution which is not classical, we may need additional assumptions. If π^* satisfies (2.5) where $\partial_x V^*$ and $\partial_{xx} V^*$ exist, and there exist corresponding superdifferential replacements when they do not, then π^* might be optimal under some additional assumptions [1, Chapter 5, Theorem 6.2]. We can satisfy most of these assumptions by assuming (A3) [1, Chapter 5, Theorem 6.6].

For ease of presentation we assume that the underlying C-SOC of interest satisfies assumption (A1) and that the on-policy value function parameters satisfy (A2) and (A3), or (A1) and (A4) (for a reasonably small ε), such that (C-HJ) yields a classical solution and σ^{-1} is uniformly bounded. Later we will discuss how the regularization of (A4) might effect the accuracy of the proposed methods.

Given the previous discussion, we now formally state the C-SOC problem.

Continuous-Time Stochastic Optimal Control Problem (C-SOC)

The C-SOC problem is to determine or approximate the optimal value function V^* (C-V^*) and the optimal feedback policy π^* on $[0,T] \times \mathbb{R}^n$.

Of course, fully approximating the function over the entire space is usually unreasonable for numerical methods applied to generalized problems because nonlinear dynamics and nonquadratic costs can introduce locally irregular topology in the value function. In the next chapter we will discuss how this assumption may be relaxed to produce a more realistic problem.

CHAPTER 3

ON/OFF-POLICY FBSDE AND THE FBSDE SOC PROBLEM

In this chapter we discuss the on-policy and off-policy FBSDE representations of the on-policy value function. We also discuss a few tools which are useful for numerically solving FBSDEs, and conclude with a reformulation of the C-SOC problem presented at the end of the previous section, which is more sensitive to the numerical challenges of FBSDE methods.

Henceforth, we use the following example problem to illustrate the various concepts.

1-D Nonlinear SOC Example

We define the costs and dynamics as

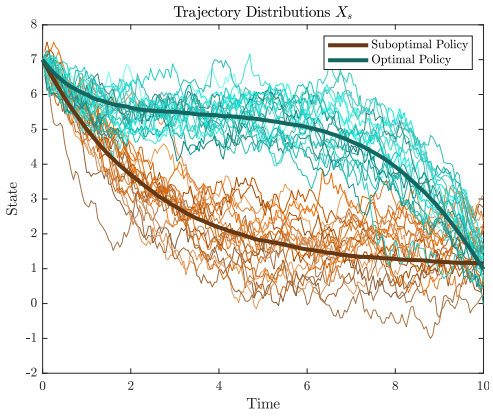
$$\mathrm{d}X_s = \big(0.1(X_s - 3)^2 + 0.2u_s\big)\mathrm{d}s + 0.8\,\mathrm{d}W_s, \qquad x_0 = 7,$$
$$J(t, x_t; u_{[t,T]}) = \mathbf{E}_{\mathsf{Q}}\bigg[\int_t^T \big(12\,|X_s - 6| + 0.4\,u_s^2\big)\,\mathrm{d}s + 25\,X_T^2\bigg],$$

over a time interval of length $T = 10$. We alternately consider the target policy μ as either the optimal policy $\mu = \pi^*$ or the suboptimal policy $\mu = \widetilde{\pi}$, defined as

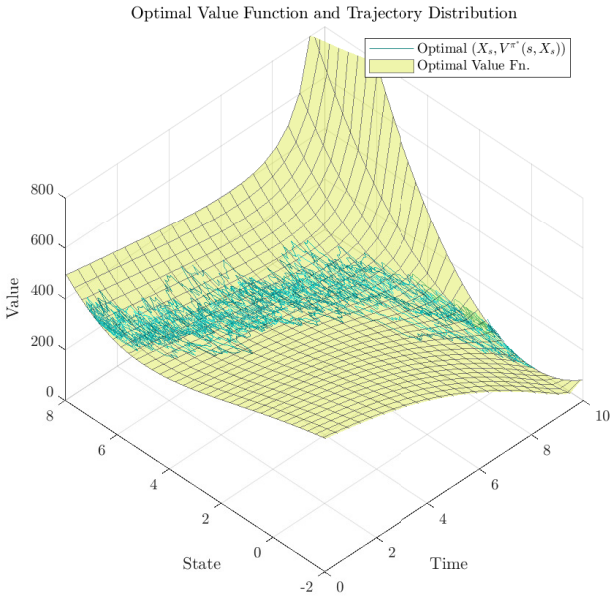
$$\widetilde{\pi}(t, x) = -0.5(x - 3)^2 - 2(x - 1).$$

3.1 On-Policy FBSDE

As discussed in the previous section, we first focus on solving the on-policy value function V^μ, for an arbitrary policy μ. To that end, we begin our investigation of Feynman-Kac FBSDE numerical methods by introducing the on-policy representation. The positivity of



(a) FSDE distributions



(b) Optimal value function with $(X_s, V^*(s, X_s))$ under π^*

Figure 3.1: Visualizing the 1-D Nonlinear SOC Example. The red/green trajectories are generated from samples of (2.3) following the suboptimal policy $\widetilde{\pi}$ or optimal policy π^*. Thick lines are mean values. The cyan trajectories follow π^*.

$\sigma\sigma^\top$ yields that (C-HJ) is a parabolic PDE and, hence, we can apply the Feynman-Kac Theorem, originally attributed to [56].

On-Policy Feynman-Kac FBSDEs

Theorem 3.1 (On-Policy Feynman-Kac Theorem)**.** *Letting V^μ be the solution of* (C-HJ)*, consider the pair of FBSDEs*

$$\mathrm{d}X_s = f_s^\mu \,\mathrm{d}s + \sigma_s \,\mathrm{d}W_s^{\mathsf{Q}}, \qquad X_0 = x_0, \tag{3.1}$$

$$\mathrm{d}Y_s = -\ell_s^\mu \,\mathrm{d}s + Z_s^\top \,\mathrm{d}W_s^{\mathsf{Q}}, \qquad Y_T = g(X_T), \tag{3.2}$$

where Y_s and Z_s are, respectively, one and n-dimensional adapted processes. There ▶

On-Policy Feynman-Kac FBSDEs (cont)

exists a unique solution (X_s, Y_s, Z_s) *of this FBSDE system that satisfies*

$$\begin{aligned} Y_s &= V^\mu(s, X_s), && s \in [0, T], \\ Z_s &= \sigma_s^\top \partial_x V^\mu(s, X_s), && \textit{a.e. } s \in [0, T], \end{aligned} \tag{3.3}$$

Q*-almost surely (a.s.). In particular,* $Y_0 = V^\mu(0, x_0)$. □

Proof. See [1, Chapter 7, Theorem 4.5, (4.29)]. □

We call (3.1) the forward SDE (FSDE) and (3.2) the backward SDE (BSDE). Henceforth, we assume (X_s, Y_s, Z_s) is a solution to the FBSDE system (3.1) and (3.2).

This theorem demonstrates that there is an intrinsic relationship between the solution of FBSDEs following a target policy μ and the smooth surfaces of its on-policy value function V^μ, as is illustrated in Figure 3.2. The relationship works both ways in how it can be used numerically: solving the FBSDEs can inform the solution of the value function, and solving the value function can inform the solution of the FBSDEs. The following result makes this more clear, demonstrating how the BSDE relationship applies over short time intervals $[t, \tau]$, where $0 \le t \le \tau \le T$. We define

$$\widehat{Y}_{t,\tau} := Y_\tau - \Delta\widehat{Y}_{t,\tau}, \tag{3.4}$$

as an estimator for Y_t, where $\Delta\widehat{Y}_{t,\tau}$ is an estimator for the difference $\Delta Y_t = Y_\tau - Y_t$. We have the following two results for the choice of this difference estimator.

Corollary 3.2. *If*

$$\Delta\widehat{Y}^{\text{noisy}}_{t,\tau} := -\int_t^\tau \ell_s^\mu \mathrm{d}s + \int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}}, \tag{3.5}$$

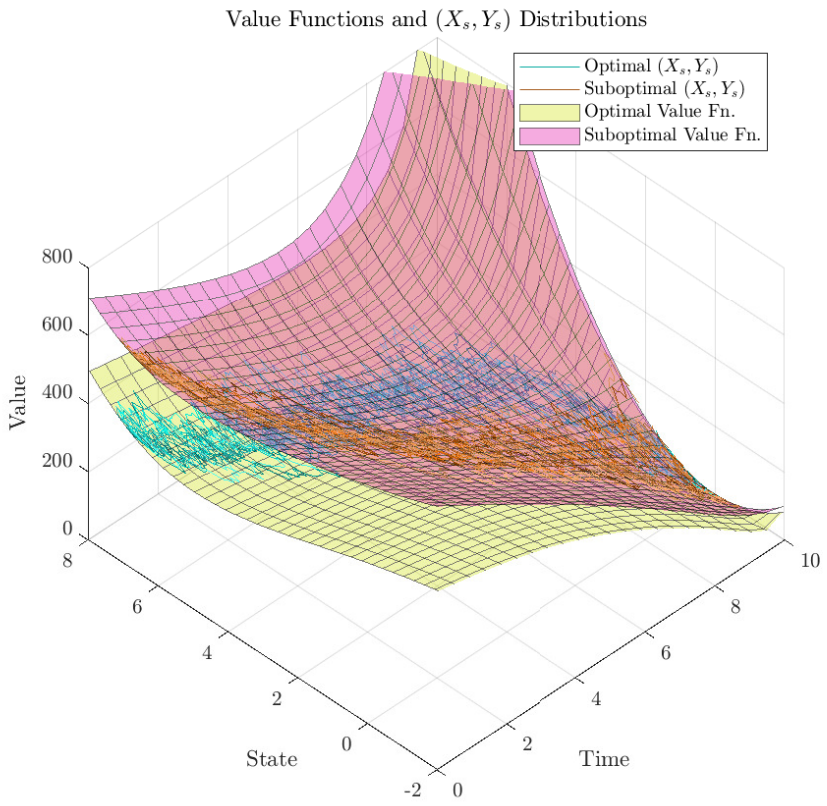


Figure 3.2: Illustrating the on-policy Feynman-Kac representation theorem. The two distributions (X_s, Y_s) solve the FBSDE system (3.1) and (3.2) starting at $X_0 = x_0 = 7$, where the target policy is either optimal $\mu = \pi^*$ or suboptimal $\mu = \widetilde{\pi}$. The Feynman-Kac theorem indicates that each distribution will Q-a.s. lie on the surface of its respective value function V^μ.

then,

$$Y_t = \widehat{Y}_{t,\tau}^{\text{noisy}} = \mathbf{E}_{\mathsf{Q}}[\widehat{Y}_{t,\tau}^{\text{noisy}} | X_t] = V^\mu(t, X_t), \tag{3.6}$$

Q-*a.s.*. □

Proof. By the definition of a BSDE solution [1, Chapter 7, Definition 3.1],

$$\begin{aligned} Y_t &= g(X_T) + \int_t^T \ell_s^\mu \mathrm{d}s - \int_t^T Z_s^\top \mathrm{d}W_s^{\mathsf{Q}} \\ &= g(X_T) + \int_\tau^T \ell_s^\mu \mathrm{d}s - \int_\tau^T Z_s^\top \mathrm{d}W_s^{\mathsf{Q}} + \int_t^\tau \ell_s^\mu \mathrm{d}s - \int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}} \\ &= Y_\tau - \Delta \widehat{Y}_{t,\tau}^{\text{noisy}} \\ &= \widehat{Y}_{t,\tau}^{\text{noisy}}, \end{aligned}$$

Q-a.s.. Since Y_t (and consequently $\widehat{Y}_{t,\tau}^{\text{noisy}}$) is X_t-measurable due to (3.3), it follows that $Y_t = \mathbf{E}_{\mathsf{Q}}[\widehat{Y}_{t,\tau}^{\text{noisy}} | X_t]$. □

Corollary 3.3. *If*

$$\Delta \widehat{Y}_{t,\tau}^{\text{noiseless}} := -\int_t^\tau \ell_s^\mu \mathrm{d}s, \tag{3.7}$$

then,

$$Y_t = \mathbf{E}_{\mathsf{Q}}[\widehat{Y}_{t,\tau}^{\text{noiseless}} | X_t] = V^\mu(t, X_t), \tag{3.8}$$

Q-*a.s.*. □

Proof. The equality $\mathbf{E}_{\mathsf{Q}}[\widehat{Y}_{t,\tau}^{\text{noisy}} | X_t] = \mathbf{E}_{\mathsf{Q}}[\widehat{Y}_{t,\tau}^{\text{noiseless}} | X_t]$ follows immediately from the standard property of Itô integrals [1, Chapter 7, Theorem 3.2] (and the tower property of conditional expectation) yielding $\mathbf{E}_{\mathsf{Q}}[\int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}} | X_t] = \mathbf{E}_{\mathsf{Q}}[\int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}} | \mathcal{F}_t] = 0$. □

In the language of estimation theory, $\widehat{Y}_{t,\tau}^{\text{noisy}}$ and $\widehat{Y}_{t,\tau}^{\text{noiseless}}$ are unbiased estimators of Y_t. While unbiasedness is a good property of an estimator, low variance is also good.

Proposition 3.4. *The conditional variance of these estimators is*

$$\mathrm{Var}_{\mathsf{Q}}[\widehat{Y}^{\text{noisy}}_{t,\tau}|X_t] = 0 \tag{3.9}$$

$$\mathrm{Var}_{\mathsf{Q}}[\widehat{Y}^{\text{noiseless}}_{t,\tau}|X_t] = \mathbf{E}_{\mathsf{Q}}\left[\int_t^\tau \|Z_s\|^2 \,\mathrm{d}s \middle| X_t\right] \tag{3.10}$$

Q-*a.s.*. □

Proof. Note that the deviation can be reduced to

$$\widehat{Y}^{\text{noisy}}_{t,\tau} - \mathbf{E}_{\mathsf{Q}}[\widehat{Y}^{\text{noisy}}_{t,\tau}|X_t] = Y_t - \mathbf{E}_{\mathsf{Q}}[Y_t|X_t] = 0$$

and

$$\begin{aligned}
\widehat{Y}^{\text{noiseless}}_{t,\tau} - \mathbf{E}_{\mathsf{Q}}[\widehat{Y}^{\text{noiseless}}_{t,\tau}|X_t] &= Y_\tau - \Delta\widehat{Y}^{\text{noiseless}}_{t,\tau} - \mathbf{E}_{\mathsf{Q}}[Y_\tau - \Delta\widehat{Y}^{\text{noiseless}}_{t,\tau}|X_t] \\
&= Y_\tau - \Delta\widehat{Y}^{\text{noisy}}_{t,\tau} - \mathbf{E}_{\mathsf{Q}}[Y_\tau - \Delta\widehat{Y}^{\text{noisy}}_{t,\tau}|X_t] + \int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}} \\
&= \widehat{Y}^{\text{noisy}}_{t,\tau} - \mathbf{E}_{\mathsf{Q}}[\widehat{Y}^{\text{noisy}}_{t,\tau}|X_t] + \int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}} \\
&= \int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}}
\end{aligned}$$

We plug both of these into the definition of conditional variance

$$\mathrm{Var}_{\mathsf{Q}}[\widehat{Y}_{t,\tau}|X_t] := \mathbf{E}_{\mathsf{Q}}[(\widehat{Y}_{t,\tau} - \mathbf{E}_{\mathsf{Q}}[\widehat{Y}_{t,\tau}|X_t])^2|X_t],$$

and the result for the noisy estimator follows easily. The result for the noiseless estimator follows from the Itô isometry [1, Chapter 1, Proposition 5.3]. □

In theory, the noiseless estimator is a better estimator because it has less variance, but in numerical methods this requires accurate computation of $\int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{Q}}$ for the estimator to be unbiased. By excluding the term entirely, we may introduce some variance in the estimator, but we will introduce less bias into the estimator through numerical error. Also consider

that the variance of the noiseless estimator (and likely the bias in a numerical estimate of the noisy estimator) scales linearly with the duration of the time interval $\Delta t := \tau - t$, justifying that time intervals be kept short.

3.2 Least Squares Monte Carlo

Least squares Monte Carlo (LSMC) is a scheme for obtaining the parameters of a parametric model of the value function V^μ, originally credited to [25] for use in BSDE problems. We first state a general property of conditional expectation that arises from the fact that $V^\mu(t, X_t) = \mathbf{E}_\mathsf{Q}[\widehat{Y}_{t,\tau}|X_t]$.

Corollary 3.5. *The minimizer ϕ^* of*

$$\inf_{\phi \in L_2} \mathbf{E}_\mathsf{Q}[(\widehat{Y}_{t,\tau} - \phi)^2], \tag{3.11}$$

over X_t-measurable square integrable variables ϕ coincides with the value function, that is, $\phi^* = V^\mu(t, X_t)$. □

Proof. This follows from the L_2-projective properties of conditional expectation [57, Chapter 10.3, Property 11] applied to (3.6). □

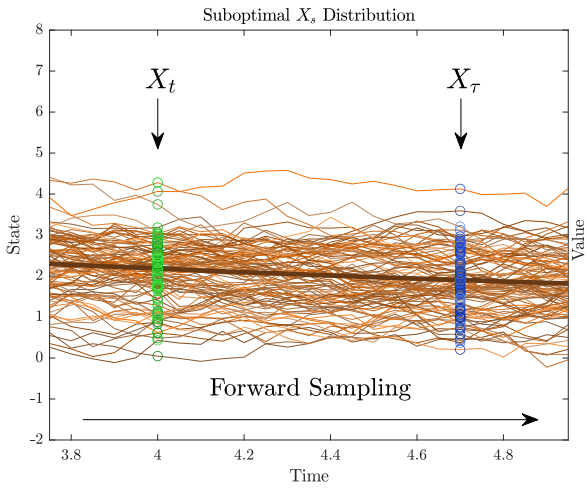
In LSMC numerical methods, we approximate the minimization in (3.11) over the subspace of X_t-measurable variables $\{\phi(X_t; \alpha) : \alpha \in \mathcal{A}\}$, where $\phi(x; \alpha)$ is a function representation with parameters $\alpha \in \mathcal{A}$ (we assume henceforth that $\phi(x; \alpha) \in C^2(\mathbb{R}^n)$ for all $\alpha \in \mathcal{A}$). Let $\{(x_t^k, \widehat{y}_t^k)\}_{k=1}^M$ be a set of samples approximating the joint distribution $(X_t, \widehat{Y}_{t,\tau})$, denoted as $\widetilde{\mathsf{Q}}$. The optimal parameters for this representation are found by minimizing

$$\begin{aligned}\arg\min_{\alpha \in \mathcal{A}} \mathbf{E}_\mathsf{Q}[(\widehat{Y}_{t,\tau} - \phi(X_t; \alpha))^2] &\approx \arg\min_{\alpha \in \mathcal{A}} \mathbf{E}_{\widetilde{\mathsf{Q}}}[(\widehat{Y}_{t,\tau} - \phi(X_t; \alpha))^2] \\ &= \arg\min_{\alpha \in \mathcal{A}} \sum_{k=1}^M \frac{1}{M} (\widehat{y}_t^k - \phi(x_t^k; \alpha))^2 =: \alpha_t^*. \end{aligned} \tag{3.12}$$

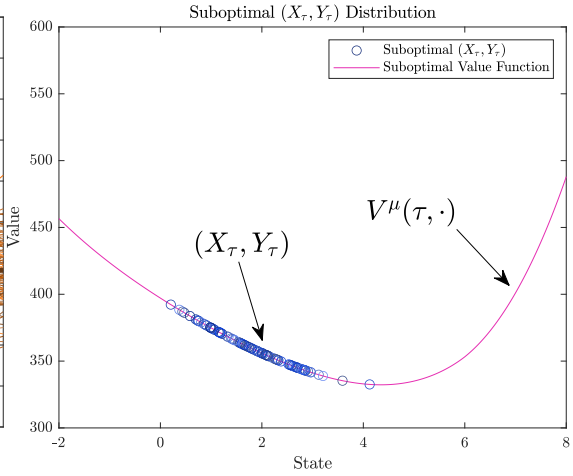
When the function representation is linear in the parameters $\phi(x;\alpha) = \Phi(x)\alpha$ this optimization is a linear least squares regression problem in $\mathcal{A}$. The optimal parameters define the new approximate representation of the value function, by

$$V^{\mu}(t,x) \approx \widetilde{V}^{\mu}(t,x) := \phi(x;\alpha_t^*). \tag{3.13}$$

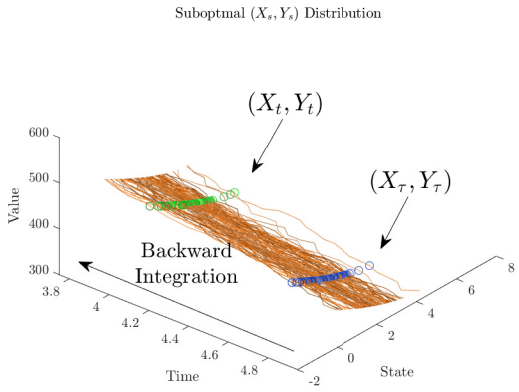
Figure 3.3 illustrates, from a theoretical perspective, how Feynman-Kac FBSDEs and LSMC are used in numerical methods to approximate the on-policy value function V^{μ}.



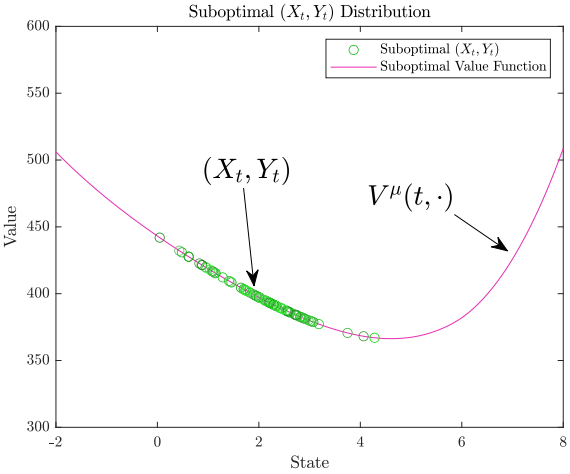
(a) FSDE (3.1) sampled



(b) Y_τ determined from relationship (3.3)



(c) Y_t estimated from backward integrating (3.2)



(d) $V^\mu(t, \cdot)$ determined from LSMC (3.12)

Figure 3.3: Feynman-Kac FBSDE methods over short intervals for the suboptimal policy $\mu = \widetilde{\pi}$. **(a)** First, the FSDE is sampled to the end of the interval $s = T$. **(b-d)** In a series of backward steps from $s = \tau$ to $s = t$, starting at $\tau = T$, we estimate the value function $V^\mu(t, \cdot)$. **(b)** The distribution (X_τ, Y_τ) is determined from $Y_\tau = V^\mu(\tau, X_\tau)$, then, **(c)** backward integrated to produce $\widehat{Y}_{t,\tau}$, an estimator for Y_t. **(d)** Finally, using LSMC regression the value function $V^\mu(t, \cdot)$ is approximated using a parametric optimization. In numerical methods, the distribution (X_t, Y_t) obtained by backward integration will incur numerical error and will not lie exactly on the curve.

We make a brief comment about conditional expectation $\mathbf{E}_\mathsf{Q}[\cdot | X_t = x_t]$ given $X_t = x_t \in \mathbb{R}^n$. Let $\mathsf{Q}_{X_t} := \mathsf{Q} \circ X_t^{-1}$ denote the distribution of the random vector X_t in the

probability measure Q. The statement $Y_t = V^\mu(t, X_t)$, Q-a.s., implies

$$\mathbf{E}_\mathsf{Q}[Y_t | X_t = x_t] = \mathbf{E}_\mathsf{Q}[V^\mu(t, X_t) | X_t = x_t] = V^\mu(t, x_t), \quad \mathsf{Q}_{X_t}\text{-a.s.},$$

but not for all $x_t \in \mathbb{R}^n$. We must use the qualification Q_{X_t}-a.s. because $\mathbf{E}_\mathsf{Q}[\cdot | X_t = x_t]$ is not well-defined over Q_{X_t}-null sets. In other words, when x_t is a point where the distribution of X_t is non-trivial in Q, the relationship above can be assumed. Although for many systems the distribution of X_t might be broadly non-trivial in the same sense that Gaussian distributions are everywhere non-trivial, the statement still has an important interpretation in numerical methods, especially when the LSMC approximate optimization (3.12) is used. Numerical on-policy Feynman-Kac FBSDE methods are only capable of determining the value function $V^\mu(t, \cdot)$ over the Monte Carlo approximation of the distribution Q_{X_t}. We must rely on the smoothness of the value function and its approximation to extrapolate to values far from the distribution used to approximate it. In Section 3.3 we will show that although this relationship will always be restricted to a distribution over X_t, it need not necessarily be the distribution Q_{X_t} associated with the FSDE (3.1).

3.3 Off-Policy Drifted FBSDE

As discussed at the end of Section 3.2, applying LSMC to the on-policy pair of FBSDEs (3.1) and (3.2) only allows for us to solve for $V^\mu(t, \cdot)$ in the distribution Q_{X_t}, generated by the FSDE governed by policy μ. This is a very rigid constraint, considering the value function is defined on all of $[0, T] \times \mathbb{R}^n$. Further, we are iteratively looking to improve the policy and thus our target policy μ (and thus the distribution Q_{X_t}) is constantly changing. It is useful, in general, to solve for the value function in regions outside of this distribution.

One potential solution is to solve the value function at (t, x_t) with low density by repeating the FBSDE method on the interval $[t, T]$ for different start states x_t, but there are several problems with this approach. First of all, a number of samples must be produced to

represent each start state, potentially wasting their ability to contribute information to each other. Secondly, it is hard to determine exactly where these points should be located and how they should be distributed. Finally, such an approach potentially breaks the continuity of distribution that a forward SDE provides. In backward steps, we are trying to produce estimates of the value function which eventually lead back to x_0. If we start over at new (t, x_t) which is unreachable by x_0 we will waste computation on subproblems which are unhelpful for the original problem.

Instead of creating a complicating set of subproblems, we can utilize a result from stochastic control theory to simply change the probability measure over the trajectory distributions, and still maintain the results from the Feynman-Kac theorem. We now present a result based on Girsanov's theorem, namely, that an alternative pair of *drifted* FBSDEs with a different trajectory distribution can be used to estimate the same value function V^μ. This result will be used to disentangle the drift of the forward distribution from the policy associated with the value function.

Off-Policy Feynman-Kac-Girsanov FBSDEs

Theorem 3.6 (Feynman-Kac-Girsanov Theorem)**.** *Let* $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\in[0,T]}, \mathsf{P})$ *be a new filtered probability space on which* W_s^{P} *is Brownian and let* K_s *be any* $\mathcal{F}_s$*-progressively measurable process on the interval* $[0, T]$ *such that*

$$D_s := \sigma_s^{-1}(f_s^\mu - K_s), \tag{3.14}$$

is bounded and

$$\mathrm{d}X_s = K_s\,\mathrm{d}s + \sigma_s\,\mathrm{d}W_s^{\mathsf{P}}, \qquad X_0 = x_0, \tag{3.15}$$

admits a unique square-integrable solution X_s *(see e.g. [1, Chapter 1, Theorem 6.16]). Then, the Hamilton-Jacobi PDE* (C-HJ) *has a representation as the unique square-* ▶

Off-Policy Feynman-Kac-Girsanov FBSDEs (cont)

integrable solution (X_s, Y_s, Z_s) *to the FBSDE system consisting of* (3.15) *and*

$$\mathrm{d}Y_s = -(\ell_s^\mu + Z_s^\top D_s)\,\mathrm{d}s + Z_s^\top \mathrm{d}W_s^{\mathsf{P}}, \qquad Y_T = g(X_T), \tag{3.16}$$

in the sense that

$$\begin{aligned} Y_s &= V^\mu(s, X_s), && s \in [0,T], \\ Z_s &= \sigma_s^\top \partial_x V^\mu(s, X_s), && \textit{a.e. } s \in [0,T], \end{aligned} \tag{3.17}$$

P-*a.s.*. □

Proof. The existence of a square-integrable solution to (3.15) allows the conditions of [1, Chapter 7, Theorem 3.2] to be satisfied for (3.16), guaranteeing a unique square-integrable solution (Y_s, Z_s). Now define the processes

$$W_t^{\mathsf{Q}} := W_t^{\mathsf{P}} - \int_0^t D_s\,\mathrm{d}s, \tag{3.18}$$

$$\Theta_t := \exp\left(-\frac{1}{2}\int_0^t \|D_s\|^2\,\mathrm{d}s + \int_0^t D_s^\top \mathrm{d}W_s^{\mathsf{P}} \right), \tag{3.19}$$

for $t \in [0,T]$. Since D_s is bounded, Girsanov's theorem [7, Chapter 5, Theorem 10.1] implies that the process W_s^{Q} defined by (3.18) is Brownian in some measure Q derived from P in the form of

$$\mathrm{d}\mathsf{Q} = \Theta_T\,\mathrm{d}\mathsf{P}, \tag{3.20}$$

where Θ_T is the Radon-Nikodym derivative. With a simple algebraic reduction (checked with the substitution $\mathrm{d}W_s^{\mathsf{P}} = \mathrm{d}W_s^{\mathsf{Q}} + D_s\mathrm{d}s$), Girsanov's theorem also guarantees separately that X_s (weakly) solves the on-policy FSDE (2.3), and that (X_s, Y_s, Z_s) solves the on-policy

BSDE (3.2). Here, the idea is that the sample functions for the processes are the same, but the probability measure (acting on sets of trajectory samples ω) which characterizes their distributions changes.

Since D_s is bounded, it satisfies Novikov's criterion [58, Theorem 15.4.2] and thus, it follows that Θ_t is P-a.s. strictly positive, and further that the measures P and Q are equivalent, that is, they are absolutely continuous with respect to the other [59]. Since (3.3) holds Q-a.s., there exists an $N \in \mathcal{F}$ such that $E^{\mathsf{c}} \subseteq N$, where $E := \{\omega \in \Omega : Y_t(\omega) = V^{\mu}(t, X_t(\omega))\}$, and $\mathsf{Q}(N) = 0$. It subsequently follows from the definition of absolute continuity that $\mathsf{P}(N) = 0$. Thus, (3.3) holds P-a.s. as well. □

As before, we have the following relationships over short intervals, given (X_s, Y_s, Z_s), a solution to the drifted FBSDE system (3.15) and (3.16), and the definition

$$\widehat{Y}_{t,\tau} := Y_\tau - \Delta\widehat{Y}_{t,\tau},$$

identical to the previous definition (3.4).

Corollary 3.7. *If*

$$\Delta\widehat{Y}^{\text{noisy}}_{t,\tau} := -\int_t^\tau (\ell^{\mu}_s + Z_s^\top D_s)\mathrm{d}s + \int_t^\tau Z_s^\top \mathrm{d}W^{\mathsf{P}}_s, \tag{3.21}$$

then,

$$Y_t = \widehat{Y}^{\text{noisy}}_{t,\tau} = \mathbf{E}_{\mathsf{P}}[\widehat{Y}^{\text{noisy}}_{t,\tau} | X_t] = V^{\mu}(t, X_t), \tag{3.22}$$

P-*a.s.*. □

Proof. The proof follows similarly to the proof of Corollary 3.2. □

Corollary 3.8. *If*

$$\Delta \widehat{Y}_{t,\tau}^{\text{noiseless}} := -\int_t^\tau (\ell_s^\mu + Z_s^\top D_s)\mathrm{d}s, \tag{3.23}$$

then,

$$Y_t = \mathbf{E}_{\mathsf{P}}[\widehat{Y}_{t,\tau}^{\text{noiseless}} | X_t] = V^\mu(t, X_t), \tag{3.24}$$

P-*a.s.*. □

Proof. The proof follows similarly to the proof of Corollary 3.3. □

The analysis of conditional variance of the drifted estimators given in Proposition 3.4 holds true when the measure Q is replaced with P, as well as the discussion about LSMC in Section 3.2. Since the approximation of Z_s is required in both of these estimators (as opposed to just the noisy estimator in the on-policy formulation), there is additional bias and variance introduced in either case.

We can interpret the Feynman-Kac-Girsanov theorem in the following sense. As long as the diffusion function σ is the same as in the problem formulation, we can pick an arbitrary process K_s to be the drift term, which generates a distribution for the forward process X_s in the corresponding measure P. The BSDE yields an expression for Y_t using the same process W_s^{P} as used in the FSDE. The term $Z_s^\top D_s$ acts as a correction in the BSDE to compensate for changing the drift of the FSDE. We can again use the minimization (3.12) to approximate the value function V^μ, the only difference being that $(x_t^k, \widehat{y}_t^k)$ are now samples approximating the distribution $\mathsf{P}_{(X_t, \widehat{Y}_{t,\tau})}$.

Figure 3.4 illustrates the results of this theorem. Note that the drifted, off-policy formulation is a generalization of the on-policy formulation. Indeed, when $K_s \equiv F_s^\mu$, we have $D_s \equiv 0$ and $W_s^{\mathsf{Q}} \equiv W_s^{\mathsf{P}}$. When comparing this figure to Figure 3.2, we see that since the optimal drift $K_s = F_s^{\pi^*}$ corresponds to the target policy $\mu = \pi^*$, the off-policy result is the

same as the on-policy result. The forward SDE distributions X_s for the suboptimal case are also identical in both figures, that is, if we were to remove the value axis from the figures the distributions would be the same for the orange trajectories. The only difference between the figures is that the joint process (X_s, Y_s) now lies on the value function associated with the target policy V^{π^*} (the yellow surface) instead of its on-policy value function $V^{\widetilde{\pi}}$ (the magenta surface). In practice it is not likely that we will have access to the optimal policy π^*, but we use it in these examples for ease of illustration.

It should be highlighted that K_s need not be a deterministic function of the random variable X_s, as is the case with f_s^μ. For instance, it can be selected as the function $K_s(\omega) = h(s, X_s(\omega), \omega)$ for some appropriate function h, producing a non-trivial distribution for the joint process (X_s, K_s). Further, K_s need not satisfy a Markovian property, as long as the smoothness properties in [1, Chapter 1, Theorem 6.16] hold. This insight will be explored more in the next chapter.

One of the key insights behind this application of Girsanov's theorem is the fact that it is applied simultaneously to both the forward and backward SDE in the FBSDE system. Although the application has been described as an *importance sampling* technique (e.g., [16]), intuitively, this description is misleading. First of all, no likelihood weights are involved in LSMC because we have cast the whole problem into another measure. We can draw identically distributed samples from P because W_s^{P} is Brownian in this measure, and we can do LSMC expectations on the backward process because the integration is over W_s^{P}. Secondly, the change of measure is not being performed to reduce variance, but rather to change the distribution over which function regressions are being performed. The goal of this change is to approximate the value function V^μ along trajectories not necessarily governed by the policy μ.

3.4 Weighted-Drifted LSMC

Once we have produced a forward SDE distribution P we may want to further concentrate approximation accuracy in the backward pass using weighted regression. For example, during the backward pass we can form heuristics from approximate values for the cost-to-go and the cost-to-come for sampled states. These heuristics can then be converted into weights for the regression.

Consider the short time horizon characterization of the FBSDEs, noting that X_s, Y_s, Z_s, K_s, W_s^{P}, and $\widehat{Y}_{t,\tau}$ are $\mathcal{F}_\tau$-measurable for $s, t \in [0, \tau]$. These variables are fully characterized by the probability measure P_τ, the restriction of P to the sigma-algebra $\mathcal{F}_\tau$. For any weighting variable (Radon-Nikodym derivative) $\Theta_\tau^{\mathsf{R}|\mathsf{P}}$ which is P_τ-a.s. strictly positive, we can define the equivalent measure R_τ as

$$\mathrm{d}\mathsf{R}_\tau = \Theta_\tau^{\mathsf{R}|\mathsf{P}} \mathrm{d}\mathsf{P}_\tau, \tag{3.25}$$

[60, Chapter 10, Remark 10.4]. In the context of LSMC for FBSDE methods we offer the following theorem.

Theorem 3.9. *Assume $\Theta_\tau^{\mathsf{R}|\mathsf{P}}$ is selected such that W_s^{P} is Brownian on the interval $[t, \tau]$ with respect to the induced measure R_τ. It holds that*

$$Y_t = \mathbf{E}_{\mathsf{R}_\tau}[\widehat{Y}_{t,\tau} | X_t] = V^\mu(t, X_t), \qquad \mathsf{R}_\tau\text{-a.s.}. \tag{3.26}$$

Furthermore, the minimizer ϕ^* of the optimization problem

$$\inf_{\phi \in L^2} \mathbf{E}_{\mathsf{R}_\tau}[(\widehat{Y}_{t,\tau} - \phi)^2] = \inf_{\phi \in L^2} \mathbf{E}_{\mathsf{P}_\tau}[\Theta_\tau^{\mathsf{R}|\mathsf{P}} (\widehat{Y}_{t,\tau} - \phi)^2], \tag{3.27}$$

over X_t-measurable square integrable variables ϕ coincides with the value function $\phi^* = V^\mu(t, X_t)$ R_τ-a.s.. □

Proof. The proof of Theorem 3.6 shows that since P_τ and R_τ are equivalent, (3.17) holds P_τ-a.s. iff it holds R_τ-a.s.. Since W_s^{P} is Brownian in R_τ over the interval, we have $\mathbf{E}_{\mathsf{R}_\tau}[\int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{P}}|X_t] = 0$. The rest of the proof of (3.26) follows similarly to the proof of Corollary 3.2.

Equation (3.27) follows similarly to the proof of Theorem 3.5, followed by a change of measure (3.25). □

Continuing the discussion of approximate LSMC in Section 3.2, let $\{(x_t^k, \widehat{y}_t^k, \theta_\tau^k)\}_{k=1}^M$ be a set of point-samples approximating the joint distribution $(X_t, \widehat{Y}_{t,\tau}, \Theta_\tau^{\mathsf{R}|\mathsf{P}})$, denoted as $\widetilde{\mathsf{P}}$. For drifted, weighted LSMC we instead use the approximate optimization

$$\begin{aligned}\arg\min_{\alpha\in\mathcal{A}} \mathbf{E}_{\mathsf{P}}[\Theta_\tau^{\mathsf{R}|\mathsf{P}}(\widehat{Y}_{t,\tau} - \phi(X_t;\alpha))^2] &\approx \arg\min_{\alpha\in\mathcal{A}} \mathbf{E}_{\widetilde{\mathsf{P}}}[\Theta_\tau^{\mathsf{R}|\mathsf{P}}(\widehat{Y}_{t,\tau} - \phi(X_t;\alpha))^2] \\ &= \arg\min_{\alpha\in\mathcal{A}} \sum_{k=1}^M \frac{\theta_\tau^k}{M}(\widehat{y}_t^k - \phi(x_t^k;\alpha))^2 =: \alpha_t^*. \quad (3.28)\end{aligned}$$

3.5 Policy Improvement

For the purposes of finding an approximate solution to the (C-V^*) problem we are not only interested in the on-policy value function V^μ, but also how its solution can be used to improve the policy. Supposing we have a policy μ and its corresponding on-policy value function V^μ, we now discuss how we may produce a feedback policy π which is more optimal and thus a closer approximation of the optimal policy π^*.

Proposition 3.10 (Policy Comparison Principle)**.** *Take Assumption* (A2) *and Assumption* (A3) *for feedback policies μ and π.*[1] *If the inequality*

$$(\partial_x V^\mu)^\top f^\pi + \ell^\pi \leq (\partial_x V^\mu)^\top f^\mu + \ell^\mu, \quad (3.29)$$

is satisfied on on $[0,T]\times\mathbb{R}^n$ then $V^\pi \leq V^\mu$ is as well.

[1] We mainly need continuity and boundedness of the derivatives $\partial_t \ell^\mu$ and $\partial_x \ell^\mu$ and boundedness of the solutions.

Proof. By the assumptions we have classical, bounded solutions for both (C-HJ) PDEs corresponding to μ and π. Substituting the inequality into the PDE corresponding to μ yields

$$\begin{aligned} 0 = &\quad \partial_t V^\mu + \frac{1}{2}\operatorname{tr}[\sigma\sigma^\top \partial_{xx} V^\mu] + (\partial_x V^\mu)^\top f^\mu + \ell^\mu \\ \geq &\quad \partial_t V^\mu + \frac{1}{2}\operatorname{tr}[\sigma\sigma^\top \partial_{xx} V^\mu] + (\partial_x V^\mu)^\top f^\pi + \ell^\pi, \end{aligned}$$

confirming that V^μ is a supersolution of the (C-HJ) PDE corresponding to π. By [6, Chapter 5, Theorem 9.1] and the fact that $V^\pi(T,\cdot) \equiv V^\mu(T,\cdot)$, we have $V^\pi \leq V^\mu$. □

If we take

$$\pi(t,x) \in \arg\min_{u\in\mathcal{U}}\{(\partial_x V^\mu(t,x))^\top f(t,x,u) + \ell(t,x,u)\}, \tag{3.30}$$

to be this policy we satisfy (3.29) immediately. Further, if $V^\mu \equiv V^*$, we obtain an optimal policy π^*.

3.6 Revised SOC Problem

Now that we have discussed the Feynman-Kac FBSDE approach, we revise the C-SOC problem proposed at the end of Section 2.2. Define P^* as the probability measure over the on-policy FBSDE solution (X_s, Y_s, Z_s) associated with the target policy $\mu = \pi^*$ for some optimal policy π^*.

FBSDE SOC Problem

Let $\mathcal{T} := (t_0 = 0, t_1, \ldots, t_{N-1}, t_N = T)$ be a partition of the interval $[0,T]$. The FBSDE SOC problem is, for each $t_i \in \mathcal{T}$, to determine or approximate the optimal value function $V^*(t_i,\cdot)$ (C-V^*) and the optimal feedback policy $\pi^*(t_i,\cdot)$ in the weighted distribution $\mathsf{R}_{\tau_i, X_{t_i}}$ for appropriately selected measures R_{τ_i} for each t_i. An ▶

FBSDE SOC Problem (cont)

appropriate measure R_{τ_i} will be close to $\mathsf{P}^*_{\tau_i}$, but not necessarily coincide.

There are two primary changes to the original problem, which sought to solve the value function and policy on all $[0, T] \times \mathbb{R}^n$. First, we only seek to solve for the value function at a finite set of discrete times t_i. As formulated, the LSMC method performs a regression over a distribution at particular time based on short time interval estimators. The question of how $V^*(s, \cdot)$ and $\pi^*(s, \cdot)$ are approximated over the open interval $s \in (t_i, t_{i+1})$ is left to other methods such as interpolation.[2]

The second change is the vaguely worded, by intention, qualification that these functions be determined in an appropriate distribution $\mathsf{R}_{\tau_i, X_{t_i}}$. Numerical Feynman-Kac FBSDE methods are primarily useful for SOC problems because they concentrate approximation of the optimal value function and policy in a local region around the distribution of optimal trajectories. The optimal trajectory distribution is unlikely to be the optimal distribution for approximating the value function and policy for two reasons. First, especially if the diffusion in the system is small, the optimal distribution will have near-degenerate density, that is, all trajectories will clump together in some dimension. In this case, function regression is likely to be ill-posed, resulting in instability in the determination of function parameters due to a singular Gram matrix. Second, optimizing over the average $\mathsf{P}^*_{X_t}$ will concentrate all approximation near the mean, resulting in poor sensitivity in the tails of the distribution. Better approximation near the mean has diminishing returns when the function is smooth, but the cost of having one trajectory diverge due to poor approximation of the value function on the tails is very high.

The idea that the distribution over which the optimal value function is approximated should be included in the fundamental problem of optimal control is crucial for modern

[2]The theory presented might be expanded to allow any time $t \in [0, T]$, and perform one large LSMC over the full interval instead of a series of LSMC optimizations at each $t_i \in \mathcal{T}$. We have not investigated such generalizations and choose not to include them for ease of presentation.

methods. In fact, the idea that broader exploration produces more robust value function approximation is the key concept in "soft"-reinforcement learning (RL) literature [45].

3.7 Chapter Summary and Contributions

In this section we discussed the continuous-time methodology for solving drifted Feynman-Kac FBSDEs with weighted LSMC. We discussed the various assumptions about the dynamics and costs, what smoothness guarantees they offer, and how we can use the on-policy value function as the iteratively-improved approximation of the value function. We introduced the on-policy FBSDEs and showed how the LSMC method can be used to approximate the value function. We showed how the value function estimate can be used to improve the policy in an iterative method. The weighted-drifted FBSDE framework allows us to arbitrarily choose the distribution over which to solve the value function. Finally, we reframed the generalized SOC problem in the context of Feynman-Kac FBSDEs.

In this chapter we introduced three measures: (a) Q, the measure associated with the target policy μ for the value function V^{μ}, (b) P, the sampling measure used in the forward pass to explore the state space, and (c) R_{τ}, the weighted measure used in the backward pass to control function approximation accuracy. In the following chapters we will further develop how we choose these measures and how we approximate them with numerical methods.

This presentation of Feynman-Kac FBSDEs for SOC is novel in the following ways:

- The design choice to solve for a general on-policy value function V^{μ} and not just the optimal value function V^{*}.
- The characterization of Feynman-Kac FBSDEs as on-policy vs. off-policy.
- The analysis of variance of the value function estimators $\widehat{Y}_{t,\tau}$.
- The integration of weighted LSMC with drifted FBSDEs to further change the measure.

In previous methods featuring Girsanov-drifted FBSDEs, authors directly apply the Feynman-Kac theorem to the HJB equation and then add a drift term to reduce variance [23, 16]. The problem with this presentation is first, that it requires knowledge of the gradient of the value function to compute the optimization over control appearing in (C-HJB) to perform the backward integration, before the value function is known. Our presentation does not require such an assumption, recognizing that whatever policy μ we are currently targeting is always based on an approximation and does not require any guarantee of being optimal. Further, the presentation allows us to distinguish between on-policy methods which are associated with high accuracy since Z_t may not need to be computed, and off-policy methods, which allow us to choose a different drift, e.g., the drift associated with a different policy. For example, we may, as a part of a numerical method, want to determine the value function associated with the current policy, as well as a different policy.

Although the reduction of $\widehat{Y}_{t,\tau}^{\text{noisy}}$ to $\widehat{Y}_{t,\tau}^{\text{noiseless}}$ was presented in [16], the variance analysis presented here is novel. Weighted LSMC has been represented elsewhere (e.g. [61]), but has not been applied to drifted FBSDEs.

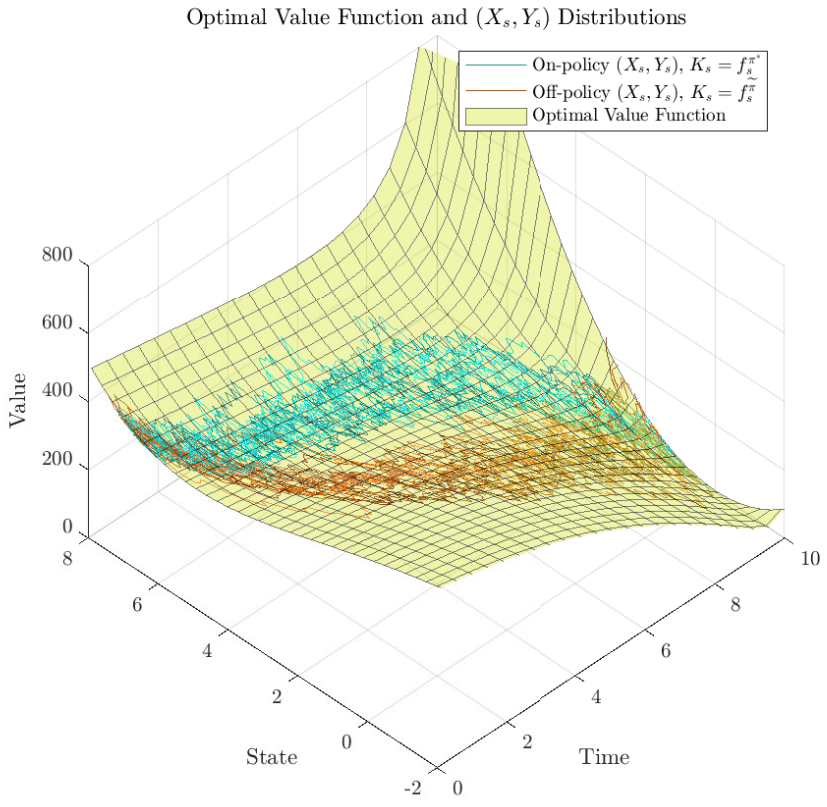


Figure 3.4: Illustrating the off-policy Feynman-Kac-Girsanov representation theorem. The two distributions (X_s, Y_s) solve the drifted FBSDE system (3.15) and (3.16) starting at $X_0 = x_0 = 7$, where the target policy is the optimal policy $\mu = \pi^*$ and the drift is either $K_s = f_s^{\pi^*}$ (cyan trajectories) or $K_s = f_s^{\tilde{\pi}}$ (orange trajectories). The Feynman-Kac-Girsanov theorem indicates that each distribution will P-a.s. lie on the surface of the on-policy value function V^{μ}. In this example, since the target policy is incidentally the optimal policy, the on-policy value function is the optimal $V^{\mu} = V^{\pi^*} = V^*$. Further, when the drift is $K_s = f_s^{\pi^*}$, then $D_s \equiv 0$ and the off-policy FBSDE system becomes equivalent to the on-policy system.

CHAPTER 4

IMPROVING FBSDE ESTIMATORS WITH DISCRETE-TIME ANALYSIS

In Chapter 3 we showed how LSMC methods ultimately resolve to a regression problem, determining an approximation of the on-policy value function $\widetilde{V}^{\mu}(t,\cdot) \approx V^{\mu}(t,\cdot)$ from an estimator $\widehat{Y}_{t,\tau}$ given the known relationship $\mathbf{E}_{\mathsf{P}}[\widehat{Y}_{t,\tau}|X_t] = V^{\mu}(t,X_t)$.[1] For the drifted forward SDE over the interval $[t,\tau]$,

$$\mathrm{d}X_s = K_s\,\mathrm{d}s + \sigma_s\,\mathrm{d}W_s^{\mathsf{P}},$$

this estimator is characterized as one of two backwards integrated SDEs:

$$\widehat{Y}_{t,\tau}^{\text{noisy}} := Y_\tau + \int_t^\tau (\ell_s^\mu + Z_s^\top D_s)\mathrm{d}s - \int_t^\tau Z_s^\top \mathrm{d}W_s^{\mathsf{P}},$$

or

$$\widehat{Y}_{t,\tau}^{\text{noiseless}} := Y_\tau + \int_t^\tau (\ell_s^\mu + Z_s^\top D_s)\mathrm{d}s.$$

Although, in theory, we can use a simplistic approximation of these SDEs over short intervals, such as Euler-Maruyama [62, Section 10.2], arguing that a sufficiently small $\Delta t = \tau - t$ will provide a decent approximation, over this chapter we will demonstrate that careful treatment of the estimator can significantly improve accuracy of the full FBSDE method. Since the backward pass of FBSDE methods integrates error in each backward step, small errors accumulate quickly and cannot be recovered from. We propose novel estimators with guarantees, offered from error analysis and confirmed in numerical simulation. In fact, we will show that on linear-quadratic-regulator (LQR) problems our proposed

[1] In this chapter we ignore the weighted measure R_τ. The results will naturally generalize.

estimators are nearly optimal, while previously proposed methods diverge in the backward pass.

Henceforth, following the revised SOC problem in Section 3.6 we adjust the notation over short intervals from $[t, \tau]$ to $[t_i, t_{i+1}]$, where

$$t_i, t_{i+1} \in \mathcal{T} := (t_0 = 0, t_1 = \Delta t, t_2 = 2\Delta t, \ldots, t_{N-1} = T - \Delta t, t_N = T),$$

a partition of the interval $[0, T]$ with constant step size Δt. Further, for variables we denote $X_i := X_{t_i}$, and for functions $V^\mu(t, x)$ denote $V_i^\mu(x) := V^\mu(t_i, x)$, and similarly for other variables and functions, for brevity.

4.1 Euler-Maruyama FBSDE Approximation

Many approaches to solving the FBSDEs propose approximating both the forward and backward steps with Euler-Maruyama-like SDE approximations (see, for instance, [23], [16], and the survey in [63]). For the drifted FSDE the approximation is

$$X_{i+1} - X_i = K_i \, \Delta t + \sqrt{\Delta t} \, \sigma_i \, \Delta W_i^{\mathsf{P}}, \tag{4.1}$$

where ΔW_i^{P} is an n-dimensional normal random variable $\Delta W_i^{\mathsf{P}} \sim \mathcal{N}(0, I_n)$. For the drifted BSDE step we have

$$\widehat{Y}_i = Y_{i+1} - \Delta \widehat{Y}_i, \tag{4.2}$$

where $\Delta \widehat{Y}_i$ is either

$$\Delta \widehat{Y}_i^{\text{noisy}} = -(\ell_i^\mu + Z_{i+1}^\top D_i) \, \Delta t + Z_{i+1}^\top \sqrt{\Delta t} \, \Delta W_i, \tag{4.3}$$

or

$$\Delta \widehat{Y}_i^{\text{noiseless}} = -(\ell_i^\mu + Z_{i+1}^\top D_i)\, \Delta t. \tag{4.4}$$

The variable Z_{i+1} is evaluated at the end of the interval so that it can utilize the latest approximation of the value function gradient. The primary contribution of this chapter is to propose new estimators for $\widehat{Y}_i$ to be used in the LSMC function regression step.

4.2 Motivation of the Proposed Approach

In Chapter 3 we presented results from continuous-time FBSDE theory, then, in the previous section, used standard methods in SDE approximation to form a discrete-time approximation of the forward and backward SDEs. In the remainder of this chapter we propose the converse approach: we begin by forming a discrete-time approximation of the dynamics and the value function, then we derive relationships which resemble those arrived at previously. In doing so, we make two contributions: first, we arrive at better estimators compared to the direct discretization of the continuous time relations because we are able to exploit characteristics of the discrete-time formulation obscured by the continuous-time problem, and, secondly, we provide a discrete-time intuition for the continuous-time theory by using familiar theorems in its derivation.

4.2.1 Insights from Continuous-Time FBSDE Theory

Before detailing our approach, we briefly analyze the mechanisms of how continuous-time FBSDE theory arrives at its result, that is, the question of how the continuous-time theory can inform a discrete-time theory. The first insight is that the local smoothness of value functions can be leveraged in backward integration, using the derivatives of V_{i+1}^μ to estimate V_i^μ. Consider that the proof of the Feynman-Kac representation theorem (Theorem 3.1) utilizes Itô's formula, a classic result in stochastic control theory which represents

the extrapolation of smooth functions of stochastic processes [1, p. 378]. For the on-policy dynamics $\mathrm{d}X_s = f_s^\mu \,\mathrm{d}s + \sigma_s \mathrm{d}W_s^{\mathsf{Q}}$ on the interval $[t_i, t_{i+1}]$, Itô's formula yields

$$\begin{aligned} &V^\mu(t_{i+1}, X_{i+1}) \\ &= V^\mu(t_i, X_i) + \int_{t_i}^{t_{i+1}} \partial_t V^\mu(s, X_s) + (\partial_x V^\mu(s, X_s))^\top f_s^\mu + \frac{1}{2}\operatorname{tr}(\sigma_s \sigma_s^\top \partial_{xx} V^\mu(s, X_s))\,\mathrm{d}s \\ &\qquad + \int_{t_i}^{t_{i+1}} (\partial_x V^\mu(s, X_s))^\top \sigma_s \,\mathrm{d}W_s^{\mathsf{Q}}, \end{aligned} \tag{4.5}$$

where tr is the trace operator [1, Chapter 1, Theorem 5.5]. FBSDE methods are founded on the intrinsic relationship between $Y_i = V^\mu(t_i, X_i)$, the random variable we are trying to estimate and $V^\mu(t_{i+1}, X_{i+1})$, but also on the gradient $\partial_x V^\mu$ and the Hessian $\partial_{xx} V^\mu$. In the Feynman-Kac theorem, the (C-HJ) equation is substituted in for the $\partial_t V^\mu$ term, conveniently cancelling out the other two terms in the first integral and yielding the on-policy BSDE. However, in a discrete-time approximation, such cancellations may be inappropriate since the integral will be replaced with a zero-order hold. Instead of Itô's formula, we can rely on Taylor's theorem, its deterministic counterpart, to include higher order derivatives in the estimator.

The second insight comes from the continuity of the FBSDE processes. When we find an approximation of the value function $V^\mu(t_{i+1}, \cdot)$, the accuracy of the approximation will be generally concentrated about some distribution $\mathsf{P}_{X_{i+1}}$ (or $\mathsf{R}_{X_{i+1}}$ in the weighted case). When we compute the approximation of the value function $V^\mu(t_i, \cdot)$ at the previous time step using backward integration, we will rely on the fact that the distribution $\mathsf{P}_{\overline{X}_{i+1}}$ over which $V^\mu(t_{i+1}, \cdot)$ will be evaluated will be close to, or covered by, $\mathsf{P}_{X_{i+1}}$. This ensures that the backward integration accumulates little error from extrapolation in the backward pass. Further, since each backward integration is performed pathwise, the error in approximation should be low because the Y_s process will be continuous over the interval. That is, $Y_{i+1}(\omega) - Y_i(\omega)$ will always be small for sufficiently small intervals, so each estimate $Y_i(\omega)$ is unlikely to introduce significant error, even if the Y_i's distribution has much higher

variance. The continuity of distributions and paths continues backwards in time, eventually collapsing to a deterministic point mass starting state, concentrating approximation accuracy on trajectories reachable from this state. The continuity of paths and distributions is further examined in Chapter 5.

The third insight comes from the drifted formulation, the idea that we can use a trajectory distribution which is not strictly related to the policy associated with the value function. Said another way, we can use a change of measure to produce off-policy estimators with better accuracy than biased off-policy estimators which do not compensate for using a drift different from the policy. Girsanov's theorem shows that we can change SDEs by substituting in the relationship $\mathrm{d}W_s^{\mathsf{Q}} = \mathrm{d}W_s^{\mathsf{P}} - D_s \mathrm{d}s$, where W_s^{Q} is Brownian in Q and W_s^{P} is Brownian in P. By changing both the forward and backward SDEs, we can change the measure (and particularly the forward SDE distribution) without introducing weights in the LSMC regression step. This change of measure is necessitated by the fact that we would like to improve policies after we sample forward distributions over X_s, and it is very inefficient to resample every time the policy changes in order to be in compliance with the on-policy formulation.

4.2.2 Discrete-Time FBSDE Simplified Example

In this section we motivate the more fully developed methods of later sections with a greatly simplified example that illustrates the approach taken. Define a 1-dimensional C-SOC problem with $f \equiv \ell \equiv 0$, $\sigma = \Delta t = 1$, and $W_i^{\mathsf{Q}} := W_{t_{i+1}}^{\mathsf{Q}} - W_{t_i}^{\mathsf{Q}} \sim \mathcal{N}(0, 1)$. The short interval on-policy FBSDE relationships for this system are

$$X_{i+1} = X_i + W_i^{\mathsf{Q}},$$
$$Y_{i+1} = Y_i + W_i^{\mathsf{Q}}.$$

Recalling that $Y_i = V_i^\mu(X_i)$, we take the expectation of both sides of the BSDE to arrive at a discrete on-policy Bellman equation,

$$V_i^\mu(X_i) = \mathbf{E}_\mathsf{Q}[V_{i+1}^\mu(X_{i+1})|X_i].$$

Upon substitution of X_{i+1} into this equation, we can perform a second-order Taylor expansion around X_i,

$$\begin{aligned} V_i^\mu(X_i) &= \mathbf{E}_\mathsf{Q}[V_{i+1}^\mu(X_i + W_i^\mathsf{Q})|X_i] \\ &\approx \mathbf{E}_\mathsf{Q}[V_{i+1}^\mu(X_i) + \partial_x V_{i+1}^\mu(X_i) W_i^\mathsf{Q} + \frac{1}{2}\partial_{xx} V_{i+1}^\mu(X_i)(W_i^\mathsf{Q})^2|X_i] \\ &= V_{i+1}^\mu(X_i) + \frac{1}{2}\partial_{xx} V_{i+1}^\mu(X_i). \end{aligned}$$

Thus, we have arrived at the estimator

$$\widehat{Y}_i = V_{i+1}^\mu(X_i) + \frac{1}{2}\partial_{xx} V_{i+1}^\mu(X_i),$$

for $Y_i = V_i^\mu(X_i)$. The estimator's bias depends on higher order terms in the Taylor expansion (when they exist) and its variance is $\mathrm{Var}_\mathsf{Q}[\widehat{Y}_i|X_i] = 0$. In fact, in this on-policy example, since odd multiples of normal random variables have an expectation $\mathbf{E}_\mathsf{Q}[(W_i^\mathsf{Q})^{2j+1}|X_i] = 0$, $j = 0, 1, \ldots$, only even-ordered derivatives affect the bias (so our estimator's bias is actually accurate up to the third order Taylor expansion).

In Figure 4.1 we illustrate how our on-policy Taylor-expansion estimator performs on a given value function V_{i+1}^μ. The estimator is fairly accurate and does not require a convolution to evaluate.

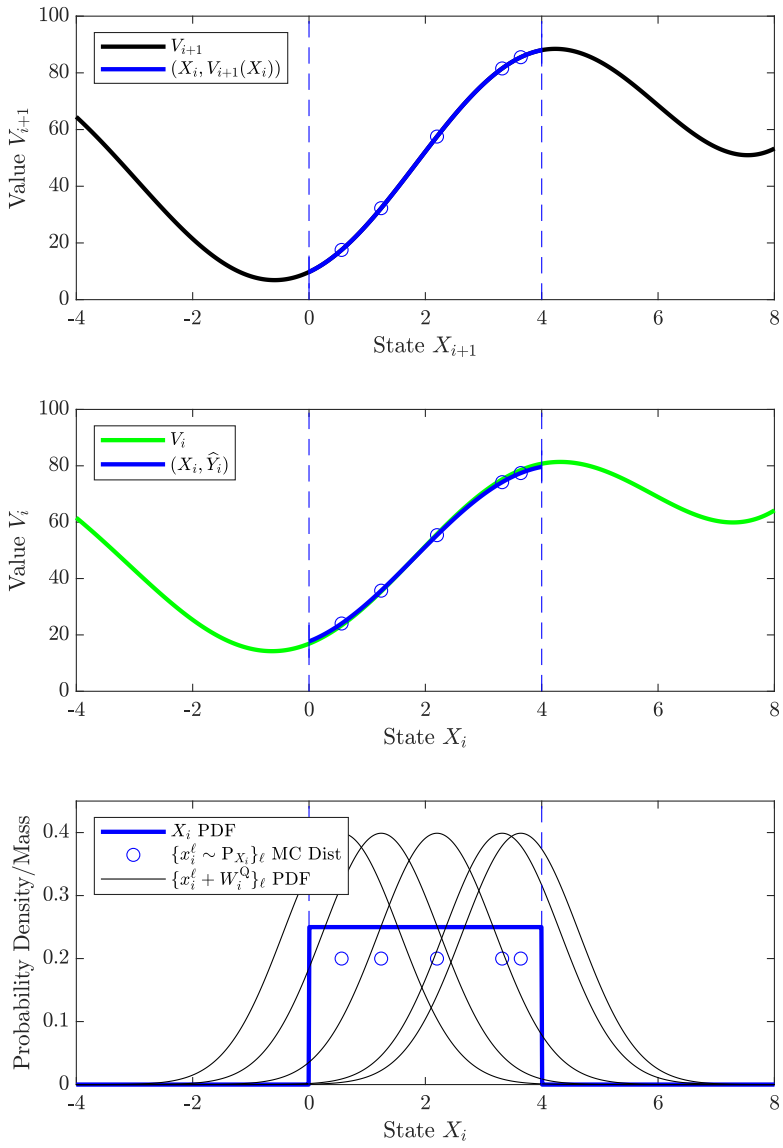


Figure 4.1: Illustrating the on-policy 1-step discrete-time example. We assume that we are given the value function V_{i+1}^μ (black curve in top figure) and probability density $q_{X_i}(x_i) = \mathrm{d}\mathsf{Q}_{X_i}/\mathrm{d}x_i$ (blue curve in bottom figure). From V_{i+1}^μ we can compute the ground truth value of V_i^μ (green curve in middle figure) via the convolution $V_i^\mu(x) = \mathbf{E}_{\mathsf{Q}}[V_{i+1}^\mu(X_i + W_i^{\mathsf{Q}})|X_i = x] = \int_{-\infty}^{\infty} V_{i+1}^\mu(x+w)(2\pi)^{-1/2}\exp(-1/2w^2)\mathrm{d}w$. The blue section of the curve in the top figure, along with its second derivative, is used to compute the estimator variable $\widehat{Y}_i$ in the middle figure (also in blue). In LSMC methods distributions are represented by Monte Carlo samples, illustrated by the circle markers. Each of the markers in the middle figure is an approximation of the expected value of the function in the top figure over the respective PDF in the bottom figure.

Suppose in Figure 4.1 the value function V_{i+1}^μ is not available on the interval $[0, 4]$, but instead a shifted interval $[k, 4+k]$ for some $k \in \mathbb{R}$. We can produce an estimator again

using the Taylor-expansion as

$$\begin{aligned}
V_i^{\mu}(X_i) &= \mathbf{E}_{\mathsf{Q}}[V_{i+1}^{\mu}(X_{i+1})|X_i] \\
&= \mathbf{E}_{\mathsf{Q}}[V_{i+1}^{\mu}(X_i + W_i^{\mathsf{Q}})|X_i] \\
&= \mathbf{E}_{\mathsf{Q}}[V_{i+1}^{\mu}(X_i + k + (W_i^{\mathsf{Q}} - k))|X_i] \\
&\approx \mathbf{E}_{\mathsf{Q}}[V_{i+1}^{\mu}(X_i + k) + \partial_x V_{i+1}^{\mu}(X_i + k)(W_i^{\mathsf{Q}} - k) \\
&\qquad\qquad + \frac{1}{2}\partial_{xx} V_{i+1}^{\mu}(X_i + k)(W_i^{\mathsf{Q}} - k)^2|X_i] \\
&= V_{i+1}^{\mu}(X_i + k) - \partial_x V_{i+1}^{\mu}(X_i + k)\, k + \frac{1}{2}\partial_{xx} V_{i+1}^{\mu}(X_i + k)(1 + k^2).
\end{aligned}$$

Notice that the value function and its derivatives are only evaluated on the distribution $X_i + k$, relying on Taylor expansion extrapolation to compute each $V_i^{\mu}(x_i)$. We call this formulation drifted off-policy because it is associated with a new forward difference equation

$$X_{i+1} = X_i + k + W_i^{\mathsf{P}},$$

where $W_i^{\mathsf{P}} = W_i^{\mathsf{Q}} - k$ is normal in another measure. Consider the probability density functions (PDFs) for these variables, where W_i^{P} is normally distributed in P and W_i^{Q} is normally distributed in Q,

$$\begin{aligned}
\mathrm{d}\mathsf{Q}_{W_i^{\mathsf{Q}}} &= (2\pi)^{-1/2} \exp(-\frac{1}{2}(w_i^{\mathsf{Q}})^2)\, \mathrm{d}w_i^{\mathsf{Q}} \\
&= (2\pi)^{-1/2} \exp(-\frac{1}{2}(w_i^{\mathsf{P}} + k)^2)\, \mathrm{d}w_i^{\mathsf{P}} \\
&= \exp(-\frac{1}{2}k^2 + (-k)w_i^{\mathsf{P}})(2\pi)^{-1/2} \exp(-\frac{1}{2}(w_i^{\mathsf{P}})^2)\, \mathrm{d}w_i^{\mathsf{P}} \\
&= \exp(-\frac{1}{2}k^2 + (-k)w_i^{\mathsf{P}})\mathrm{d}\mathsf{P}_{W_i^{\mathsf{P}}}.
\end{aligned}$$

It is a property of log-normal distributions that $\mathbf{E}_{\mathsf{P}}[\exp(-\frac{1}{2}k^2 + (-k)W_i^{\mathsf{P}})] = 1$, for any $k \in \mathbb{R}$, when W_i^{P} is normally distributed [64], confirming that P is a probability measure

iff Q is one. We also see that we have arrived at a weighing variable which coincides with (3.19), used in Girsanov's theorem, for $D_s \equiv -k$.

Figure 4.2 illustrates how the estimator performs for different values of k. For smaller values of k, the bias is not unreasonable, but eventually the bias grows larger. The magnitude of this bias is related to how well a local second order Taylor expansion might represent the value function V_{i+1}^{μ}. When the value function is a quadratic polynomial this representation is exact for any distribution over X_i and any drift k.

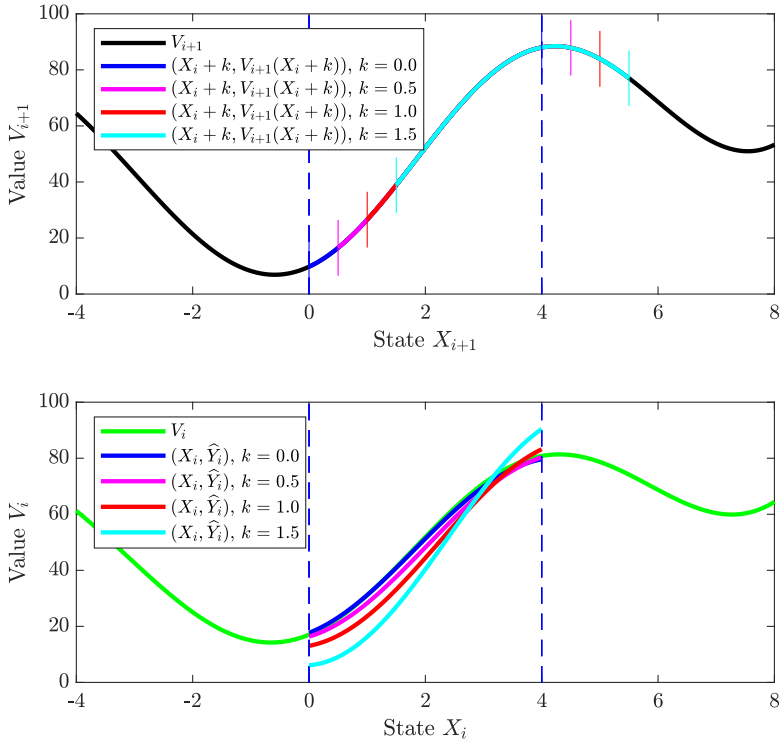


Figure 4.2: Illustrating the off-policy 1-step discrete-time example. The shifted uniform intervals of $X_i + k$, for $k = 0, 0.5, 1, 1.5$ are used to query the value function V_{i+1}^{μ} as seen in the top figure. The resulting estimators $\widehat{Y}_i$ approximating $V_i(X_i)$, for different values of k, are visualized in the bottom figure.

It is important to recall that introducing drift is not desirable if V_{i+1}^{μ} is accurately known everywhere. The purpose of introducing drift is to align the distribution over which V_{i+1}^{μ} is queried for computing the estimator, with the distribution over which V_{i+1}^{μ} was approximated in the previous LSMC backward step. Extrapolation far from the edges of the distribution used to approximate the function is often undesirable. For example, in the case

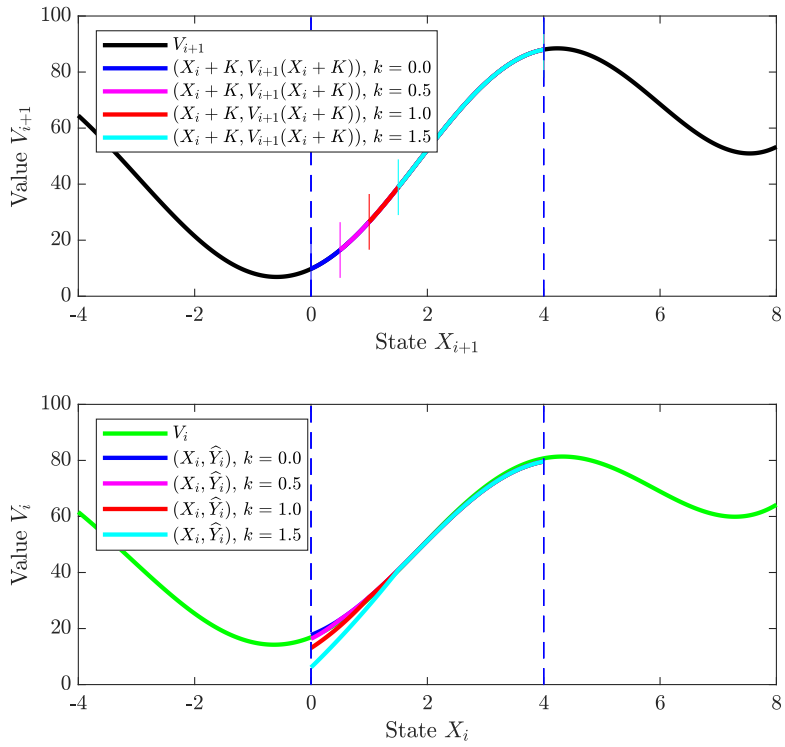


Figure 4.3: The off-policy 1-step discrete-time example with random drift K. For this example, we consider the scenario where V^{μ}_{i+1} is only known on the distribution $X_i + k$, for $k = 0, 0.5, 1, 1.5$, as illustrated in the top plot in Figure 4.2. Differently from that example, we choose the drift term to be $K = \begin{cases} k - X_i & \text{if } X_i < k \\ 0 & \text{o.w.} \end{cases}$. In the top figure we illustrate only the segments of the value function used to compute the estimators in the bottom figure. By reducing the average magnitude of K, the accuracy of the estimators improves.

of high-degree polynomial regression, Runge's phenomenon sometimes arises, resulting in large oscillations near the edges of the distribution. Instead of extrapolation, we propose using a localized low-order Taylor expansion to arrive at values we do not have direct access to. Such an approach will tame the high-frequency elements and thus reduce the impact of problems associated with extrapolation.

This method has several avenues of generalization away from this simplified presentation, the first being that the drift need not be a constant term k, but instead can be a random variable K. Figure 4.3 illustrates how this generalization might be used, for example, to reduce error in the approximation. Although in this example K is a deterministic function of X_i, it can also have non-trivial variance as long as it is independent of W^{P}_i.

If we are willing to introduce more variance in the estimator, we are able to produce an estimator with less bias. If we directly compute the backward difference

$$\begin{aligned} Y_{i+1} - Y_i &= V_{i+1}^{\mu}(X_{i+1}) - V_i^{\mu}(X_i) \\ &= V_{i+1}^{\mu}(X_{i+1}) - \mathbf{E}_{\mathsf{Q}}[V_{i+1}^{\mu}(X_{i+1})|X_i], \end{aligned}$$

then apply the drifted Taylor expansion simultaneously to both $V_{i+1}^{\mu}(X_{i+1})$ terms, we get

$$\begin{aligned} &\approx \left(V_{i+1}^{\mu}(X_i + k) + \partial_x V_{i+1}^{\mu}(X_i + k)(W_i^{\mathsf{Q}} - k) + \frac{1}{2}\partial_{xx} V_{i+1}^{\mu}(X_i + k)(W_i^{\mathsf{Q}} - k)^2\right) \\ &\quad - \left(V_{i+1}^{\mu}(X_i + k) - \partial_x V_{i+1}^{\mu}(X_i + k)\, k + \frac{1}{2}\partial_{xx} V_{i+1}^{\mu}(X_i + k)(1 + k^2)\right) \\ &= \partial_x V_{i+1}^{\mu}(X_i + k)(W_i^{\mathsf{P}} + k) + \frac{1}{2}\partial_{xx} V_{i+1}^{\mu}(X_i + k)((W_i^{\mathsf{P}})^2 - 1 - k^2). \end{aligned}$$

The estimator for the higher variance, lower bias Y_i estimator is thus,

$$V_{i+1}^{\mu}(X_{i+1}) - \partial_x V_{i+1}^{\mu}(X_i + k)(W_i^{\mathsf{P}} + k) - \frac{1}{2}\partial_{xx} V_{i+1}^{\mu}(X_i + k)((W_i^{\mathsf{P}})^2 - 1 - k^2),$$

where $X_{i+1} = X_i + k + W_i^{\mathsf{P}}$ so as to keep the variables in the measure P. We refer to this estimator as the *noisy* formulation because it now has non-trivial variance. The reason this formulation has lower bias is because the higher order terms cancel out, to some degree, in the difference between the Taylor expansions outside and under the conditional expectation. In fact, the bias is zero in the on-policy case where $k = 0$. Naturally, however, since the normal variable W_i^P is involved in the computation, variance is introduced in the estimator.

The performance of the noisy estimator over the previous estimator (denoted *noiseless*) is visualized in Figure 4.4. For high numbers of samples the approximation can improve, but it cannot overcome bias introduced from the drift. The highest relative gains of the noisy over the noiseless come from the on-policy estimator, when the bias is zero. This effect suggests the respective roles for these estimators in numerical methods. When the policy is far from optimal and the drift is large, noiseless estimators provide an estimate

which is roughly on-par with the quality of an asymptotically sampled noisy estimator, with the convenient property of being zero variance, so the noiseless estimator should be used. When the policy is near optimal and drift is low or zero, noisy estimators can be utilized to refine accuracy.

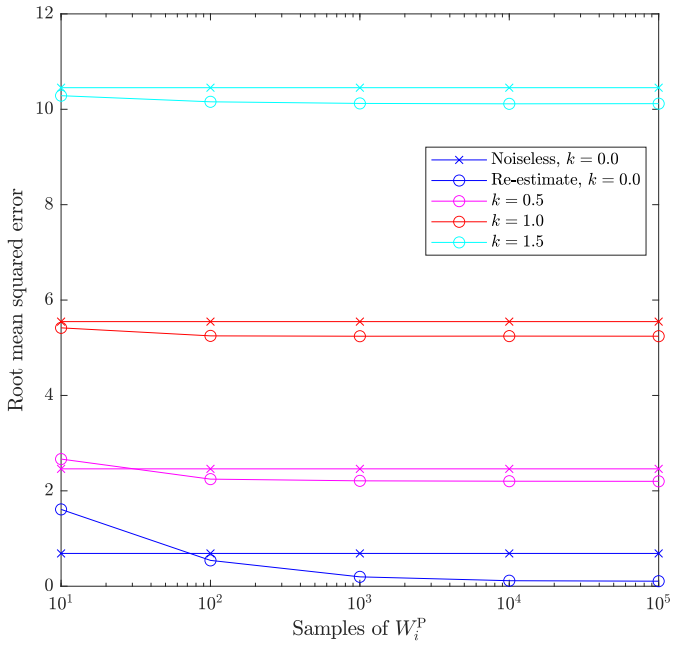


Figure 4.4: Charting performance of the noisy estimator applied to the problem visualized in Figure 4.2. The noisy estimator is computed by averaging over a number of samples of W_i^{P}, as indicated in the x-axis, for each given value in the uniform distribution over X_i. We then form the root mean squared error (RMSE) statistic $(\mathbf{E}_{\mathsf{P}}[(\widehat{Y}_i - Y_i)^2])^{1/2}$, which averages over the uniform distribution of X_i. These statistics are compared to the previous noiseless estimator for differing values of k and numbers of samples. The noiseless estimator does not vary with the number of samples because it has zero variance.

The remainder of this chapter is devoted to generalizing the approach introduced here to n-dimensional discrete-time FBSDEs with non-trivial, non-linear dynamics and costs. We also compare this approach to the Euler-Maruyama-type approaches discussed in the previous section, demonstrating how they reflect an improved approximation of the same approach. The theorems utilized in this construction are the well-studied on-policy Bellman equation and Taylor's theorem, and the less-studied discrete-time Girsanov theorem.

Similarly to its continuous-time counterpart, this theorem is very flexible in how it can be applied. To illustrate this, consider this single-step Girsanov result:

Lemma 4.1 (Discrete Girsanov 1-Step)**.** *Let W^{P} be a normal random vector in $\mathbb{R}^n$, let D be an independent, bounded random vector, and let P be the product measure which represents their joint distribution. Then, the measure Q defined as*

$$\mathrm{d}\mathsf{Q} = \exp\left(-\frac{1}{2}\|D\|^2 + D^\top W^{\mathsf{P}} \right) \mathrm{d}\mathsf{P}, \tag{4.6}$$

is a probability measure and the variable

$$W^{\mathsf{Q}} := W^{\mathsf{P}} - D, \tag{4.7}$$

is a normal random vector in Q. □

Proof. See Appendix A.1. □

The random vector D need only be bounded and independent of W^{P} for the change of measure to produce an alternative normal vector W^{Q}. It can be a discrete collection of point masses, a continuous random variable with density, or anything in between. We can expand this result even further to produce the following:

Discrete-Time Girsanov

Lemma 4.2 (Discrete-Time Girsanov Theorem)**.** *Let $(\Omega, \{\mathcal{F}_i\}_{i=0}^N, \mathsf{P})$ be a filtered probability space and let $\{\xi_i\}_{i=0}^N$ be an adapted process where $\xi_0 := (0_n, 0_n)$ and $\xi_{i+1} := (D_i, W_i^{\mathsf{P}})$ for $i = 0, \ldots, N-1$, such that, in P, D_i is a bounded random vector, W_i^{P} is normal random vector, and D_i is independent of W_i^{P}. If Q is the measure* ▶

Discrete-Time Girsanov (cont)

defined by

$$\mathrm{d}\mathsf{Q} = \prod_{i=0}^{N-1} \exp\left(-\frac{1}{2}\|D_i\|^2 + D_i^\top W_i^{\mathsf{P}} \right) \mathrm{d}\mathsf{P}, \tag{4.8}$$

then Q *is a probability measure and*

$$W_i^{\mathsf{Q}} := W_i^{\mathsf{P}} - D_i, \tag{4.9}$$

are $\mathcal{F}_{i+1}$*-measurable, independent normal random vectors in* Q*.* □

Proof. See Appendix A.2. □

So long as the random vector D_i is independent of W_i^{P}, it can depend on the full history of the process, the joint distribution $\{(D_j, W_j^{\mathsf{P}})\}_{j=0}^{i-1}$, which means, for example, it can be a function of $(X_0, X_1, \ldots, X_i)$. Now that we have introduced the approach, the remainder of the chapter is a self-contained study of the method in general.

4.3 Discrete-Time Forward-Backward Difference Equations

4.3.1 Discrete-Time SOC Approximation

We begin by discretizing the C-SOC problem. Let $(\widetilde{\Omega}, \widetilde{\mathcal{F}}, \{\widetilde{\mathcal{F}}_i\}_{i\in\{0,\ldots,N\}}, \widetilde{\mathsf{Q}})$ be the discrete-time filtered probability space and let $\{W_i^{\mathsf{Q}}\}_{i=0}^{N-1}$ be a discrete time Brownian process in $\widetilde{\mathsf{Q}}$, that is, $W_i^{\mathsf{Q}} \sim \mathcal{N}(0, I_n)$ is normally distributed, $\widetilde{\mathcal{F}}_{i+1}$-measurable, and $\{W_i^{\mathsf{Q}}\}$ are mutually independent. The on-policy forward stochastic difference equation is

$$X_{i+1} - X_i = F_i^{\mu} + \Sigma_i W_i^{\mathsf{Q}}, \qquad X_0 = x_0, \tag{4.10}$$

where, using the Euler-Maruyama approximation method,[2]

$$F_i^\mu = f(t_i, X_i, \mu_i(X_i))\Delta t, \qquad \Sigma_i = \sigma(t_i, X_i)\sqrt{\Delta t}, \tag{4.11}$$

and the on-policy value function is

$$V_i^\mu(X_i) = \mathbf{E}_{\widetilde{\mathbb{Q}}}[\sum_{j=i}^{N-1} L_j^\mu + g(X_N)\,|X_i], \tag{4.12}$$

where

$$L_j^\mu = \ell(t_j, X_j, \mu_j(X_j))\Delta t. \tag{4.13}$$

According to [62, Chapter 10, Theorem 10.2.2], when a linear growth condition in x is imposed on f_s^μ, σ_s, and ℓ_s^μ along with a few other conditions, then it can be shown that the absolute error between the Euler-Maruyama approximation X_i and the continuous forward process X_t is of order $\mathcal{O}((\Delta t)^{1/2})$. When σ_s is constant with respect to x, the error bound improves to $\mathcal{O}(\Delta t)$ [62, Chapter 10, Theorem 10.3.5].

4.3.2 Discrete-Time BSDE Approximation

For the discrete-time value function $\{V_i^\mu\}$ and forward process $\{X_i\}$ we define the process $\{Y_i := V_i^\mu(X_i)\}$. Further, we define the term ΔY_i as one that satisfies the backward difference,

$$\Delta Y_i := Y_{i+1} - Y_i, \tag{4.14}$$

where we use separate estimators $\widehat{Y}_{i+1} \approx Y_{i+1}$ and $\Delta\widehat{Y}_i \approx \Delta Y_i$ to obtain a combined estimator

[2] Or some other approximation scheme that results in the form (4.10), (4.12).

Combined Backward Step Estimator

$$\widehat{Y}_i := \widehat{Y}_{i+1} - \Delta\widehat{Y}_i. \tag{4.15}$$

with the interpretation $\widehat{Y}_i \approx V_i^\mu(X_i)$. Both $\widehat{Y}_{i+1}$ and $\Delta\widehat{Y}_i$ can be chosen according to different approximation schemes; these choices are investigated below. These approximation schemes assume the availability of some approximate representation of the value function at the next step $\widetilde{V}_{i+1}^\mu \approx V_{i+1}^\mu$, as well as its derivatives, and they produce a representation $\widetilde{V}_i^\mu \approx V_i^\mu$ using LSMC.

4.3.3 On-Policy Taylor-Expanded Backward Difference

We now propose an estimator for $\Delta\widehat{Y}_i$, the discrete analogue to the on-policy terms defined in (3.5) and (3.7). We begin by noting that the on-policy value function satisfies the on-policy Bellman equation

$$V_i^\mu(X_i) = L_i^\mu + \mathbf{E}_{\widetilde{\mathbb{Q}}}[V_{i+1}^\mu(X_{i+1})|X_i]. \tag{4.16}$$

Consider the second-order Taylor expansion of the approximation $\widetilde{V}_{i+1}^\mu \approx V_{i+1}^\mu$ of the term inside the conditional expectation,

$$\widetilde{V}_{i+1}^\mu(X_{i+1}) = \widetilde{V}_{i+1}^\mu(\overline{X}_{i+1}^{\mathsf{Q}} + \Sigma_i W_i^{\mathsf{Q}}) = \widetilde{Y}_{i+1} + \delta_{i+1}^{\text{h.o.t.}}, \tag{4.17}$$

$$\widetilde{Y}_{i+1} := \overline{Y}_{i+1} + \overline{Z}_{i+1}^\top W_i^{\mathsf{Q}} + \frac{1}{2} W_i^{\mathsf{Q}\top} \overline{M}_{i+1} W_i^{\mathsf{Q}}, \tag{4.18}$$

centered at the conditional mean,

$$\overline{X}_{i+1}^{\mathsf{Q}} := \mathbf{E}_{\widetilde{\mathbb{Q}}}[X_{i+1}|X_i] = X_i + F_i^\mu, \tag{4.19}$$

where

$$\overline{Y}_{i+1} := \widetilde{V}^{\mu}_{i+1}(\overline{X}^{\mathsf{Q}}_{i+1}), \tag{4.20}$$

$$\overline{Z}_{i+1} := \Sigma_i^{\top} \partial_x \widetilde{V}^{\mu}_{i+1}(\overline{X}^{\mathsf{Q}}_{i+1}), \tag{4.21}$$

$$\overline{M}_{i+1} := \Sigma_i^{\top} \partial_{xx} \widetilde{V}^{\mu}_{i+1}(\overline{X}^{\mathsf{Q}}_{i+1}) \Sigma_i, \tag{4.22}$$

and $\delta^{\text{h.o.t.}}_{i+1}$ includes the third and higher order terms in the Taylor series expansion. Substituting $\widetilde{Y}_{i+1}$ in for $V^{\mu}_{i+1}(X_{i+1})$ in (4.16) and rearranging terms, and in light of (4.15), we arrive at an estimator for the backward step

On-Policy Taylor $\Delta\widehat{Y}_i$ Estimator

$$\Delta\widehat{Y}^{\text{taylor}}_i := -L^{\mu}_i + \overline{Z}^{\top}_{i+1} W^{\mathsf{Q}}_i + \frac{1}{2} \operatorname{tr}(\overline{M}_{i+1}(W^{\mathsf{Q}}_i W^{\mathsf{Q}\top}_i - I)). \tag{4.23}$$

For the purposes of comparison we restate the on-policy Euler-Maruyama estimators derived in Section 4.1,

$$\Delta\widehat{Y}^{\text{noisy-em}}_i := -L^{\mu}_i + \widetilde{Z}^{\top}_{i+1} W^{\mathsf{Q}}_i, \tag{4.24}$$

$$\Delta\widehat{Y}^{\text{nless-em}}_i := -L^{\mu}_i, \tag{4.25}$$

where,

$$\widetilde{Z}_{i+1} := \Sigma_i^{\top} \partial_x \widetilde{V}^{\mu}_{i+1}(X_{i+1}). \tag{4.26}$$

There are two differences in the proposed Taylor series expansion approach compared to the Euler-Maruyama approach. First, the gradient of the value function is evaluated at $\overline{X}^{\mathsf{Q}}_{i+1}$ instead of X_{i+1}. This effect can be exploited because in the discrete-time ap-

proach the difference equation separates the drift step and the diffusion step, whereas in the continuous-time approach the drift and diffusion are considered inseparable. However, if the continuous-time SDEs are eventually discretized using Euler-Maruyama, this assumption is broken over short intervals anyways. Secondly, the trace term now appears in the Taylor-expansion estimator. While in the continuous-time counterpart second-order effects are infinitesimally small, they can no longer be ignored in the discrete-time approximation. Note, however, that $\mathbf{E}_{\mathsf{Q}}\left[\frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}(W_i^{\mathsf{Q}}W_i^{\mathsf{Q}\top} - I))\middle| X_i\right] = 0$ since $\mathbf{E}_{\mathsf{Q}}[W_i^{\mathsf{Q}}W_i^{\mathsf{Q}\top}|X_i] = I$ and $\overline{M}_{i+1}$ is X_i-measurable.

The following theorem suggests that this choice of approximation of ΔY_i has relatively small residual error.

Error in On-Policy Taylor Estimator $\Delta\widehat{Y}_i$

Theorem 4.3. *The choice $\Delta\widehat{Y}_i^{\text{taylor}}$ in* (4.23) *is an unbiased estimator of the actual value function difference ΔY_i, i.e.,*

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[\Delta\widehat{Y}_i|X_i] = \mathbf{E}_{\widetilde{\mathsf{Q}}}[\Delta Y_i|X_i]. \tag{4.27}$$

Further, the residual error is

$$\Delta Y_i - \Delta\widehat{Y}_i = \delta_{i+1}^{\Delta\widehat{Y}} - \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i], \tag{4.28}$$

$$\delta_{i+1}^{\Delta\widehat{Y}} := \delta_{i+1}^{\widetilde{V}} + \delta_{i+1}^{\text{h.o.t.}}, \tag{4.29}$$

where $\delta_{i+1}^{\widetilde{V}} := V_{i+1}^{\mu}(X_{i+1}) - \widetilde{V}_{i+1}^{\mu}(X_{i+1})$ is the error in the $(i+1)^{st}$ step value function representation. □

Proof. The relationship (4.27) follows directly from taking the conditional expectation $\mathbf{E}_{\widetilde{\mathsf{Q}}}[\,\cdot\,|X_i]$ of both sides of (4.28). We now show (4.28).

Comparing (4.23) with (4.18), it can be easily shown that

$$\Delta\widehat{Y}_i = -L_i^\mu + \widetilde{Y}_{i+1} - \mathbf{E}_{\widetilde{\mathbb{Q}}}[\widetilde{Y}_{i+1}|X_i], \tag{4.30}$$

and, similarly, the Taylor expansion (4.17) immediately yields $Y_{i+1} = \widetilde{Y}_{i+1} + \delta_{i+1}^{\Delta\widehat{Y}}$. Combining these two expressions yields

$$\Delta\widehat{Y}_i = -L_i^\mu + Y_{i+1} - \delta_{i+1}^{\Delta\widehat{Y}} - \mathbf{E}_{\widetilde{\mathbb{Q}}}[Y_{i+1} - \delta_{i+1}^{\Delta\widehat{Y}}|X_i]. \tag{4.31}$$

Substituting in the Bellman equation (4.16) and rearranging we arrive at (4.28). □

In general, the on-policy Taylor expansion residual $\delta_{i+1}^{\text{h.o.t.}}$ has a small mean due to the following result.

Proposition 4.4. *Of the higher order terms in the Taylor expansion residual $\delta_{i+1}^{\text{h.o.t.}}$, the terms with odd order, starting with the third order term, have zero conditional expectations given X_i.* □

Proof. See Appendix A.3. □

Further, under a very basic function approximation scheme, we can entirely dismiss the term $\delta_{i+1}^{\text{h.o.t.}}$.

On-Policy Taylor $\Delta\widehat{Y}_i$ Estimator Exact on LQR Problems

Proposition 4.5. *If the value function approximation $\widetilde{V}_{i+1}^\mu$ is quadratic then $\delta_{i+1}^{\text{h.o.t.}} \equiv 0$. Thus, the residual error is determined entirely by the residual error of the function approximation of V_{i+1}^μ,*

$$\Delta Y_i - \Delta\widehat{Y}_i^{\text{taylor}} = \delta_{i+1}^{\widetilde{V}} - \mathbf{E}_{\widetilde{\mathbb{Q}}}[\delta_{i+1}^{\widetilde{V}}|X_i]. \tag{4.32}$$

▶

On-Policy Taylor $\Delta\widehat{Y}_i$ Estimator Exact on LQR Problems (cont)

If $\widetilde{V}_{i+1}^{\mu}$ is exact $\widetilde{V}_{i+1}^{\mu} \equiv V_{i+1}^{\mu}$ then the estimator is exact,

$$\Delta Y_i = \Delta \widehat{Y}_i^{\text{taylor}}. \tag{4.33}$$

Proof. This is a direct consequence of the fact that if $\widetilde{V}_{i+1}^{\mu}$ is quadratic then its second order Taylor expansion is exact. □

Note that this does not require the true value function to be quadratic, only its approximation. Although using a less expressive representation improves the error coming from the term $\delta_{i+1}^{\text{h.o.t.}}$, there may be a trade-off in terms of increasing the magnitude of the error in $\delta_{i+1}^{\widetilde{V}}$, since the function V_{i+1}^{μ} might be less appropriately modeled.

The most remarkable aspect of Proposition 4.5 is that it suggests that for linear-quadratic-regulator (LQR) problems these estimators are exact up to function approximation error, due to the fact that for LQR problems V_i^{μ} itself is in the class of quadratic functions. This provides a fundamental guarantee for these estimators. On the contrary, the Euler-Maruyama estimators are not exact when applied to LQR problems.

Remark 4.1. *If the value function approximation $\widetilde{V}_{i+1}^{\mu}$ is quadratic, the residual error of the Euler-Maruyama estimators is*

$$\begin{aligned}\Delta Y_i - \Delta \widehat{Y}_i^{\text{noisy-em}} = \delta_{i+1}^{\widetilde{V}} - \mathbf{E}_{\widetilde{\mathbb{Q}}}[\delta_{i+1}^{\widetilde{V}} | X_i] + (\overline{Z}_{i+1} - \widetilde{Z}_{i+1})^{\top} W_i^{\mathbb{Q}} \\ + \frac{1}{2} \operatorname{tr}(\overline{M}_{i+1}(W_i^{\mathbb{Q}} W_i^{\mathbb{Q}\top} - I)).\end{aligned} \tag{4.34}$$

□

Though all three ΔY_i estimators are unbiased, the Taylor-expansion estimator is theoretically far superior on the baseline LQR problem. In numerical experiments illustrated

later we confirm this near-machine precision performance of the Taylor estimator and the divergence of the EM estimators on the same LQR problem.

4.3.4 Estimators of $\widehat{Y}_{i+1}$

We propose two potential estimators for $\widehat{Y}_{i+1} \approx V_{i+1}^{\mu}(X_{i+1})$.

Proposed $\widehat{Y}_{i+1}$ Estimators

First, we propose using the value function approximation associated with the previous backward step to re-estimate the $\widehat{Y}_{i+1}$ values,

$$\widehat{Y}_{i+1}^{\text{re-est}} := \widetilde{V}_{i+1}^{\mu}(X_{i+1}). \tag{4.35}$$

Alternatively, we can also use the estimator

$$\widehat{Y}_{i+1}^{\text{noiseless}} := \widetilde{Y}_{i+1}, \tag{4.36}$$

which ends up cancelling out the terms with W_i^{Q} in them, so that (4.15) reduces to

$$\widehat{Y}_{i}^{\text{noiseless}} = L_i^{\mu} + \overline{Y}_{i+1} + \frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}). \tag{4.37}$$

The following theorem establishes the error analysis of the two Taylor-expansion-based estimators.

On-Policy Estimator $\widehat{Y}_i$ Bias and Variance

Theorem 4.6. *For the estimator $\widehat{Y}_i := \widehat{Y}_{i+1} - \Delta\widehat{Y}_i$, where $\Delta\widehat{Y}_i$ is defined in* (4.23) *and* ▶

On-Policy Estimator $\widehat{Y}_i$ Bias and Variance (cont)

$\widehat{Y}_{i+1}$ is defined in (4.35) *or* (4.36)*, the bias is*

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_i - \widehat{Y}_i^{\text{re-est}}|X_i] = \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\widetilde{V}}|X_i], \tag{4.38}$$

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_i - \widehat{Y}_i^{\text{noiseless}}|X_i] = \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\widetilde{V}} + \delta_{i+1}^{\text{h.o.t.}}|X_i]. \tag{4.39}$$

The variances of these estimators are

$$\text{Var}_{\widetilde{\mathsf{Q}}}[\widehat{Y}_i^{\text{re-est}}|X_i] = \text{Var}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\text{h.o.t.}}|X_i], \tag{4.40}$$

$$\text{Var}_{\widetilde{\mathsf{Q}}}[\widehat{Y}_i^{\text{noiseless}}|X_i] = 0. \tag{4.41}$$

Proof. See Appendix A.4. □

This theorem shows that the *re-estimate* condition has less bias than the *noiseless* condition, but it is a higher variance estimator. We also observe that when $\delta_{i+1}^{\text{h.o.t.}} = 0$ the bias and variance of these two estimators are identical. However, since it is not immediately clear which condition is superior when this is not true, we examine both methods and compare the results in Section 4.5.

4.3.5 Drifted Taylor-Expanded Backward Difference

We now offer a discrete-time approximation of the drifted off-policy FBSDEs. Let $(\widetilde{\Omega}, \widetilde{\mathcal{F}}, \{\widetilde{\mathcal{F}}_i\}_{i\in\{0,\dots,N\}}, \widetilde{\mathsf{P}})$ be an alternative discrete-time filtered probability space where W_i^{P} is the associated Brownian process. Define on this space the difference equation

$$X_{i+1} - X_i = K_i + \Sigma_i W_i^{\mathsf{P}}, \qquad X_0 = x_0, \tag{4.42}$$

where the process $\{K_i\}_{i=0}^{N-1}$ is chosen at will, $\widetilde{\mathcal{F}}_{i+1}$-measurable, and independent of W_i^{P}. For example, K_i can be constructed using the function $K_i(\omega) = \mathcal{K}_i(X_i(\omega), \xi_i(\omega))$, where $\{\xi_i\}$ is some random process where ξ_i is $\widetilde{\mathcal{F}}_{i+1}$-measurable and independent of W_i^{P} (but not necessarily of W_{i-1}^{P}). Each K_i must also be selected such that

$$D_i := \Sigma_i^{-1}(F_i^{\mu} - K_i), \tag{4.43}$$

is bounded.

The discrete-time version of Girsanov's theorem, Lemma 4.2, can be used to produce the measure $\widetilde{\mathsf{Q}}$, defined as

$$\mathrm{d}\widetilde{\mathsf{Q}} = \prod_{i=0}^{N-1} \exp\left(-\frac{1}{2}\|D_i\|^2 + D_i^{\top} W_i^{\mathsf{P}} \right) \mathrm{d}\widetilde{\mathsf{P}}, \tag{4.44}$$

which satisfies the assumptions of Section 4.3.1. Under the theorem,

$$W_i^{\mathsf{Q}} := W_i^{\mathsf{P}} - D_i, \tag{4.45}$$

for $i = 0, \ldots, N-1$ is an $\mathcal{F}_{i+1}$-measurable process of independent, normally distributed random vectors. It is easy to see that the drifted forward difference (4.42) and the on-policy forward difference (4.10) are identical under the substitution (4.45). Thus, we conclude that the drifted process $\{X_i\}$ still satisfies the on-policy Bellman equation (4.16) for the same on-policy value function V^{μ}.

To derive the backward step, we perform a Taylor expansion centered at

$$\overline{X}_{i+1}^{\mathsf{P}} := \mathbf{E}_{\widetilde{\mathsf{P}}}[X_{i+1} | X_i, K_i] = X_i + K_i, \tag{4.46}$$

instead of $\overline{X}_{i+1}^{\mathsf{Q}}$. The expressions defining $\widetilde{Y}_{i+1}$, $\overline{Y}_{i+1}$, $\overline{Z}_{i+1}$, and $\overline{M}_{i+1}$ (4.17), (4.18), (4.20), (4.21), (4.22) are all identical except for replacing $\overline{X}_{i+1}^{\mathsf{Q}}, W_i^{\mathsf{Q}}$ with $\overline{X}_{i+1}^{\mathsf{P}}, W_i^{\mathsf{P}}$. Again,

substituting $\widetilde{Y}_{i+1}$ in for $V_{i+1}^{\mu}(X_{i+1})$ in (4.16) and rearranging terms in light of (4.15), we arrive at an estimator for the backward step.

Off-Policy Drifted Taylor $\Delta\widehat{Y}_i$ Estimator

The off-policy drifted Taylor-expanded $\Delta\widehat{Y}_i$ estimator is defined as

$$\begin{aligned}\Delta\widehat{Y}_i^{\text{drift}} := &-L_i^{\mu} + \overline{Z}_{i+1}^{\top} W_i^{\mathsf{P}} - \overline{Z}_{i+1}^{\top} D_i \\ &+ \frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}(W_i^{\mathsf{P}} W_i^{\mathsf{P}\top} - I - D_i D_i^{\top})).\end{aligned} \tag{4.47}$$

where

$$\begin{aligned}\overline{X}_{i+1}^{\mathsf{P}} &:= \mathbf{E}_{\widetilde{\mathsf{P}}}[X_{i+1}|X_i, K_i] = X_i + K_i, \\ \overline{Y}_{i+1} &:= \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{P}}), \\ \overline{Z}_{i+1} &:= \Sigma_i^{\top}\partial_x \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{P}}), \\ \overline{M}_{i+1} &:= \Sigma_i^{\top}\partial_{xx}\widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{P}})\Sigma_i, \\ D_i &:= \Sigma_i^{-1}(F_i^{\mu} - K_i).\end{aligned}$$

Recognize that this is a generalization of (4.23), by noting that when $K_i = F_i^{\mu}$ then $D_i = 0$ and the drifted forward difference (4.42) and the backward step reduce to their on-policy form (4.10), (4.23).

Error in Off-Policy Drifted Taylor Estimator $\Delta\widehat{Y}_i$

Lemma 4.7. *The choice* (4.47) *yields the residual error*

$$\Delta Y_i - \Delta\widehat{Y}_i = \delta_{i+1}^{\Delta\widehat{Y}} - \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]. \tag{4.48}$$

Proof. Substituting (4.45) into

$$\widetilde{Y}_{i+1} := \overline{Y}_{i+1} + \overline{Z}_i^\top W_i^{\mathsf{P}} + \frac{1}{2} W_i^{\mathsf{P}\top} \overline{M}_i W_i^{\mathsf{P}}, \tag{4.49}$$

yields

$$\begin{aligned}\widetilde{Y}_{i+1} &= \overline{Y}_{i+1} + \overline{Z}_{i+1}^\top (W_i^{\mathsf{Q}} + D_i) + \frac{1}{2}(W_i^{\mathsf{Q}} + D_i)^\top \overline{M}_{i+1}(W_i^{\mathsf{Q}} + D_i) \\ &= \overline{Y}_{i+1} + \overline{Z}_{i+1}^\top W_i^{\mathsf{Q}} + \overline{Z}_{i+1}^\top D_i + D_i^\top \overline{M}_{i+1} W_i^{\mathsf{Q}} + \frac{1}{2} \operatorname{tr}\big(\overline{M}_{i+1}(W_i^{\mathsf{Q}} W_i^{\mathsf{Q}\top} + D_i D_i^\top)\big).\end{aligned}$$

Note that D_i, $\overline{Y}_{i+1}$, $\overline{Z}_{i+1}$, and $\overline{M}_{i+1}$, are (X_i, K_i)-measurable. Taking the conditional expectation in the on-policy measure $\widetilde{\mathsf{Q}}$ yields

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[\widetilde{Y}_{i+1} | X_i, K_i] = \overline{Y}_{i+1} + \overline{Z}_{i+1}^\top D_i + \frac{1}{2} \operatorname{tr}\big(\overline{M}_{i+1}(I + D_i D_i^\top)\big). \tag{4.50}$$

Comparing (4.47), (4.49), and (4.50), it can be easily shown that

$$\Delta\widehat{Y}_i = -L_i^\mu + \widetilde{Y}_{i+1} - \mathbf{E}_{\widetilde{\mathsf{Q}}}[\widetilde{Y}_{i+1} | X_i, K_i]. \tag{4.51}$$

The Taylor expansion (4.17) immediately yields $Y_{i+1} = \widetilde{Y}_{i+1} + \delta_{i+1}^{\Delta\widehat{Y}}$. Combining these two expressions yields

$$\Delta\widehat{Y}_i = -L_i^\mu + Y_{i+1} - \delta_{i+1}^{\Delta\widehat{Y}} - \mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_{i+1} - \delta_{i+1}^{\Delta\widehat{Y}} | X_i, K_i]. \tag{4.52}$$

Substituting in the Bellman equation $Y_i = L_i^\mu + \mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_{i+1} | X_i]$ (4.16) and rearranging, we have

$$\Delta Y_i - \Delta\widehat{Y}_i = \delta_{i+1}^{\Delta\widehat{Y}} + \mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_{i+1} - \delta_{i+1}^{\Delta\widehat{Y}} | X_i, K_i] - \mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_{i+1} | X_i]. \tag{4.53}$$

Under the measure $\widetilde{\mathsf{Q}}$, Y_{i+1} is independent of K_i given X_i, so we have

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_{i+1}|X_i, K_i] = \mathbf{E}_{\widetilde{\mathsf{Q}}}[Y_{i+1}|X_i],$$

and by subsituting into the previous equation we arrive at (4.48). □

The distribution of the residual error $\Delta Y_i - \Delta\widehat{Y}_i$ depends on the measure we use to interpret it. For numerical applications we sample from the measure $\widetilde{\mathsf{P}}$ instead of $\widetilde{\mathsf{Q}}$, and thus this estimator is no longer unbiased with respect to the sampled distribution. The conditional expectation with respect to $\widetilde{\mathsf{P}}$ of the right hand side of (4.48) is

$$\begin{aligned}\varepsilon_{i+1}^{\mathsf{P}|\mathsf{Q}} &:= \mathbf{E}_{\widetilde{\mathsf{P}}}[\delta_{i+1}^{\Delta\widehat{Y}} - \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]|X_i, K_i] \\ &= \mathbf{E}_{\widetilde{\mathsf{P}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i] - \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]. \end{aligned} \tag{4.54}$$

The two estimators for $\widehat{Y}_{i+1}$, (4.35) (4.36), presented in Section 4.3.4, can be used without modification, given that in the noiseless condition, $\widetilde{Y}_{i+1}$ is taken to be (4.49). The drifted *noiseless* estimator now resolves to

$$\widehat{Y}_i^{\text{noiseless}} = L_i^{\mu} + \overline{Y}_{i+1} + \overline{Z}_{i+1}^{\top} D_i + \frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}(I + D_i D_i^{\top})). \tag{4.55}$$

Off-Policy Estimator $\widehat{Y}_i$ Bias and Variance

Theorem 4.8. *For the estimator $\widehat{Y}_i := \widehat{Y}_{i+1} - \Delta\widehat{Y}_i$ where $\Delta\widehat{Y}_i$ is defined in* (4.47) *and $\widehat{Y}_{i+1}$ is defined in* (4.35) *or* (4.36) *the bias is*

$$\mathbf{E}_{\widetilde{\mathsf{P}}}[Y_i - \widehat{Y}_i^{\text{re-est}}|X_i, K_i] = \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i] - \mathbf{E}_{\widetilde{\mathsf{P}}}[\delta_{i+1}^{\text{h.o.t.}}|X_i, K_i], \tag{4.56}$$

$$\mathbf{E}_{\widetilde{\mathsf{P}}}[Y_i - \widehat{Y}_i^{\text{noiseless}}|X_i, K_i] = \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]. \tag{4.57}$$

►

Off-Policy Estimator $\widehat{Y}_i$ Bias and Variance (cont)

The variances of the estimators are

$$\mathrm{Var}_{\widetilde{\mathsf{P}}}[\widehat{Y}_i^{\text{re-est}}|X_i, K_i] = \mathrm{Var}_{\widetilde{\mathsf{P}}}[\delta_{i+1}^{\text{h.o.t.}}|X_i, K_i] \tag{4.58}$$

$$\mathrm{Var}_{\widetilde{\mathsf{P}}}[\widehat{Y}_i^{\text{noiseless}}|X_i, K_i] = 0. \tag{4.59}$$

Proof. See Appendix A.4. □

Since $\widetilde{\mathsf{Q}}$ is not available during computation, we characterize $\mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i]$ exclusively in the measure $\widetilde{\mathsf{P}}$ using the next result.

Proposition 4.9. *The bias term appearing in Theorem 4.8 is bounded as*

$$|\mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]| \leq \exp(\frac{1}{2}\|D_i\|^2)\ \mathbf{E}_{\widetilde{\mathsf{P}}}[(\delta_{i+1}^{\Delta\widehat{Y}})^2|X_i, K_i]^{1/2}. \tag{4.60}$$

Proof. See Appendix A.5. □

Although the error bound in Proposition 4.9 suggests that the bias grows rapidly with the magnitude $\|D_i\|$, when this magnitude is small ($\|D_i\| \leq 1$) the first term in the product on the right hand side of (4.60) is bounded by $\sqrt{e} \approx 1.65$. To illustrate the effect of $\|D_i\|$ on the error bound, consider a one-dimensional problem where we select $K_i = F_i^\mu + a$ for some random variable a with bounded magnitude $|a| \leq \Sigma_i$ a.s.. It subsequently follows that $\exp(\|D_i\|^2) \leq \sqrt{e}$ a.s.. This suggests that, in general, the magnitude of the difference $F_i^\mu - K_i$ should be proportional to the diffusion Σ_i. Further, it is still the case that if the value function approximation is quadratic then the higher order terms $\delta_{i+1}^{\text{h.o.t.}}$ drop out.

These analytical results justify the assumption that for appropriately chosen K_i, the choice of (4.47) represents a low bias, low variance approximator for the backward difference step. It also provides guidance for how to select K_i.

4.4 Policy Improvement

In this section we discuss how policies can be improved based on the value function parameters obtained from the backward passes. First, we discuss a naïve continuous-time approximation approach arising from the Hamiltonian used in HJB equations. Continuous-time analysis of the Hamiltonian suggests that the optimal control policy π^* satisfies the inclusion (2.5), so a naïve approach to improving the policy would be to use the Euler-Maruyama approximation of the dynamics and costs along with the gradient of the recent approximation of the value function to evaluate this policy optimization. This Hamiltonian-based approach

$$\begin{aligned}\widetilde{\pi}_i^*(x) &\in \arg\min_{u\in U}\{\ell(t_i,x,u) + f(t_i,x,u)^\top \partial_x \widetilde{V}_i(x)\} \\ &\equiv \arg\min_{u\in U}\{L_i(x,u) + F_i(x,u)^\top \partial_x \widetilde{V}_i(x)\},\end{aligned} \tag{4.61}$$

is used for estimating the optimal policy in [16, 17].[3]

According to the discussion in the previous section, we propose an alternative Taylor-based approach to (4.61) as follows. We begin with a discrete approximation of the continuous problem and form the Q-value function at time i, given the value function V_{i+1}^μ,

$$Q_i^\mu(x,u) := L_i(x,u) + \mathbf{E}_{\widetilde{\mathsf{Q}}_i}[V_{i+1}^\mu(X_{i+1})|X_i = x], \tag{4.62}$$

where $\widetilde{\mathsf{Q}}_i$ is the measure corresponding to the forward difference step

$$X_{i+1} - x = F_i(x,u) + \Sigma_i W_i^{\mathsf{Q}}. \tag{4.63}$$

[3]The equivalence is in the case the Euler-Maruyama approximation is used.

The optimal Bellman equation indicates that the optimal policy satisfies

$$\pi_i^*(x) \in \arg\min_{u \in U} Q_i^{\pi^*}(x,u),$$

and the optimal value function satisfies

$$V_i^{\pi^*}(x) = \min_{u \in U} Q_i^{\pi^*}(x,u).$$

Notice that when $V_{i+1}^{\pi} \leq V_{i+1}^{\mu}$ and $Q_i^{\pi}(x,\pi_i(x)) \leq Q_i^{\mu}(x,\mu_i(x))$ then $V_i^{\pi} \leq V_i^{\mu}$, so π will be an improved policy over μ. Letting

$$\overline{X}_{i+1}^{x,u} := \mathbf{E}_{\widetilde{\mathbf{Q}}_i}[X_{i+1}|X_i = x] = x + F_i(x,u), \tag{4.64}$$

and performing the same Taylor expansion approach as in (4.17), (4.18), we arrive at the approximation $\widetilde{Q}_i^{\mu} \approx Q_i^{\mu}$ defined as

Approximate Taylor Q-Value Function

$$\widetilde{Q}_i^{\mu}(x,u) := L_i(x,u) + \overline{Y}_{i+1}^{x,u} + \frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}^{x,u}), \tag{4.65}$$

where

$$\begin{aligned}\overline{Y}_{i+1}^{x,u} &:= \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{x,u}),\\ \overline{M}_{i+1}^{x,u} &:= \Sigma_i^{\top}\partial_{xx}\widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{x,u})\Sigma_i.\end{aligned}$$

Proposition 4.10. *The error when using* (4.65) *to approximate the Q-value function is*

$$Q_i^{\mu}(x,u) - \widetilde{Q}_i^{\mu}(x,u) = \mathbf{E}_{\widetilde{\mathbf{Q}}_i}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i = x]. \tag{4.66}$$

□

Proof. Due to (4.17) we have $Y_{i+1} = \widetilde{Y}_{i+1} + \delta_{i+1}^{\Delta\widehat{Y}}$. We arrive at (4.66) by subtracting (4.65) from (4.62), and then substituting in Y_{i+1} for $V_{i+1}^{\mu}(X_{i+1})$. □

In general, we seek a policy that minimizes this Q-value function,

$$\mu_i^*(x; \widetilde{V}_{i+1}^{\mu}) := \min_{u \in U} \widetilde{Q}_i^{\mu}(x, u). \tag{4.67}$$

When the function L_i is quadratic in terms of u and/or contains an L_1 regularization term like $\sum_{j=1}^{n} |u^j|$, U is an interval set, F_i is affine in the control, and $\widetilde{V}_{i+1}^{\mu}$ is quadratic, then the optimization (4.67) has an analytic solution. Also, similarly to the previous section, when $\widetilde{V}_{i+1}^{\mu}$ is quadratic, as is the case in LQR problems, the Taylor expansion of the Q-value function is exact. Thus, this optimization will yield the exact optimal control solution for the LQR problem.

4.5 Numerical Results

Next, we numerically evaluate and compare the proposed Taylor estimators to the naïve Euler-Maruyama estimators on two problems, a nonlinear 1-dimensional problem and an LQR 4-dimensional problem. The estimators discussed in this work are summarized in Table 4.1.

4.5.1 Nonlinear 1D Problem

Consider the scalar optimal control problem with the dynamics and cost

$$\begin{aligned} &\mathrm{d}X_s = \big(0.1(X_s - 3)^2 + 0.2u_s\big)\mathrm{d}s + 0.8\,\mathrm{d}W_s, \qquad x_0 = 7, \\ &J(t, x_t; u_{[t,T]}) = \mathbf{E}_{\mathsf{Q}}\bigg[\int_t^T \big(12\,|X_s - 6| + 0.4\,u_s^2\big)\,\mathrm{d}s + 25\,X_T^2 \bigg| X_t = x_t\bigg], \end{aligned}$$

Table 4.1: Expressions for the proposed noiseless and re-estimate estimators, as well as the competing Euler-Maruyama estimators (4.3), and (4.4) (used in [16]).

Estimator	$\widehat{Y}_i =$
Taylor Noiseless	$L_i^\mu + \overline{Y}_{i+1} + \overline{Z}_{i+1}^\top D_i$ $+\frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}(I + D_i D_i^\top))$
Taylor Reestimate	$\widetilde{V}_{i+1}^\mu(X_{i+1}) + L_i^\mu - \overline{Z}_{i+1}^\top W_i^{\mathsf{P}} + \overline{Z}_{i+1}^\top D_i$ $+\frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}(I + D_i D_i^\top - W_i^{\mathsf{P}} W_i^{\mathsf{P}\top}))$
Euler-Maruyama Noiseless [16]	$\widetilde{V}_{i+1}^\mu(X_{i+1}) + L_i^\mu + \widetilde{Z}_{i+1}^\top D_i$
Euler-Maruyama Noisy	$\widetilde{V}_{i+1}^\mu(X_{i+1}) + L_i^\mu - \widetilde{Z}_{i+1}^\top W_i^{\mathsf{P}} + \widetilde{Z}_{i+1}^\top D_i$

over a time interval of length $T = 10$, with $N = 200$ discrete timesteps. We compute a ground-truth optimal value function V_i^* by directly evaluating the optimal Bellman equation using a finely-gridded state space, control space, and noise space, and set the optimal policy as the target $\mu \equiv \pi^*$. The optimal value function is visualized in Figure 4.5 (the yellow surface), along with two forward-backward trajectory distributions $\{(X_i, Y_i)\}$ considered for evaluation: (a) the optimal $K_i^{\text{optimal}} = F_i^{\pi^*}$ (the cyan trajectories in Figure 4.5(a)), and (b) the suboptimal $K_i^{\text{subopt}} = -0.2X_i$ (the orange trajectories).

Although FBSDE methods are typically iterative methods, to investigate performance of estimators in a controlled setting, each trial performs one forward pass, and then uses Chebyshev polynomials to locally approximate the optimal value function in a single backward pass. For evaluation, we are interested in an interpretation of accuracy which weighs accuracy in the center of the distribution X_i equally to accuracy a few standard deviations away from the mean. We are interested in this because future iterations should shift the distribution away from the mean as the policy improves. To this end, we first compute the

time-varying interval set

$$\begin{aligned}\mathcal{C}_i &:= [\underline{c}_i, \overline{c}_i] := [\overline{x}_i - \max\{3\sigma_i, 1\}, \overline{x}_i + \max\{3\sigma_i, 1\}] \\ &\approx \{\underline{c}_i, \underline{c}_i + \Delta x, \ldots, \overline{c}_i\} =: \widetilde{\mathcal{C}}_i, \end{aligned} \tag{4.68}$$

for $i = 0, \ldots, N$, where $\overline{x}_i, \sigma_i$ are the mean and standard deviation of X_i for the optimal or suboptimal forward trajectory distribution, which we denote *confidence regions* and visualize in Figure 4.5(b). For the uniform distribution over $\mathcal{C}_i$, we use the relative absolute error (RAE) metric [65, Chapter 5]

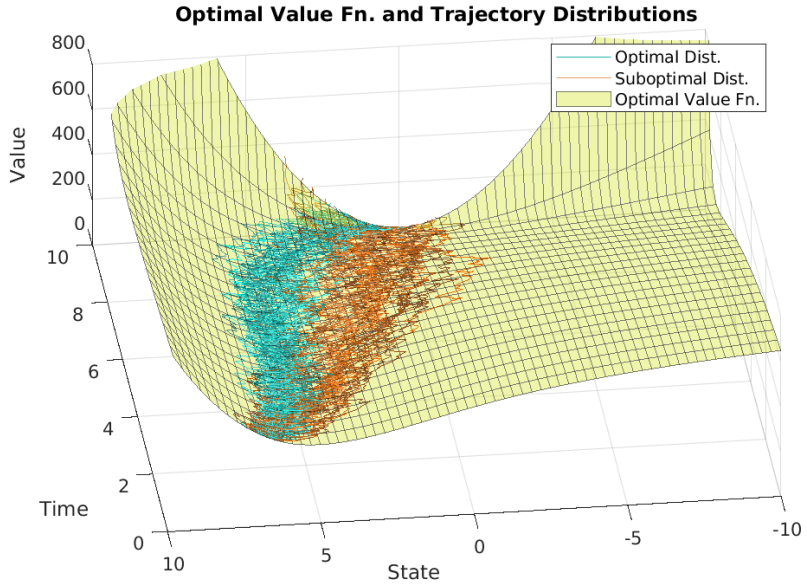
$$\frac{\int_{\mathcal{C}_i} |\widetilde{V}_i(x) - V_i^*(x)| \, \mathrm{d}x}{\int_{\mathcal{C}_i} |\int_{\mathcal{C}_i} V_i^*(y) \, \mathrm{d}y - V_i^*(x)| \, \mathrm{d}x} \approx \frac{\sum_{x \in \widetilde{\mathcal{C}}_i} |\widetilde{V}_i(x) - V_i^*(x)|}{\sum_{x \in \widetilde{\mathcal{C}}_i} |\sum_{y \in \widetilde{\mathcal{C}}_i} \frac{1}{|\widetilde{\mathcal{C}}_i|} V_i^*(y) - V_i^*(x)|}, \tag{4.69}$$

to quantify accuracy.

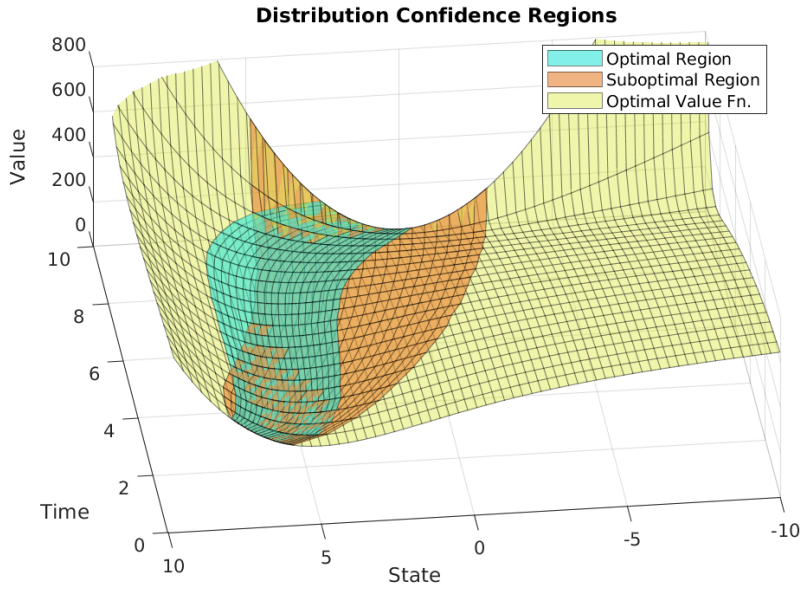
In Figures 4.6(a)-(b) we illustrate the approximate value functions for each estimator, restricted to the optimal confidence region $\mathcal{C}_i^{\text{optimal}}$, in a trial where the forward distribution is generated with the suboptimal drift K_i^{subopt}. The Taylor estimators match the ground truth with relatively no error, and, while the EM noiseless condition largely matches the ground truth, some fluctuation is apparent. The EM noisy estimator diverges significantly from the ground truth, though its curvature still somewhat matches the curvature of the ground truth.

We also performed a series of trials to compare accuracy over time for a 2×2 set of forward sampling conditions (K_i^{optimal} and K_i^{subopt}), and accuracy metrics (RAE over $\mathcal{C}_i^{\text{optimal}}$ and $\mathcal{C}_i^{\text{subopt}}$). For each estimator/sampling condition/accuracy metric 10 independent trials were run.

For all conditions, the Taylor-based estimators show little difference in performance between each other, and, in general, significantly outperform the Euler-Maruyama (EM) estimators. The EM noiseless estimator is, in general, far more accurate than its noisy counterpart, though it could be argued that the noisy estimator is better behaved. For all estimators the plots illustrated at the top-left and bottom-right of Figure 4.6(c) perform bet-

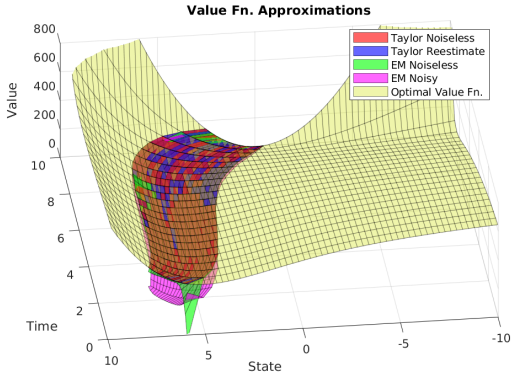


(a) Optimal and suboptimal trajectory distributions (with corresponding drifts K_i^{optimal} / K_i^{subopt}) used as the forward pass

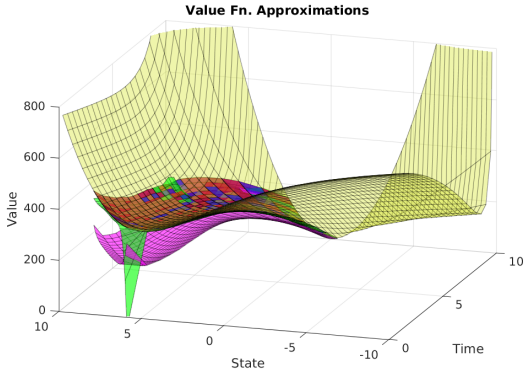


(b) Optimal and suboptimal confidence regions ($\mathcal{C}_i^{\text{optimal}}$ / $\mathcal{C}_i^{\text{subopt}}$) used for accuracy evaluation

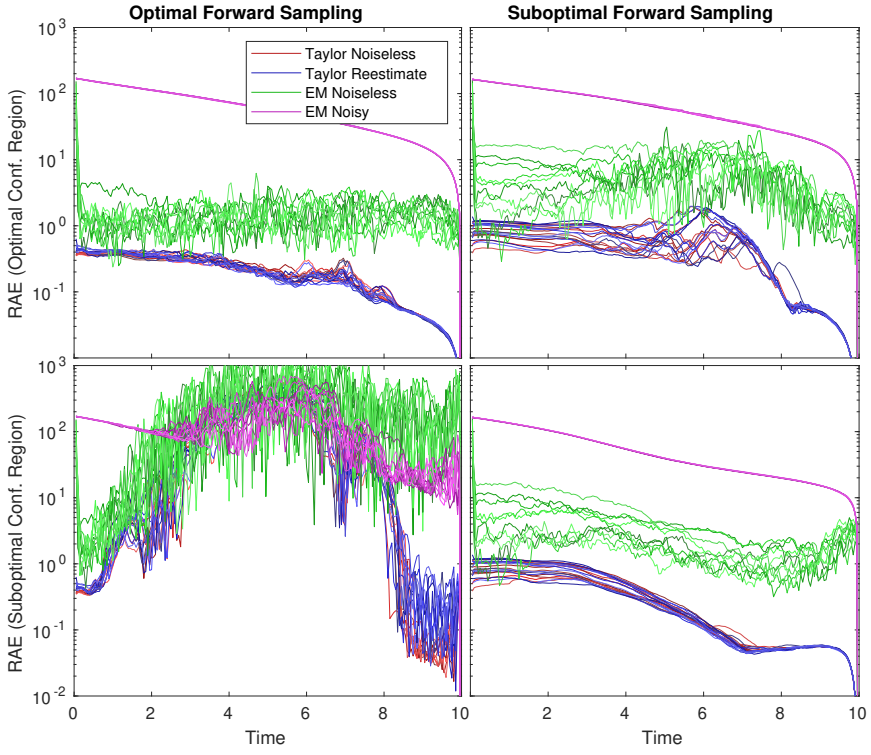
Figure 4.5: Optimal value function and trajectory distributions for the 1-dimensional nonlinear problem. The yellow surface is the ground truth optimal value function and the cyan and orange trajectories are the optimal and suboptimal trajectory distributions, respectively, used as forward distributions for evaluation. The regions in (b) are computed from the statistics of the trajectory distributions $\{X_i\}$ visualized in (a). The trajectories and regions are projected onto the value function so they can be easily compared to forthcoming figures.



(a) Value function approximations, restricted to the optimal confidence region $\mathcal{C}_i^{\text{optimal}}$. Each estimator's parameters are computed with the suboptimal trajectory distribution condition (drift is K_i^{subopt})



(b) Rotated view of (a)



(c) Relative absolute error (RAE) (4.69) for the two sampling conditions (K_i^{optimal} / K_i^{subopt}) and two evaluation distributions ($\mathcal{C}_i^{\text{optimal}}$ / $\mathcal{C}_i^{\text{subopt}}$)

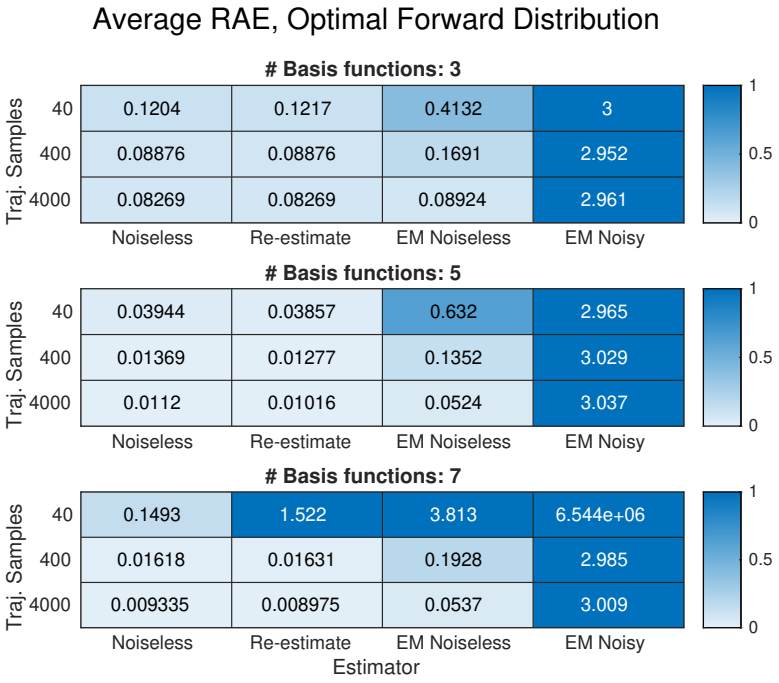
Figure 4.6: Value function accuracy experiments for each estimator on the 1-dimensional nonlinear control problem. We use Chebyshev polynomials to represent the value function basis functions, using 7 basis functions (which can represent degree 6 polynomials). Each of the two figure columns in (c) refer to the sampling condition used in the FBSDE trial to compute the value function parameters. The rows refer to the confidence regions over which the value function approximation is evaluated for accuracy.

ter than the those at top-right and bottom-left because the confidence region completely overlaps the sampling distribution. Thus, the top-right and bottom-left plots are largely measures of extrapolation accuracy on a distribution far different from the one used to approximate the value function. The $K_i^{\text{optimal}}/\mathcal{C}_i^{\text{subopt}}$ condition in the bottom-left has relatively high error for all estimators, even compared to the other extrapolation condition (top-right plot). If we compare the distributions and regions in Figure 4.5, it appears that the suboptimal distribution better covers the optimal region than the optimal distribution covers the suboptimal region. The high error in this condition is likely due to this fact, coupled with the fact that Runge's phenomenon begins to dominate outside the region covered by forward distribution.

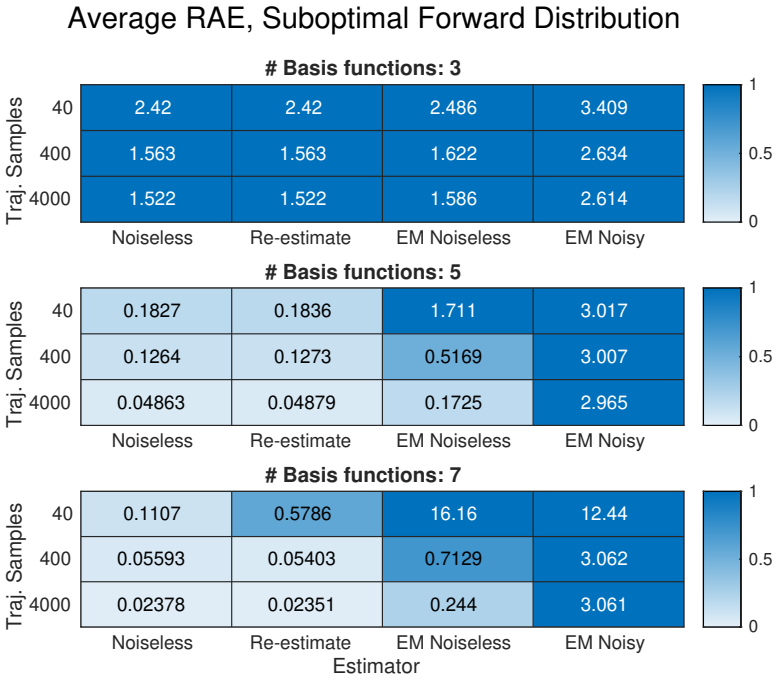
The poor performance in the bottom-left plot suggests that while an on-policy estimator with forward sampling distribution following the optimal policy may produce accurate function estimates within the tight region of the optimal trajectories, off-policy estimators with sampling distributions which cover a broader region of the state space may be more robust (i.e., accurately represent the value function over a broader region of the state space) without significantly reducing approximation accuracy. The idea that broader exploration, especially of regions for which the value would not significantly increase, produces more robust value function approximation is a key idea in "soft"-reinforcement learning (RL) literature [45]. However, while soft-RL methods propose changing the objective function to incentivize exploration, the proposed methods in this chapter demonstrate that exploration can be achieved without changing the target value function.

We also ran a series of simulations to investigate how each estimator performs under different algorithmic conditions, visualized in Figure 4.7. For each element in Figure 4.7 we average the RAE approximations (4.69) over both 20 trials and $N = 200$ timesteps.

The results show that in all cases the proposed Taylor-based estimators perform as well as the Euler-Maruyama estimators, and for the vast majority perform significantly better. Although the Taylor-based estimators generally perform equally well, there are slight dif-



(a) Optimal forward sampling distribution generated with K^{optimal} (On-policy estimators).

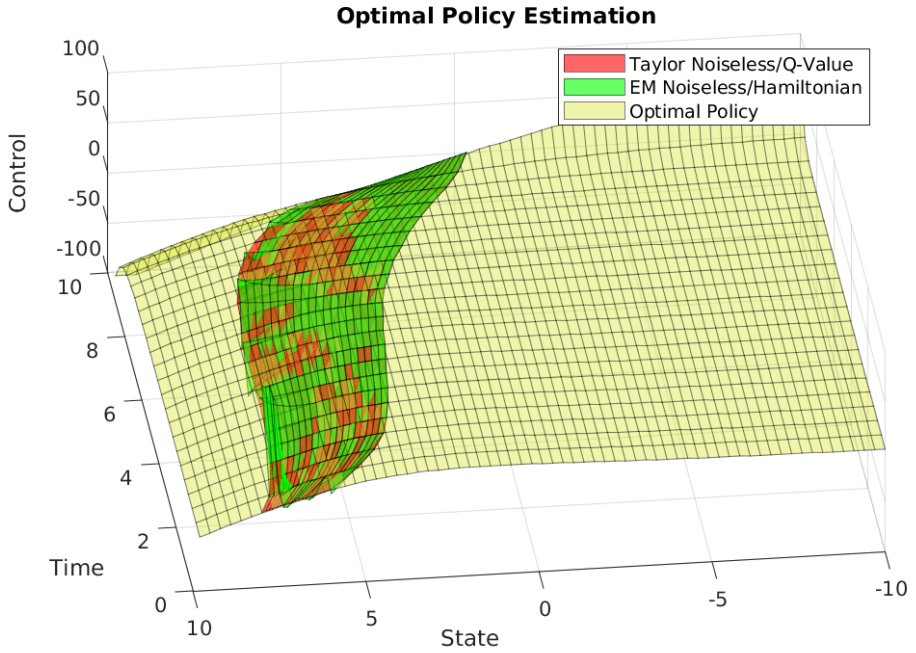


(b) Suboptimal forward sampling distribution generated with K^{subopt} (Off-policy estimators).

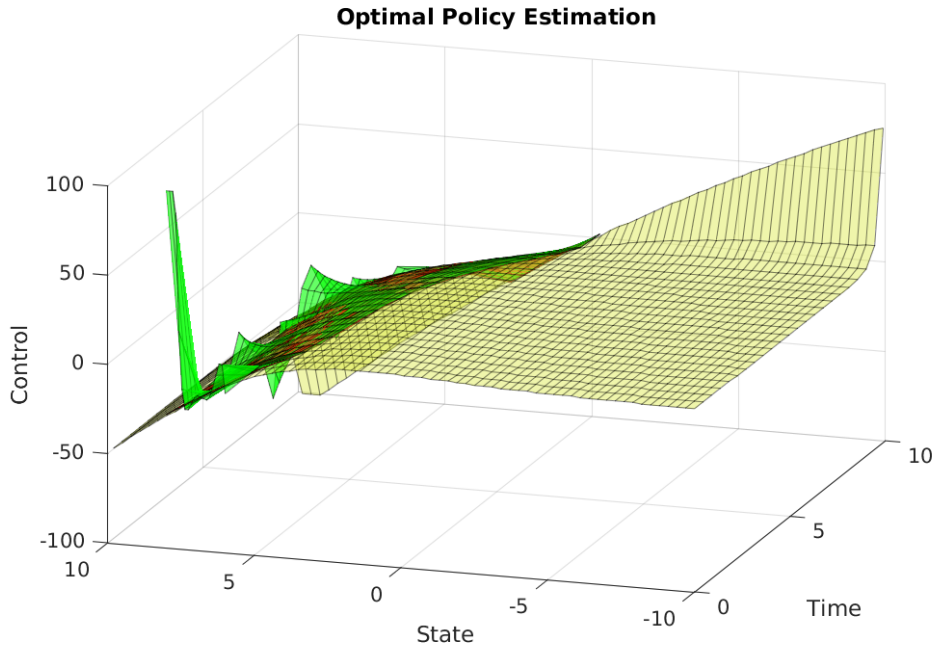
Figure 4.7: Heatmaps of experiments comparing the proposed estimators (Noiseless/Re-estimate) against naïve estimators (EM Noiseless/EM Noisy), with varying numbers of basis functions and numbers of trajectory samples. Each matrix element is the RAE in the $\mathcal{C}_i^{\text{optimal}}$ distribution, averaged over both 20 trials and $N = 200$ timesteps.

ferences in how they perform in different conditions. The Taylor-noiseless estimator seems to outperform the re-estimate estimator when the number of trajectory samples is low, and vice versa when the number is high. Recall that the error analysis suggests that the re-estimate estimator has lower bias but higher variance than the Taylor-noiseless estimator. The simulated results confirm the theoretical results, that is, when the number of trajectory samples is low, high variance makes the re-estimate estimator perform poorly, but when there are enough samples to overcome the variance in the estimator, the low bias properties can result in better accuracy. In practice, however, it is likely that the low variance of the Taylor-noiseless estimator is preferable to the slightly more bias it introduces.

To evaluate the policy improvement scheme proposed in Section 4.4, we took the value function approximation produced in the Taylor noiseless condition (visualized as the red surface in Figures 4.6(a)-(b)) and used it for $\widetilde{V}_{i+1}^{\mu}$ in the Q-value function approximation (4.65). We then produce the policy optimization (4.67) based on these parameters (the red surface in Figure 4.8). We compare this to the Hamiltonian-based policy (4.61) (the green surface in Figure 4.8), where the value function $\widetilde{V}_i$ comes from the EM-noiseless condition (the green surface in Figures 4.6(a)-(b)). This method replicates the policy optimization used in [16, 17]. Although the Hamiltonian-based optimization shows reasonable performance, our proposed Taylor-based approximation is far more accurate and well-behaved.



(a) Local representations of estimated optimal policy



(b) Rotated view of (a)

Figure 4.8: Comparing the proposed policy optimization to the method utilized in [16] [17], (4.61) using the EM-noiseless value function approximation, and the ground truth (yellow surface). The red surface is computed as (4.67) where $\widetilde{V}_{i+1}^{\mu}$ is based on the results of the Taylor-noiseless estimator backward pass. Both policies are restricted to $\mathcal{C}_i^{\text{optimal}}$.

4.5.2 LQR 4D Problem

We also tested the proposed estimators on a linearized version of the 4-dimensional finite time cart-pole problem,

$$\begin{aligned}\mathrm{d}X_s &= \left(\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & a_1 & a_2 & a_3 \\ 0 & 0 & 0 & 1 \\ 0 & a_4 & a_5 & a_6 \end{bmatrix} X_s + \begin{bmatrix} 0 \\ b_1 \\ 0 \\ b_2 \end{bmatrix} u_s \right) \mathrm{d}s + \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 1 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \mathrm{d}W_s \\ &= (AX_s + Bu_s)\mathrm{d}s + \sigma \mathrm{d}W_s,\end{aligned}$$

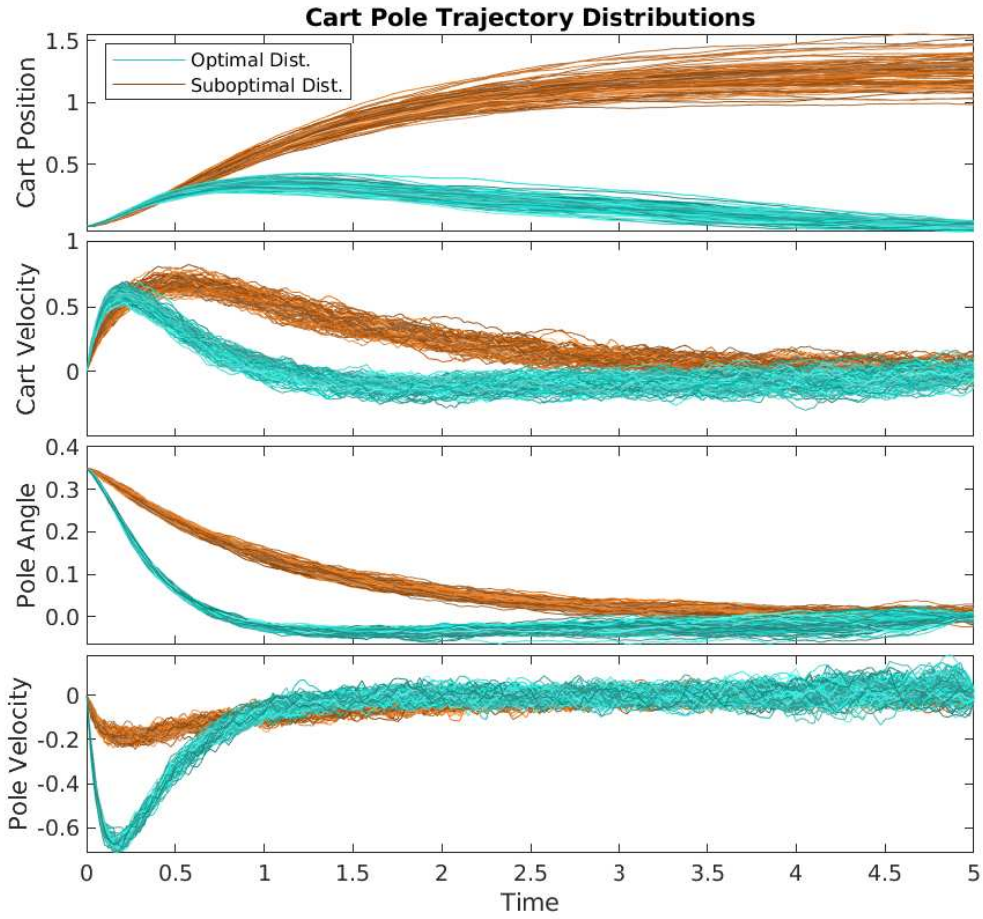
where $a_1, a_2, a_3, a_4, a_5, a_6, b_1, b_2$ are constant parameters and $x_0 = [0, 0, \pi/9, 0]^\top$. For the suboptimal sampling distribution we selected a discrete time approximation of the time-invariant feedback policy

$$K_s^{\text{subopt}} = \left(A + B \begin{bmatrix} 0 & 0 & k_1 & k_2 \end{bmatrix} \right) X_s,$$

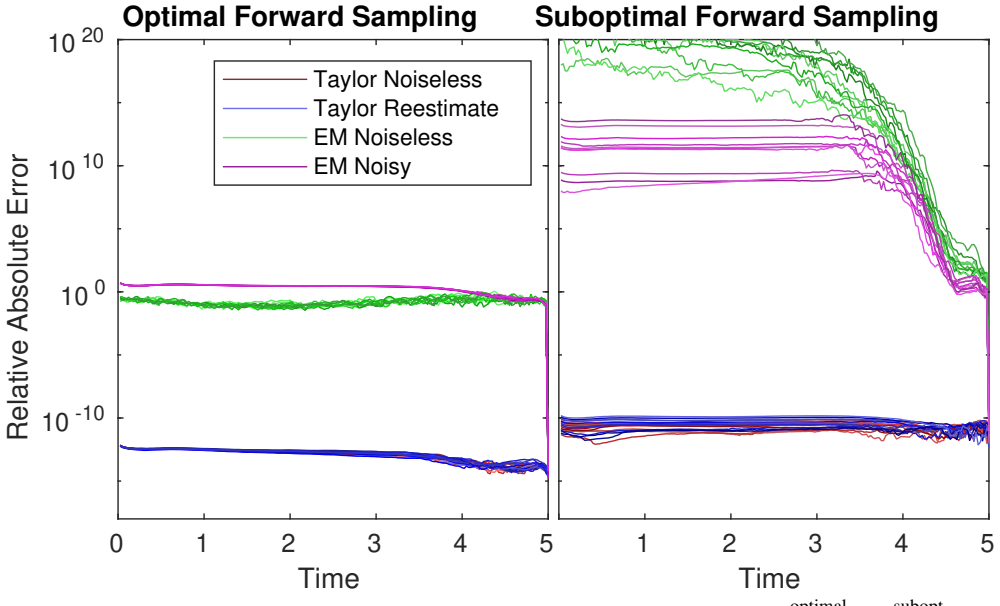
where k_1, k_2 are constant parameters. The optimal policy is found through the solution of the associated Riccati equations (distributions visualized in Figure 4.9(a)).

The value function model for $\widetilde{V}_i$ used Chebyshev functions of degree 2 and lower (15 basis functions). The RAE approximations (4.69) are visualized in Figure 4.9(b) where $\widetilde{\mathcal{C}}_i := \widetilde{\mathcal{C}}_i^1 \times \widetilde{\mathcal{C}}_i^2 \times \widetilde{\mathcal{C}}_i^3 \times \widetilde{\mathcal{C}}_i^4$ and each $\widetilde{\mathcal{C}}_i^j$ is defined similarly to (4.68) based on the mean and standard deviation of the optimal trajectories in each of the 4 dimensions.

As predicted by the error analysis, since this is an LQR problem and the value function is in the class of quadratic functions, the Taylor expansion-based estimators are able to produce approximations of the value function with accuracy near machine precision



(a) Trajectory distributions for the two sampling conditions (K_i^{optimal} / K_i^{subopt})



(b) Relative absolute error (4.69) for the two sampling conditions (K_i^{optimal} / K_i^{subopt})

Figure 4.9: Comparing the accuracy of the estimators on a 4-dimensional LQR approximation of cart-pole balancing system.

for both conditions. For the suboptimal forward sampling condition the EM estimators diverge quickly during the backward pass. For the optimal forward sampling condition, corresponding to the on-policy estimation, the EM estimators perform mediocre compared to the value function's variance and their error is still several orders of magnitudes higher than the Taylor estimators.

These results confirm that the proposed estimators are able to achieve near machine-precision performance on the most common problem in stochastic optimal control. Further, they confirm that utilizing the second-order derivatives of the value function is crucial for Girsanov-inspired off-policy estimator schemes, contrary to what naïve application of the theory would suggest.

4.6 DT-FBSDE Iterative Method

Now that we have both a method to approximate the value function and a method to generate a policy from this value function approximation, we can discuss how these methods can be combined to create an iterative method. We start with any initial drift $\{K_i^0\}$ and sample a forward pass to produce a distribution over the process $\{X_i\}$. A good choice for the initial target policy is something similar to the initial drift, since we have established that on-policy estimators are very accurate. Using this target policy $\{\mu_i^0\}$, we perform a backward pass to compute the initial value function approximation $\{\widetilde{V}_i^1\}$. Using the relation (4.67) we can obtain a new optimized policy $\mu_i^1(x) := \mu_i^*(x; \widetilde{V}_{i+1}^1)$.

We now have the iterative part of the method, assuming we start with a pair of value function approximation and a target policy at iteration j, $\{(\widetilde{V}_i^j, \mu_i^j)\}$, and would like to find an improved pair $\{(\widetilde{V}_i^{j+1}, \mu_i^{j+1})\}$. This new pair can be produced equivalently to the initial iteration, except we must choose some drift $\{K_i^j\}$ for the current forward pass. The improved selection of drift is a primary topic for the following chapter, but for the purposes of demonstration we offer a simple choice. For a given matrix $G_i(x)$, and normally distributed random vector δ_i, independent of W_i^{P}, we can choose the drift to be the dynamics driven

by the target policy with some noise added

$$K_i^j = F_i^{\mu^j} + G_i \delta_i.$$

The benefit of such a choice will be expanded upon in more detail in the next chapter, but the primary benefit is that it maintains a wide distribution about the trajectories driven by the current best policy. Thus, even if the optimal distribution collapses to a set of trajectories with relatively small variance, the value function will still have good approximation accuracy in a region around the optimal distribution.

When the curvature of the value function is relatively tame, the method described above might be sufficient, but executing a new policy $\{\mu_i^j\}$ might drive the system into regions where it is heavily extrapolating the value function computed previously. If the value function is computed, for example, with high dimensional polynomial regression, this extrapolation might make the policy very unstable due to Runge's phenomenon. Since we know the distribution over which the previous iteration's value function was approximated to some degree of accuracy, we can use $\{\widetilde{V}_i^j\}$ on this distribution, then blend to some other value function approximation which is more tame in the extrapolative regions.

One potential choice for the tame extrapolative function we test here is to compute the value function using quadratic polynomials $\{\widetilde{V}_i^{j,\text{quad}}\}$ in addition to higher dimensional polynomial approximations. Then, in (4.67), we use the blended value function

$$\widetilde{V}_i^{j,\text{blend}} := \gamma_i(x)\widetilde{V}_i^j + (1 - \gamma(i,x))\widetilde{V}_i^{j,\text{quad}}, \tag{4.70}$$

where γ_i is a weighing function

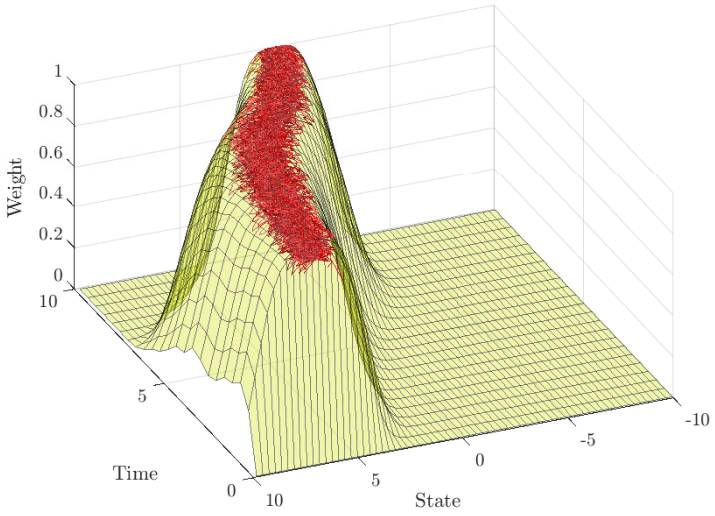
$$\gamma_i(x) \sim \exp\left(-\left(\frac{x - \overline{x}_i}{\beta \sigma_i} \right)^4 \right), \tag{4.71}$$

instead of the original value function itself $\widetilde{V}_i^j$. The constant $\beta > 0$ is tuning parameter

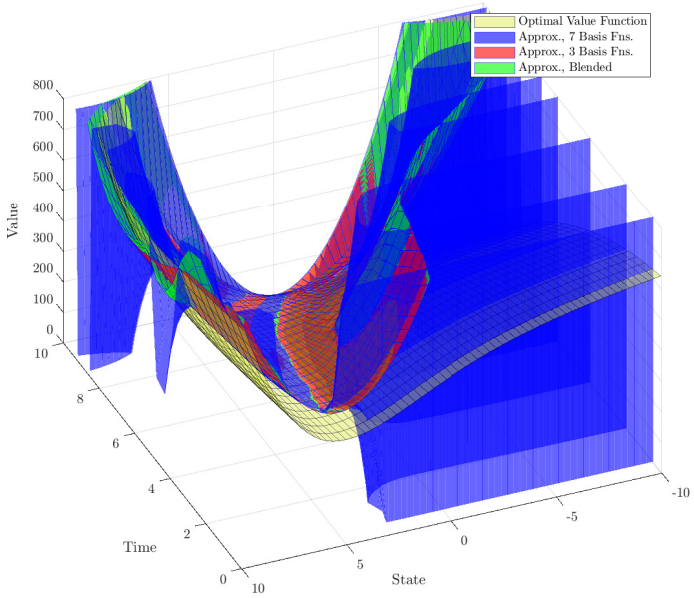
which varies the width of the plateau generated by the weighing function ($\beta = 6$ in our experiments), and thus is a control on how much we will exploit extrapolation of $\widetilde{V}_i^j$ versus how much we will rely on $\widetilde{V}_i^{j,\text{quad}}$ for extrapolated values.

This method is illustrated after 3 iterations of the iterative method in Figure 4.10. Note that, despite the fact that far from the distribution the approximation with 7 basis functions (degree 6 polynomial) has extreme oscillation, the blended approximation acts as a suitable approximate supersolution of the optimal function. By approximate supersolution we mean that, for most of the state space, the blended function overestimates the optimal. This is preferable to underestimating, since the policy optimization is less likely to hallucinate an optimal value basin and attempt to exploit it.

The performance of the iterative method is illustrated in Figure 4.11. To evaluate the solution at each iteration, we test the policy directly by sampling $M = 2000$ trajectories following the policy and computing the average cost. The curves for approximations with a number of basis functions ≥ 5 are all overlapping, suggesting that adding more basis functions seems to have diminishing returns. This is likely due to the fact that the blended method confines the value function approximation to a relatively tight region which, due to smoothness, only requires a relatively low polynomial to represent accurately.



(a) The weighing function (4.71) (in yellow) for the trajectory distribution (in red)



(b) The blended value function (4.70)

Figure 4.10: Illustrating the blended value function method for taming oscillations of the high-degree polynomial value function approximation in extrapolative regions of the state space.

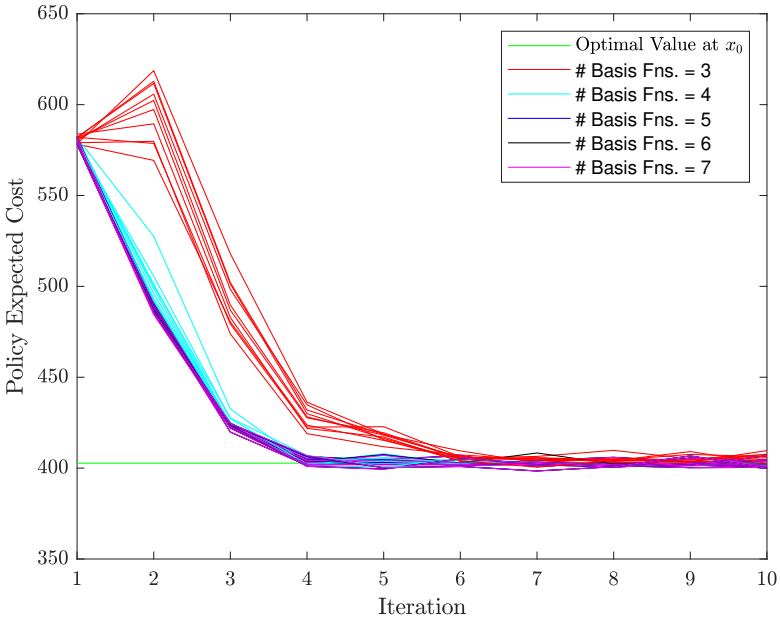


Figure 4.11: Iterative method applied to approximations with varying numbers of basis functions. Ten trials are executed per condition.

4.7 Discrete-Time FBSDE Conclusion

Taylor-based estimators for numerically solving Feynman-Kac FBSDEs have been demonstrated to be significantly more accurate than naïve Euler-Maruyama-based estimators through both error analysis and numerical simulation. These estimators are derived by using higher-order Taylor expansions and following the spirit of the continuous-time Feynman-Kac-Girsanov formulation. Both error analysis and numerical simulation confirm that these estimators have very high accuracy when applied to LQR problems. Further, in simulation, the proposed estimators are orders of magnitude more accurate than the EM estimators in both LQR and nonlinear problems. Using these results, this chapter also proposes a method to use the estimated value function parameters for generating an improved policy. Finally, a full iterative method is proposed and evaluated, demonstrating that the techniques converge to an optimal solution after a few iterations.

CHAPTER 5

SOLUTION OF FBSDES USING MCKEAN-MARKOV BRANCHED SAMPLING

5.1 Introduction

In the previous chapter we proposed methods for improving Feynman-Kac FBSDE methods at the time step-wise level, demonstrating that we can achieve a level of performance that is very accurate on LQR problems, and for non-LQR problems is improved over previous methods. In this chapter we focus on how the methods can be improved at the full time interval level.

Recall from the previous section that one of the primary problems with this method is that extrapolation far from the sampled trajectories has limited utility. Due to the iterative nature of the method, it is challenging to strike a balance between exploiting extrapolation, and controlling the instability this exploitation might introduce. Thus, the trajectory distributions between iterations cannot change significantly. For this reason, the method proposed in Section 4.6 is largely a local trajectory optimization technique, heavily reliant on a good initial sampling drift $\{K_i^0\}$ to converge to the optimal trajectory distribution.

5.1.1 Chapter Overview and Approach

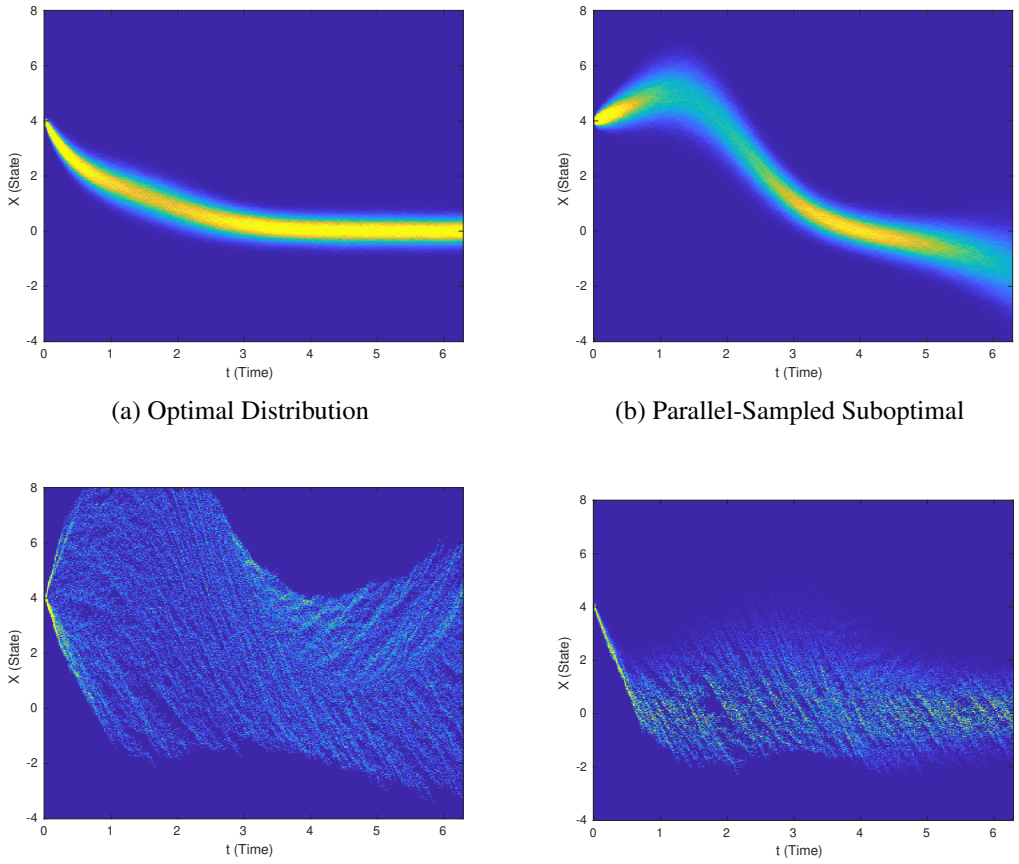
We now motivate an interpretation of the Feynman-Kac theory, which can be utilized for global exploration and local optimization simultaneously. In Section 3.7 we summarized the results of Chapter 3, that Feynman-Kac FBSDE theory requires the choice of three measures: (a) Q, the measure associated with the target policy μ for the value function V^μ, (b) P, the sampling measure used in the forward pass to explore the state space, and (c) R_{i+1}, the weighted measure used in the backward pass to control function approximation accuracy. The contributions of this chapter are first, to change the numerical representation

of the sampling measure P, second, to suggest a method by which we select R_{i+1}, and third, to show how these choices can work together to quickly find a global solution to the SOC problem.

We denote the numerical method of representing P as a collection of independent trajectory samples a *parallel-sampled* distribution. We propose instead to represent the sampling distribution as a *branch-sampled* distribution, where Monte Carlo samples are represented numerically as nodes in a tree. Figure 5.1 briefly summarizes the approach proposed in this chapter, compared to the previous chapter, how these three measures work together to rapidly find the optimal distribution. The density of optimal trajectory distribution we are interested in approximating the value function over is illustrated in Figure 5.1(a). Previously, we would use iterated executions of the parallel-sampled method applied to suboptimal policies, e.g., see the density in Figure 5.1(b). Regardless of whether noise is added to the drift, the policy (when stable) constrains trajectories to a local region. It is easy to see how, even in this simplistic one-dimensional example, little overlap with the optimal trajectory distribution will require several iterations until the iterative method will converge. If we instead choose to broadly sample the state space using space-filling, randomized sampling algorithms like rapidly exploring random trees (RRTs), we can cover the optimal trajectory distribution with Monte Carlo samples, even if they do not come from optimal trajectories themselves, without even having an initial drift process for initialization. As we solve for the value function in the backward pass, we can use these approximations in heuristics to produce the weighted measure R_{i+1}, concentrating function approximation in the regions likely to contain optimal trajectories.

5.2 Repeated Least-Squares Monte Carlo

In this section we present a new perspective on the interpretation of least-squares Monte Carlo (LSMC): that we can treat each optimization independently, with different measures for each application. Recall the LSMC optimization used to obtain the value function



(a) Optimal Distribution

(b) Parallel-Sampled Suboptimal

(c) Branch-Sampled (P)

(d) Branch-Sampled, Weighted (R)

Figure 5.1: Heatmap of different state distributions for a 1-dimensional SOC problem, illustrating how RRT-sampling and weighing can accelerate discovery of the optimal distribution.

approximation,

$$\arg\min_{\alpha\in\mathcal{A}} \mathbf{E}_{\mathsf{P}}[(\widehat{Y}_i - \phi(X_i;\alpha))^2],$$

where $(\Omega, \{\mathcal{F}_i\}_{i=0}^N, \mathsf{P})$ is the discrete-time filtered probability space.[1] For any choice of the estimator $\widehat{Y}_i$ from Chapter 4, the integrand is $\mathcal{F}_{i+1}$-measurable, and thus this optimization is equivalent to

$$\equiv \arg\min_{\alpha\in\mathcal{A}} \mathbf{E}_{\mathsf{P}_{i+1}}[(\widehat{Y}_i - \phi(X_i;\alpha))^2],$$

where P_{i+1} is the restriction of P to the algebra $\mathcal{F}^{i+1}$. Further, since the process $\{\Theta_i\}$, defined as

$$\Theta_{i+1} = \prod_{j=0}^{i} \exp\left(-\frac{1}{2}\|D_j\|^2 + D_j^\top W_j^{\mathsf{P}} \right), \tag{5.1}$$

is a P-martingale (adapted to the filtration and $\mathbf{E}_{\mathsf{P}}[\Theta_{i+k}|\mathcal{F}_i] = \Theta_i$, $\forall k \geq 0$), then we have the change of measure relationship

$$\mathrm{d}\mathsf{Q}_{i+1} = \Theta_{i+1}\,\mathrm{d}\mathsf{P}_{i+1}, \tag{5.2}$$

due to [57, Corollary 10.1.2]. Although these facts do not fundamentally change the result of the optimization, they highlight the insight that LSMC optimizations only depend on the measure P_{i+1} and do not necessarily need to be connected to the same total measure P.

Suppose we define $N-1$ pairs of drift processes and corresponding measures $\{(\{K_j^i\}_{j=0}^i, \mathsf{P}_{i+1})\}_{i=0}^{N-1}$ where each process is $\{K_j^i\}_{j=0}^i := \{K_0^i, K_1^i, \ldots, K_i^i\}$. Applying the discrete-time version of Girsanov's theorem, Lemma 4.2, separately to each pair, we can construct a series of LSMC problems, all of which are valid approximation schemes

[1] We let Q/P refer to the discrete-time FBSDE measures in this chapter instead of the $\widetilde{\mathsf{Q}}/\widetilde{\mathsf{P}}$ used previously.

for the respective value functions $\{V_i^\mu\}_{i=0}^{N-1}$. However, since each measure is constructed using a different drift, we have that each of the measures is no longer a restriction of the others, that is, for $j > 0$, the restriction of P_{i+j} to the algebra $\mathcal{F}_i$, $\mathsf{P}_{i+j}|_{\mathcal{F}_i}$, is no longer equivalent to P_i itself. As discussed in Section 4.2, continuity of the processes (X_s, Y_s, Z_s) is an important insight that the continuous-time problem imparts upon the discrete time problem, so radically changing the measure at each backward step is unlikely to succeed. If we can maintain continuity of the $\{X_i\}$ process, note that the other two processes will follow since the value function is smooth, and the estimators can be computed using only the parameters found in the previous backward step. We use the measure-theoretic notion of absolute continuity, assuming that we choose drift processes which satisfy the relation

Repeated LSMC Continuity Condition

$$\mathsf{P}_{i+1}|_{X_i} \ll \mathsf{P}_i|_{X_i}, \tag{5.3}$$

where the notation $|_{X_i}$ refers to the restriction of the measure to events in the sigma algebra $\sigma(X_i)$. This relationship effectively suggests that the support of the distribution of X_i on P_{i+1} is contained in the support of its distribution on P_i. In the context of Monte Carlo methods, it suggests that every sample of X_i in P_{i+1} coincides with some sample in P_i, although the converse is not necessarily true. Applying this assumption iteratively means that we can affirmatively trace any sample back to the initial state at time $i = 0$.

These points, and how this chapter's approach differs from the approach discussed in Section 4.6, is illustrated in Figure 5.2. We assume that we start with a continuous density over X_i in P_i with support covering the x-axis, the blue dots representing samples from this distribution. The left figure characterizes the previous approach, where the joint distribution $(X_i, \mathcal{K}_i(X_i))$ is determined by a deterministic function, and thus is restricted to a curve.

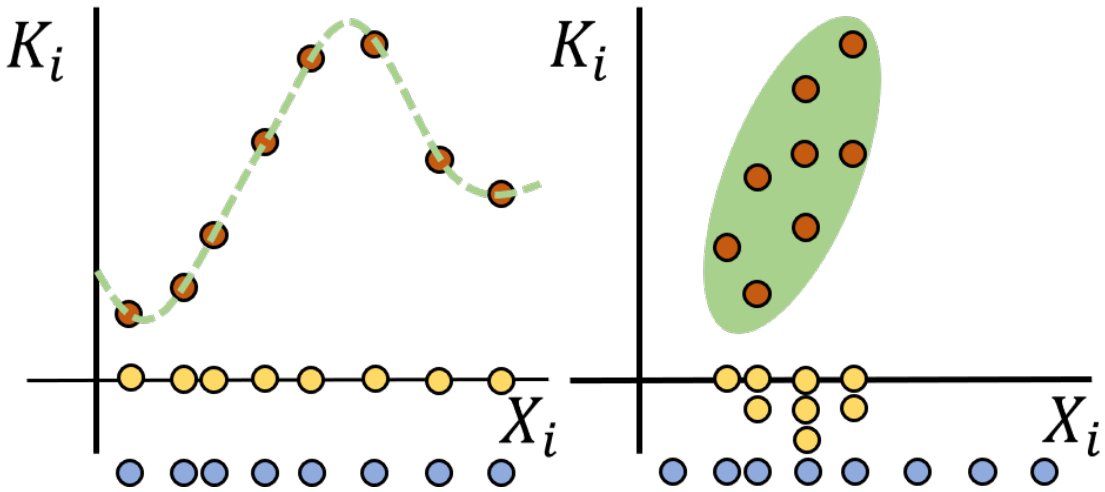


Figure 5.2: Comparing Monte Carlo representations of the joint distribution (X_i, K_i) in P_{i+1}, regions of high probability colored green. The distribution of X_i in P_i is approximated by blue dots, the distribution of X_i in P_{i+1} by yellow dots, and the distribution of (X_i, K_i) in P_{i+1} by reddish-orange dots. **Left:** $\mathsf{P}_{i+1}|_{X_i} \equiv \mathsf{P}_i|_{X_i}$ **Right:** $\mathsf{P}_{i+1}|_{X_i} \ll \mathsf{P}_i|_{X_i}$

If we use the method of adding independent Gaussian noise, the distribution will look similar, expanding to a band about the same curve and the joint samples shifted randomly up and down. On the right, the joint distribution (X_i, K_i) in P_{i+1} is given an arbitrary distribution in the support of the X_i distribution on P_i. The absolute continuity assumption $\mathsf{P}_{i+1}|_{X_i} \ll \mathsf{P}_i|_{X_i}$ holds true for the Monte Carlo approximations, and this can be seen by verifying that every yellow dot corresponds to some blue dot.

5.3 Branching Path LSMC

Now that we have motivated the approach from a theoretical perspective, we discuss more formally the numerical representation and how it forms a sufficient approximation of the underlying theory. For ease of presentation, we begin by presenting, in Section 5.3.1, the construction of a stochastically sampled tree as a data structure used to approximate the FSDE distribution. Next, we demonstrate in Section 5.3.2 how this data structure can be interpreted as a series of McKean-Markov path measures $\{\overrightarrow{\mathsf{P}}_i\}_{i=0}^N$ to approximate the forward sampling distributions. In Section 5.3.4 we discuss how these measures can be used in the backward pass to approximate the BSDE solution by estimating the value function.

5.3.1 Forward SDE Branched Sampling

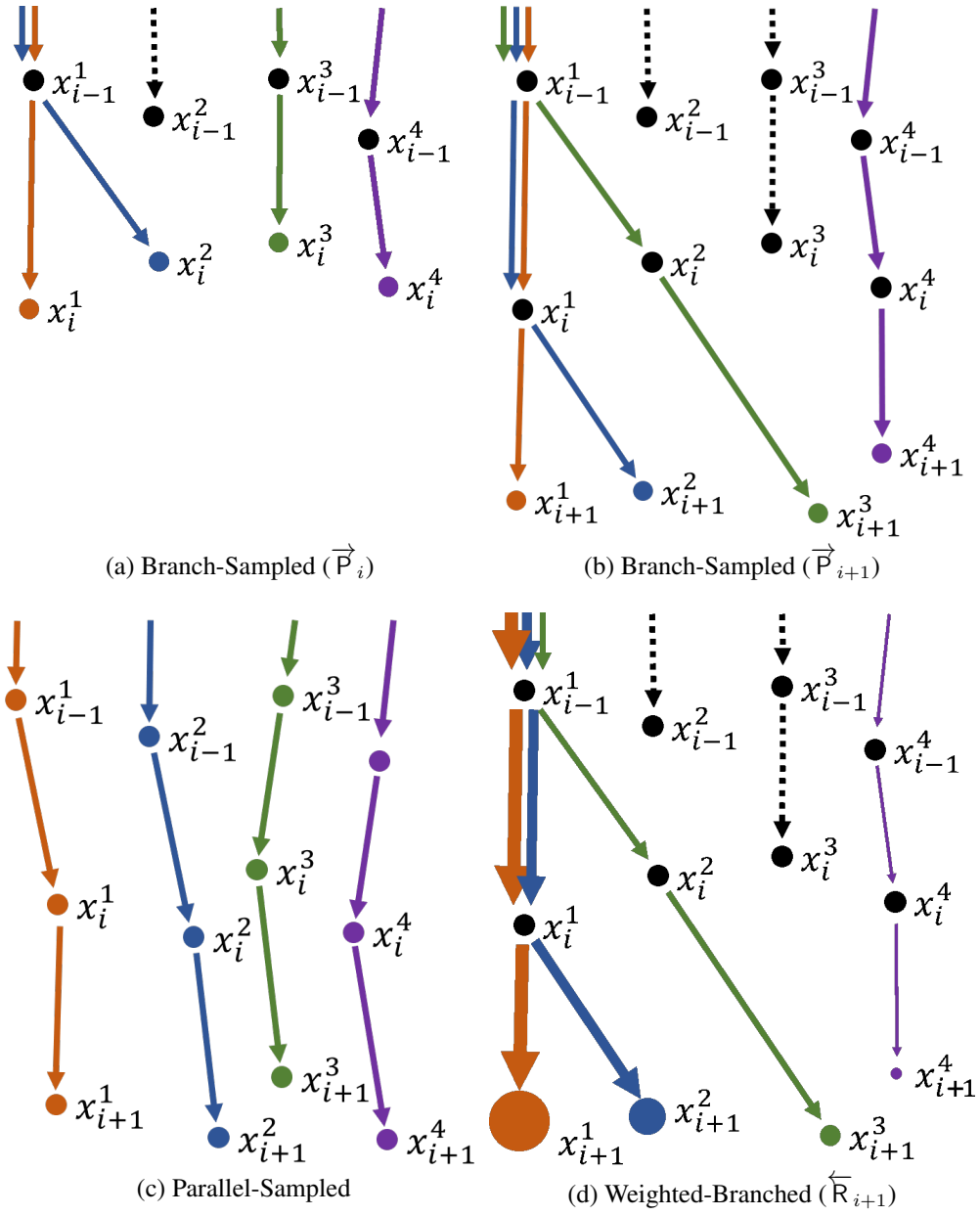
We begin by discussing the construction of the tree data structure $\mathcal{G}$ representing the forward SDE. In this section we only generally describe how edges are added and what data is stored. Later, in Section 5.4.1, we propose a specific methodology for selecting nodes for expansion and choosing the drift value. The tree is initialized with a root node x_0 and is constructed asynchronously as long as new nodes and directed edges are added using the following procedure. Let $x_i^{\text{parent}} \in \mathbb{R}^n$ be a state node in the tree at time i selected $x_i^{\text{parent}} \sim h^{\text{expand}}(\mathcal{G})$ from the tree for expansion, as the parent of a new edge. The drift $k_i \sim h^{\text{drift}}(x_i^{\text{parent}}, \{x_i^k\}_k)$ is sampled from some random function which can depend on both the state and the distribution of nodes at that time. Independently the noise is sampled $w_i \sim \mathcal{N}(0, I_n)$. The child state node is computed using an Euler-Maruyama SDE step approximation of the FSDE (3.15),

$$x_{i+1}^{\text{child}} = x_i^{\text{parent}} + k_i \Delta t + \Sigma_i(x_i^{\text{parent}}) w_i. \tag{5.4}$$

The edge $(x_i^{\text{parent}}, d_i^{\text{data}}, x_{i+1}^{\text{child}})$ is added to the tree $\mathcal{G}$, where $d_i^{\text{data}} = (k_i, w_i, \ldots)$ is the data attached to the edge. A new parent can then be selected for expansion, including selecting the same parent again. Figure 5.3 (a-b) illustrates the branching tree data structure.

5.3.2 McKean-Markov Measure Representation

We approximate the continuous-time sampling distributions with discrete-time McKean-Markov branch sampled paths as presented in [66]. McKean-Markov models encompass a wide variety of process sampling techniques, including particle filters, sequential Monte Carlo, Monte Carlo Markov chain, and others. The difference between Markov processes and McKean-Markov processes is that, for the Markov process $\{X_i\}$, the distribution of X_{i+1} depends on the singular value X_i takes at that time, x_i, and for the McKean-Markov process $\{X_i\}$, the distribution of X_{i+1} depends on the distribution of X_i at that time. From



(a) Branch-Sampled ($\overrightarrow{\mathrm{P}}_i$)

(b) Branch-Sampled ($\overrightarrow{\mathrm{P}}_{i+1}$)

(c) Parallel-Sampled

(d) Weighted-Branched ($\overleftarrow{\mathrm{R}}_{i+1}$)

Figure 5.3: **(a-b)** Illustrating how the branch sampled measures are represented based on the underlying data structure. The colored paths represent the collection of paths representing the respective measure. Dotted lines represent edges in the data structure which are not included in the path measure for that time step. **(c-d)** Comparing the unweighted parallel-sampling method from previous approaches to the proposed weighted and branch-sampled method.

a numerical standpoint, this means that the distribution of particles representing X_{i+1}'s distribution is allowed to depend on some function or method based on the set of particles representing the distribution of X_i. The tree data structure $\mathcal{G}$ represents a series of path measures $\{\overrightarrow{\mathsf{P}}_i\}_{i=0}^N$, each approximating the distribution

$$\overrightarrow{\mathsf{P}}_i \approx \mathsf{P}_i \circ \xi_i^{-1},$$

where ξ_i is the discrete-time random path defined as

$$\xi_i := (X_0, \mathcal{D}_0, X_1, \cdots, \mathcal{D}_{i-1}, X_i),$$

and ξ_i^{-1} is the inverse map from events on the path space to events on the sample space Ω [57, Chapter 3]. We use $\mathcal{D}_i$ to refer to the set of $\mathcal{F}_{i+1}$-measurable random variables associated with the edges of the tree, including K_i and W_i^{P}. The purpose of including other unspecified variables in this set is for the purposes of representing variables used in heuristic methods, presented in more detail later. The empirical measure approximations are defined as

$$\overrightarrow{\mathsf{P}}_i := \frac{1}{M}\sum_{j=1}^{M} \delta_{\xi_i^j}, \tag{5.5}$$

where δ is the Dirac-delta measure acting on sample paths

$$\xi_i^j := (x_{0,i}^j, d_{0,i}^j, x_{1,i}^j, d_{1,i}^j, \ldots, d_{i-1,i}^j, x_{i,i}^j). \tag{5.6}$$

The notation $x_{m,i}^j$ indicates that this element is the sample of a random variable X_m that is the ancestor of sample $x_{i,i}^j$ in the path ξ_i^j, and similarly for the edge variables $\mathcal{D}_m$. Each node in the tree x_i^j (alternatively called a particle) is associated with a unique path ξ_i^j whose final term is $x_{i,i}^j = x_i^j$. Figs. 5.3(a)-(b) illustrate how each colored node at a particular time

step is associated with its matching colored path, and that all of these paths collectively constitute the path measure.

Recall that in this construction there is no requirement for $\overrightarrow{\mathsf{P}}_i$ and $\overrightarrow{\mathsf{P}}_{i+1}$ to agree over the interval $\{0, \ldots, i\}$. This property is illustrated by the fact that, for example, the path ending at x_i^3 in Figure 5.3(a) is represented in $\overrightarrow{\mathsf{P}}_i$ but not represented in $\overrightarrow{\mathsf{P}}_{i+1}$ in Figure 5.3(b).

It can be observed in Figure 5.3(b) that some edges are multiply represented in the distribution. If the drift term K_i were restricted to be a deterministic function of X_i (as is the case in [16, 17, 19]), such a construction would represent an unfaithful characterization of the path distribution because samples of the Brownian process are independent and thus should be sampled as in Figure 5.3(c). However, since K_i itself is permitted to have a distribution, the overlapping of paths is justified as the drift having been selected so as to concentrate the paths in a certain part of the state space. Again, referring back to Figure 5.2, this figure illustrates why parallel sampling is naturally suited for representing deterministic functions and branch sampling is necessary for representing nontrivial joint distributions (X_i, K_i). While a faithful representation of the independent process W_s^{P} might be weakened by this construction, there exist some guarantees about the convergence of such measures with increasing numbers of samples (see, e.g., [66]). We have the following asymptotic basis for this numerical approximation scheme.

Convergence of Expectations of Path Integral Measures

Theorem 5.1. *For any arbitrary function G_{i+1} evaluated on paths ξ_{i+1}, we have the almost surely convergence*

$$\mathbf{E}_{\overrightarrow{\mathsf{P}}_{i+1}}[G_{i+1}(\xi_{i+1})] = \sum_{j=1}^{M} \frac{1}{M} G_{i+1}(\xi_{i+1}^j) \to \mathbf{E}_{\mathsf{P}_{i+1}}[G_{i+1}(\xi_{i+1})], \tag{5.7}$$

as the number of particles $M \to \infty$

Proof. See [66, Section 4.1.2]. □

Thus, our measure approximation converges with large numbers of particles and decreasing time intervals.

5.3.3 Local Entropy Weighting

The methods discussed in the previous section provide the opportunity to employ broad sampling schemes to cover the state space with potential paths. However, fitting a value function broadly to a wide support distribution might degrade the quality of the function approximation since high accuracy of function approximation is more in demand in those parts of the state space in proximity to optimal trajectories. Once forward sampling has been performed and some parts of the value function have been approximated, we can apply a heuristic in which sample paths closer to optimal trajectories are weighted more to concentrate value function approximation accuracy in those regions.

To this end, we propose using a bounded heuristic random variable ρ_{i+1} to produce a new measure R_{i+1}, the weighted counterpart to P_{i+1}. We use the $i+1$ notation, instead of just i, to coincide with the notation from previous sections, where LSMC optimization is performed over the expectation $\mathbf{E}_{\mathsf{R}_{i+1}}$. In order to avoid underdetermination of the regression by concentrating a single or few samples, we select R_{i+1} as

Local-Entropy Optimization Problem

$$\arg\min_{\mathsf{R}_{i+1}} \left\{ \mathbf{E}_{\mathsf{R}_{i+1}}[\rho_{i+1}] + \lambda \mathcal{H}(\mathsf{R}_{i+1} \| \mathsf{P}_{i+1}) \right\}, \tag{5.8}$$

with $\lambda > 0$, a tuning variable, and

$$\mathcal{H}(\mathsf{R}_{i+1} \| \mathsf{P}_{i+1}) = \mathbf{E}_{\mathsf{R}_{i+1}} \left[\log \left(\frac{\mathrm{d}\mathsf{R}_{i+1}}{\mathrm{d}\mathsf{P}_{i+1}} \right) \right], \tag{5.9}$$

the relative entropy of R_{i+1} which takes its minimum value when $\mathsf{R}_{i+1} = \mathsf{P}_{i+1}$, the distribution in which all sampled paths have equal weight. The minimizer of (5.8), which balances between minimizing the value of ρ_{i+1} and minimizing the relative entropy of its induced measure, has a solution of R^*_{i+1} determined as [67, p. 2]

Local-Entropy Optimization Solution

$$\mathrm{d}\mathsf{R}^*_{i+1} = \Theta^{\mathsf{R}|\mathsf{P}}_{i+1}\mathrm{d}\mathsf{P}_{i+1}, \qquad \Theta^{\mathsf{R}|\mathsf{P}}_{i+1} := \frac{\exp(-1/\lambda\rho_{i+1})}{\mathbf{E}_{\mathsf{P}_{i+1}}[\exp(-1/\lambda\rho_{i+1})]}. \tag{5.10}$$

Henceforth, we let R_{i+1} refer to this minimizer R^*_{i+1}. During numerical approximation we can interpret the weights as a *softmin* operation over paths according to this heuristic, a method often used in the deep learning literature [49].

5.3.4 Local-Entropy Least Squares Monte Carlo

To approximate the measure R_{i+1} in Theorem 3.9 we use

$$\theta^j_{i+1} = \exp(-\frac{1}{\lambda}\rho^j_{i+1}). \tag{5.11}$$

The heuristic value is calculated as $\rho^j_{i+1} = \rho_{i+1}(\xi^j_{i+1})$, taking care to exclude $w^j_{i,i+1}$ so that its distribution remains Brownian. Recall that Theorem 3.9 expects that W^{P}_s is Brownian over the interval $[t_i, t_{i+1}]$. Although the inclusion of $x^j_{i+1,i+1}$ in this function might violate this assumption, the amount of bias added is likely minimal because only a single time step of noise is at stake.

In each step of the backward pass, we use $\overleftarrow{\mathsf{R}}_{i+1}$ and value function approximation $\widetilde{V}^{\mu}_{i+1}(x) = \phi(x; \alpha_{i+1})$, parameterized by $\alpha_{i+1} \in \mathcal{A}$, where $\mathcal{A}$ is the parameter space, to approximate the value function at the previous time step $\phi(x; \alpha_i) \approx V^{\mu}_i(x)$, by producing some $\alpha_i \in \mathcal{A}$. We use the weighted LSMC presented in (3.28),

Local-Entropy Weighted LSMC

$$\begin{aligned}\arg\min_{\alpha\in\mathcal{A}} \mathbf{E}_{\mathsf{R}_{i+1}}[(\widehat{Y}_i - \phi(X_i;\alpha))^2] &= \arg\min_{\alpha\in\mathcal{A}} \mathbf{E}_{\mathsf{P}_{i+1}}[\Theta^{\mathsf{R}|\mathsf{P}}_{i+1}(\widehat{Y}_i - \phi(X_i;\alpha))^2] \\ &\approx \arg\min_{\alpha\in\mathcal{A}} \mathbf{E}_{\overrightarrow{\mathsf{P}}_{i+1}}[\Theta^{\mathsf{R}|\mathsf{P}}_{i+1}(\widehat{Y}_i - \phi(X_i;\alpha))^2] \\ &= \arg\min_{\alpha\in\mathcal{A}} \sum_{k=1}^{M} \frac{\theta_i^k}{M}(\widehat{y}_i^k - \phi(x_i^k;\alpha))^2 =: \alpha_i^*, \quad (5.12)\end{aligned}$$

to compute parameters for the local-entropy-weighted, path-integral LSMC approximation $\phi(\cdot;\alpha_i^*) \approx V_i^\mu(\cdot)$.[2]

The novelty of this method over classic LSMC [25], developed for parallel-sampled paths, comes from (a) the observation that solving the FBSDE problem over a changing set of measures $\{\mathsf{P}_i\}$ validates the choice of branch-sampled path distributions; (b) we can weigh regression points using a heuristic that acts on the entire path history, not just the immediate states; and (c) weighing as in (5.11) has a particular interpretation as the selection of a measure with desirable properties for robustness using (5.8).

5.4 Forward-Backward RRT

In this section we present a novel algorithm to which we refer as FBRRT since it presents forward-backward rapidly exploring random trees for solving SDEs. The FBRRT algorithm is a particular numerical application of the generalized theory presented in Section 5.3. The ultimate goal of the FBRRT algorithm is to produce the set of parameters $\{\alpha_i\}_{i=1}^N$ which approximate the optimal value function as $\phi(\cdot;\alpha_i) \approx V_i^*(\cdot)$. This is achieved by, first, generating a forward pass producing a graph representation $\mathcal{G}$ of the path measures

[2]For the remainder of this chapter, we use as the estimator for $\widehat{Y}_i$ the Euler-Maruyama noiseless estimator. In evaluation this allows for more direct comparisons to the methods in [17], since the estimator is the same. We leave the integration of Taylor estimators into the methodology in this chapter to future applications.

$\{\overrightarrow{\mathsf{P}}_i\}_{i=1}^N$. Next, the backward pass uses $\mathcal{G}$, μ_i, and ρ_{i+1} to produce α_i, backwards in time. For this method, we choose the Hamiltonian-based policy (4.61) as the target policy[3]

$$\mu_i(x) \in \arg\min_{u \in U} \{L_i(x,u) + F_i(x,u)^\top \partial_x \widetilde{V}_i(x)\}.$$

The policy cost $\overline{J}_k$ associated with a set of parameterized policies is evaluated by sampling a parallel-sampled set of trajectories and computing the mean cost $\mathbf{E}[\sum_{i=0}^{N-1} L_i^\mu + g(X_N)]$. At the end of each iteration, nodes with high heuristic value ρ_{i+1} are pruned from the tree $\mathcal{G}$, and new nodes are added in the forward pass in the next iteration. This outer loop of the FBRRT algorithm is summarized in Algorithm 3.

Algorithm 3 Forward-Backward RRT

procedure FBRRT(x_0)
  $\widetilde{\mathcal{G}}$.init(ξ_0)
  for $k = 1, \cdots, N_{\text{iter}}$ **do**
    $\mathcal{G} \leftarrow$ FORWARDPASS($\widetilde{\mathcal{G}}, (\alpha_i)_i$) ▷ Generate tree which represents $\{\overrightarrow{\mathsf{P}}_i\}_i$
    $(\alpha_i)_i \leftarrow$ BACKWARDPASS($\mathcal{G}$) ▷ Approximate value functions $\{V(\cdot;\alpha_i)\}_i$
    $\overline{J}_k \leftarrow$ POLICYCOST($x_0, (\alpha_i)_i$) ▷ Evaluate computed policy $\{\mu_i(\cdot;\alpha_{i+1})\}_i$
    $\widetilde{\mathcal{G}} \leftarrow$ ERODE($\mathcal{G}, (\alpha_i)_i$) ▷ Prune tree to remove suboptimal paths
  end for
  return $(\alpha_i)_i$
end procedure

5.4.1 Kinodynamic RRT Forward Sampling

In general, we desire sampling methods that seek to explore the whole state space, thus increasing the likelihood of sampling in the proximity of optimal trajectories. For this reason, we chose methods inspired by kinodynamic RRT, proposed in [51]. The selection procedure for this method ensures that the distribution of the chosen particles is more uniformly distributed in a user-supplied region of interest $\mathcal{X}^{\text{roi}} \subseteq \mathbb{R}^n$, more likely to select particles which explore empty space, and less likely to oversample dense clusters of particles.

[3] As discussed in the previous footnote, we use a policy comparable with [17].

With some probability $\varepsilon_i^{\text{rrt}} \in [0, 1]$ we choose the RRT sampling procedure, but otherwise we choose a particle uniformly from $\{x_i^j\}_{j=1}^M$, each particle having equal weight. This ensures dense particle clusters will still receive more attention. Thus, the choice of the parameter $\varepsilon_i^{\text{rrt}}$ balances exploring the state space against refining the area around the current distribution.

For choosing the drift values, that is, those sampled from the distribution h left unspecified in Section 5.3.1, we again choose a random combination of exploration and exploitation. For exploitation we choose

$$K_i = F_i(X_i, \mu_i(X_i; \alpha_i)). \tag{5.13}$$

For exploration we choose

$$K_i = F_i(X_i, u^{\text{rand}}). \tag{5.14}$$

where the control is sampled randomly from a user-supplied set $u^{\text{rand}} \sim U^{\text{rand}}$. For example, for minimum fuel (L_1) problems where the control is bounded as $u \in [-1, 1]$ and the running cost is $L = |u|$, we select $U^{\text{rand}} = \{-1, 0, 1\}$ because the policy is guaranteed to only return values in this discrete set.

Algorithm 4 summarizes the implementation of the RRT-based sampling procedure, producing the forward sampling tree $\mathcal{G}$. The algorithm takes as input any tree with width $\widetilde{M}$ and adds nodes at each depth until the width is M, the parameter indicating the desired width. In the first iteration there are no value function estimate parameters available to exploit, so we set $\varepsilon^{\text{rrt}} = 1$ to maximize exploration using the RRT sampling.

5.4.2 Path-Integral Dynamic Programming Heuristic

We now propose a heuristic design choice for the backward pass weighting variables ρ_{i+1}, and justify their choice with theoretical analysis. A good heuristic will give high weights to

Algorithm 4 RRT Branched-Sampling

procedure FORWARDPASS($\mathcal{G}, (\alpha_1, \ldots, \alpha_N)$)
  for $k = \widetilde{M} + 1, \cdots, M$ **do** ▷ Add node each loop
    for $i = 0, \cdots, N-1$ **do** ▷ For each time step
      $\{x_i^j\}_j \leftarrow \mathcal{G}$.nodesAtTime$(i)$
      if $\varepsilon^{\text{rrt}} > \kappa^{\text{rrt}} \sim$ Uniform$([0,1])$ **then**
        $x_i^{\text{rand}} \sim$ Uniform$(\mathcal{X}^{\text{roi}})$
        $(x_i^{\text{near}}, j^{\text{near}}) \leftarrow$ Nearest$(\{x_i^j\}_j, x_i^{\text{rand}})$
      else
        $(x_i^{\text{near}}, j^{\text{near}}) \sim$ Uniform$(\{x_i^j\}_j)$
      end if ▷ j^{near} is index of selected node
      if $\varepsilon^{\text{opt}} > \kappa^{\text{opt}} \sim$ Uniform$([0,1])$ **then**
        $u_i \leftarrow \mu_i(x_i^{\text{near}}; \alpha_{i+1})$
      else
        $u_i \sim U^{\text{rand}}$
      end if
      $k_i \leftarrow F_i(x_i^{\text{near}}, u_i)$
      $w_i \sim \mathcal{N}(0, I_n)$
      $x_{i+1}^{\text{next}} \leftarrow x_i^{\text{near}} + k_i + \Sigma_i(x_i^{\text{near}}) w_i$
      $j^{\text{next}} \leftarrow \mathcal{G}$.addEdge$(i, j^{\text{near}}, (x_i^{\text{near}}, k_i, x_{i+1}^{\text{next}}))$
      $\overrightarrow{L}_{0:i-1} \leftarrow \mathcal{G}$.getRunCost$(i-1, j^{\text{near}})$
      $\overrightarrow{L}_{0:i} \leftarrow \overrightarrow{L}_{0:i-1} + L_i(x_i^{\text{near}}, u_i)$
      $\mathcal{G}$.setRunCost$(i, j^{\text{next}}, \overrightarrow{L}_{0:i})$
    end for
  end for
  return $\mathcal{G}$
end procedure

paths likely to have low value over the whole interval. Thus, in the middle of the interval we care both about the current running cost and the expected cost. A dynamic programming principle result[4] indicates that

$$V_0^*(x_0) = \min_{\{u_j\}} E_{\mathsf{P}_{i+1}^u}[\sum_{j=0}^{i} L_j(X_j, u_j) + V_{i+1}^*(X_{i+1})],$$

where $\{u_j\}$ is any control process in $\mathcal{U}$ on the interval $j = 0, \ldots, i$ and P_{i+1}^u is the measure produced by the drift $\{K_j = F_j(X_j, u_j)\}$.

Optimal Trajectory Distribution Heuristic

Following this minimization, we choose the heuristic to be

$$\rho_{i+1} = \sum_{j=0}^{i} L_i(X_i, u_i) + \widetilde{V}_{i+1}^{\mu}(X_{i+1}), \tag{5.15}$$

where $\{u_j\}$ is chosen identically to how the control for the drift is produced.

The running cost is computed in the forward sampling in line 21 of Algorithm 4.

Algorithm 5 details the implementation of the backward pass with local entropy weighting. Line 18 does not, theoretically, have an effect on the optimization, since it will come out of the exponential as a constant multiplier, but it has the potential to improve the numerical conditioning of the exponential function computation as discussed in [49, Chapter 5, equation (6.33)]. The λ value is, in general, a parameter which must be selected by the user. For some problems we choose to search over a series of possible λ parameters, evaluating each one with a backward pass and using the one that produces the smallest expected cost over a batch of trajectory rollouts executing the computed policy.

[4] In continuous-time, this is following directly from [6, Chapter 4, Corollary 7.2].

Algorithm 5 Local Entropy Weighted LSMC Backward Pass

procedure BACKWARDPASS($\mathcal{G}$)
  $\{\xi_N^j\}_j \leftarrow \mathcal{G}$.pathsAtTime($N$)
  $\{x_N^j\}_j \leftarrow \{\xi_N^j\}_j$
  $y_N \leftarrow [g(x_N^1) \ \cdots \ g(x_N^M)]^\top$
  $\alpha_N \leftarrow \arg\min_\alpha \sum_j \theta_N(\widehat{y}_N^j - \Phi(x_N^j)\alpha)^2$
  for $i = N-1, \cdots, 1$ **do** ▷ For each time step
    $\{\xi_{i+1}^j\}_j \leftarrow \mathcal{G}$.pathsAtTime($i+1$)
    for $j = 1, \cdots, M$ **do** ▷ For each path
      $(x_i^j, k_i^j, x_{i+1}^j) \leftarrow \xi_{i+1}^j$ ▷ $x_i^j = x_{i,i+1}^j$, etc.
      $y_{i+1}^j \leftarrow \Phi(x_{i+1}^j)\alpha_{i+1}$ ▷ (3.26)
      $z_{i+1}^j \leftarrow \Sigma_i^\top(x_i^j)\partial_x\Phi(x_{i+1}^j)\alpha_{i+1}$
      $\mu_i^j \leftarrow \mu_i(x_i^j; \alpha_{i+1})$ ▷ Hamiltonian-based Policy (4.61)
      $d_i^j \leftarrow \Sigma_i^{-1}(x_i^j)(F_i(x_i^j, \mu_i^j) - k_i^j)$
      $\widehat{y}_i^j \leftarrow y_{i+1}^j + (L_i(x_i^j, \mu_i^j) + z_{i+1}^{j\top} d_i^j)$ ▷ EM Noiseless Estimator
      $\overrightarrow{L}_{0:i} \leftarrow \mathcal{G}$.getRunCost($i, j$)
      $\rho_{i+1}^j \leftarrow y_{i+1}^j + \overrightarrow{L}_{0:i}$ ▷ (5.15)
    end for
    $\rho_{i+1} \leftarrow \rho_{i+1} - \min_j\{\rho_{i+1}^j\}$ ▷ exp conditioning
    $\theta_{i+1} \leftarrow \exp(-1/\lambda\rho_{i+1})$ ▷ (5.10)
    $\alpha_i \leftarrow \arg\min_\alpha \sum_j \theta_{i+1}^j(\widehat{y}_i^j - \Phi(x_i^j)\alpha)^2$ ▷ (3.11)
  end for
  return $(\alpha_1, \ldots, \alpha_N)$
end procedure

5.4.3 Path Integral Erode

After the backward pass of the algorithm, we have updated approximations of the value function $\{\widetilde{V}_i^*(\cdot) = \phi(\cdot;\alpha_i)\}$ along with the tree $\mathcal{G}$ which represents the forward sampling path measures $\{\overrightarrow{\mathsf{P}}_{i+1}\}$. To improve our approximation, we can use our value function estimates to create a new tree $\mathcal{G}'$ with new forward sampling measures $\{\overrightarrow{\mathsf{P}}'_{i+1}\}$ via the heuristic ρ_{i+1}.

We have found experimentally that sampling a new tree from scratch is both wasteful and shows signs of catastrophic forgetting. That is, the following backward pass performs worse, since it has lost data samples which were important for forming good function estimates. On the other hand, simply adding more samples to the current tree can prove to be unsustainable in the long run. To keep the time complexity constant between iterations, we propose bounding the number of samples at each time step. After each backward pass we remove as many samples as added in the forward pass, "eroding" the tree before the forward pass "expands" it.

We begin at the end of the trajectory $i = N$ and remove the nodes $\{x_N^j\}_{j=1}^M$ with highest ρ_N^j value until there are only $\widetilde{M}$ nodes left at depth N. We proceed in a similar fashion backwards down the tree, removing nodes with with high ρ_i^j value. However, due to the tree structure of the path measures, if we remove nodes which have children we disconnect the paths and ruin the assumed structure. Thus, we only remove nodes which have no children. The implementation of this algorithm is detailed in Algorithm 6.

5.4.4 Function Approximation

In our implementation of the FBRRT algorithm, the value function is represented by 2nd order multivariate Chebyshev polynomials. Specifically, we use all products of the basis functions $\bigcup_{j=1}^n \{1, x_j, 2x_j^2 - 1\}$ with polynomial degree 2 or lower, namely,

$$\Phi(x) := (1, x_1, \cdots, x_n, 2x_1^2 - 1, \cdots, 2x_n^2 - 1, x_1x_2, \cdots, x_1x_n, x_2x_3, \cdots, x_2x_n, \cdots, x_{n-1}x_n).$$

Algorithm 6 Path Integral Erode

procedure ERODE($\mathcal{G}, (\alpha_i)_i$)
  for $i = N, \cdots, 1$ **do** ▷ For each time step
    $\{\rho_i^j\}_j \leftarrow \mathcal{G}$.getHeuristics($t_i$)
    for all $j' \in$ sortDescending($\{\rho_i^j\}_j$) **do**
      if $\mathcal{G}$.hasNoChildren($x_i^{j'}$) **then**
        $\mathcal{G}$.removeParentEdge($x_i^{j'}$)
        $\mathcal{G}$.removeNode($x_i^{j'}$)
      end if
      if $\mathcal{G}$.numNodes(t_i) $= \widetilde{M}$ **then**
        break
      end if
    end for
  end for
  return $\mathcal{G}$
end procedure

For better conditioning, points are first normalized to the interval $[-1, 1]^n$ based on a parameterized region of interest, so the basis functions are $\Phi(\ldots, (x_j - a_j^{\text{offset}})/a_j^{\text{scale}}, \ldots)$.

5.5 Numerical Results

We evaluated the FBRRT algorithm by applying it to four nonlinear stochastic optimal control problems: (a) L_1 double integrator ($n = 2$), (b) L_1 inverted pendulum ($n = 2$), (c), L_1 double inverted pendulum ($n = 4$), and (d) intersection reachability problem ($n = 5$). For the L_1/min fuel problems we used a running cost of $\ell = a|u|$, $a > 0$, and for the reachability problem we used $\ell = 0$. The number of particles per time step is $M = 1,024$ for the two-dimensional problem up to $M = 4 \times 1024$ for the five-dimensional intersection reachability problem. and $M = 3 \times 1,024$ for the four-dimensional double inverted pendulum problem. The number of time steps was set to $N = 80$ for the double inverted pendulum problem and $N = 64$ for the other problems. For all problems the control input was restricted in the set $\mathcal{U} = [-1, 1]$. The erode particle number $\widetilde{M}$ was set to $(3/4)M$ for the double inverted pendulum and $(1/2)M$ for all other problems. We implemented the FBRRT algorithm and all examples in Matlab 2019b and ran them on a

computer with an Intel G4560 3.50GHz processor and 8GB RAM.

5.5.1 L_1 Double Integrator

In order to compare the proposed FBRRT algorithm to the parallel sampled techniques in [17], which we denote below as parallel-sampled FBSDE, considered the double integrator system with

$$\begin{aligned} \mathrm{d}X_s &\equiv \begin{bmatrix} \mathrm{d}X_s^{(1)} \\ \\ \mathrm{d}X_s^{(2)} \end{bmatrix} \\ &= \begin{bmatrix} X_s^{(2)} \\ \\ u \end{bmatrix} \mathrm{d}s + \begin{bmatrix} 0.01 & 0 \\ \\ 0 & 0.1 \end{bmatrix} \begin{bmatrix} \mathrm{d}W_s^{(1)} \\ \\ \mathrm{d}W_s^{(2)} \end{bmatrix}, \end{aligned} \tag{5.16}$$

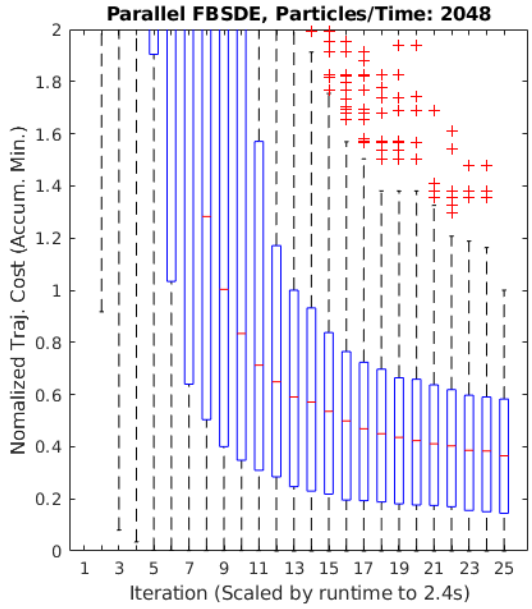
with L_1 running cost, i.e.,

$$V^*(t,x) = \inf_{u_{[t,T]}} \mathbf{E}_{\mathbf{Q}}^{t,x} \Big[\int_t^T c_0 |u_s| \mathrm{d}s + \sum_{j=1}^{n} c_j (X_T^{(j)})^2 \Big], \tag{5.17}$$

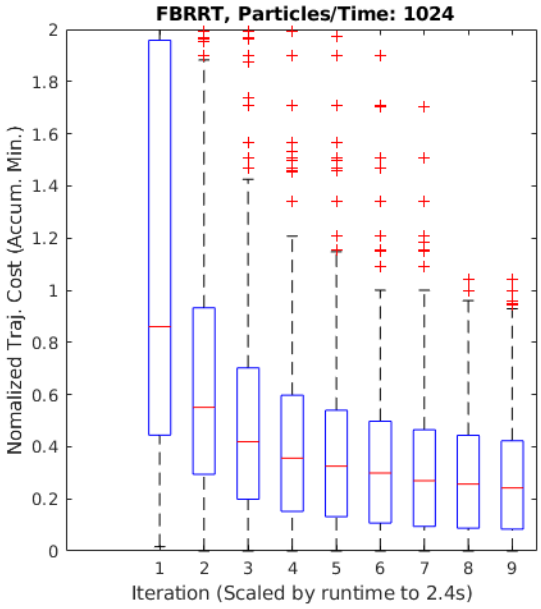
where c_0, c_1, c_2 are scalar parameters. When the system starts with positive position and velocity, the optimal policy is to decelerate to a negative velocity, coast for a period of time so that fuel is not used, then accelerate to reach the origin.

As shown in Figs. 5.4(c)-(d), the parallel-sampled FBSDE takes a significant number of iterations to begin converging to the near optimal policy, while the proposed method produces a near-optimal policy at the first iteration. The algorithm converges to the optimal policy.

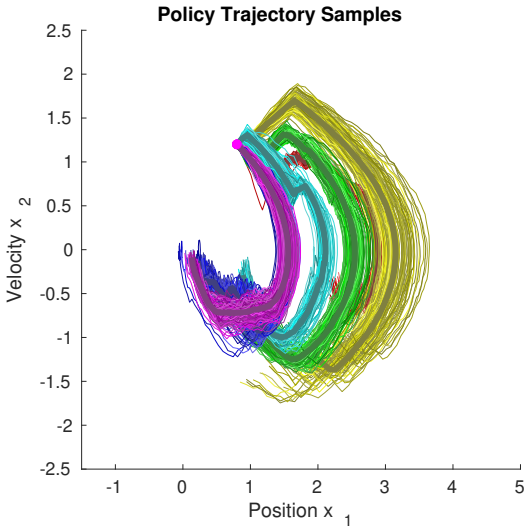
We also compared the convergence speed and robustness of the two methods by randomly sampling different starting states and evaluated their relative performance over a number of trials. For each of 30 random initial states x_0 we ran 20 trials of each method for



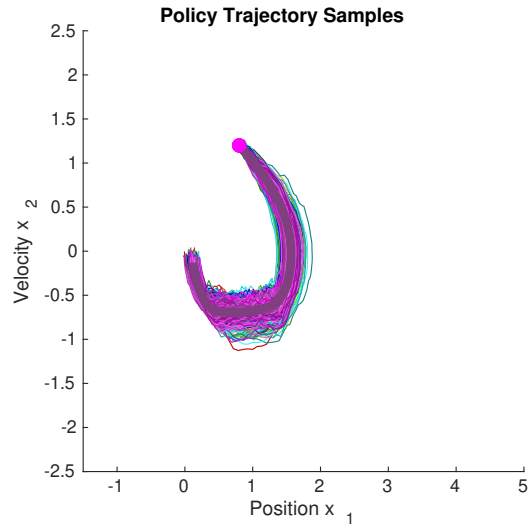
(a) Parallel-sampled FBSDE [17]



(b) FBRRT



(c) Trajectory samples for parallel-sampled FBSDE



(d) Trajectory samples for FBRRT

Figure 5.4: **(a-b)** Comparison of parallel-sampled FBSDE [17] and FBRRT for the L_1 double integrator problem for random initial states. Expected trajectory costs for the computed policies are normalized across different initial conditions. **(c-d)** Trajectory samples from policies generated after the first 6 iterations. The first iteration is colored red, followed by yellow, green, cyan, dark blue, and magenta. Thick lines are mean trajectories.

a number of iterations, each iteration producing an expected cost for the computed policy. We normalized the final costs across the initial states by dividing all costs for a particular initial state by the largest cost obtained across both methods. For each iteration, we assign the value of the accumulated minimum value across previous iterations for that trial, i.e., the value is the current best cost after running that many iterations, regardless of the current cost. We aggregated these values across initial states and trials into the box plots in Figure 5.4. Since the FBRRT is significantly slower than the FBSDE per iteration due to the nearest neighbors calculation, we scale each iteration by the runtime. Note that every iteration of FBRRT after the first one requires approximately half the runtime, since only half of the eroded tree needs resampling. In summary, the FBRRT converges faster and in fewer iterations than FBSDE, and does so with half as many particle samples.

5.5.2 L_1 Inverted Pendulum

The L_1 inverted pendulum problem attempts to rotate a bar to balance upright using torque control, but do so with minimal effort. The L_1 inverted pendulum problem is also a two-dimensional problem, but with nonlinear dynamics, as follows

$$f = \begin{bmatrix} x_2 \\ \\ a_1 x_2 + a_2 \sin x_1 + a_3 u \end{bmatrix},$$

where a_1, a_2 and a_3 are constants and $\sigma_t \equiv \mathsf{diag}(0.04, 0.4)$. This problem is further complicated by the fact that the goal is an unstable equilibrium. Despite this, the algorithm is still able to produce a decent policy after only one iteration, converging within 6 iterations (Figure 5.5). In [17], it was proposed for the parallel-sampled FBSDE algorithm to only resample a small number of paths at each iteration. Although this modification helped the algorithm from experiencing divergence, convergence was significantly slow (55 iterations) and such a technique is likely to be sensitive to entrapment in local minima. Since FBRRT

samples broadly from the beginning, it is unlikely to be affected by such problems.

Of significant interest is the fact that although the final leg of the trajectory has relatively small particle density from the first iteration of the forward sampling, it appears that particles following paths mostly different in the first half were able to help inform the policy which ended up following an entirely separate path. Specifically, note that the blue line begins by swinging backwards, then switching back to swing all the way around to the goal. However, looking more closely, note that all of the particles sampled near the origin come from paths which did not swing backwards much, if at all. Despite this, those particles have green hue, indicating that they significantly helped contribute to the shape of the policy in this region. This result demonstrates a significant benefit of our algorithm, namely, that it can incorporate into the policy information sampled from highly dissimilar paths in the tree.

5.5.3 L_1 Double Inverted Pendulum

In order to study the proposed FBRRT algorithm on a highly nonlinear system in higher dimensions, consider the double inverted pendulum with the state space dimension $n = 4$ presented in [68], but with added damping friction to the joints. Thus, the dynamics are in the form of

$$\mathrm{d}X_s = f(X_s, u_s)\,\mathrm{d}s + \begin{bmatrix} 0.03 & 0 & 0 & 0 \\ 0 & 0.03 & 0 & 0 \\ 0 & 0 & 0.18 & 0 \\ 0 & 0 & 0 & 0.18 \end{bmatrix} \begin{bmatrix} \mathrm{d}W_s^{(1)} \\ \mathrm{d}W_s^{(2)} \\ \mathrm{d}W_s^{(3)} \\ \mathrm{d}W_s^{(4)} \end{bmatrix}, \tag{5.18}$$

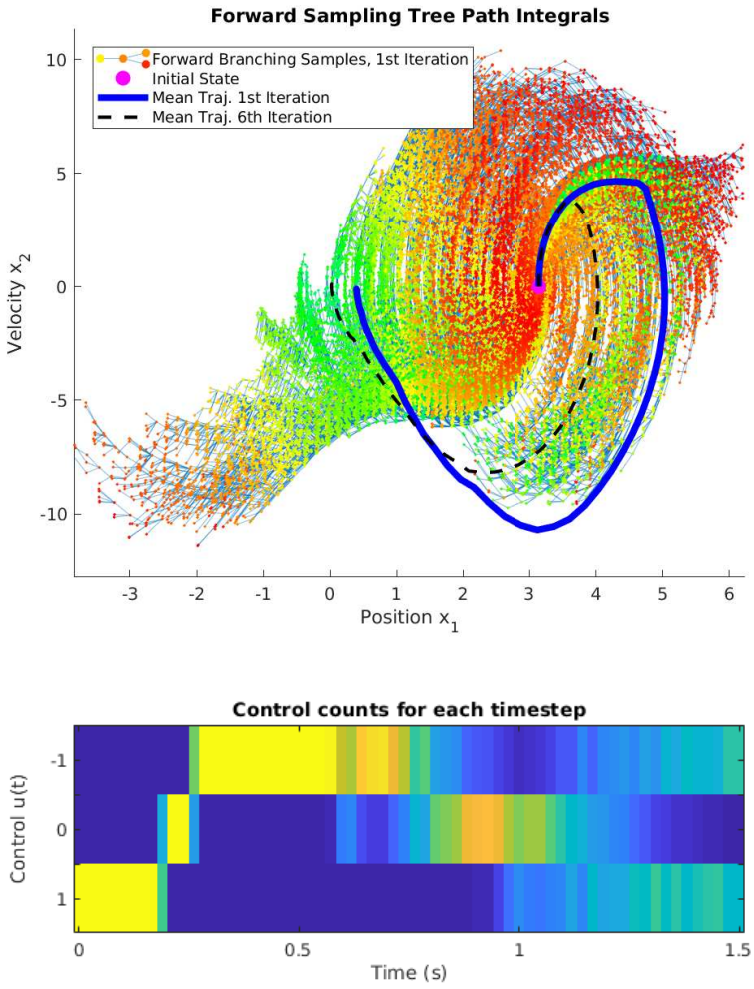


Figure 5.5: Forward sampling tree for the first iteration of the L_1 inverted pendulum problem. Hue corresponds to the path-integral heuristic ρ_i used for weighing particles in the backward pass and for pruning the tree (green values are smaller). The blue and black dashed lines are the mean of trajectory rollouts, following the policies computed at the end of the 1st and 6th iterations, respectively. Control counts are based on trajectory rollouts of the 6th iteration policy computed by FBRRT. The hue of each rectangle indicates the relative frequency of each control signal in $\{-1, 0, 1\}$ for each time step.

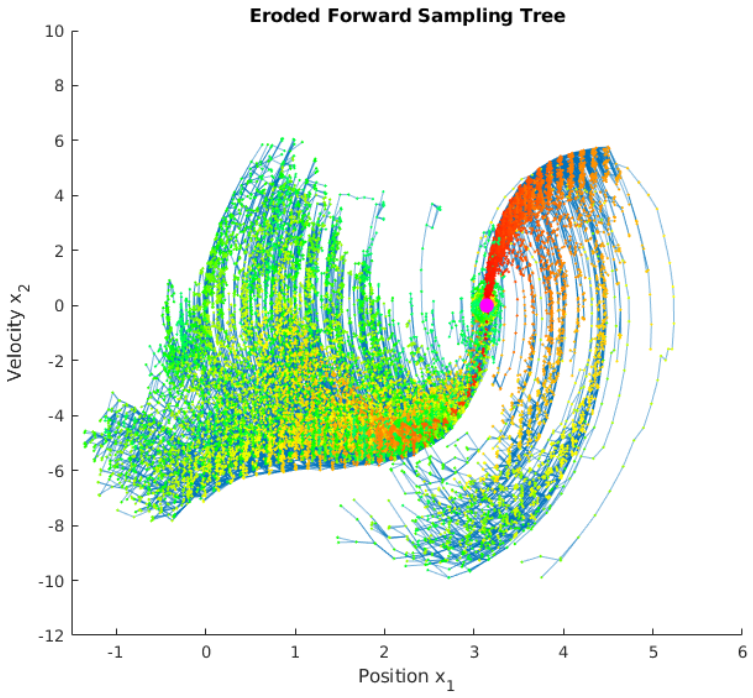


Figure 5.6: Path integral erosion method for L_1 inverted pendulum at end of first iteration. Nodes of green hue in Figure 5.5 are largely included and nodes of red hue are largely excluded. Hue in this figure corresponds to particle time t, green values are later.

$$f(x,u) \equiv f\left(\begin{bmatrix} \alpha & \beta & \omega & \psi \end{bmatrix}^\top, u\right)$$
$$= \begin{bmatrix} \omega \\ \psi \\ \frac{d_3\left(d_2\psi^2 \sin\beta + 2d_2\omega\psi\sin\beta - f_3\omega + f_2\sin(\alpha+\beta) - f_1\sin\alpha\right) + d_2\cos\beta\left(d_2\omega^2\sin\beta + f_4\psi - f_2\sin(\alpha+\beta)\right) + d_0 d_3\, u}{d_1 d_3 + 2d_2 d_3\cos\beta - d_2^2\cos^2\beta} \\ \frac{-(d_1 + 2d_2\cos\beta)\left(d_2\omega^2\sin\beta + f_4\psi - f_2\sin(\alpha+\beta)\right) - d_2\cos\beta\left(d_2\psi^2\sin\beta + 2d_2\omega\psi\sin\beta - f_3\omega + f_2\sin(\alpha+\beta) - f_1\sin\alpha\right) - d_0 d_2\cos\beta\, u}{d_1 d_3 + 2d_2 d_3\cos\beta - d_2^2\cos^2\beta} \end{bmatrix}. \tag{5.20}$$

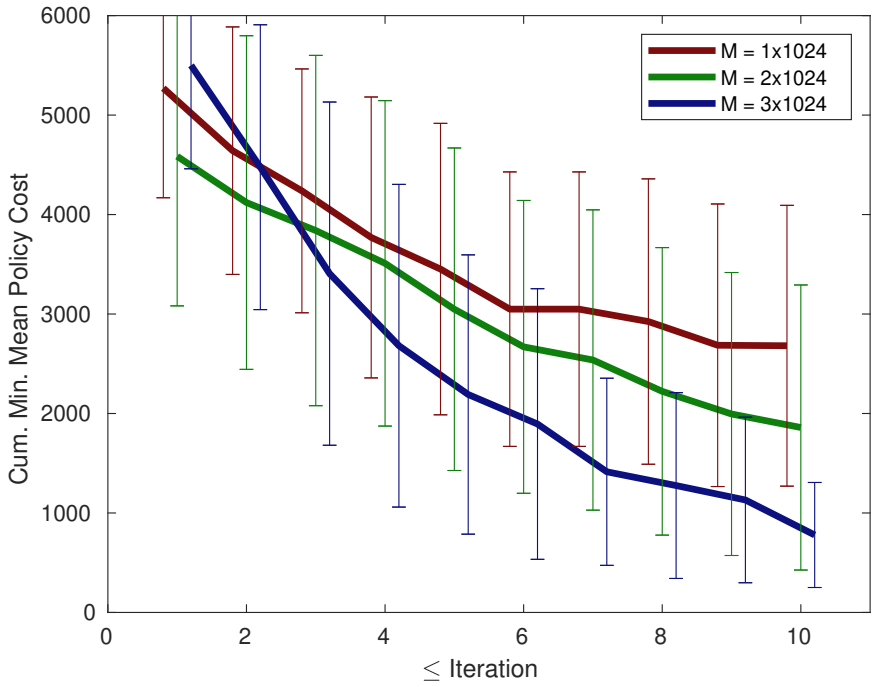
where the nonlinear function f is displayed in (5.20), within which $d_0, d_1, d_2, d_3, f_1, f_2, f_3$ are scalar parameters of the system. The associated optimal control problem is taken to be

$$V^*(t,x) = \inf_{u_{[t,T]}} \mathbf{E}_{\mathsf{Q}}^{t,x}\left[\int_t^T c_0|u_s|\mathrm{d}s + \sum_{j=1}^n c_j (X_T^{(j)})^2\right], \tag{5.19}$$

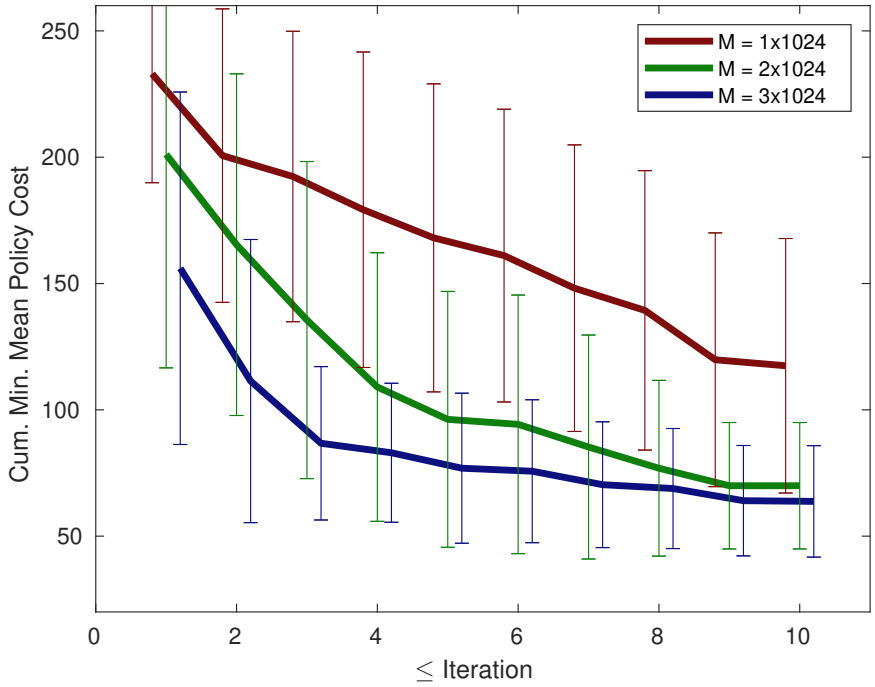
where c_0, c_1, c_2, c_3, c_4 are scalar parameters.

Two initial conditions are evaluated, $x_0^{\text{vert}} = [0,0,0,0]^\top$, where the bars are vertically down and motionless, and $x_0^{\text{off}} = [\pi/10, \pi/10, 0, 0]^\top$, where the angles of both bars are slightly perturbed from x_0^{vert} by $18°$. The number of time steps is taken to be $N = 80$ and the erode particle number is selected as $\widetilde{M} = (3/4)M$. The evaluation of these conditions over 30 trials with differing numbers of particles M is provided in Figure 5.7.

Since the initial conditions of the two experiments are close, their optimal values should also be close. Despite having comparable optimal values, the x_0^{off} condition converges far more rapidly than the x_0^{vert} condition. Slightly perturbing the initial condition vastly improved the performance of the algorithm for this problem. The reason the x_0^{vert} condition performs poorly is likely because the system is very sensitive in that region and a localized policy results in a bifurcation of trajectory densities. If the differing groups of trajectories have similar heuristic values, the value function approximation tries to fit a function to groups of particles in different sides of the state space, resulting in poor accuracy for either group. When the x_0^{off} condition is used, there is less ambiguity in which trajectory distributions are near-optimal resulting in better performance.

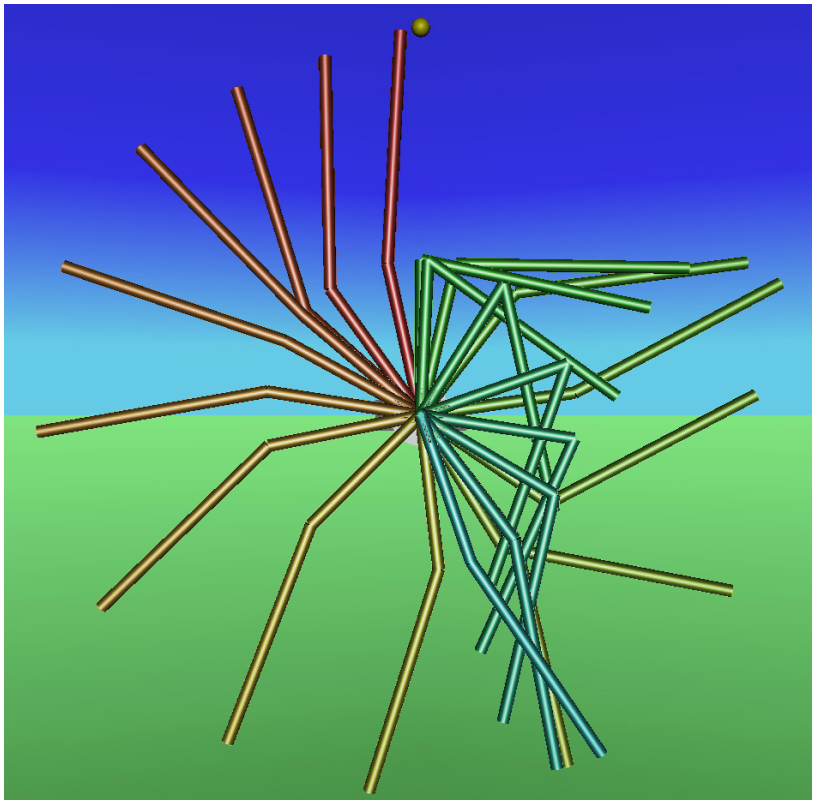


(a) Mean cost distribution for x_0^{vert}

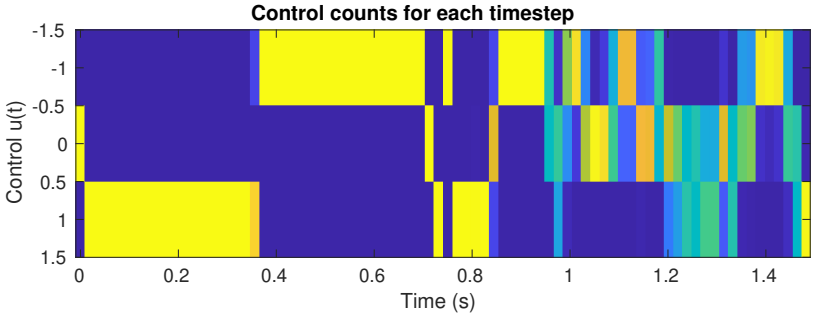


(b) Mean cost distribution for x_0^{off}

Figure 5.7: Mean policy cost statistics for L_1 double inverted pendulum problem. The mean bars and standard deviation whiskers characterize the distribution over 30 trials, where the value for each iteration is the accumulated minimum of the values over all previous iterations in that trial up to and including that iteration. M particles are used per time step in each condition.



(a) Double Inverted Pendulum Simulation



(b) Control Distribution

Figure 5.8: **(a)** Simulation of L_1 double inverted pendulum policy execution with $x_0 = (1.1\pi, 0.1\pi, 0, 0)$, guided by the best policy of 30 trials with $M = 3 \times 1024$ particles. The simulation begins in cyan, then moves to green, yellow, then red. **(b)** Control count distribution of sampled trajectories following the policy.

5.5.4 Intersection Collision Reachability

We also applied the algorithm to a reachability problem involving two vehicles in an intersection, visualized in Figure 5.9. We assume that the red vehicle in Figure 5.9 is autonomous and follows a hard-coded collision avoidance policy with respect to the yellow vehicle, which is driven by a human driver. The collision avoidance policy continuously accelerates or decelerates the red vehicle, governed by double integrator dynamics, depending on the course trajectory of the human's vehicle, governed by bicycle dynamics [69] with constant velocity and controlled by the rate of change of steering. The combined system has five dimensions and has the nonlinear dynamics

$$
\begin{aligned}
f &= \begin{bmatrix} a_1 \cos(x_3 + x_4) \\ a_1 \sin(x_3 + x_4) - x_5 \\ \frac{a_1}{a_2} \sin(x_4) \\ \text{clip}(a_3 u, -a_4, a_4) \\ \gamma(x) \end{bmatrix}, \\
\gamma(x) &= \text{clip}(\gamma_1, a_5, a_6), \\
\gamma_1 &= \begin{cases} \min(a_7(\gamma_2 - a_8), 0) & \gamma_2 \geq 0 \\ \max(a_7(\gamma_2 + a_8), 0) & \text{o.w.} \end{cases}, \\
\gamma_2 &= x_2 + -x_1 \tan(x_3) - \gamma_3 x_5, \\
\gamma_3 &= \frac{-x_1}{v_1 \cos(x_3)},
\end{aligned}
$$

where a_i are constants, (x_1, x_2) is the relative position of the two cars, x_3 is the orientation of yellow, x_4 is the steering angle of yellow, and x_5 is the velocity of red, driven by the feedback collision avoidance policy γ.

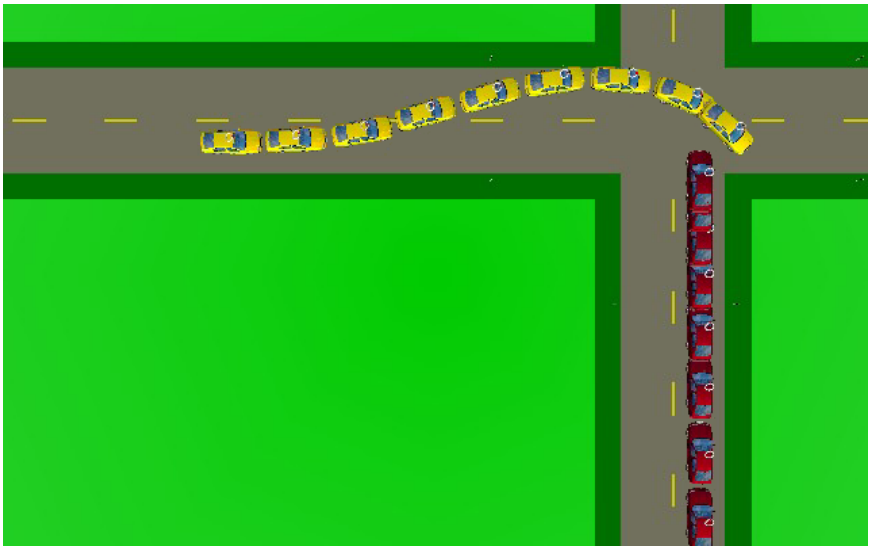


Figure 5.9: Simulation of intersection reachability policy execution.

When the human vehicle's trajectory is predictable, the autonomous vehicle's collision avoidance policy easily avoids the human vehicle, even if they begin on a collision course. However, the reachability problem seeks the worst-case human driver policy, given the autonomous vehicle's collision avoidance policy and the initial system state. That is, it seeks the optimal policy the human can execute which brings the two vehicles as close as possible at the end of the time horizon, on average. Solving problems such as these can aid in the verification of autonomous vehicle controllers or to evaluate whether the current controller should be switched for something more cautious.

The nature of the problem guarantees a bifurcation in trajectory rollouts, since the human vehicle either passes in front of the autonomous vehicle or behind it. This bifurcation can be easily seen in Figure 5.10. The FBRRT algorithm discovers that the optimal policy for causing a near-collision is to begin by swerving left in order to encourage the autonomous vehicle to begin braking, then to swerve back to the right to approach the slowed down autonomous vehicle.

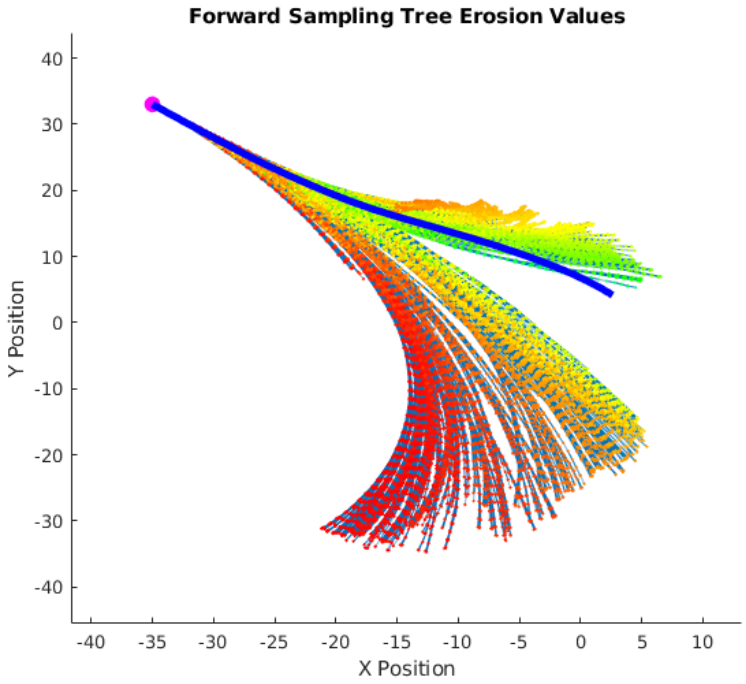


Figure 5.10: Forward sampling tree for the intersection reachability problem for the first iteration. Hue corresponds to the path integral heuristic ρ_i (green values are smaller) used for weighing particles in the backward pass and for pruning the tree. The blue line is the mean of sample trajectories, following the policies computed at the end of the 1st iteration.

5.6 FBRRT Conclusion

In this chapter, we have proposed a novel generalization of the FBSDE approach to solve stochastic optimal control problems, combining branched sampling techniques with weighted least squares function approximation to greatly expand the flexibility of these methods. Leveraging the efficient space-filling properties of RRT methods, we have demonstrated that our method significantly improves the convergence properties of previous FBSDE numerical methods. We have shown how the proposed method works hand-in-hand with a local entropy-weighted LSMC method, concentrating function approximation in the regions where optimal trajectories are most likely to be dense. Finally, we have demonstrated that FBRRT can generate feedback control policies for high-dimensional nonlinear stochastic optimal control problems.

CHAPTER 6
CONCLUSION AND FUTURE WORK

In this work, we have presented a novel methodology for approaching SOC problems using Feynman-Kac FBSDE theory, and have contributed techniques which significantly increase the accuracy and rate of convergence of numerical algorithms for obtaining the solution. The proposed methodology is generalized in such a way as to expose four primary design choices for every iteration of the method:

- Choose the target policy μ associated with the on-policy value function V^{μ} and on-policy measure Q.
- Construct the branch-sampled tree $\mathcal{G}$, which represents the set of sampling measures $\{\mathsf{P}_{i+1}\}$.
- Design the local-entropy heuristic ρ_{i+1} to be minimized for the purposes of concentrating approximation accuracy in regions likely to have optimal trajectories, associated with the weighted measures $\{\mathsf{R}_{i+1}\}$.
- Establish the value function model $\varphi(x;\alpha)$ used to approximate V^{μ}.

A fifth design choice, not explored in this work, would involve establishing and updating a policy model $\mu(x;\beta)$. By using the Taylor-noiseless estimator, we can obtain accuracy improvements shown to work well on problems like the LQR problem. Once convergence of the iterative method is achieved and a near-optimal policy is obtained, dense sampling of the policy and application of the Taylor-noisy estimator can produce a refined, more accurate value function.

Although we have offered design choices for all of these in our FBRRT implementation, we acknowledge that there is plenty of room for improvement. Employing more recently

developed RRT methods may improve the forward sampling method. In addition, in this work we did not discuss state constraints or obstacles. Since RRT methods are naturally designed to accommodate obstacles, the methods proposed here should be extendable to those problems as well. The challenge here will be how to properly characterize obstacles in a stochastic control framework, though for application in real-world problems one might opt to ignore a formal interpretation.

The erode procedure could use additional care to improve robustness. One of the primary sources of instability is removing important support for representing the value function. Part of the problem is that RRT tree exploration will frequently contain long, single-path branches which are hard to erode since it must be kept in the tree as long as just one of its descendants is marked for keeping. Although continuity in the $\{X_i\}$ process distributions is clearly important for accurate backward integration, requiring that each node be fully connected back to the root might not be strictly necessary, as long as there are nearby nodes which sufficiently represent the distribution continuity. This problem might also be mitigated with rewiring strategies used in RRT* and many of its derivatives [52, 53].

Another problem, as discussed in Section 5.5.3, has largely to do with the heuristic and systems which might be multi-modal. If we are using a localized approximation method for the value function, like polynomial basis functions, these methods are unlikely to perform well as the regression tries unsuccessfully to fit a single function to multiple modes. Encoding into the heuristic some commitment to a single mode might improve these methods, but how to achieve this remains an open research question.

Robustly evaluating the value function in extrapolative regions remains a difficult problem. When the sampled trajectory distribution is well-behaved, the blending method discussed in Section 4.6 works well, but this method has not been tested on the FBRRT implementation, and might suffer from the multi-modal problems described previously. One potential solution to this problem is to constrain the drift sampling to regions not far from the regions where the approximation is accurate, that is, the weighted distributions $\mathsf{R}_{i+1,X_{i+1}}$.

One of the primary questions moving forward is whether these methods are better suited for local value function approximation with models like polynomial basis functions, or for global approximation with models like deep neural nets. Regardless of the model used, we can, instead of using full linear regression applied to the LSMC optimization problem, use a gradual optimization technique such as stochastic gradient descent (SGD) to refine the value function and policy approximation. The arguments in the minimizations (5.12) and (4.67) need not be fully minimized, but can be differentiated with respect to the value function and policy parameters to produce a step of SGD. Such techniques are central to deep RL techniques such as deep deterministic policy gradient (DDPG) [70], but when combined with the proposed estimators might improve robustness of convergence and, if deep neural nets are used for representation, produce broader approximations.

One of the key problems with using more complicated models for the value function is the necessity to compute gradients and Hessians. This is especially concerning in high-dimensional spaces where the number of partial derivatives scales with $\mathrm{O}(n^2)$. A potential solution to the problem may come from modifying the vanishing viscosity method discussed in Chapter 3 to regularize the problem. If there are only a few dimensions of noise with significant diffusion, we can choose to use the nominal, sparse version of diffusion $\widehat{\sigma}$ in the computation of $\overline{Z}_{i+1}$ and $\overline{M}_{i+1}$. Such a choice will reduce the number of partials which need evaluation to $\mathrm{O}(m^2)$, where m is the number of dimensions with significant noise.

One of the insights to consider in any adaptation to deep RL techniques is that the *replay buffer* can be potentially interpreted as a joint distribution over (X_i, K_i, X_{i+1}). Approaches like DDPG are considered off-policy in the RL community because the controls leading to these tuples come from a previous version of the policy. The difference between our estimators and the updates performed in DDPG is that DDPG chooses to incur the bias without directly addressing it, while FBSDE estimators attempt to reduce the bias of the update using a motion model. For this reason, any application of this theory to deep RL will

be considered a model-based RL method (a survey of leading methods is found in [71]).

Another avenue of exploration is combining these methods with MPPI. The MPPI method maintains a distribution of forward trajectories which can be very easily be interpreted as the forward pass of an FBSDE method. In fact, the local-entropy weighing theory used in weighing the function approximation is also used in MPPI to weigh its trajectories, though the heuristics are different. If we take any set of trajectories generated by MPPI, potentially along with the weights computed by its algorithm, we can perform a backwards pass and locally approximate the value function. The open question in this line of research is how, or if, the value function approximation should inform the MPPI algorithm. One potential application is to use the nominal MPPI policy as the target policy and use a different cost function in the FBSDE method, taking the MPPI-generated trajectories as the sampling distribution, as a means to gather broader information about the state space without the need to sample more trajectories. For example, we may be able to answer questions about stochastic reachability under a nominal policy, which could then be used to bias the MPPI cost function away from regions likely to be unsafe.

Appendices

APPENDIX A

PROOFS OF STATED THEOREMS

A.1 Proof of Lemma 4.1

Lemma A.1 (Discrete Girsanov 1-Step)**.** *Let W^{P} be a normal random vector in $\mathbb{R}^n$, let D be an independent, bounded random vector, and let P be the product measure which represents their joint distribution. Then the measure Q defined as*

$$\mathrm{d}\mathsf{Q} = \exp\left(-\frac{1}{2}\|D\|^2 + D^\top W^{\mathsf{P}} \right) \mathrm{d}\mathsf{P}, \tag{A.1}$$

is a probability measure and the variable

$$W^{\mathsf{Q}} := W^{\mathsf{P}} - D, \tag{A.2}$$

is a normal random vector in Q.

Proof. Let $W^{\mathsf{P}} : (\Omega_1, \mathcal{B}_1) \mapsto (\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$ be a normal random vector in the probability space $(\Omega_1, \mathcal{B}_1, \mathsf{P}_1)$, and let $D : (\Omega_2, \mathcal{B}_2) \mapsto (\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$ be a bounded random vector in $(\Omega_2, \mathcal{B}_2, \mathsf{P}_2)$, where $\mathcal{B}(\mathbb{R}^n)$ is the Borel σ-field over open sets in the metric space $(\mathbb{R}^n, \|\cdot\|_2)$ and $\mathcal{B}_1, \mathcal{B}_2$ are the σ-algebras generated by their maps. Define the product measure space $(\Omega_1 \times \Omega_2, \mathcal{B}_1 \times \mathcal{B}_2, \mathsf{P})$ where $\mathcal{B}_1 \times \mathcal{B}_2$ is the σ-field over measurable rectangles $A_1 \times A_2$, such that $A_1 \in \mathcal{B}_1$, $A_2 \in \mathcal{B}_2$, and $\mathsf{P}(A_1 \times A_2) = \mathsf{P}_1(A_1)\mathsf{P}_2(A_2)$ for each rectangle [57, p. 148]. The random variable $\Theta := \varphi(D, W^{\mathsf{P}})$ where

$$\varphi(d, w) := \exp\left(-\frac{1}{2}\|d\|^2 + d^\top w \right)$$

is $\mathcal{B}_1 \times \mathcal{B}_2$-measurable, due to the continuity of the function φ, and strictly positive. We

have by Fubini's Theorem that

$$\begin{aligned}\int_{\Omega_1\times\Omega_2}\Theta\,\mathrm{d}\mathsf{P} &= \int_{\Omega_2}\int_{\Omega_1}\Theta_{\omega_2}(\omega_1)\,\mathrm{d}\mathsf{P}_1(\omega_1)\,\mathrm{d}\mathsf{P}_2(\omega_2)\\ &= \int_{\Omega_2}\int_{\Omega_1}\varphi(D_{\omega_2},W^{\mathsf{P}}(\omega_1))\,\mathrm{d}\mathsf{P}_1(\omega_1)\,\mathrm{d}\mathsf{P}_2(\omega_2)\end{aligned}$$

where Θ_{ω_2} is the section of Θ with respect to ω_2 [57, p. 152]. It is a property of log-normal distributions [64] that for any $d\in\mathbb{R}^n$,

$$\int_{\Omega_1}\varphi(d,W^{\mathsf{P}}(\omega_1))\,\mathrm{d}\mathsf{P}_1(\omega_1)=1,$$

so we immediately obtain that

$$\int_{\Omega_1\times\Omega_2}\Theta\,\mathrm{d}\mathsf{P}=1.$$

Since Θ is strictly positive and has a mean of 1 in P, we can use the Radon-Nikodym theorem to define an equivalent probability measure Q as

$$\mathrm{d}\mathsf{Q}=\Theta\,\mathrm{d}\mathsf{P}.$$

Define the random element $W^{\mathsf{Q}}:(\Omega_1\times\Omega_2,\mathcal{B}_1\times\mathcal{B}_2)\mapsto(\mathbb{R}^n,\mathcal{B}(\mathbb{R}^n))$ as $W^{\mathsf{Q}}(\omega_1,\omega_2)=W^{\mathsf{P}}(\omega_1)-D(\omega_2)$, which is measurable because the subtraction map is continuous. We now show that W^{Q} is a normal distribution by proving that its density is $p_{\mathcal{N}}(w^{\mathsf{Q}})$ where the function is

$$p_{\mathcal{N}}(w)=(2\pi)^{-n/2}\exp\left(-\frac{1}{2}\|w\|^2\right).$$

We will prove this is the density by showing the equality

$$\int_{(W^{\mathsf{Q}})^{-1}(B)} \mathrm{d}\mathsf{Q} = \int_B p_{\mathcal{N}}(w^{\mathsf{Q}})\,\mathrm{d}w^{\mathsf{Q}},$$

for any $B \in \mathcal{B}(\mathbb{R}^n)$, where $\mathrm{d}w^{\mathsf{Q}}$ is the Lebesgue measure. By a change of measure and Fubini's Theorem we have

$$\begin{aligned}\int_{(W^{\mathsf{Q}})^{-1}(B)} \mathrm{d}\mathsf{Q} &= \int_{\Omega_1\times\Omega_2} \mathbf{1}_{(W^{\mathsf{Q}})^{-1}(B)}\,\mathrm{d}\mathsf{Q} \\ &= \int_{\Omega_1\times\Omega_2} \mathbf{1}_{(W^{\mathsf{Q}})^{-1}(B)}\Theta\,\mathrm{d}\mathsf{P} \\ &= \int_{\Omega_2}\int_{\Omega_1} (\mathbf{1}_{(W^{\mathsf{Q}})^{-1}(B)})_{\omega_2}(\omega_1)\varphi(D_{\omega_2}, W^{\mathsf{P}}(\omega_1))\,\mathrm{d}\mathsf{P}_1(\omega_1)\,\mathrm{d}\mathsf{P}_2(\omega_2),\end{aligned}$$

where $\mathbf{1}$ is the indicator function. The section of the indicator function can be reworked as

$$= \int_{\Omega_2}\int_{\Omega_1} \mathbf{1}_B(W^{\mathsf{P}}(\omega_1) - D_{\omega_2})\varphi(D_{\omega_2}, W^{\mathsf{P}}(\omega_1))\,\mathrm{d}\mathsf{P}_1(\omega_1)\,\mathrm{d}\mathsf{P}_2(\omega_2),$$

then we evaluate the normal density in P_1 [57, Proposition 5.5.2]

$$= \int_{\Omega_2}\int_{\mathbb{R}^n} \mathbf{1}_B(w^{\mathsf{P}} - D_{\omega_2})\varphi(D_{\omega_2}, w^{\mathsf{P}})p_{\mathcal{N}}(w^{\mathsf{P}})\,\mathrm{d}w^{\mathsf{P}}\,\mathrm{d}\mathsf{P}_2(\omega_2).$$

The following reduction is due to the relation $\varphi(d, w)p_{\mathcal{N}}(w) = p_{\mathcal{N}}(w - d)$,

$$= \int_{\Omega_2}\int_{\mathbb{R}^n} \mathbf{1}_B(w^{\mathsf{P}} - D_{\omega_2})p_{\mathcal{N}}(w^{\mathsf{P}} - D_{\omega_2})\,\mathrm{d}w^{\mathsf{P}}\,\mathrm{d}\mathsf{P}_2(\omega_2).$$

We can now use a change of variables in $\mathbb{R}^n$, $w^{\mathsf{Q}}_{\omega_2} = w^{\mathsf{P}} - D_{\omega_2}$, [1]

$$= \int_{\Omega_2}\int_{\mathbb{R}^n} \mathbf{1}_B(w^{\mathsf{Q}}_{\omega_2})p_{\mathcal{N}}(w^{\mathsf{Q}}_{\omega_2})\,\mathrm{d}w^{\mathsf{Q}}_{\omega_2}\,\mathrm{d}\mathsf{P}_2(\omega_2),$$

[72, Theorem 3.7.1]. Note that for each ω_2 the measure $\mathrm{d}w^{\mathsf{Q}}_{\omega_2}$ is translation invariant and

[1] The mapping $F_{\omega_2}(w^{\mathsf{P}}) \mapsto w^{\mathsf{P}} - D_{\omega_2}$ is injective on $\mathbb{R}^n$ because D_{ω_2} is a.s. bounded.

that the integrand no longer varies with ω_2. Thus we can reduce to the desired result,

$$\begin{aligned} &= \int_{\mathbb{R}^n} \mathbf{1}_B(w^{\mathsf{Q}}) p_{\mathcal{N}}(w^{\mathsf{Q}}) \, \mathrm{d}w^{\mathsf{Q}} \\ &= \int_B p_{\mathcal{N}}(w^{\mathsf{Q}}) \, \mathrm{d}w^{\mathsf{Q}}. \end{aligned}$$

□

A.2 Proof of Lemma 4.2

Lemma A.2 (Discrete-Time Girsanov Theorem)**.** *Let* $(\Omega, \{\mathcal{F}_i\}_{i=0}^N, \mathsf{P})$ *be a filtered probability space and let* $\{\xi_i\}_{i=0}^N$ *be an adapted process where* $\xi_0 := (0_n, 0_n)$ *and* $\xi_{i+1} := (D_i, W_i^{\mathsf{P}})$ *for* $i = 0, \ldots, N-1$*, such that, in* P*,* D_i *is a bounded random vector,* W_i^{P} *is normal random vector, and* D_i *is independent of* W_i^{P}*. If* Q *is the measure defined by*

$$\mathrm{d}\mathsf{Q} = \prod_{i=0}^{N-1} \exp\left(-\frac{1}{2}\|D_i\|^2 + D_i^\top W_i^{\mathsf{P}} \right) \mathrm{d}\mathsf{P}, \tag{A.3}$$

then Q *is a probability measure and*

$$W_i^{\mathsf{Q}} := W_i^{\mathsf{P}} - D_i, \tag{A.4}$$

are $\mathcal{F}_{i+1}$*-measurable, independent normal random vectors in* Q*.*

Proof. We assume that $\Omega := \Omega_{0:N}$ where $\Omega_{0:i} := \Omega_0 \times \cdots \times \Omega_i$, with $A_{0:i} \subseteq \Omega_{0:i}$, $\forall A_{0:i} \in \mathcal{F}_i$. We also assume that there exists a series of transition kernels $\mathsf{P}_{i+1|i}(\omega_{0:i}, \mathrm{d}\omega_{i+1})$ which defines P_{i+1} as

$$\mathsf{P}_{i+1}(A_{0:i+1}) = \int_{A_{0:i}} \mathsf{P}_{i+1|i}(\omega_{0:i}, (A_{0:i+1})_{\omega_{0:i}}) \mathsf{P}_i(\mathrm{d}\omega_{0:i}),$$

for $A_{0:i} \in \mathcal{F}_i$, $A_{0:i+1} \in \mathcal{F}_{i+1}$, such that $\mathsf{P}_N = \mathsf{P}$. We use an inductive argument to prove

the result. Define the variable

$$\Theta_{i+1} = \prod_{j=0}^{i} \exp\left(-\frac{1}{2}\|D_j\|^2 + D_j^\top W_j^{\mathsf{P}} \right), \tag{A.5}$$

and define the measures Q_{i+1} as

$$\mathrm{d}\mathsf{Q}_{i+1} = \Theta_{i+1}\mathrm{d}\mathsf{P}_{i+1},$$

for $i = 0, \ldots, N-1$ and $\mathsf{Q}_0 \equiv \mathsf{P}_0$. We have Q_1 is a probability measure and W_0^{Q} is a normal random vector in Q_1 by Lemma 4.1. Assume Q_i is a probability measure and $W_{i-1}^{\mathsf{Q}}, \ldots, W_0^{\mathsf{Q}}$ are independent normal random vectors in Q_i.

For $B_{0:i} \in \mathbb{B}((\mathbb{R}^n)^i)$ and $C_{0:i+1} = (W_{0:i}^{\mathsf{Q}})^{-1}(B_{0:i})$, we have

$$\begin{aligned}
\int_{C_{0:i+1}} \mathrm{d}\mathsf{Q}_{i+1} &= \int_{C_{0:i+1}} \Theta_{i+1}\mathrm{d}\mathsf{P}_{i+1} \\
&= \int_{\Omega_{0:i+1}} \mathbf{1}_{C_{0:i+1}} \Theta_{i+1}\mathrm{d}\mathsf{P}_{i+1} \\
&= \int_{\Omega_{0:i}} \left[\int_{\Omega_{i+1}} \mathbf{1}_{(C_{0:i+1})_{\omega_{0:i}}} (\Theta_{i+1})_{\omega_{0:i}} \mathsf{P}_{i+1|i}(\omega_{0:i}, \mathrm{d}\omega_{i+1}) \right] \mathsf{P}_i(\mathrm{d}\omega_{0:i}) \\
&= \int_{\Omega_{0:i}} \left[\int_{\Omega_{i+1}} \mathbf{1}_{(C_{0:i+1})_{\omega_{0:i}}} \exp\left(-\frac{1}{2}\|D_i\|^2 + D_i^\top W_i^{\mathsf{P}} \right) \mathsf{P}_{i+1|i}(\omega_{0:i}, \mathrm{d}\omega_{i+1}) \right] \\
&\qquad\qquad \Theta_i \mathsf{P}_i(\mathrm{d}\omega_{0:i}),
\end{aligned}$$

Similarly to Lemma 4.1, the inside integral reduces to an integral over B_i,

$$= \int_{\Omega_{0:i}} \mathbf{1}_{C_{0:i}} \left[\int_{B_i} p_{\mathcal{N}}(w_i^{\mathsf{Q}})\, \mathrm{d}w_i^{\mathsf{Q}} \right] \Theta_i \mathsf{P}_i(\mathrm{d}\omega_{0:i}),$$

and change of measure yields

$$
\begin{aligned}
&= \int_{\Omega_{0:i}} \mathbf{1}_{C_{0:i}} \left[\int_{B_i} p_{\mathcal{N}}(w_i^{\mathsf{Q}}) \, \mathrm{d}w_i^{\mathsf{Q}} \right] \mathsf{Q}_i(\mathrm{d}\omega_{0:i}) \\
&= \left[\int_{B_i} p_{\mathcal{N}}(w_i^{\mathsf{Q}}) \, \mathrm{d}w_i^{\mathsf{Q}} \right] \int_{\Omega_{0:i}} \mathbf{1}_{C_{0:i}} \mathsf{Q}_i(\mathrm{d}\omega_{0:i}).
\end{aligned}
$$

By our inductive assumption, we arrive at the result

$$
\begin{aligned}
&= \left[\int_{B_i} p_{\mathcal{N}}(w_i^{\mathsf{Q}}) \, \mathrm{d}w_i^{\mathsf{Q}} \right] \prod_{j=0}^{i-1} \int_{B_j} p_{\mathcal{N}}(w_j^{\mathsf{Q}}) \, \mathrm{d}w_j^{\mathsf{Q}} \\
&= \prod_{j=0}^{i} \int_{B_j} p_{\mathcal{N}}(w_j^{\mathsf{Q}}) \, \mathrm{d}w_j^{\mathsf{Q}}.
\end{aligned}
$$

□

A.3 Proof of Proposition 4.4

Proof. In the following, the variable

$$\alpha := (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n,$$

is used as multi-index notation,

$$
\begin{aligned}
|\alpha| &:= \alpha_1 + \cdots + \alpha_n, & \alpha! &:= \alpha_1! \cdots \alpha_n!, \\
x^\alpha &:= x_1^{\alpha_1} \cdots x_n^{\alpha_n}, & \partial_{x_\alpha} &:= \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \cdots \partial x_n^{\alpha_n}}.
\end{aligned}
$$

Let $j \geq 3$ be an odd number and suppose $\widetilde{V}_{i+1}^{\mu} \in C^k(\mathbb{R}^n)$ for some $k \geq j$. The j-th order term of the Taylor expansion residual is given by Taylor's theorem as

$$\sum_{|\alpha|=j} \frac{1}{\alpha!} \partial_{x_\alpha} \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{Q}}) (\Sigma_i W_i^{\mathsf{Q}})^\alpha. \tag{A.6}$$

It can be shown with algebra and the multinomial theorem that there exists functions γ_α such that the W_i^{Q} terms can be linearly separated from the others,

$$\sum_{|\alpha|=j} \gamma_\alpha(\partial_{x_\alpha} \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{Q}}), \Sigma_i)(W_i^{\mathsf{Q}})^\alpha. \tag{A.7}$$

Since both $\partial_{x_\alpha} \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{Q}})$ and Σ_i are X_i-measurable, when taking the conditional expectation the operator passes inside

$$\sum_{|\alpha|=j} \gamma_\alpha(\partial_{x_\alpha} \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{Q}}), \Sigma_i)\mathbf{E}_{\widetilde{\mathsf{Q}}}[(W_i^{\mathsf{Q}})^\alpha | X_i]. \tag{A.8}$$

Due to the independence of the different dimensions of W_i^{Q}, the conditional expectation inside (A.8) can be expanded into the product

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[(W_i^{\mathsf{Q}})_1^{\alpha_1} | X_i] \cdots \mathbf{E}_{\widetilde{\mathsf{Q}}}[(W_i^{\mathsf{Q}})_n^{\alpha_n} | X_i]. \tag{A.9}$$

Since $|\alpha|$ is odd, there exists an l such that α_l is odd. The properties of the standard normal distribution guarantee

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[(W_i^{\mathsf{Q}})_l^{\alpha_l} | X_i] = 0, \tag{A.10}$$

and thus, (A.8) is zero as well, so we arrive at the result

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[\sum_{|\alpha|=j} \gamma_\alpha(\partial_{x_\alpha} \widetilde{V}_{i+1}^{\mu}(\overline{X}_{i+1}^{\mathsf{Q}}), \Sigma_i)(W_i^{\mathsf{Q}})^\alpha | X_i] = 0. \tag{A.11}$$

□

A.4 Proof of Theorem 4.6 & Theorem 4.8

Proof. Using the result (4.48) of Lemma 4.7 we have

$$\begin{aligned}\widehat{Y}_i &:= \widehat{Y}_{i+1} - \Delta\widehat{Y}_i \\ &= \widehat{Y}_{i+1} - \Delta Y_i + (\delta_{i+1}^{\Delta\widehat{Y}} - \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]),\end{aligned}$$

and so the expression for the bias is

$$\mathbf{E}_{\widetilde{\mathsf{P}}}[Y_i - \widehat{Y}_i|X_i, K_i] = \mathbf{E}_{\widetilde{\mathsf{P}}}[Y_{i+1} - \widehat{Y}_{i+1}|X_i, K_i] - \varepsilon_{i+1}^{\mathsf{P}|\mathsf{Q}}.$$

The variance of the estimator is

$$\begin{aligned}\mathrm{Var}_{\widetilde{\mathsf{P}}}[\widehat{Y}_i|X_i, K_i] &= \mathrm{Var}_{\widetilde{\mathsf{P}}}[\widehat{Y}_{i+1} - \Delta Y_i \\ &\qquad + (\delta_{i+1}^{\Delta\widehat{Y}} - \mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i])|X_i, K_i] \\ &= \mathrm{Var}_{\widetilde{\mathsf{P}}}[\delta_{i+1}^{\Delta\widehat{Y}} - (Y_{i+1} - \widehat{Y}_{i+1})|X_i, K_i],\end{aligned}$$

noting that we can drop the terms Y_i and $\mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]$ because they are (X_i, K_i)-measurable.

For the re-estimate estimator we have

$$\begin{aligned}Y_{i+1} - \widehat{Y}_{i+1}^{\text{re-est}} &= V_{i+1}^{\mu}(X_{i+1}) - \widetilde{V}_{i+1}^{\mu}(X_{i+1}) \\ &= \delta_{i+1}^{\widetilde{V}},\end{aligned} \tag{A.12}$$

and for the noiseless estimator we have

$$\begin{aligned} Y_{i+1} - \widehat{Y}_{i+1}^{\text{noiseless}} &= V_{i+1}^{\mu}(X_{i+1}) - \widetilde{Y}_{i+1} \\ &= V_{i+1}^{\mu}(X_{i+1}) - (\widetilde{V}_{i+1}^{\mu}(X_{i+1}) - \delta_{i+1}^{\text{h.o.t.}}) \\ &= \delta_{i+1}^{\Delta\widehat{Y}}, \end{aligned} \tag{A.13}$$

due to (4.17). Plugging these two equalities into the general expressions for the bias and variance and doing simple reductions yields the theorem results. Note that Theorem 4.6 is proved by setting $\widetilde{\mathsf{P}} \equiv \widetilde{\mathsf{Q}}$, which entails $\varepsilon_{i+1}^{\mathsf{P}|\mathsf{Q}} \equiv 0$, and by excluding K_i from the conditional expectations. □

A.5 Proof of Theorem 4.9

Proof. First note that the process $\{\Theta_i\}$ is a martingale in $\widetilde{P}$, that is, $\mathbf{E}_{\widetilde{\mathsf{P}}}[\Theta_j|\mathcal{F}_i] = \Theta_i$ for $j \geq i$. This can be shown by defining the measures

$$\mathrm{d}\widetilde{\mathsf{Q}} = \Theta_j \mathrm{d}\widetilde{\mathsf{P}}_j = \frac{\Theta_j}{\Theta_i}\mathrm{d}\widetilde{\mathsf{R}}_i,$$

and noting that, for all $B_i \in \mathcal{F}_i$,

$$\int_{B_i} \Theta_i \mathrm{d}\widetilde{\mathsf{P}}_j = \int_{B_i} \mathrm{d}\widetilde{\mathsf{R}}_i = \int_{B_i} \mathrm{d}\widetilde{\mathsf{Q}} = \int_{B_i} \Theta_j \mathrm{d}\widetilde{\mathsf{P}}_j,$$

where the inner equality is due to $\widetilde{\mathsf{R}}_i$ and $\widetilde{\mathsf{Q}}$ agreeing on B_i.

From [59, Lemma 1] we have

$$\begin{aligned}\mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|\mathcal{F}_i] &= \frac{\mathbf{E}_{\widetilde{\mathsf{P}}}[\Theta_N\delta_{i+1}^{\Delta\widehat{Y}}|\mathcal{F}_i]}{\mathbf{E}_{\widetilde{\mathsf{P}}}[\Theta_N|\mathcal{F}_i]} \\ &= \frac{\Theta_i\mathbf{E}_{\widetilde{\mathsf{P}}}[\Theta_i^{-1}\Theta_N\delta_{i+1}^{\Delta\widehat{Y}}|\mathcal{F}_i]}{\Theta_i} \\ &= \mathbf{E}_{\widetilde{\mathsf{P}}}[(\Theta_N\Theta_{i+1}^{-1})\varphi(D_i, W_i^{\mathsf{P}})\delta_{i+1}^{\Delta\widehat{Y}}|\mathcal{F}_i] \\ &= \mathbf{E}_{\widetilde{\mathsf{P}}}[\varphi(D_i, W_i^{\mathsf{P}})\delta_{i+1}^{\Delta\widehat{Y}}|\mathcal{F}_i],\end{aligned}$$

where the final equality is due to the tower property of conditional expectation and the $\mathcal{F}_{i+1}$ measurability of the remaining terms. Again by the tower property of conditional expectation we have

$$\mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i] = \mathbf{E}_{\widetilde{\mathsf{P}}}[\varphi(D_i, W_i^{\mathsf{P}})\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i].$$

By the Cauchy-Schwartz inequality, we have that

$$\begin{aligned}&|\mathbf{E}_{\widetilde{\mathsf{Q}}}[\delta_{i+1}^{\Delta\widehat{Y}}|X_i, K_i]| \\ &\quad\leq \mathbf{E}_{\widetilde{\mathsf{P}}}[\varphi(D_i, W_i^{\mathsf{P}})^2|X_i, K_i]^{1/2}\mathbf{E}_{\widetilde{\mathsf{P}}}[(\delta_{i+1}^{\Delta\widehat{Y}})^2|X_i, K_i]^{1/2}.\end{aligned}$$

Using properties of log-normal distributions [64] we have

$$\begin{aligned}\mathbf{E}_{\widetilde{\mathsf{P}}}[\varphi(D_i, W_i^{\mathsf{P}})^2|X_i, K_i] &= \mathbf{E}_{\widetilde{\mathsf{P}}}[\exp(\|D_i\|^2)|X_i, K_i] \\ &= \exp(\|D_i\|^2),\end{aligned}$$

which, upon substitution, yields the desired result. □

APPENDIX B

USER'S GUIDE TO IFBSDE METHODS AND FBRRT

The purpose of this chapter is twofold: (1) to provide a condensed and more accessible summary of the methods proposed in this work for future research, and (2) to contextualize these methods in real-world systems and control-design methods.

B.1 Problem Setup

We begin by discussing which continuous-time problems are suited to the methods of this work. We first restate the SOC problem from (2.1), (C-V^*),

$$\begin{aligned} \mathrm{d}X_s &= f(s, X_s, u_s)\,\mathrm{d}s + \sigma(s, X_s)\,\mathrm{d}W_s^{\mathsf{Q}}, \\ J(t, x_t; u_{[t,T]}) &:= \mathbf{E}_{\mathsf{Q}}\left[\int_t^T \ell(s, X_s, u_s)\,\mathrm{d}s + g(X_T)\right], \\ V^*(t, x_t) &= \inf_{u_{[t,T]}} J(t, x_t; u_{[t,T]}). \end{aligned}$$

One of the primary requirements of the application of this theory is local smoothness of the value function. This is generally guaranteed by assumption (A1), and by f, σ, ℓ, and g being Lipschitz continuous in x. Thus, discontinuous functions like relays, quantization functions, and indicator functions might be poorly suited for inclusion in components of these functions. Even if a smoothed version is used, FBSDE methods may breakdown if the cost functions have zero-valued derivatives in most parts of the state space, similar to the "sparse rewards" concept often investigated in the reinforcement-learning community. An illustration of this issue is available in the "go right problem".

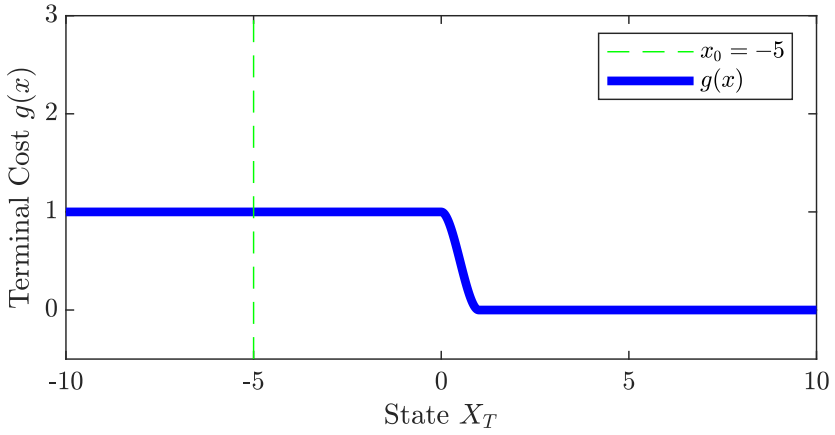
Go Right Problem

SOC Problem Definition:

$$\mathrm{d}X_s = \overbrace{10u_s}^{\text{velocity control}}\,\mathrm{d}s + \overbrace{\mathrm{d}W_s}^{\text{noise}}, \qquad u_s \in [-1,1] \qquad x_0 = -5,$$

$$V^*(t,x) = \inf_{u_s} \mathbf{E}[\underbrace{g(X_T)}_{\substack{\text{smooth}\\ \text{Heaviside}}}], \qquad T = 1,$$

$$g(x) = \begin{cases} 1 & x \le 0 \\ 1 - 3x^2 + 2x^3 & 0 \le x \le 1 \\ 0 & x \ge 1 \end{cases}$$



The optimal value function is $V^*(0,-5) \approx 0$ with optimal policy $u_s^* \equiv 1$.

Feynman-Kac FBSDEs (from HJB equations):

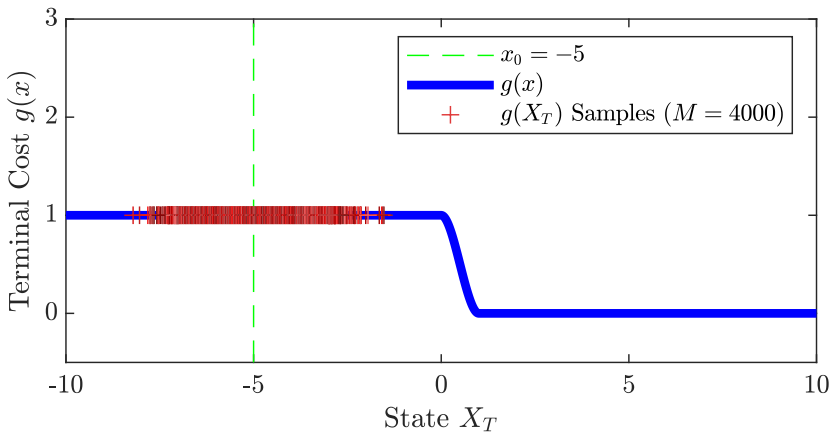
$$\mathrm{d}X_s = \mathrm{d}W_s, \qquad x_0 = -5,$$

$$\mathrm{d}Y_s = -10 \min_{u\in[-1,1]} \{Z_s u\}\,\mathrm{d}s + Z_s\,\mathrm{d}W_s, \qquad Y_T = g(X_T),$$

Numerical samples ($M = 4000$) of $g(X_T)$ simulated:

▶

Go Right Problem (cont)

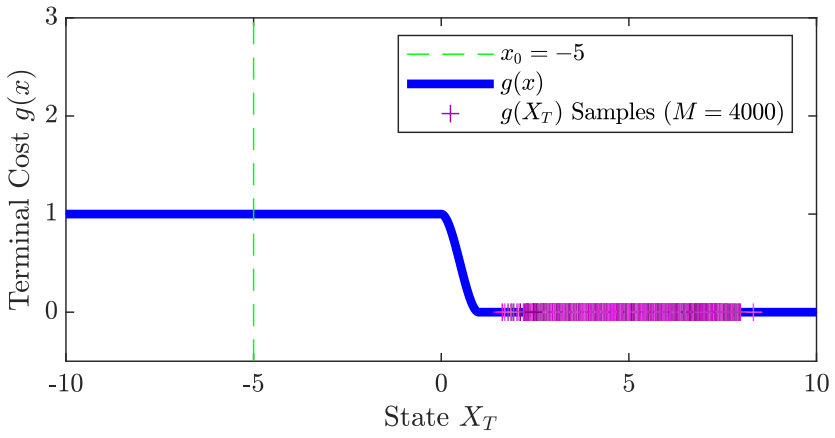


According to numerical sampling, $(Y_t = V^*(t, X_t) \equiv 1, Z_t \equiv 0)$ is a solution of the FBSDE system, but the optimal value function should be $V^*(0, -5) \approx 0$. Suppose instead the forward SDE corresponds to the optimal policy distribution.

Feynman-Kac FBSDEs (from HJB equations, optimal forward SDE distribution):

$$\begin{aligned} \mathrm{d}X_s &= 10\,\mathrm{d}s + \mathrm{d}W_s, & x_0 &= -5, \\ \mathrm{d}Y_s &= -10 \min_{u\in[-1,1]} \{Z_s(u-1)\}\,\mathrm{d}s + Z_s\,\mathrm{d}W_s, & Y_T &= g(X_T), \end{aligned}$$

Numerical samples ($M = 4000$) of $g(X_T)$ simulated:



According to numerical sampling, $(Y_t = V^*(t, X_t) \equiv 0, Z_t \equiv 0)$ is a solution of ▶

Go Right Problem (cont)

the FBSDE system, but $\partial_x V^* \equiv 0$ will result in a singular policy. That is, there is no gradient to use for determining the optimal control associated with this value function.

Issues like those encountered in the "go right" problem can be overcome by changing the cost functions or by choosing the drift process K_s appropriately enough for the value function to have meaningful curvature in the forward process distribution.

For deterministic optimal control problems ($\sigma \equiv 0$) and SOC problems where σ is sometimes singular, we can approximate the problem using the vanishing viscosity method (A4). Often, this simply amounts to adding independent noise to each dimension in the state space not represented in the rows of σ, and for deterministic problems can be achieved by making σ a constant diagonal matrix with positive elements. As discussed in the text surrounding (A4), there is always a method to produce a σ which is an ε-modification of a nominal $\widehat{\sigma}$ which is also nonsingular.

Although all theoretical results utilize a diffusion term σ which depends on state x, all numerical results involved constant additive noise $\sigma(t, x) \equiv \sigma \in \mathbb{R}^{n \times n}$. While multiplicative noise (e.g. $\sigma(t, x) = \sum_i x^i A^i$ for $A^i \in \mathbb{R}^{n \times n}$) is theoretically consistent under the proposed methods, the significant increase in variance introduced by such problems deserves extra care and consideration (see, e.g., [73]). Since the drift term K_s can be selected arbitrarily, conditions can be placed on it to ensure the forward distribution's variance does not explode with time.

In general, the smoothness of the value function determines how well these methods will perform. The discrete-time analysis shows that off-policy estimators are Taylor-expansions whose error grows with the higher order terms of the expansion. Adding noise to the problem can smooth the value function, but sometimes this approximation is unacceptable for representing the underlying problem. For example, the mountain-car problem, often studied in reinforcement literature, has a significant discontinuity in its value function

which, if smoothed over, might produce a poor solution [50].

B.2 Discretization

Although the Euler-Maruyama method can be used to discretize a continuous-time problem into a discrete-time problem, other methods could be used, including avoiding the continuous-time representation altogether. While this work began by proposing a method to solve continuous-time problems, the discrete-time formulation begins with a discrete-time representation (4.10), (4.12)

$$X_{i+1} = X_i + F_i^\mu + \Sigma_i W_i^{\mathsf{Q}}, \quad X_0 = x_0,$$
$$V_i^\mu(X_i) = \mathbf{E}_{\tilde{\mathsf{Q}}}[\sum_{j=i}^{N-1} L_j^\mu + g(X_N)\,|X_i],$$

and thus, we can start with this representation alone. If we choose to use the Euler-Maruyama approximation, the functions can be obtained as

$$F_i^\mu := F_i(X_i, \mu_i(X_i)), \qquad\qquad L_i^\mu := L_i(X_i, \mu_i(X_i)),$$

where $\mu_i : \mathbb{R}^n \to \mathcal{U}$ is a discrete-time policy and

$$F_i(x,u) = \Delta t\, f(t_i, x, u), \qquad L_i(x,u) = \Delta t\, \ell(t_i, x, u), \qquad \Sigma_i(x) = \sqrt{\Delta t}\, \sigma(t_i, x),$$

where Δt is the discretization time interval.

For systems whose dynamics are highly nonlinear, F_i^μ could be a multiple-step progression of deterministic dynamics over finer time-interval steps, i.e.,

$$F_i(x,u) = x_{\overline{M}},$$
$$x_{j+1} = F_{i,j}(x_j, u), \quad j = 0, \ldots, \overline{M} - 1, \quad x_0 = x,$$

and/or a deterministic Runge-Kutta scheme. Although the SDE might not be precisely represented, faithfulness to the deterministic dynamics in many applications is probably more important than faithfulness to the SDE, especially in problems where noise is arbitrarily added for the purposes of applying the method properly.

B.2.1 Control-Dependent Diffusion

As a brief note, the choice to make Σ_i a function dependent only on state and not control is somewhat arbitrary and, with appropriate care, we could reintroduce this dependence on the policy, substituting Σ_i with Σ_i^μ. The primary modification to the results of this work would pertain to policy optimization schemes and related comparison theorems.

B.3 Forward Sampling

In this section we discuss selection of the off-policy forward sampling distributions, characterized by the choice of drift process $\{K_i\}$ in the off-policy FSDE

$$X_{i+1} = X_i + K_i + \Sigma_i W_i^{\mathsf{P}}, \quad X_0 = x_0.$$

In Chapter 5 we discussed two types of sampling schemes, parallel-sampled, where each trajectory is sampled independently of the others, and branch-sampled, where the trajectory samples arise from a tree structure. It is important to note that the branch-sampling methodology is a generalization of the parallel-sampled methodology and so we can always use the former to represent the latter. In fact, it is sometimes useful to warm-start the FBRRT sampling with parallel-sampled trajectories following constant input, then expand the tree from these initial trajectories.

There are costs and benefits to using parallel-sampled or branch-sampled distributions. Multi-processing frameworks like CUDA can sample large numbers of parallel-sampled trajectories very quickly, whereas the nearest-neighbor methods, like KD-trees, relied upon

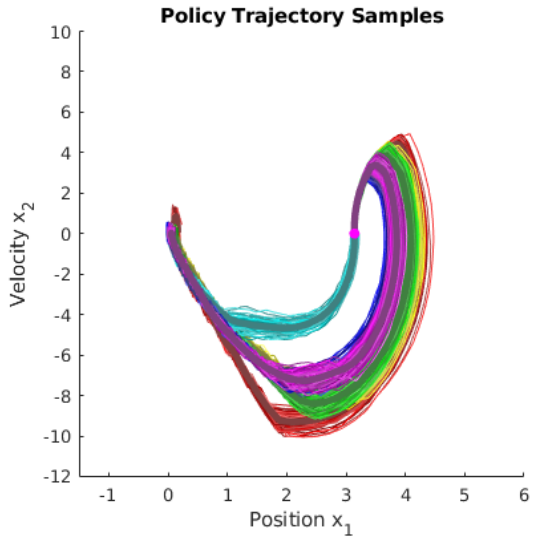


Figure B.1: Trajectory distributions following the approximated optimal policy for the L^1-inverted pendulum problem after different iterations of the FBRRT algorithm. The first iteration is colored red, followed by yellow, green, cyan, dark blue, and magenta. Thick lines are mean trajectories.

by algorithms like RRT are very challenging to parallelize and thus become a bottleneck. Further, RRT sampling frequently encounters the problem of encountering nodes with poor local support for function regression. On the other hand, as was demonstrated in the results of Chapter 5, we can often produce a near-optimal policy after just a handful of iterations by more broadly sampling the reachable space. Thus, the choice of parallel or branched-sampling depends largely on how much initial knowledge can be incorporated into the algorithm about the likely distribution of optimal trajectories.

In underactuated systems, especially for problems where costs are non-quadratic, it might be challenging to provide a near-optimal policy because there might be several locally optimal policies. Consider, for example, the trajectory distributions illustrated in Figure B.1 for the underactuated L^1-inverted pendulum problem. The expected cost of these trajectory distributions are relatively similar across iterations, and thus the cyan distribution is in a locally optimum basin, separated from the globally optimal magenta basin. The FBRRT algorithm maintains sampling distributions in different parts of the state space and thus can potentially consider multiple optimal basins during its backward pass. A parallel-

sampled distribution might get stuck near the cyan distribution, unable to explore broadly enough to find the policy which begins with controls moving in the opposite direction.

If speed is of higher concern, locally optimal policies are sufficient, and a good initial guess is available, parallel-sampling is likely to be faster and more robust for those purposes. Densely clustered collocation points are more likely to robustly produce quality function approximations in regression. Exploration can be better achieved by adding Brownian noise to the current policy dynamics since all that is needed is local exploration.

B.3.1 Feedback Linearizable Systems

We say a system is feedback linearizable if we have a feedback policy π which can cancel out the nonlinear dynamics, yielding a linear system. Feedback linearizable systems are especially well suited for iFBSDE methods under the assumption that the optimal policy is near some linearized version of the controller. To see this, suppose we have the feedback policy $\pi_i(x, v)$ which feedback linearizes the dynamics,

$$F_i(x, \pi_i(x, v)) = A_i x + B_i v,$$

where v is some alternative input. We can select the drift $\{K_i\}$ by choosing v as a linear feedback controller $v = \nu_i(x) = G_i x + g_i$ designed according to any linear control design method, e.g., LQR. The drift $K_i = A_i X_i + B_i(G_i X_i + g_i) = \widetilde{A}_i X_i + \widetilde{b}_i$ is thus parameterized by the gain matrices of the linearized controller (G_i, g_i). The improvement of the forward sampling after every iteration of iFBSDE is the selection of these linear gains (G_i, g_i), which is easy to choose so as to guarantee good behavior using basic linear control theory. The drift distribution selected in this way is realizable since it is sampled according to a controller which could be executed under the feedback linearizable assumption.

Though the selection of drift can be constrained to the selection of linear gains, the value function will still correspond to the nonlinear problem thanks to the off-policy estimator

construction. That is, the optimal policy produced by iFBSDE methods will be optimal with respect to the original nonlinear dynamics, not the transformed system. Further research must be performed on how to update the gain matrices $(\widetilde{A}_i, \widetilde{b}_i)$ with each iteration based on the estimated value function $\widetilde{V}_i^\mu$.

B.4 Estimators

The two proposed estimators for the value function are Taylor noiseless

$$\widehat{Y}_i^{\text{noiseless}} = L_i^\mu + \overline{Y}_{i+1} + \overline{Z}_{i+1}^\top D_i + \frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}(I + D_i D_i^\top)),$$

and Taylor re-estimate

$$\widehat{Y}_i^{\text{re-est}} = \widetilde{V}_{i+1}^\mu(X_{i+1}) + L_i^\mu - \overline{Z}_{i+1}^\top W_i^{\mathsf{P}} + \overline{Z}_{i+1}^\top D_i + \frac{1}{2}\operatorname{tr}(\overline{M}_{i+1}(I + D_i D_i^\top - W_i^{\mathsf{P}} W_i^{\mathsf{P}\top})),$$

where

$$\begin{aligned}
\overline{X}_{i+1}^{\mathsf{P}} &:= \mathbf{E}_{\widetilde{\mathsf{P}}}[X_{i+1}|X_i, K_i] = X_i + K_i, \\
\overline{Y}_{i+1} &:= \widetilde{V}_{i+1}^\mu(\overline{X}_{i+1}^{\mathsf{P}}), \\
\overline{Z}_{i+1} &:= \Sigma_i^\top \partial_x \widetilde{V}_{i+1}^\mu(\overline{X}_{i+1}^{\mathsf{P}}), \\
\overline{M}_{i+1} &:= \Sigma_i^\top \partial_{xx} \widetilde{V}_{i+1}^\mu(\overline{X}_{i+1}^{\mathsf{P}})\Sigma_i, \\
D_i &:= \Sigma_i^{-1}(F_i^\mu - K_i).
\end{aligned}$$

The noiseless estimator is preferable in general since its variance is low. When D_i is small or zero-valued and a large number of samples are included in the Monte Carlo sampling, the re-estimate estimator provides lower bias.

When the dimension of the problem is high, one of the challenges with using estimators like these is the computation of the Hessian $\partial_{xx}\widetilde{V}_{i+1}^\mu$, since it is a matrix with $O(n^2)$

elements. When the value function representation is quadratic and Σ_i is constant this is not a problem because $\overline{M}_{i+1}$ will be constant for all collocation points. However, for many problems, diffusion is not large in every dimension, especially since it has been added arbitrarily using the vanishing viscosity method. Thus a potential solution to this problem is to set the small elements in Σ_i to zero in the computation of $\overline{M}_{i+1}$, and only evaluate the relevant second derivatives.

B.5 Local-Entropy Weighted LSMC

The method proposed for approximating the value function is local-entropy weighted LSMC, (5.12),

$$\alpha_i^* = \arg\min_{\alpha \in \mathcal{A}} \sum_{k=1}^{M} \exp(-\frac{1}{\lambda}\rho_{i+1}^k)(\widehat{y}_i^k - \phi(x_i^k;\alpha))^2,$$

where $\{(\rho_{i+1}^k, x_i^k, \widehat{y}_i^k)\}_{k=1}^M$ are the sample heuristic to be minimized, collocation points in the state space, and estimator values, respectively, and $\lambda > 0$ is a tuning variable. Setting the heuristic is an open research question, but the path-integrated heuristic, (5.15),

$$\rho_{i+1} = \sum_{j=0}^{i} L_i(X_i, u_i) + \widetilde{V}_{i+1}^{\mu}(X_{i+1}),$$

where $\{u_i\}$ is the control sampled by the drift, has the convenient interpretation as being the equivalent to minimizing a dynamic programming problem resulting in the optimal value at the initial time.

In practice, much care and attention must be afforded tuning λ. One strategy applied to higher dimensional problems was to try different values in separate backward passes and using the backward pass which results in the best performance. Finding more principled methods of tuning this variable is a topic of future research, but structuring the weighted LSMC in this way is likely to be crucial for branch-sampled distributions.

B.6 Value Function Representation

Although not a primary focus of this work, how the value function is represented has a significant impact on the efficacy of these methods. The only type of model studied in this work are linear combinations of polynomial basis functions, namely Chebyshev functions. For most cases we used a basis with polynomials up to degree 2. For the 1-dimensional case, we demonstrated that using higher order polynomials can significantly improve accuracy, up to a point of diminishing returns. When the dimension of the problem is high, using higher order polynomials begins to become challenging since the number of basis functions is $O(n^k)$ where n is the state space dimension and k is the order of polynomial desired.

Especially for high-dimensional systems, another model which could be explored is deep neural networks. The time required to train such large models changes the paradigm from a method which might take on the order of seconds or minutes to hours or days. As discussed in the conclusion, other parts of the algorithm can be modified to produce a policy gradient algorithm.

B.7 Policy Improvement

Again, methods for improving the policy between iterations were not a primary focus for this work and require problem-based attention. When dynamics are control-affine and running costs are L^2 or L^1 with respect to u (or independent of u), there is theoretically an analytic solution for the optimal policy with respect to an optimal value function.

One of the biggest challenges iFBSDE faces is large jumps in policy, which results in instability since the trajectories enter a part of the state space for which the value function is poorly approximated. Although not explored in this work, methods which change the policy by small amounts instead of trying to optimize the policy completely might perform better, especially for parallel-sampled algorithms. Again, this motivates the reorienting of

the methods in this work to a policy-gradient method from deep reinforcement learning.

REFERENCES

[1] J. Yong and X. Y. Zhou, *Stochastic Controls: Hamiltonian Systems and HJB Equations*. Springer, 1999, vol. 43.

[2] E. Pardoux and S. G. Peng, "Adapted solution of a backward stochastic differential equation," *Systems and Control Letters*, vol. 14, no. 1, pp. 55–61, 1990.

[3] S. Peng, "Backward stochastic differential equations and applications to optimal control," *Applied Mathematics and Optimization*, vol. 27, no. 2, pp. 125–144, 1993.

[4] N. El Karoui, S. Peng, and M. C. Quenez, "Backward stochastic differential equations in finance," *Mathematical Finance*, vol. 7, no. 1, pp. 1–71, 1997.

[5] M. Bardi and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*. 1997, vol. 1, pp. 1689–1699, ISBN: 978-0-8176-4754-4. arXiv: arXiv:1011.1669v3.

[6] W. H. Fleming and H. M. Soner, *Controlled Markov Processes and Viscosity Solutions*. Springer, 2006, vol. 25.

[7] W. H. Fleming and R. W. Rishel, "Deterministic and stochastic optimal control," *Bulletin of the American Mathematical Society*, vol. 82, pp. 869–870, 1976.

[8] C. Pierre, "Introduction to differential games," Tech. Rep., 2010.

[9] R. Buckdahn and J. Li, "Stochastic Differential Games and Viscosity Solutions of Hamilton-Jacobi-Bellman-Isaacs Equations," no. 10426022, 2007. arXiv: 0702131 [math].

[10] L. Evans and P. E. Souganidis, *Differential games and representation formulas for solutions of Hamilton-Jacobi-Isaacs equations*, 1984.

[11] S. Osher, "A Level Set Formulation for the Solution of the Dirichlet Problem for Hamilton–Jacobi Equations," *SIAM Journal on Mathematical Analysis*, vol. 24, no. 5, pp. 1145–1152, 1993.

[12] R. Buckdahn and T. Nie, "Generalized Hamilton-Jacobi-Bellman equations with Dirichlet boundary and stochastic exit time optimal control problem," pp. 1–29, 2015. arXiv: arXiv:1412.0730v4.

[13] A. B. Kurzhanski and P. Varaiya, "Reachability Under Uncertainty," in *Proceedings of the 41st lEEE Conference on Decision and Control*, 2002, pp. 1982–1987, ISBN: 0780375165. arXiv: 1206.5253.

[14] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–957, 2005.

[15] J. Calder, "Some notes on viscosity solutions of Hamilton-Jacobi equations," 2018.

[16] I. Exarchos and E. A. Theodorou, "Stochastic optimal control via forward and backward stochastic differential equations and importance sampling," *Automatica*, vol. 87, pp. 159–165, 2018.

[17] I. Exarchos, E. A. Theodorou, and P. Tsiotras, "Stochastic L^1-optimal control via forward and backward sampling," *Systems and Control Letters*, vol. 118, pp. 101–108, 2018.

[18] ——, "Game-theoretic and risk-sensitive stochastic optimal control via forward and backward stochastic differential equations," in *Conference on Decision and Control*, Las Vegas, NV, 2016, pp. 6154–6160, ISBN: 9781509018376.

[19] ——, "Stochastic Differential Games: A Sampling Approach via FBSDEs," *Dynamic Games and Applications*, 2018.

[20] J. Zhang, "A numerical scheme for BSDEs," *Annals of Applied Probability*, vol. 14, no. 1, pp. 459–488, 2004.

[21] C. Bender and R. Denk, "A forward scheme for backward SDEs," *Stochastic Processes and their Applications*, 2007.

[22] D. Becherer and P. Turkedjiev, "Multilevel approximation of backward stochastic differential equations," pp. 1–42, 2014. arXiv: 1412.3140.

[23] C. Bender and T. Moseler, "Importance sampling for backward SDEs," *Stochastic Analysis and Applications*, vol. 28, no. 2, pp. 226–253, 2010.

[24] E. Gobet, J. P. Lemor, and X. Warin, "A regression-based Monte Carlo method to solve backward stochastic differential equations," *Annals of Applied Probability*, vol. 15, no. 3, pp. 2172–2202, 2005.

[25] F. A. Longstaff and E. S. Schwartz, "Valuing American options by simulation: A simple least-squares approach," *Review of Financial Studies*, vol. 14, pp. 113–147, 2001.

[26] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*. 2002, ISBN: 0387954821.

[27] I. M. Mitchell, "A toolbox of level set methods," *Department of Computer Science, University of British Columbia, Vancouver, BC, Canada, Tech. Rep. TR-2004-09, July*, pp. 177–247, 2004.

[28] S. Yensiri, "An Investigation of Radial Basis Function-Finite Difference (RBF-FD) Method for Numerical Solution of Elliptic Partial Differential Equations," *Mathematics*, vol. 5, no. 54, 2017.

[29] M. Li, W. Chen, and C. S. Chen, "The localized RBFs collocation methods for solving high dimensional PDEs," *Engineering Analysis with Boundary Elements*, vol. 37, no. 10, pp. 1300–1304, 2013.

[30] D. H. Jacobson and D. Q. Mayne, *Differential dynamic programming*. New York, NY: North-Holland, 1970.

[31] E. A. Theodorou, Y. Tassa, and E. Todorov, "Stochastic differential dynamic programming," in *American Control Conference*, Baltimore, MD, 2010, pp. 1125–1132.

[32] M. Giftthaler, M. Neunert, M. Stauble, J. Buchli, and M. Diehl, "A Family of Iterative Gauss-Newton Shooting Methods for Nonlinear Optimal Control," *IEEE International Conference on Intelligent Robots and Systems*, pp. 6903–6910, 2018. arXiv: 1711.11006.

[33] Y. Tassa, T. Erez, and W. D. Smart, "Receding Horizon Differential Dynamic Programming," in *Advances in Neural Information Processing Systems 20*, 2008, pp. 1465–1472.

[34] Y. Tassa, *Theory and Implementation of Biomimetic Motor Controllers (Ph.D. Thesis)*. Hebrew University of Jerusalem, 2011.

[35] G. Williams, N. Wagener, B. Goldfain, P. Drews, J. M. Rehg, B. Boots, and E. A. Theodorou, "Information theoretic MPC for model-based reinforcement learning," in *International Conference on Robotics and Automation, Singapore*, IEEE, 2017, pp. 1714–1721.

[36] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: theory and practice—a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[37] S. Peng, "A general stochastic maximum principle for optimal control problems," *SIAM Journal on control and optimization*, vol. 28, no. 4, pp. 966–979, 1990.

[38] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[39] M. Herceg, M. Kvasnica, C. N. Jones, and M. Morari, "Multi-Parametric Toolbox 3.0," in *Proc. ̃of the European Control Conference*, Zürich, Switzerland, 2013, pp. 502–510.

[40] H. J. Kappen, "Linear theory for control of nonlinear stochastic systems," *Physical review letters*, vol. 95, no. 20, p. 200 201, 2005.

[41] V. Gómez, H. J. Kappen, J. Peters, and G. Neumann, "Policy search for path integral control," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8724 LNAI, no. PART 1, pp. 482–497, 2014.

[42] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Aggressive driving with model predictive path integral control," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2016-June, pp. 1433–1440, 2016.

[43] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[44] N. Heess, G. Wayne, D. Silver, T. Lillicrap, Y. Tassa, and T. Erez, "Learning continuous control policies by stochastic value gradients," *Advances in Neural Information Processing Systems*, vol. 2015-Janua, pp. 2944–2952, 2015. arXiv: 1510.09142.

[45] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *35th International Conference on Machine Learning (ICML)*, vol. 5, pp. 2976–2989, 2018. arXiv: arXiv:1801.01290v2.

[46] T. Wang, X. Bao, I. Clavera, J. Hoang, Y. Wen, E. Langlois, S. Zhang, G. Zhang, P. Abbeel, and J. Ba, "Benchmarking model-based reinforcement learning," *arXiv preprint arXiv:1907.02057*, 2019.

[47] P. Morere, G. Francis, T. Blau, and F. Ramos, "Reinforcement Learning with Probabilistically Complete Exploration," 2020. arXiv: 2001.06940.

[48] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Advances in neural information processing systems*, 2000, pp. 1057–1063.

[49] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[50] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[51] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *The International Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.

[52] S. Karaman and E. Frazzoli, "Incremental sampling-based algorithms for optimal motion planning," *Robotics Science and Systems VI*, vol. 104, no. 2, 2010.

[53] I. Noreen, A. Khan, and Z. Habib, "Optimal path planning using RRT* based approaches: a survey and future directions," *Int. J. Adv. Comput. Sci. Appl*, vol. 7, no. 11, pp. 97–107, 2016.

[54] O. Arslan, E. A. Theodorou, and P. Tsiotras, "Information-theoretic stochastic optimal control via incremental sampling-based algorithms," in *IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning, Orlando, FL*, 2014.

[55] I. Exarchos, E. A. Theodorou, and P. Tsiotras, "Stochastic L1-optimal control via forward and backward sampling," *Systems and Control Letters*, vol. 118, pp. 101–108, 2018.

[56] S. Peng, "Probabilistic interpretation for systems of quasilinear parabolic partial differential equations," *Stochastics Stochastics Rep*, vol. 37, no. 1-2, pp. 61–74, 1991.

[57] S. Resnick, *A Probability Path*. Birkhäuser Verlag AG, 2003.

[58] S. N. Cohen and R. J. Elliott, *Stochastic calculus and applications*. Springer, 2015, vol. 2.

[59] G. Lowther, *Girsanov transformations*, May 2010.

[60] A. Pascucci, *PDE and Martingale Methods in Option Pricing*. Springer Science & Business Media, 2011.

[61] F. J. Fabozzi, T. Paletta, and R. Tunaru, "An improved least squares Monte Carlo valuation method based on heteroscedasticity," *European Journal of Operational Research*, vol. 263, no. 2, pp. 698–706, 2017.

[62] P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*. Springer Science and Business Media, 2013, vol. 23.

[63] D. J. Higham, "An introduction to multilevel Monte Carlo for option valuation," *International Journal of Computer Mathematics*, vol. 92, no. 12, pp. 2347–2360, 2015. arXiv: 1505.00965.

[64] E. L. Crow and K. Shimizu, *Lognormal Distributions*. Marcel Dekker New York, 1987.

[65] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd. Elsevier Inc., 2011.

[66] P. Del Moral, *Mean Field Simulation for Monte Carlo Integration*. Chapman and Hall/CRC, 2013.

[67] E. A. Theodorou and E. Todorov, "Relative entropy and free energy dualities: Connections to path integral and KL control," in *IEEE Conference on Decision and Control, Maui, Hawaii*, IEEE, 2012, pp. 1466–1473.

[68] R. Tedrake, "Underactuated Robotics: Learning, Planning, and Control for Efficient and Agile Machines: Course Notes for MIT 6.832," *Working Draft Edition*, vol. 3, 2009.

[69] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles," *arXiv preprint arXiv:*, pp. 1–27, 2016. arXiv: 1604.07446.

[70] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *4th International Conference on Learning Representations (ICLR)*, 2016. arXiv: 1509.02971.

[71] T. Wang, X. Bao, I. Clavera, J. Hoang, Y. Wen, E. Langlois, S. Zhang, G. Zhang, P. Abbeel, and J. Ba, "Benchmarking Model-Based Reinforcement Learning," pp. 1–25, 2019. arXiv: 1907.02057.

[72] V. I. Bogachev, *Measure theory*. Springer Science & Business Media, 2007, vol. 1.

[73] B. Gravell, P. M. Esfahani, and T. H. Summers, "Learning optimal controllers for linear systems with multiplicative noise via policy gradient," *IEEE Transactions on Automatic Control*, 2020.