Abstract—We propose a numerical method to solve forward-backward stochastic differential equations (FBSDE) arising in stochastic optimal control problems. Instead of sampling forward paths independently, we demonstrate how a rapidly-exploring random tree (RRT) method can be utilized for the forward integration pass, as long as the controlled drift terms are appropriately compensated in the backward integration pass. We show how a value function approximation is produced by solving a series of function approximation problems backwards in time along the edges of the constructed RRT tree. We employ a local entropy-weighted least squares Monte Carlo (LSMC) method to concentrate function approximation accuracy in regions most likely to be visited by optimally controlled trajectories. We demonstrate the proposed method and evaluate it on two nonlinear stochastic optimal control problems with non-quadratic running costs, showing that it can greatly improve convergence over previous FBSDE numerical solution methods.

I. INTRODUCTION

The theory of forward-backward stochastic differential equations (FBSDEs) has been gaining traction as a framework to solve nonlinear stochastic control problems, including optimal control problems with quadratic cost [6], minimum-fuel ($L_1$-running cost) problems [8], differential games [7], and reachability problems [6, 23]. Although initial results demonstrate promise in terms of flexibility and theoretical validity, numerical algorithms which leverage this theory have not yet matured. For even modest problems, state-of-the-art algorithms often have issues with slow and unstable convergence to the optimal policy. Producing more robust numerical methods is critical for the broader adoption of FBSDE methods in real-world tasks.

FBSDE numerical solution methods broadly consist of two steps, a forward pass, which generates Monte Carlo samples of the forward stochastic process, and a backward pass, which iteratively approximates the value function backwards in time. Typically, FBSDE methods perform this approximation with a scheme called least-squares Monte Carlo (LSMC), which uses conditional expectation projection to implicitly solve the backward SDE by turning the problem into a parametric function approximation problem [15]. Often, the approximate value function fit in the backward pass is used to improve sampling in an updated forward pass, leading to an iterative algorithm which, ideally, improves the approximation till convergence. Although FBSDE methods share a distinct similarity to differential dynamic programming (DDP) techniques [13, 27, 25], DDP is generally less flexible. For most DDP applications, a strictly positive definite running cost with respect to the control is required for convergence [24, Section 2.2.3]. Furthermore, in DDP, the computation of first and second order derivatives of both dynamics and costs is necessary for the backward pass, making it challenging to apply this approach to problems where these derivatives are not known analytically. In contrast, FBSDE techniques only require a good fit of the value function and evaluation of the gradient of this value function to obtain the optimal control.

The flexibility of FBSDE algorithms stems from their relation with Feynman-Kac-type formulae, which connects the solution of a broad class of second-order parabolic or elliptic

Fig. 1: Forward sampling tree for the first iteration of the $L_1$ inverted pendulum problem. Hue corresponds to the path integral heuristic $\mathcal{Y}^P$ 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.
PDEs to the solution of FBSDEs \[28\], brought to prominence in Pardoux and Peng \[19\], Peng \[20\], El Karoui et al. \[5\]. Both Hamilton-Jacobi-Bellman (HJB) and Hamilton-Jacobi-Isaacs (HJI) second order PDEs, utilized for solving stochastic optimal control and stochastic differential game equations respectively, can thus be solved via FBSDE methods, even when the costs and dynamics are nonlinear. This theoretically allows for the solution of PDEs typically solved using grid-based methods such as the Level Set Toolbox \[17\], known for poor scaling with higher dimensional systems \((n \geq 4)\).

Although FBSDE methods have received some attention in the past, especially in the mathematical finance community \[2 \[15\] [16]\], application of these methods to physical systems has only seen significant progress recently \[6 \[9\] [8]\]. The primary advantage of these methods is that they produce an unbiased estimator for the value function associated with the HJB equations. The challenge with their utilization is that naïve application of the theory leads to estimators with high variance. Simply put, if the forward pass never produces sample trajectories close to the optimal trajectories, then the value function cannot be estimated accurately in those regions. Recent work has shown that Girsanov’s theorem can be used to change the sampling measure of the forward pass without adding intrinsic bias to the estimator \[6 \[8\] [9]\]. That is, a change over probability spaces corresponds to the introduction of a drift to the forward SDE that can be employed to modify the sampling in the forward pass; this, in turn, requires appropriate accommodation of the change of measures in the backward pass.

In this work we expand upon this result, by showing that the forward sampling measure can be modified at will, which enables us to incorporate methods from other domains, namely, RRTs \[14\] (recent survey in Noreen et al. \[18\]), in order to more efficiently explore the state space in the forward pass. RRTs are frequently applied to reachability-type motion planning problems, biasing the samples towards regions of the state space that have low density. 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.

The primary contributions of this paper are as follows:

- Framing forward drift selection as a McKean-Markov branched particle sampler, allowing the use of RRTs and similar methods in the forward pass of FBSDE methods.
- Showing how local-entropy-weighted LSMC techniques can be used to concentrate value function approximation accuracy in regions likely to have optimal trajectories, to compensate for the broader search induced by the forward pass.
- Demonstrating how the forward sampling RRT tree can be pruned and re-expanded to create an iterative method.

We call the proposed method forward-backward rapidly exploring random trees (FBRRT). After we describe the approach in both theory and numerical implementation, we apply FBRRT to a few problems, compare it to Exarchos et al. \[8\], and demonstrate its ability to solve nonlinear stochastic optimal control problems with non-quadratic running costs. Although local-entropy path integral theory and RRTs have been used together in \[11\], called PI-RRT, this method is more closely related to the path-integral approach to control \[27\]. Our method similarly performs forward passes to broadly sample the state space, but follows them with backward passes to obtain approximations for the value functions, and consequently obtaining closed loop policies over the full horizon.

**II. FORWARD BACKWARDS STOCHASTIC DIFFERENTIAL EQUATIONS AND OPTIMAL CONTROL**

In this section, we briefly introduce the theoretical foundation behind FBSDE methods and demonstrate how they allow us to solve stochastic optimal control problems. We start with a complete, filtered probability space \((\Omega, \mathcal{F}, \mathbb{P})\), on which \(W^P\) is an \(n\)-dimensional standard Brownian (Wiener) process with respect to the probability measure \(\mathbb{P}\) and adapted to the filtration \(\mathcal{F}_t\), \(t \in [0,T]\). Consider a stochastic nonlinear system governed by the control-affine Itô differential equation

\[
\mathrm{d}X_s = (f(s, X_s) + G(s, X_s)u_s)\mathrm{d}s + \Sigma(s, X_s)\mathrm{d}W_s^P, \tag{1}
\]

where \(X_s\) is a \(\mathcal{F}_s\)-progressively measurable state process on the interval \(s \in [0, T]\) taking values in \(\mathbb{R}^n\), \(u_s\) is a progressively measurable input process taking values in the convex compact set \(U \subseteq \mathbb{R}^m\), and \(f: [0,T] \times \mathbb{R}^n \to \mathbb{R}^n\), \(G: [0,T] \times \mathbb{R}^n \to \mathbb{R}^{n \times m}\), \(\Sigma: [0,T] \times \mathbb{R}^n \to \mathbb{R}^{n \times n}\) are bounded functions, continuously differentiable in \(s\), and twice continuously differentiable in \(x\). We further assume that \(\Sigma\) is uniformly strictly positive definite.

Associated with (1) is the stochastic optimal control problem, seeking to determine the value function \(v^*: [0, T] \times \mathbb{R}^n \to \mathbb{R}^+\) defined by

\[
v^*(t, x_t) = \inf_{u(\cdot)} \{ J_{P_T}(t, x_t; u(\cdot)) \}, \tag{SOP}
\]

\[
J_{P_T}(t, x_t; u(\cdot)) := E_{P_T}^{t,x_t}[g(X_T) + \int_t^T L(s, X_s, u_s)\mathrm{d}s],
\]

where \(E_{P_T}[\cdot] := E_{P_T}[\cdot | X_t = x_t]\) denotes the conditional expectation given \(X_t = x_t\) under the probability measure \(P_T\), and where \(L: [0, T] \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^+\) is bounded, continuously differentiable in \(s\), twice continuously differentiable in \(x\), and \(g: \mathbb{R}^n \to \mathbb{R}^+\) is bounded and thrice continuously differentiable in \(X_s\). Below, and for brevity of exposition, the arguments \((s, X_s)\) are omitted from the functions and we write, for instance, \(f_s \equiv f(s, X_s)\) and \(L_n(u) \equiv L(s, X_s, u_s)\).
It is well known, that the solution \( v^* \) to the (SOP) problem satisfies the Hamilton-Jacobi-Bellman (HJB) PDE \[11\], Chapter 4, Corollary 3.2]

\[
v_t + v_x f_s + \frac{1}{2} \text{tr}(\Sigma_s^T v_{xx} \Sigma_s) + h_x(x, \Sigma_s v_x) = 0, \quad (HJB)
\]

where, by a slight abuse of notation, we denote the partial derivative with respect to state \( v_x := \frac{\partial}{\partial x} \), and the Hessian with respect to state \( v_{xx} := \frac{\partial^2}{\partial x^2} \). By Fleming and Soner \[11\], Chapter 4, Theorem 4.2, Theorem 4.4], under the parameter assumptions and especially the uniform positive definiteness of \( \Sigma \), the (HJB) equations possess a unique classical solution \( v^* \in C^{1,2}(\Omega, \mathbb{F}_T) \times \mathbb{R}^n \) which coincides with the value function defined by the (SOP) problem. To satisfy the theorem in the following paragraph, in addition to the assumptions above, we further assume that \( L \) is selected such that \( h \) is uniformly continuous in \((t, x)\) and Lipschitz continuous in \( \lambda \).

By the Feynman-Kac theorem \[28\], Chapter 7, Theorem 4.5, (4.29)], the solution \( v^* \) to (HJB) has a representation

\[
Y_s = v^*(s, X_s), \quad s \in [0, T],
\]

\[
Z_s = \Sigma_s^T v_x^*(s, X_s), \quad \text{a.e.} s \in [0, T],
\]

\(P_0^T\)-almost surely (a.s.), where \((X_s, Y_s, Z_s)\) is the solution to the forward-backward SDEs

\[
dX_s = f_s ds + \Sigma_s dW_s^P, \quad X_0 = x_0, \quad (4)
\]

\[
dY_s = -h_x(X_s, Z_s) ds + Z_s^T dW_s^P, \quad Y_T = g(X_T), \quad (5)
\]

and \(W_s^P\) is a Brownian process in the filtered probability space \((\Omega, \mathcal{F}_t, \mathbb{P}_T, P_0^P)\).

\[A. \text{ Drifted FBSDEs}\]

We now show a result first discussed in Bender and Moseler \[3\] and later in Exarchos and Theodorou \[6\], which states that adding a drift term to the forward SDE \((4)\) and appropriately compensating in the backward SDE \((5)\) only changes the measure under which \(P_0^T\) holds. To this end, let \(W_s^P\) be a Brownian process in \((\Omega, \mathcal{F}_t, \mathbb{P}_T, P_0^P)\), and let both \(K_s^Q\) and \(K_s^P\) be any bounded adapted processes.

Define the Doléans-Dade exponential process

\[
\Theta_t^Q := \exp(\int_0^t (K_s^Q - K_s^P)^T dW_s^P - \frac{1}{2} \int_0^t \|K_s^Q - K_s^P\|^2 ds), \quad t \in [0, T],
\]

\[1\]

where \(\Theta_t^Q\) is the Radon-Nikodym derivative between the two measures \(Q_t\) and \(P_t\), restrictions of \(Q_T\) and \(P_T\) to \(\mathcal{F}_t\) (i.e. trajectories over the interval \([0, t]\)). By Girsanov’s theorem (see e.g., Fleming and Rishel \[10\], Chapter 5, Theorem 10.1), \((\Omega, \mathcal{F}_t, \mathbb{F}_T, Q_T)\) is a probability space and, furthermore,

\[
dW_s^Q := -(K_s^Q - K_s^P) ds + dW_s^P, \quad (8)
\]

defines a standard Brownian process with respect to the measure \(Q\). A further consequence of Girsanov’s theorem is illustrated, through an abuse of notation, by substituting this relationship into the equations \((4, \[5\])\), taking \(P_t\) to be \(P_0^T\) such that \(K_t^P = 0\). Performing this substitution shows that the solution \((X_s, Y_s, Z_s)\) of the FBSDE defined by \((4, \[5\])\) in the filter probability space \((\Omega, \mathcal{F}_t, \mathbb{F}_T, P_0^T)\) corresponds to the solution of the drifted FBSDE

\[
dX_s = (f_s + \Sigma_s K_s^Q) ds + \Sigma_s dW_s^Q, \quad (9)
\]

\[
dY_s = -(h_x(X_s, Z_s) - Z_s^T K_s^Q) ds + Z_s^T dW_s^Q, \quad (\text{FBSDE})
\]

\[X_0 = x_0, \quad Y_T = g(X_T), \quad \text{in the space} \quad (\Omega, \mathcal{F}_t, \mathbb{F}_T, Q_T). \]

Henceforth, we refer to \(K_s^Q\) as the drift process.

\[B. \text{ Equivalent Measures}\]

The random variable \(\Theta_t^Q\) can be interpreted as the density of the path measure \(Q_t\) with respect to the path measure \(P_t\). Further, since it is strictly positive with \(E_{P_t}[\Theta_t^Q] = 1\), the measure \(Q_t\) is called an equivalent measure to \(P_t\), that is, \(Q_t\) is absolutely continuous with respect to \(P_t\) and vice versa \[11\], Remark, p. 142]. Equivalent probability measures have the characteristic that whenever a property holds a.s. in terms of one measure, the same property holds a.s. in the equivalent measure. Thus, the equalities in \((3)\) hold \(Q_t\)-a.s., in addition to \(P_0^T\)-a.s.. An important consequence of the equivalence under this change of measure is that, by applying the conditional expectation to both sides of the equalities,

\[
E_{Q_t}[v^*(t, x_t, \mathcal{F}_t)] = \mathbb{E}_{Q_t}[\mathbb{E}_{P_t}[\Theta_t^Q] v^*(t, x_t, \mathcal{F}_t)] = \mathbb{E}_{Q_t}[\mathbb{E}_{P_t}[\Theta_t^Q] v^*(t, x_t, \mathcal{F}_t)] = 1.
\]

for \((t, x_t) \in [0, T] \times \mathbb{R}^n\), and any \(Q_t\) defined according to \((7)\).

A related result to Girsanov’s theorem (see Proposition 1.1 in the appendix) shows that if we, conversely, define the random variable \(\Theta_t^Q\) directly, there exists some drift \(K_s^Q\) with the same relationships discussed above. This demonstrates that we can produce measures equivalent to \(P_0^T\) in two different ways, either by choosing any drift process \(K_s^Q\), or reweighing the measure directly via a normalized likelihood random variable \(\Theta_t^Q\) in an importance sampling-like scheme. Further, using the transitive property of equivalent measures (following from Shiryaev \[22\], Lemma (b), p. 231]), we can choose a drift process to produce the measure \(Q_t\), then reweigh this measure to produce another measure \(R_t\), both equivalent to \(P_t\).

\[III. \text{ McKean–Markov Formulation}\]

We now demonstrate how FBSDEs can be approximated by a branching particle process, drawing on the McKean-Markov theory presented in Del Moral \[4\]. We begin by introducing a
discretization of the SDEs and discuss how the forward process can be approximated by a scheme similar to a particle filter, where some particle trajectories terminate before the horizon and others branch into multiple paths. Next, we show how the backward pass estimates the value function by solving a series of weighted least squares problems in a path integral-like method.

A. Discretization of the FBSDE

The FBSDE is discretized according to a given partition of the interval \([0, T]\), \((t_0 = 0, t_1, t_2, \ldots, t_{N-1}, t_N = T)\). For the process \(X_t\) we shorten \(X_{t_i}\) to \(X_i\) for brevity, and similarly for the other processes. Let \((X_s, Y_s, Z_s)\) be any solution to the FBSDE computed using the drift \(K_0^Q\). By integrating the SDEs over each time step, we have that the relations

\[
X_{i+1} = X_i + \int_{t_i}^{t_{i+1}} (f_s + \Sigma_s K_s^Q)ds + \int_{t_i}^{t_{i+1}} \Sigma_s dW_s^Q, \tag{10}
\]

\[
Y_i = Y_i^Q - \int_{t_i}^{t_{i+1}} Z_s^T dW_s^Q, \tag{11}
\]

hold \(Q\)-a.s. for \(i = 0, \ldots, N - 1\), where

\[
Y_i^Q := Y_{i+1} + \int_{t_i}^{t_{i+1}} (h_s(X_s, Z_s) - Z_s^T K_s^Q)ds. \tag{12}
\]

Using \(\xi\), the boundary conditions at the end of the interval yield

\[
Y_N = g(X_N), \quad Z_N = \Sigma_N g_z(X_N).
\]

Furthermore, using \(\xi\) it can be shown (Proposition 1.2 in the appendix) that the random variable \(\nu^*(t_i, X_i)\) is the conditional expectation

\[
\nu^*(t_i, X_i) = E_Q[Y_i^Q|X_i]. \tag{13}
\]

The conditional expectation eliminates the Itô integral in (11) because it is zero mean when conditioned at the beginning of the step. For the purposes of approximating (12), note that if given the value function at the end of the step, then \(\xi\) allows us to induce the value

\[
Z_{i+1} = \Sigma_{i+1} v^*_{i+1}(t_{i+1}, X_{i+1}). \tag{14}
\]

B. McKean-Markov Branched Sampling

In this subsection, we show how the spatial branching models presented in Del Moral [4, Section 4.1] can be used to represent the distribution constructed by the forward SDE (10). We denote by \(Q_i\) the time-discrete approximation of \(Q_T\), and by \(Q_i\), its restriction to the interval \([0, t_i]\), and introduce the joint random variable \(\xi_i := (K_{i-1}^Q, X_i)\). We approximate the distribution of \(\xi_i\) with the empirical measure

\[
q_i^M := \frac{1}{M} \sum_{\ell=1}^{M} \delta_{\xi_i^\ell}, \tag{15}
\]

for \(i = 0, \ldots, N\), where \(\{\xi_i^\ell\}_{\ell=1}^{M}\) is a finite set of independent sample particles \(\xi_i^\ell := (k_{i-1}^\ell, x_i^\ell) \in \mathbb{R}^n \times \mathbb{R}^n\) and \(\delta_{\xi_i^\ell}\) is the Dirac-delta measure. We interpret the random variable \(K_{i-1}^Q\) as the constant value the piecewise constant drift process takes on the interval \([t_{i-1}, t_i]\) and \(X_i\) as the instantaneous value the process \(X_s\) takes at the end of this interval \(t_i\).

The distribution of the Markov chain \((\xi_0, \xi_1, \ldots, \xi_i)\) is approximated by a finite set of sample paths

\[
\{(\xi_0^\ell, \xi_1^\ell, \ldots, \xi_i^\ell)\}_{\ell=1}^{M},
\]

defined as

\[
Q_i^M := \frac{1}{M} \sum_{\ell=1}^{M} \delta_{(\xi_0^\ell, \xi_1^\ell, \ldots, \xi_i^\ell)}. \tag{16}
\]

Each sample path in \(Q_i^M\) begins with \(\xi_0^\ell = (0, x_0)\) and ends with one of the particles constituting \(q_i^M\), that is, \(\xi_i^\ell \in (\xi_i)^M\). For each particle \(\xi_i^\ell \in \{\xi_i^\ell\}_{\ell=1}^{M}\), the particles \(\xi_i^\ell \in \{\xi_i^\ell\}_{\ell=1}^{M}\) refer to the ancestral history of the particle \(\xi_i^\ell\).

In previous FBSDE methods each path was sampled independently such that the particles never affect each others’ movement, a design choice we call parallel drift. Although this is known to produce a high-fidelity approximation of an SDE whose dynamics are well known (see, for example, the discussion in Del Moral [4, p. 4]), for solving optimal control problems FBSDE methods may benefit greatly by relaxing this restriction. More specifically, having a good approximation of the forward SDE is not useful if the drift does not resemble the optimal control trajectory distribution. We generalize this choice by allowing paths to branch and represent the probability measures in a McKean-Markov chain model.

McKean-Markov chains are a generalization of the construction used in particle filters, genetic algorithms, etc. Over each time interval, a measure \(q_i^{M,i+1}\) is constructed by first starting with a measure \(q_i^M\) and applying a selection step followed by a mutation step as follows

\[
q_i^{M,i+1} \xrightarrow{\text{Selection}} Q_i^M \xrightarrow{\text{Mutation}} q_i^{M,i+1}.
\]

In the selection step \(M\) particles \(\{\xi_i^\ell\}_{\ell=1}^{M}\) are chosen from \(\{\xi_i^\ell\}_{\ell=1}^{M}\) via an arbitrarily-designed heuristic method for expansion, some multiple times and others not at all. For each chosen particle \(\xi_i^\ell\) a “mutation” is applied, that is, a drift value \(k_i^\ell\) is produced arbitrarily which can incorporate both state information from the originating particle and randomly sampled auxiliary values. Using the selected particle and corresponding drift, the forward SDE (10) is approximated with an Euler-Maruyama step to produce \(x_i^\ell\). Combining these two values into \(\xi_i^\ell = (k_i^\ell, x_i^\ell)\), this represents one of the particles in the distribution \(q_i^{M,i+1}\). Tracing back the selection history of each particle \(\xi_i^\ell\) represents the set of sample paths used in approximate path measure \(Q_i^{M,i+1}\).

Figure \[\text{Figure 2}\] illustrates how the representations of the forward measures in parallel-sampled FBSDE compare to the proposed branched-sampled method. Some particles may share parents (e.g., in Figure \[\text{Figure 2}\] the branch-sampled path measure \(Q_i^{M,i+1}\) has
uses one of them at a time. The path measure $\overrightarrow{Q}_{i+1}^M$, converges to an ideal Feynman-Kac path integral measure $\overrightarrow{Q}_{i+1}$ with large number of sample paths [4 Sec. 4.1.2], where this measure is defined as

$$d\overrightarrow{Q}_{i+1} := \frac{1}{Z_{i+1}} \prod_{j=0}^{k-1} \psi_j(x_j) d\overrightarrow{P}_{i+1},$$

where $Z_{i+1}$ is a normalization constant, $\psi_j$ is a potential weighing function, and $\overrightarrow{P}_{i+1}$ is a measure where no selection is performed. As noted in Section II-B choosing the density weighing function directly like this produces an equivalent measure, so the selection procedure produces a measure equivalent to a measure with no selection. This confirms the validity of this procedure for FBSDE problems. A direct consequence of the convergence of the path measure is that, for any function $F_i$ on paths,

$$E_{\overrightarrow{Q}_i^M}[F_i(\xi_0, \xi_1, \ldots, \xi_i)] \approx E_{\overrightarrow{Q}_i[M]}[F_i(\xi_0, \xi_1, \ldots, \xi_i)]$$

$$= \sum_{t=1}^{M} \frac{1}{M} F_i(\hat{c}_i^t, \xi_i^t, \ldots, \xi_i^t) \quad (17)$$

is a good numerical approximation for expectations.

C. Weighted Path Integral LSMC

In the backward pass, we use the sequence of path integral measures $\{\overrightarrow{Q}_i\}_{i=1}^N$ to fit a value function. Each backward step assumes an approximation of the value function known at $t_{i+1}$, $v(t_{i+1}, \cdot; \alpha_{i+1}) \approx v^*(t_{i+1}, \cdot)$, where $\alpha_{i+1}$ is a vector of parameters and the function $v : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable. The goal of the backward step is to find the parameters $\alpha_i$ which approximate $v(t_i, \cdot; \alpha_i) \approx v^*(t_i, \cdot)$.

Suppose we are provided a potential weighting function, $\rho_{i+1} : \mathbb{R}^n \times \mathbb{R}^n \to (0, 1]$. We can again use the analysis of Section II-B to show that the random variables $\rho_{i+1}(\xi_{i+1})$, when normalized, produce an equivalent measure

$$d\overrightarrow{R}_{i+1} := \frac{\rho_{i+1}(\xi_{i+1})}{E_{\overrightarrow{Q}_{i+1}^M}[\rho_{i+1}(\xi_{i+1})]} d\overrightarrow{Q}_{i+1} = \theta_{i+1} \overrightarrow{Q}_{i+1} d\overrightarrow{Q}_{i+1}.$$ 

The weighted LSMC method relates to the relationship (13), solving for an approximation $v(t_i, X_i; \alpha_i^*)$ to the value function by solving the least squares problem

$$\alpha_i^* = \arg\min_{\alpha_i} E_{\overrightarrow{Q}_{i+1}^M}[(\hat{Y}_{t_i} - v(t_i, X_i; \alpha_i))^2]. \quad (18)$$

by applying a well known projection property of conditional expectations [21 Chapter 10.3, Property 11]. Here, we use an approach similar to importance sampling. That is, although we take the expectation of (18) in the measure $\overrightarrow{R}_{i+1}$ (i.e. the desired distribution), we evaluate it in $\overrightarrow{Q}_{i+1}$ (i.e. the proposal distribution), arriving at the weighted least squares problem

$$\arg\min_{\alpha_i} E_{\overrightarrow{Q}_{i+1}^M}[\rho_{i+1}(\xi_{i+1})(\hat{Y}_{t_i} - v(t_i, X_i; \alpha_i))^2]. \quad (19)$$

The approximation in (17) demonstrates how the expectation can be evaluated by averaging over sample paths, though we note that only one step backward in the tree is needed for each path $(F_{i+1}(\xi_{i+1}, \hat{c}_{i+1,i+1}^t))$. 

---

**Fig. 2:** Comparing parallel sampled FBSDE to branched sampled FBRRT. Each colored point represents the sample particles for $q_{i+1}^M$ and $q_i^M$, while the colored edges represent path samples for $\overrightarrow{Q}_{i+1}^M$ and $\overrightarrow{Q}_i^M$. Dashed black edges are ignored for that particular path measure.
IV. FORWARD-BACKWARD RRT

In the previous section, we proposed a generalization of FBSDE methods which greatly expands their flexibility in searching for an optimal control policy. In doing so, we have exposed a few design choices, namely, the particle selection procedure, the drift values $K^Q_i$, and the weighted LSMC potential function $\rho_i$. In this section, we propose forward-backward RRT (FBRRRT) as a particular implementation of this generalization, although other methods could be developed within the same framework discussed in the previous section.

The FBRRRT algorithm takes as input an initial state $x_0$ and produces an array of parameter vectors $(\alpha_1, \ldots, \alpha_N)$, each of which parameterizes an approximation of the value function at discrete time instances, $v^*(t_i, \cdot) \approx v(t_i, \cdot; \alpha_i)$. The algorithm begins with a forward pass, producing a branching tree in a manner similar to kinodynamic RRT. This is followed by a backward pass, approximating the value function at each time step with weighted least squares polynomial regression. Following the backward pass, particles with low weight as determined by $\{x_i^\ell\}$ are pruned from the tree, removing a significant fraction of them. The pruned tree is again fed to the forward pass procedure to grow the tree to its original width, alternating forward and backward passes until convergence.

A. Kinodynamic RRT Forward Sampling

In general, we desire resampling methods which seek to explore the whole state space, increasing the likelihood of sampling in the proximity of optimal trajectories. For this reason, we chose kinodynamic RRT methods, proposed in LaValle and Kuffner [14], as a way to achieve this goal. The selection procedure for this method ensures that the distribution of chosen particles is more uniformly distributed in a user-supplied region of interest $\mathcal{X}_{roi} \subseteq \mathbb{R}^n$, more likely to select particles which explore empty space and less likely to oversample dense clusters of particles.

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

For drift generation we again choose a random combination of exploration and exploitation. The functions $v(t_i, \cdot; \alpha_i)$ define a closed-loop policy

$$
\pi_i(x_i; \alpha_i) := \arg\min_{u \in U} \{L_i(u) + v_\ell(t_i, x_i; \alpha_i)^\top G_i u\},
$$

over the full time horizon, coinciding with the optimal control policy when the value function approximation is exact [11 Chapter 4, Corollary 3.2]. Thus, the optimal drift approximation proposed in Exarchos and Theodorou [6],

$$
K^Q_i = \Sigma_i^{-1} G_i \pi_i(x_i; \alpha_i),
$$

is good for exploitation. However, for exploration, we choose random controls, $u^\text{rand} \sim \mathcal{U}^\text{rand}$, where $\mathcal{U}^\text{rand}$ is a distribution over $U$ provided as a design parameter. For example, for minimum fuel ($L_1$) problems where control is bounded $u \in [-1, 1]$ and the running cost is $L = |u|$, we select $\mathcal{U}^\text{rand} = \{-1, 0, 1\}$ because the optimal policy [20] is guaranteed to only return values in this discrete set.

Algorithm [1] sketches out the implementation of particle selection and drift generation, producing the forward sampling tree $G$. The algorithm takes as input any tree with width $M$ and adds nodes at each depth until the width is $M$, the parameter indicating the desired width. On the first iteration there are no value function estimate parameters available to exploit, so we set $\varepsilon^mt = 1$ to maximize exploration using the RRT sampling.

Algorithm 1 RRT Branched Particle Forward Sampling

1. procedure FORWARDEXPAND($G, (\alpha_1, \ldots, \alpha_N)$)
2. for $j = M + 1, \ldots, M$ do $\triangleright$ Add node each loop
3. for $i = 0, \ldots, N - 1$ do $\triangleright$ For each step time
4. $\{x_i^t\} \leftarrow G$.nodesAtTime$_{x_i^t}$(t$_i$)
5. if $\varepsilon^mt > k^mt \sim \text{Uniform}(0, 1)$ then
6. $x^\text{rand}_i \sim \text{Uniform}(\chi_{\pi^*})$
7. $(x^\text{nearly}_i, \text{nearly}) \leftarrow \text{Nearest}(\{x_i^t\}, x^\text{rand}_i)$
8. else
9. $(x^\text{nearly}_i, \text{nearly}) \sim \text{Uniform}(\{x_i^t\})$
10. end if $\triangleright$ $\text{nearly}$ is index of selected node $x^\text{nearly}_i$
11. if $\varepsilon^mt > k^mt \sim \text{Uniform}(0, 1)$ then
12. $u_i \leftarrow \pi_i(x^\text{nearly}_i, \Phi_x(x^\text{nearly}_i)\alpha_i)$
13. else
14. $u_i \sim \text{Uniform}(U)^\pi$
15. end if
16. $k_i \leftarrow \Sigma_i^{-1} G_i(x^\text{nearly}_i) u_i$\n17. $w_i \sim N(0, \Delta t_i I_{n \times n})$
18. $x^\text{next}_i \leftarrow x^\text{nearly}_i + (f_i(x^\text{nearly}_i) + \Sigma_i k_i) \Delta t_i + \Sigma_i w_i$
19. $G$.addEdge($x^\text{next}_i, x^\text{nearly}_i$) $\triangleright$ $G$.setInt($i+1, x^\text{nearly}_i, \Delta t_i$
20. $\mathcal{L}_{i+1} \leftarrow G$.getInt($i+1, x^\text{nearly}_i, \Delta t_i$
21. end for
22. end for
23. return $G$

B. Local Entropy Backwards Weighing

We now propose a heuristic design choice for the backward pass potential weighing functions $\rho_{i+1}$, and justify its choice with some theoretical results. We pose a measure-theoretic optimization problem which selects a measure, balancing between selecting a target measure and keeping the selected measure close to the forward sampling measure for regularization. We follow an approach which draws from the theory presented in Theodorou and Todorov [26] and apply it to the problem of weighted LSMC.

A dynamic programming principle result following directly from Fleming and Soner [11 Chapter 4, Corollary 7.2] indi-
along the path
expectation further yields that
\[ \hat{\Sigma}_\lambda > \tilde{\lambda} \]
The random variable \( x \) cost for the first term and using the current approximation of \( u \) as
is the measure produced by the drift \( K_s \):= \( \Sigma_s^{-1}G_su_s \), and \( P_{\ell+1} := P_{\ell+1}(u(\cdot)) \), the distribution of trajectories controlled by the optimal policy. The law of total expectation further yields that
\[ v(0, x_0) = \min_{u(\cdot)} E_{P_{\ell+1}(u(\cdot))} [\tilde{Y}_{\ell+1}^*] \] (22)
where
\[ \tilde{Y}_{\ell+1}^* := E_{P_{\ell+1}(u(\cdot))} [\int_0^{t_{\ell+1}} L_s(u_s) ds + v^* (t_{\ell+1}, X_{\ell+1})] \]
The random variable \( \tilde{Y}_{\ell+1}^* \) can be numerically estimated at any particle \( x_{\ell+1} \) by computing the path integral over the running cost for the first term and using the current approximation of the value function \( v(t_{\ell+1}, x_{\ell+1}; \alpha_{\ell+1}) \) for the second term. By “path integral”, we indicate that the integral is approximated along the path \( (\xi_{\ell+1}^i, \xi_{\ell+1}^i, \ldots, \xi_{\ell+1}^i) \), to produce the value \( \tilde{Y}_{\ell+1}^* \) associated with the particle \( \xi_{\ell+1}^i \), as is shown in line 20 of Algorithm 1. The above analysis suggests that minimizing (22) over the possible measures \( P(u(\cdot)) \) produces \( P^* \). Thus, we use the random variable \( \tilde{Y}_{\ell+1}^* \) as a heuristic value to be minimized.
Consider the local entropy minimization problem
\[ \min_{P_{\ell+1}} \{ E_{P_{\ell+1}} [\tilde{Y}_{\ell+1}^*] + \lambda H(P_{\ell+1} \| Q_{\ell+1}) \} \] (23)
where \( \lambda > 0 \) is a design parameter and
\[ H(P_{\ell+1} \| Q_{\ell+1}) := E_{P_{\ell+1}} [\log \frac{dP_{\ell+1}}{dQ_{\ell+1}}] \]
is relative entropy, also called Kullback-Leibler divergence, a semi-metric which measures the closeness of measures (the equality is proved in Proposition I.3 in the appendix). The measure \( Q_{\ell+1} \) is approximated by \( Q_{\ell+1}^M \), representing the forward path measure produced by the previous section, while a minimizer of the problem, \( R_{\ell+1} \), approximated by \( R_{\ell+1}^M \), is the measure considered as the desired distribution used for weighing in Section 112.1. Keeping the weighted measure \( \tilde{R}_{\ell+1}^M \) relatively close to \( \tilde{Q}_{\ell+1} \) acts as a regularization technique, ensuring the least squares regression does not become ill-conditioned by introducing too many small weights. Indeed, as \( \lambda \to \infty \) we can see that \( R_{\ell+1} \to Q_{\ell+1} \), so the weighted LSMC becomes equally weighted for every particle.
The problem (23) can be shown (Proposition I.5 in the appendix) to have a minimizing measure when the equality
\[ \Theta_{\ell+1}^{Q} = \exp(\frac{-1}{\lambda \tilde{Y}_{\ell+1}^*}) \] (24)
is satisfied. Thus, we choose the weighing potential functions to be
\[ \rho_{\ell+1}(\xi_{\ell+1}) = \exp(-1/\lambda \tilde{Y}_{\ell+1}^*). \] (25)

Algorithm 2 Local Entropy Weighted LSMC Backward Pass

1: procedure BACKWARDWLSMC(\( G \))
2: \{ \( x_N \) \} \( \ell \) := \( G \).nodesAtTime(\( X \), \( t_N \))
3: \( y_N \) := \[ g(\mathcal{x}_N^{r}) \cdots g(\mathcal{x}_N^{r}) \]
4: \( \alpha_N \) := \arg \min_{\alpha} \{ \| \Phi(\mathcal{x}_N^{r}) \alpha - y_N \| ^2 \}
5: for \( i = N - 1, \ldots, 1 \) do \( \triangleright \) For each time step
6: \{ \( \xi_{\ell+1}^i \) := \( k_i \cdot \mathcal{x}_N^{r} \) \} \( \ell \) := \( G \).nodesAtTime(\( t_{\ell+1} \))
7: for \( \ell = 1, \ldots, M \) do
8: \( \mathcal{x}_{\ell+1}^{r} := \sum_{i}^{T} \Phi(\mathcal{x}_N^{r}) \alpha_{\ell+1} \)
9: \( \tilde{y}_{\ell+1}^{f} := y_{\ell+1}^{f} + (h_{\ell+1}(\mathcal{x}_{\ell+1}^{r}, \mathcal{x}_{\ell+1}^{r}) - \mathcal{x}_{\ell+1}^{r} k_{\ell}^{T} \Delta t_{\ell}) \)
10: \( \mathcal{y}_{\ell+1}^{f} := \mathcal{y}_{\ell+1}^{f} + \mathcal{G} \).getLIn(i + 1, \( \ell \))
11: \( \mathcal{x}_{\ell+1}^{r} := \mathcal{G} \).parentNode(\( i + 1, \ell \)) \( \triangleright \) for good exp conditioning
12: \( \lambda_{\ell} \) := \( \mathcal{G} \).getLIn(i + 1, \( \ell \)) \( \triangleright \) for good exp conditioning
13: \( \alpha \) := \arg \min_{\alpha} \{ \| \Phi(\mathcal{x}_N^{r}) \alpha - \tilde{y}_{\ell+1}^{f} \| ^2 \}
14: end for
15: return \( (\alpha_1, \ldots, \alpha_N) \)
16: end procedure

Algorithm 2 details the implementation of the backward pass with local entropy weighting. The value function is represented by a linear combination of multivariate Chebyshev polynomials up to the 2nd order, \( v(t_i, X_i; \alpha_i) = \Phi(X_i) \alpha_i \). The proposed local weighing method is structurally similar to a softmax operation, often used in deep learning literature [12]. Line 13 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 Goodfellow et al. [12] Chapter 5, equation (6.33)). The \( \lambda \) 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 \( \lambda \) parameters, evaluating each with a backward pass and using the one which produces the smallest expected cost over a batch of trajectory rollouts executing the computed policy.

V. NUMERICAL RESULTS

We evaluated the FBVRT algorithm by applying it to a pair of nonlinear stochastic optimal control problems. For both problems, we used a minimum fuel (\( L_1 \)) running cost of \( L(u) = a||u||, a > 0 \), where the terminal cost is a quadratic function centered at the origin. We implemented the FBVRT algorithm and all examples in Matlab 2019b and ran them on an Intel G4560 3.50GHz processor with 8GB RAM.

Figure 1 illustrates our method applied to the \( L_1 \) inverted pendulum problem. Note that even though there were no paths in the tree that continued along the 1st iteration’s mean trajectory (blue line) from beginning to end, the algorithm was still able to produce a policy in regions where no particles
were produced. The green particles along the backward swing inform the policy in the beginning of the trajectory while the green particles near the origin inform it near the end, despite taking different paths in the tree.

The policies computed after the first few iterations are visualized in Figure 3. Of significant note is that the policy obtained after only one iteration (red hue) does significantly well in general. For the $L_1$ inverted pendulum problem evaluated in [8], convergence required 55 iterations, but for our method only a handful of iterations were needed to get comparable performance.

We also compared the convergence speed and robustness of FBRRT to parallel-sampled FBSDE [8] by randomly sampling different starting states and evaluating their relative performance over a number of trials. 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 4. Since the FBRRT is significantly slower than the FBSDE per iteration due to the RRT nearest neighbors calculation, we scale each iteration by the runtime. By nearly every comparison, FBRRT converges faster and in fewer iterations than FBSDE, and does so with half as many particle samples.

VI. Conclusions and Future Work

In this work, we have proposed a novel generalization of the FBSDE approach to solve stochastic optimal control problems, combining both 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 methods. We have shown how the proposed method works hand in hand with a proposed local entropy-weighted LSMC method, concentrating function approximation in the regions where optimal trajectories are most likely to be dense. We have demonstrated that FBRRT can generate feedback control policies for nonlinear stochastic optimal control problems with non-quadratic costs.

In future work, we plan to incorporate more current RRT algorithms, since nearly any method could be adapted to this approach with the proper book keeping. Further, though we did not explicitly address obstacles, handled naturally by RRT algorithms, we plan to investigate how the FBSDE theory can represent obstacle avoidance. Another significant area of research worth investigating is to find better methods of value

![Policy Trajectory Samples](a) $L_1$ Double Integrator ($n = 2$)

![Policy Trajectory Samples](b) $L_1$ Inverted Pendulum ($n = 2$)

Fig. 3: Trajectory samples from policies generated after the first 6 iterations, the first iteration colored red, followed by yellow, green, cyan, dark blue, and magenta. All terminal costs are centered at $(0, 0)$. Dark thick lines are the mean trajectories.

Fig. 4: Comparison of FBRRT and FBSDE for the $L_1$ double integrator problem for random initial states. Expected trajectory costs for the computed policies are normalized across different initial conditions.
function representation. Although 2nd-order polynomials generally produce nice policy functions, they are unlikely to produce a good approximation of the value function outside of a local region. In addition, we plan to evaluate our method on higher dimensional problems to demonstrate how our method can solve problems out of reach for most methods.

REFERENCES

[1] Oktay Arslan, Evangelos A. Theodorou, and Panagiotis Tsiotras. Information-theoretic stochastic optimal control via incremental sampling-based algorithms. In IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning, Orlando, FL, 2014.

[2] Christian Bender and Robert Denk. A forward scheme for backward SDEs. Stochastic Processes and their Applications, 2007. ISSN 03044149. doi: 10.1016/j.spa.2007.03.005.

[3] Christian Bender and Thilo Moseler. Importance sampling for backward SDEs. Stochastic Analysis and Applications, 28(2):226–253, 2010. ISSN 07362294. doi: 10.1080/07362290903546405.

[4] Pierre Del Moral. Mean Field Simulation for Monte Carlo Integration. Chapman and Hall/CRC, 2013.

[5] Nicole El Karoui, Shige Peng, and Marie Claire Quenez. Backward stochastic differential equations in finance. Mathematical Finance, 7(1):1–71, 1997.

[6] Ioannis Exarchos and Evangelos A. Theodorou. Stochastic optimal control via forward and backward stochastic differential equations and importance sampling. Automatica, 87:159–165, 2018.

[7] Ioannis Exarchos, Evangelos A. Theodorou, and Panagiotis Tsiotras. Game-theoretic and risk-sensitive stochastic optimal control via forward and backward stochastic differential equations. In Conference on Decision and Control, Las Vegas, Nevada, pages 6154–6160, 2016. ISBN 9781509018376. doi: 10.1109/CDC.2016.7799215.

[8] Ioannis Exarchos, Evangelos A. Theodorou, and Panagiotis Tsiotras. Stochastic L1-optimal control via forward and backward sampling. Systems and Control Letters, 118:101–108, 2018. ISSN 01676911. doi: 10.1016/j.sysconle.2018.06.005.

[9] Ioannis Exarchos, Evangelos A. Theodorou, and Panagiotis Tsiotras. Stochastic Differential Games: A Sampling Approach via FBSDEs. Dynamic Games and Applications, 2018. ISSN 2153-0785. doi: 10.1007/s13235-018-0268-4.

[10] Wendell H. Fleming and Raymond W. Rishel. Deterministic and stochastic optimal control. Bulletin of the American Mathematical Society, 82:869–870, 1976.

[11] Wendell H. Fleming and Halil M. Soner. Controlled Markov Processes and Viscosity Solutions, volume 25. Springer Science and Business Media, 2006.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2016.

[13] David H. Jacobson and David Q. Mayne. Differential dynamic programming. North-Holland, New York, NY, 1970.

[14] Steven M. LaValle and James J. Kuffner. Randomized kinodynamic planning. The International Journal of Robotics Research, 20(5):378–400, 2001.

[15] Francis A. Longstaff and Eduardo S. Schwartz. Valuing American options by simulation: A simple least-squares approach. Review of Financial Studies, 2001. ISSN 08939454. doi: 10.1093/rfs/14.1.113.

[16] Jin Ma and Jiongmin Yong. Forward-Backward Stochastic Differential Equations and their Applications. Springer, 2007.

[17] Ian 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, pages 177–247, 2004. ISSN 0962-9343.

[18] Iram Noreen, Amna Khan, and Zulfiquar Habib. Optimal path planning using RRT* based approaches: a survey and future directions. Int. J. Adv. Comput. Sci. Appl, 7(11):97–107, 2016.

[19] E. Pardoux and S. G. Peng. Adapted solution of a backward stochastic differential equation. Systems and Control Letters, 14(1):55–61, 1990. ISSN 01676911. doi: 10.1016/0167-6911(90)90082-6.

[20] Shige Peng. Backward stochastic differential equations and applications to optimal control. Applied Mathematics and Optimization, 27(2):125–144, 1993.

[21] Sidney Resnick. A Probability Path. Birkhäuser Verlag AG, 2003.

[22] A. N. Shiryaev. Probability. Springer, 2nd edition, 1996.

[23] Halil M. Soner and Nizar Touzi. A stochastic representation for the level set equations. Communications in Partial Differential Equations, 27(9-10):2031–2053, 2002.

[24] Yuval Tassa. Theory and Implementation of Biomimetic Motor Controllers (Ph.D. Thesis). Hebrew University of Jerusalem, 2011.

[25] Yuval Tassa, Tom Erez, and William D. Smart. Receding Horizon Differential Dynamic Programming. Advances in Neural Information Processing Systems 20, pages 1465–1472, 2008.

[26] Evangelos A. Theodorou and Emanuel Todorov. Relative entropy and free energy dualities: Connections to path integral and KL control. In IEEE Conference on Decision and Control, Maui, Hawaii, pages 1466–1473. IEEE, 2012.

[27] Evangelos A. Theodorou, Yuval Tassa, and Emo Todorov. Stochastic differential dynamic programming. In American Control Conference, Baltimore, Maryland, pages 1125–1132. IEEE, 2010.

[28] Jiongmin Yong and Xun Yu Zhou. Stochastic Controls: Hamiltonian Systems and HJB Equations, volume 43. Springer Science and Business Media, 1999.