Feynman-Kac Neural Network Architectures for Stochastic Control Using Second-Order FBSDE Theory

Marcus A. Pereira [*] [†] MPEREIRA30@GATECH.EDU
Ziyi Wang[*†] ZIYIWANG@GATECH.EDU
Tianrong Chen[*†] TIANRONG.CHEN@GATECH.EDU
Emily A. Reed[*] [‡] EMILYREE@USC.EDU
Evangelos A. Theodorou[*] EVANGELOS.THEODOROU@GATECH.EDU

Editors: A. Bayen, A. Jadbabaie, G. J. Pappas, P. Parrilo, B. Recht, C. Tomlin, M. Zeilinger

Abstract

We present a deep recurrent neural network architecture to solve a class of stochastic optimal control problems described by fully nonlinear Hamilton Jacobi Bellman partial differential equations. Such PDEs arise when considering stochastic dynamics characterized by uncertainties that are additive, state dependent, and control multiplicative. Stochastic models with these characteristics are important in computational neuroscience, biology, finance, and aerospace systems and provide a more accurate representation of actuation than models with only additive uncertainty. Previous literature has established the inadequacy of the linear HJB theory for such problems, so instead, methods relying on the generalized version of the Feynman-Kac lemma have been proposed resulting in a system of second-order Forward-Backward SDEs. However, so far, these methods suffer from compounding errors resulting in lack of scalability. In this paper, we propose a deep learning based algorithm that leverages the second-order FBSDE representation and LSTM-based recurrent neural networks to not only solve such stochastic optimal control problems but also overcome the problems faced by traditional approaches, including scalability. The resulting control algorithm is tested on a high-dimensional linear system and three nonlinear systems from robotics and biomechanics in simulation to demonstrate feasibility and out-performance against previous methods.

1. Introduction

Stochastic Optimal Control (SOC) is at the center of decision making under uncertainty with extensive prior work in both theory and algorithms (Stengel, 1994; Fleming and Soner, 2006). One of the most celebrated formulations is for linear dynamics and additive noise, known as the Linear Quadratic Gaussian (LQG) case (Stengel, 1994). For control affine nonlinear stochastic systems, the SOC formulation results in the Hamilton-Jacobi-Bellman (HJB) equation, which is a nonlinear Partial Differential Equation (PDE) that evolves backwards in time. Solving the HJB PDE for high dimensional systems is challenging and suffers from the curse of dimensionality (Bellman, 2003).

Algorithms so far that solve the HJB equation can be mostly classified into two groups: (i) linearization-based and (ii) sampling-based. Linearization-based algorithms rely on either first-order (iLQG) or second-order (Stochastic Differential Dynamic Programming) Taylor's approximation of dynamics and quadratic approximation of the cost function (Todorov and Li, 2005b; Theodorou et al., 2010b). These algorithms exhibit high sensitivity to time discretization and require application of

[*] Autonomous Control and Decision Systems Laboratory (ACDS Lab), School of Aerospace Engineering, Georgia Institute of Technology, Atlanta GA 30313, USA
[†] Equal Contribution
[‡] Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089-2905, USA

special linearization schemes. Sampling-based algorithms, such as the Markov-Chain Monte Carlo (MCMC) approximation of the HJB equation (Kushner and Dupuis, 1992; Huynh et al., 2016), rely on backward propagating the value function on a pre-specified grid. Recently, tensor-train decomposition techniques (Gorodetsky et al., 2015) were proposed to improve scalability. However, these have demonstrated limited applicability so far. Other methods include transforming the problem into a deterministic one by employing the Fokker-Planck PDE (Annunziato and Borzi, 2010; Annunziato and Borzì, 2013). This approach is also grid-based and therefore unscalable to high dimensions.

Previous sampling-based methods rely on the linear Feynman-Kac lemma (Karatzas and Shreve, 1991) and its generalized nonlinear version (Pardoux and Rascanu, 2014). The former relies on exponentiating the value function and an assumption coupling control authority to the noise variance, which prohibits tuning of the control cost independent of the noise. Controls are computed using forward sampling of Forward Stochastic Differential Equations (FSDEs) (Kappen, 2005; Todorov, 2007, 2009; Theodorou et al., 2010a). The nonlinear Feynman-Kac lemma originates from the theory of backward Stochastic Differential Equations (SDEs) that connects the solution of a parabolic PDE to a set of Forward-Backward Stochastic Differential Equations (FBSDEs) (Pardoux and Peng, 1990; El Karoui et al., 1997) and does not require any of the above transformations or assumptions. However, solving FBSDEs requires backpropagating SDEs to satisfy a terminal condition. Since the uncertainty that enters the dynamics evolves forward in time, only conditional expectations can be back-propagated (Shreve, 2004, Chapter 2) thereby requiring complicated numerical schemes (Exarchos and Theodorou, 2018, 2016; Exarchos et al., 2018, 2019). The major limitation of these methods is the compounding regression errors at every time step rendering them unscalable.

Recent efforts in Deep Learning for solving nonlinear PDEs demonstrate potential solutions to overcome the problem of compounding errors. An algorithm introduced by Han et al. (2018) mitigates the problem by using the terminal condition as the prediction target for a forward propagated backward SDE. This is enabled by randomly initializing the value function and its gradient at the start time and treating them as trainable parameters of a self-supervised deep learning problem. As a result, the approximation errors at each time step are compensated for by backpropagation during training. However, their algorithm relies on forward processes driven purely by noise, which is not only inefficient for exploration but cannot yield any solution for controlling highly nonlinear systems. In Pereira et al. (2019), we extended this work by firstly incorporating importance sampling using Girsanov's theorem (Shreve, 2004, Chapter 5) allowing efficient sampling from controlled dynamics. Secondly, we introduced a more efficient Long-Short Term Memory (LSTM)-based architecture and a framework that can optimally handle control constraints. However, the approach is only applicable to stochastic systems wherein noise is either additive or state dependent.

Hence, in this paper, we address SOC problems wherein controls have a multiplicative effect on the noise entering the system resulting in fully nonlinear HJB PDEs. Literature for SOC of such systems is sparse and existing algorithms (iLQG/SDDP) require special linearization schemes for stochastic dynamics (Torre and Theodorou., 2015). The aforementioned methods relying on the linear Feynman-Kac lemma cannot be derived in the case of control multiplicative noise and those relying on the nonlinear Feynman-Kac lemma are limited to first-order FBSDEs. Bakshi et al. (2017) used the second-order FBSDEs (2FBSDEs) to solve the fully nonlinear HJB PDE, but their approach suffers from the same aforementioned approximations errors due to regression. To overcome these drawbacks, we propose a novel LSTM-based architecture tailored to solve such SOC problems, which are critical in biomechanics, computational neuroscience, autonomous systems, and finance (Todorov and Li, 2005a; Mitrovic et al., 2010; Primbs, 2007; McLane, 1971; Phillis, 1985).

The paper is organized as follows: in Section 2 we introduce notation, mathematical preliminaries, and problem formulation. Section 3 provides the 2FBSDE formulation followed by the Deep 2FBSDE algorithm in Section 4. Results from our simulation experiments are provided in Section 5 followed by a discussion, contributions, and future directions in Section 6.

2. Notation, mathematical preliminaries, and problem statement

Note that hereon bold faced notation will be abused for both vectors and matrices, while non-bold faced will be used for scalars. We consider SDEs that are nonlinear functions of the state but are affine in the controls. Let $\mathbf{x} \in \mathbb{R}^{n_x}$ be the state, and $\mathbf{u} \in \mathbb{R}^{n_u}$ be the controls taking values in the set of all admissible controls $\mathcal{U}([0,T])$, for a fixed finite time horizon $T \in [0,\infty)$. Let $\big([v(t)^{\mathrm{T}}\ \mathbf{w}(t)^{\mathrm{T}}]^{\mathrm{T}}\big)_{t\in[0,T]}$ be a vector of mutually independent Brownian motions in $\mathbb{R}^{n_w+1}$. We assume that the functions $\mathbf{f} : [0,T] \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$, $\mathbf{G} : [0,T] \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_x \times n_u}$, $\mathbf{\Sigma} : [0,T] \times \mathbb{R}^{n_x} \to \mathbb{R}^{n_x \times n_w}$ and $\sigma \in \mathbb{R}^+$ satisfy certain Lipschitz and growth conditions (Wang et al., 2019, SM A). Given this assumption, for every initial condition $\xi \in \mathbb{R}^{n_x}$, there exists a unique solution $\big(\mathbf{x}(t)\big)_{t\in[0,T]}$ to the FSDE,

$$\begin{aligned} \mathrm{d}\mathbf{x}(t) &= \big(\mathbf{f}(t,\mathbf{x}(t)) + \mathbf{G}(t,\mathbf{x}(t))\mathbf{u}(t)\big)\mathrm{d}t + \sigma\mathbf{G}(t,\mathbf{x}(t))\mathbf{u}(t)\mathrm{d}v(t) + \mathbf{\Sigma}(t,\mathbf{x}(t))\mathrm{d}\mathbf{w}(t) \\ &= \big(\mathbf{f}(t,\mathbf{x}(t)) + \mathbf{G}(t,\mathbf{x}(t))\mathbf{u}(t)\big)\mathrm{d}t + \mathbf{C}\big(t,\mathbf{x}(t),\mathbf{u}(t)\big)\mathrm{d}\boldsymbol{\epsilon}(t), \end{aligned} \tag{1}$$

with $\mathbf{x}(0) = \xi$, $\mathbf{C}(t,\mathbf{x}(t),\mathbf{u}(t)) = \big[\sigma\mathbf{G}(t,\mathbf{x}(t))\mathbf{u}(t), \mathbf{\Sigma}(t,\mathbf{x}(t))\big]$ and $\boldsymbol{\epsilon}(t) = \begin{bmatrix} v(t) & \mathbf{w}(t) \end{bmatrix}^{\mathrm{T}}$.

Problem Statement and HJB PDE: We formulate the SOC problem as finding the optimal control ($\mathbf{u}^*$) that minimizes the following expected cost, which is quadratic in control

$$J\big(t,\mathbf{x}(t);\mathbf{u}(t)\big) = \mathbb{E}\left[\int_t^T \big(q(s,\mathbf{x}(s)) + \frac{1}{2}\mathbf{u}(s)^{\mathrm{T}}\mathbf{R}\mathbf{u}(s)\big)\mathrm{d}s + \phi\big(\mathbf{x}(T)\big)\right], \tag{2}$$

where $q : \mathbb{R}^{n_x} \to \mathbb{R}^+$ is the running state cost, $\mathbf{R}$ is a symmetric positive definite matrix of size $n_u \times n_u$ and $\phi : \mathbb{R}^{n_x} \to \mathbb{R}^+$ is the terminal state cost. q and ϕ are differentiable with continuous derivatives up to the second order. We can define the value function as

$$V\big(t,\mathbf{x}(t)\big) = \inf_{\mathbf{u}(t)\in\mathcal{U}([t,T])} J\big(t,\mathbf{x}(t);\mathbf{u}(t)\big), \quad V\big(T,\mathbf{x}(T)\big) = \phi\big(\mathbf{x}(T)\big). \tag{3}$$

Assuming that $V(t,\mathbf{x})$ is once differentiable in t and twice differentiable in $\mathbf{x}$ with continuous derivatives, there exists a unique solution to the following fully nonlinear HJB PDE

$$V_t + q + V_{\mathbf{x}}^{\mathrm{T}}\mathbf{f} - \frac{1}{2}V_{\mathbf{x}}^{\mathrm{T}}\mathbf{G}\hat{\mathbf{R}}^{-1}\mathbf{G}^{\mathrm{T}}V_{\mathbf{x}} + \frac{1}{2}\operatorname{tr}(V_{\mathbf{xx}}\mathbf{\Sigma}\mathbf{\Sigma}^{\mathrm{T}}) = 0, \quad V(T,\mathbf{x}(T)) = \phi(\mathbf{x}(T)), \tag{4}$$

with the optimal control $\mathbf{u}^*(t,\mathbf{x}) = -\hat{\mathbf{R}}^{-1}\mathbf{G}(t,\mathbf{x})^{\mathrm{T}}V_{\mathbf{x}}(t,\mathbf{x})$, where, $\hat{\mathbf{R}} \triangleq (\mathbf{R} + \sigma^2\mathbf{G}^{\mathrm{T}}V_{\mathbf{xx}}\mathbf{G})$. See (Wang et al., 2019, SM B) for the detailed derivation.

3. A Controlled 2FBSDE Solution to the fully nonlinear HJB PDE

Bakshi et al. (2017) considered neither controls nor control multiplicative noise in their numerical algorithm. This lack of control leads to insufficient exploration of possible solutions and for highly

nonlinear systems renders it potentially impossible to find a solution to complete the task. In light of this, we introduce a new set of 2FBSDEs that includes control in the FSDE

$$\begin{cases} \mathrm{d}\mathbf{x} & = \mathbf{f}\mathrm{d}t + \mathbf{G}\mathbf{u}^*\mathrm{d}t + \mathbf{C}\mathrm{d}\boldsymbol{\epsilon} & \text{(FSDE)} \\ \mathrm{d}V & = \mathcal{H}(V)\mathrm{d}t + V_{\mathbf{x}}^{\mathrm{T}}\mathbf{G}\mathbf{u}^*\mathrm{d}t + V_{\mathbf{x}}^{\mathrm{T}}\mathbf{C}\mathrm{d}\boldsymbol{\epsilon} & \text{(BSDE 1)} \\ \mathrm{d}V_{\mathbf{x}} & = \mathcal{H}(V_{\mathbf{x}})\mathrm{d}t + V_{\mathbf{xx}}\mathbf{G}\mathbf{u}^*\mathrm{d}t + V_{\mathbf{xx}}\mathbf{C}\mathrm{d}\boldsymbol{\epsilon} & \text{(BSDE 2)} \\ \mathbf{x}(0) & = \xi,\ V(T) = \phi(\mathbf{x}(T)),\ V_{\mathbf{x}}(T) = \phi_{\mathbf{x}}(\mathbf{x}(T)), \end{cases} \tag{5}$$

where $\mathbf{u}^*$ is the optimal control and the operator $\mathcal{H}$ is defined as $\mathcal{H}(\cdot) \triangleq \partial_t(\cdot) + \partial_{\mathbf{x}}(\cdot)^{\mathrm{T}}\mathbf{f} + \frac{1}{2}\mathrm{tr}(\partial_{\mathbf{xx}}(\cdot)\mathbf{C}\mathbf{C}^{\mathrm{T}})$. We can obtain this particular set of 2FBSDEs by substituting for the optimal control in the forward process (1). The backward processes are obtained by applying the Itô-Doeblin formula (Shreve, 2004, Chapter 4), which is essentially the second order Taylor series expansion, to the value function V and its gradient $V_{\mathbf{x}}$. Additionally, we substitute the expression for V_t from (4) into BSDE 1 and obtain the final form of the 2FBSDEs as

$$\begin{cases} \mathrm{d}\mathbf{x} & = \mathbf{f}\mathrm{d}t + \mathbf{G}(\mathbf{u}^*\mathrm{d}t + \sigma\mathbf{u}^*\mathrm{d}v) + \boldsymbol{\Sigma}\mathrm{d}\mathbf{w} \\ \mathrm{d}V & = -\big(q - \frac{1}{2}V_{\mathbf{x}}^{\mathrm{T}}\mathbf{G}\hat{\mathbf{R}}^{-T}(\mathbf{R} + 2\sigma^2\mathbf{G}^{\mathrm{T}}V_{\mathbf{xx}}\mathbf{G})\hat{\mathbf{R}}^{-1}\mathbf{G}^{\mathrm{T}}V_{\mathbf{x}}\big)\mathrm{d}t \\ & \quad + V_{\mathbf{x}}^{\mathrm{T}}\mathbf{G}(\mathbf{u}^*\mathrm{d}t + \sigma\mathbf{u}^*\mathrm{d}v) + V_{\mathbf{x}}^{\mathrm{T}}\boldsymbol{\Sigma}\mathrm{d}\mathbf{w} \\ \mathrm{d}V_{\mathbf{x}} & = (\mathbf{A} + V_{\mathbf{xx}}\mathbf{f})\mathrm{d}t + V_{\mathbf{xx}}\mathbf{G}(\mathbf{u}^*\mathrm{d}t + \sigma\mathbf{u}^*\mathrm{d}v) + V_{\mathbf{xx}}\boldsymbol{\Sigma}\mathrm{d}\mathbf{w} \\ \mathbf{x}(0) & = \xi,\ V(T) = \phi(\mathbf{x}(T)),\ V_{\mathbf{x}}(T) = \phi_{\mathbf{x}}(\mathbf{x}(T)), \end{cases} \tag{6}$$

with $\mathbf{A} = \partial_t(V_{\mathbf{x}}) + \frac{1}{2}\mathrm{tr}(\partial_{\mathbf{xx}}(V_{\mathbf{x}})\mathbf{C}\mathbf{C}^{\mathrm{T}})$. The full derivation can be found in Wang et al. (2019, SM B,C). In contrast to our prior approaches (Exarchos and Theodorou, 2018; Pereira et al., 2019) that first introduce an uncontrolled system of FBSDEs and then use Girsanov's theorem for imporatance sampling (i.e. introducing controls in the forward and backward SDEs), our new derivation considers controlled dynamics from the very beginning, circumventing the need to use Girsanov's theorem. Additionally, it must be emphasized that Girsanov's theorem cannot be naively used in the case of control multiplicative noise to obtain controlled forward and backward processes.

4. Deep 2FBSDE Controller

In this section, we introduce a novel Deep Neural Network (DNN) architecture called the *Deep 2FBSDE Controller*, which works with the discretized version of the continuous time solution (6) (Wang et al. (2019, Sec. 4) contains details regarding numerical integration scheme used).

Network architecture: The network architecture proposed by Han et al. (2018); Beck et al. (2019) samples from uncontrolled dynamics, which in the current context eliminates the effect of control multiplicative noise resulting in sampling from dynamics with additive noise alone. As a result, the trained $V(t, \mathbf{x})$ and $V_{\mathbf{x}}(t, \mathbf{x})$ won't be effective in the presence of controls and hence control multiplicative noise. Therefore similar to our previous work in Pereira et al. (2019), one could consider the architecture consisting of Feed-forward Neural Networks (FNNs) at every time step as in Han et al. (2018); Beck et al. (2019), with additional connections (2FBSDE-FNNs) that use $V_{\mathbf{x}}$ and $V_{\mathbf{xx}}$ at every time step for optimal feedback control. This however, introduces additional gradient paths requiring backprop through time. Alternatively, we propose the LSTM-based architecture in Fig. 1 (2FBSDE-LSTM) inspired by the successful results in our work (Pereira et al., 2019). This architectural design is the most practical in terms of memory requirements while potentially

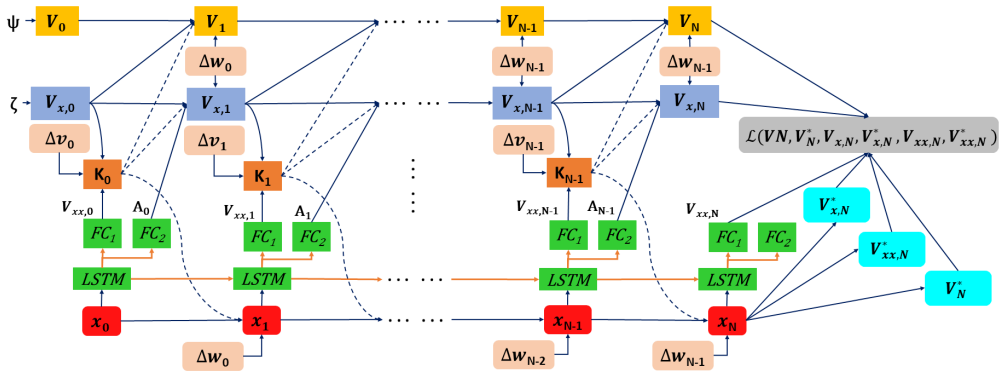

Figure 1: **Deep 2FBSDE neural network architecture.** The blocks $FC_{1,2}$ are fully connected layers with linear activations while the $LSTM$ block represents recurrent layers of stacked LSTM cells with standard nonlinear activations. These layers are parameterized by θ and shared temporally. Additionally, $V(\mathbf{x}_0, t_0)$ and $V_{\mathbf{x}}(\mathbf{x}, t_0)$ are parameterized by ψ and ζ respectively. The self-supervised targets $V_N^*, V_{\mathbf{x},N}^*$ are the terminal conditions from (6) and $V_{\mathbf{xx},N}^* = \phi_{\mathbf{xx}}(\mathbf{x}_N)$.

avoiding the vanishing gradient problem. Instead of predicting the gradient of the value function $V_{\mathbf{x}}$ at every time step, the output of the LSTM is used to predict the Hessian of the value function $V_{\mathbf{xx}}$ and $\mathbf{A} = \partial_t(V_{\mathbf{x}}) + \frac{1}{2}\mathrm{tr}(\partial_{\mathbf{xx}}(V_{\mathbf{x}})\mathbf{C}\mathbf{C}^{\mathrm{T}})$ using two separate fully connected (FC) layers with linear activations. Notice that $\partial_{\mathbf{xx}}(V_{\mathbf{x}})$ is a rank 3 tensor and using neural networks to predict V and compute this term using automatic differentiation would render this method unscalable as in Raissi (2018). We, however, bypass this problem by instead predicting the trace of the tensor product, which is a vector allowing linear growth in output size with state dimensionality.

Algorithm*: We refer the reader to Wang et al. (2019, Algorithm 1) for details regarding the training procedure of the Deep 2FBSDE Controller. Here, we present the loss function $\mathcal{L}$, which is computed using the propagated value function (V), its gradient ($V_{\mathbf{x}}$) and Hessian ($V_{\mathbf{xx}}$) at the final time step, compared against their terminal conditions denoted by *, as follows

$$\mathcal{L} = c_1 \, ||V^* - V_N||_2^2 + c_2 \, ||V_{\mathbf{x}}^* - V_{\mathbf{x},N}||_2^2 + c_3 \, ||V_{\mathbf{xx}}^* - V_{\mathbf{xx},N}||_2^2 + c_4 \, ||V^*||_2^2 + \lambda||\theta||_2^2. \quad (7)$$

The network can be trained using any variant of Stochastic Gradient Descent (SGD) such as Adam (Kingma and Ba, 2014) until convergence.

5. Simulation Results*

In this section, we demonstrate the capability of the Deep 2FBSDE Controller (2FBSDE) on four different systems in simulation. We first consider a high-dimensional linear system consisting of 100 states to demonstrate the scalablity of our algorithm using the 2FBSDE-FNNs and 2FBSDE-LSTM architectures and compare the performance with an analytical solution. For nonlinear systems, we consider a cart-pole swing-up task and a reaching task for a 12-state quadcopter. Finally, we tested on a 2-link 6-muscle (10-state) biomechanical human arm model for a planar reaching task. We chose the 2FBSDE-LSTM architecture as it is more memory efficient (fewer parameters especially for longer horizons) and consumes less training time compared to 2FBSDE-FNNs. The results were

*Refer to Wang et al. (2019), for our proposed algorithm and simulation details (hyperparameter values, state-space SDE models for each system and additional experiments) in the supplementary materials (SM G and SM H)

compared against both the Deep FBSDE Controller (referred to as FBSDE hereon) in Pereira et al. (2019), wherein the effect of control multiplicative noise was ignored by only considering the model with additive noise resulting in first-order FBSDEs, and the iLQG controller in Li and Todorov (2007), which explicitly considers control multiplicative noise. Our experiments were programmed and executed using Tensorflow-cpu (Abadi et al., 2015), on a desktop computer with an Intel Xeon E5-1607 v3 CPU (3.1Ghz x 4) with 64 GB RAM. All the code to reproduce the results presented in this section is available on our github repository [†].

All the comparison plots contain statistics gathered over 128 test trials. For 2FBSDE and FBSDE, we used a time discretization of $\Delta t = 0.02$ seconds for cart-pole, quadcopter, and human arm. For iLQG, $\Delta t = 0.01$ seconds for cartpole and quadcopter and $\Delta t = 0.001$ seconds for the human arm were used to avoid numerical instability. Regarding the exact choice of time discretization, the values were hand-tuned until we observed numerical stability, convergence, and reasonable task performance. We would like to re-iterate that in case of iLQG, finer time discretizations were required as compared to FBSDE and 2FBSDE. In all plots, the solid lines represent mean trajectories, and the shaded regions represent the 68% confidence region (mean $\pm$ standard deviation).

High Dimensional Linear System: We consider a linear time-invariant system given by $\mathrm{d}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t)\mathrm{d}t + \mathbf{B}\mathbf{u}(t)\mathrm{d}t + \sigma\mathbf{B}\mathbf{u}(t)\mathrm{d}v(t) + \mathbf{\Sigma}\mathrm{d}\mathbf{w}(t), \mathbf{x} \in \mathbb{R}^{100}$ with quadratic running state cost and terminal state cost $\mathbf{q}\big(\mathbf{x}(t)\big) = \phi\big(\mathbf{x}(T)\big) = (\mathbf{x} - \boldsymbol{\eta})^T\mathbf{Q}(\mathbf{x} - \boldsymbol{\eta})$. The dynamics and cost function parameters are set as $\mathbf{A} = \mathbf{0}, \mathbf{B} = \mathbf{I}, \Sigma = 0.5\mathbf{I}, \sigma = 0.1, \mathbf{Q} = 0.8\mathbf{I}, \mathbf{R} = 0.02\mathbf{I}, \mathbf{x}_0 = 1.0\mathbf{I}, \Delta t = 0.01$. The task is to drive all the states to $\boldsymbol{\eta} = 0$. Although these dynamics may seem trivial comprising of 100 independent scalar systems, such dynamics arise in high-dimensional multi-agent systems wherein the independent agents are only coupled through the cost function. We refer the reader to Wang et al. (2019, SM E, SM G.1) for derivation of the Riccati equations to compute the analytical solution and additional simulation details. Simulating the system for 1.5 seconds (despite task completion in 0.3 seconds) highlights the stability of 2FBSDE-LSTM compared to 2FBSDE-FNN. The divergence of 2FBSDE-FNN (*green*) results from the increase in variance of state costs across trials.

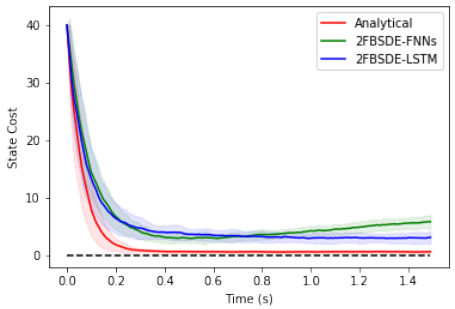

Figure 2: **Linear System Reaching Task:** 2FBSDE-LSTM has lower average cost compared to 2FBSDE-FNNs

Cartpole: This experiment was a swing-up task with a horizon of $T = 2.0$ seconds. We tested two different control cost coefficients and observed qualitatively different behavior. We present the results for $R = 0.5$ and refer the reader to Wang et al. (2019, SM H.1) for $R = 0.1$. We did not impose any cost or target for the cart position state. We would like to highlight in Fig. 3 that both 2FBSDE and iLQG (which take into account control multiplicative noise) chose to pre-swing the pendulum and took advantage of using the system's momentum to achieve the swing-up task. They thus respect the presence of control multiplicative noise entering the system as compared to FBSDE, which tries to drive the pendulum to the desired vertical position with large control effort.

Quadcopter: The controller was also tested on a 12-state quadcopter dynamics model for a task of reaching and hovering at a target position with a horizon of 2.0 seconds. Only linear and angular states are included in Fig. 4 since they most directly reflect the task performance (velocity plots

[†] GitHub repository (control_multi branch)

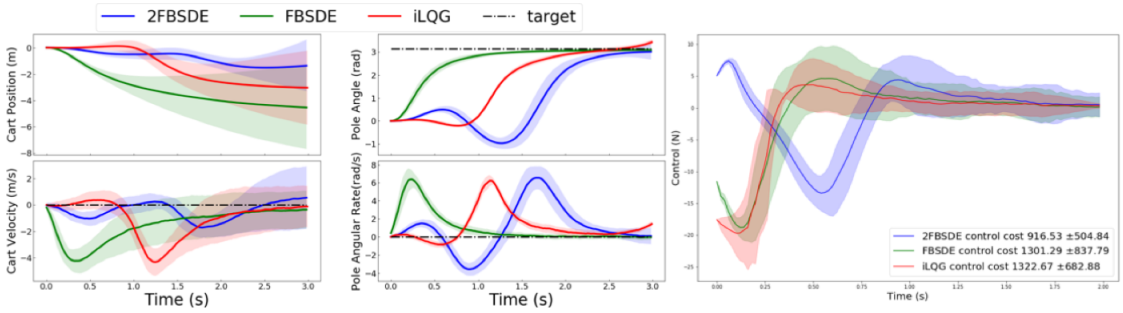

Figure 3: **Cartpole Swing-up Task:** 2FBSDE is most aware of the presence of control multiplicative noise and uses least control effort to perform the task. This is evident from the cart velocity plot, a directly actuated state, which tries to stay as close to zero as possible.

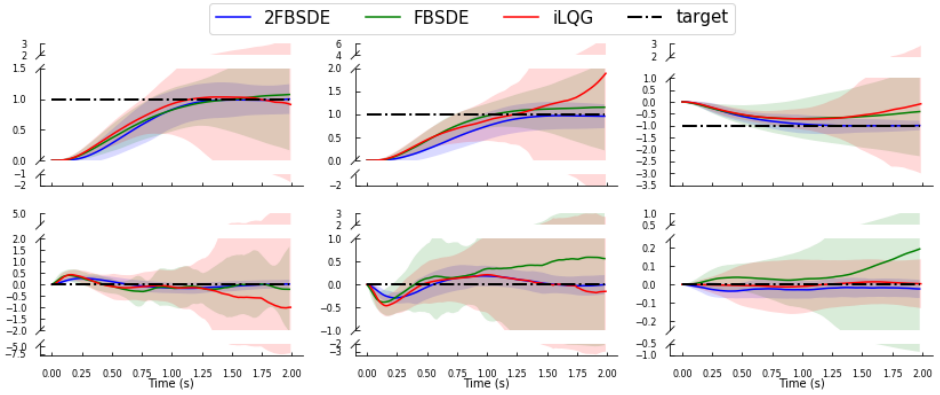

Figure 4: Above plots, from left to right, in the top row are x, y and z positions and botton row are roll, pitch and yaw angles. 2FBSDE clearly out-performs both FBSDE and iLQG, as indicated by the smaller variance in all states and stability while others diverge due to high noise.

included in Wang et al. (2019, SM H.3)). The figure demonstrates superior performance of 2FBSDE over FBSDE and iLQG controllers in reaching faster and maintaining lower variance. Moreover, these results also convey the importance of taking into account multiplicative noise in the design of optimal controllers as the state dimensionality and system complexity increases. We also tested the 2FBSDE controller for the same task (reach & hover) for a longer horizon of 3.0 seconds Wang et al. (2019, SM H.4)) to verify its stabilizing capability in the presence of control multiplicative noise.

2-link 6-muscle Human Arm: This is a 10-state bio-mechanical system wherein control multiplicative noise models have been found to closely mimic empirical observations (Todorov, 2005). The controllers were tasked with reaching a target position in 1.0 second. Additive noise in the joint-angle acceleration channels and multiplicative noise in muscle activations were considered. In Fig. 5, we show the log of terminal state cost ($\log \phi(\mathbf{x}_N)$) versus different values of additive noise standard deviations $\mathbf{\Sigma}$, (while holding multiplicative noise standard deviation fixed at $\sigma = 0.1$) on top and versus different values of σ (keeping $\mathbf{\Sigma}$ fixed at 0) on the bottom. The FBSDE results are omitted as failure occurred in all trials. As seen in the figure, when additive noise is varied (*top*), 2FBSDE outperforms iLQG at lower $\mathbf{\Sigma}$ and performance deteriorates much slower than iLQG as $\mathbf{\Sigma}$ increases. When multiplicative noise is varied (*bottom*), iLQG exhibits erratic

behaviour while 2FBSDE worsens gradually. The difference in behaviors can be attributed to the fact that iLQG is only aware of the first two moments of the uncertainty entering the system while 2FBSDE, being a sampling-based controller is exposed to the true uncertainty entering the system. We present the single control trajectory as well as mean and variance of 2FBSDE on Human Arm in (Wang et al., 2019, SM H.2), which demonstrates the practical feasibility of the resulting controls.

Discussion: Although the performance of iLQG for the cartpole and human arm tasks seems competitive to 2FBSDE, we would like to highlight the fact that iLQG becomes brittle as σ (multiplicative noise std. dev.) is increased and requires very fine time discretizations, proper regularization scheduling and fine state cost coefficient tuning for convergence. We observed that it is markedly more difficult to tune iLQG when there is both a terminal and running cost as compared to when there is only a terminal cost. To achieve the performance in Fig. 5, running cost had to be reduced by an order of magnitude compared to terminal cost for iLQG. Additionally, $\mathbf{R}$ had to be tuned to a high value to prevent divergence for higher σ. We implemented the regularization scheme for Q_{uu} as detailed in Tassa et al. (2012) in addition to back-tracking line search. We hypothesize that iLQG's sensitivity to linearization becomes a bigger problem in the presence of uncertainties causing decline in performance. Inspired by this, Torre and Theodorou. (2015) introduced a structured linearization approach, improving the performance of SDDP for simple systems in the presence of control multiplicative noise at the cost of increased complexity.

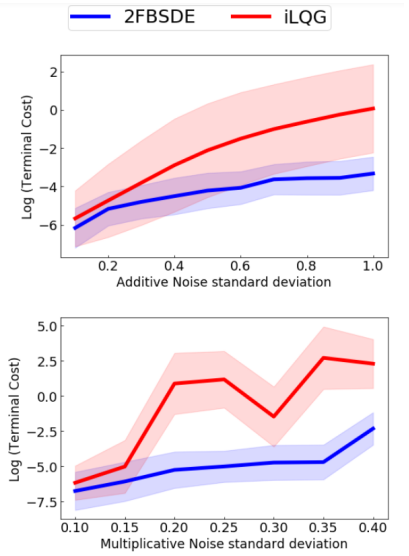

Figure 5: **Human Arm Planar Reaching Task:** 2FBSDE has lower mean and variance of terminal cost compared to iLQG.

6. Conclusion and Contributions

In this paper, we proposed novel DNN architectures tailored to solve SOCs corresponding to fully nonlinear HJB PDEs for stochastic systems with control multiplicative noise resulting in a system of 2FBSDEs. We introduced a new derivation for FBSDEs that incorporates importance sampling without relying on Girsanov's change of measure theorem, which cannot be used for systems with control multiplicative noise. We demonstrated scalability to high dimensional systems and robustness to time-discretization and successfully applied our method for SOC of nonlinear systems with additive and control multiplicative noise. Our simulations clearly illustrate the importance of considering the nature of stochastic disturbances in the problem formulation as well as in the architecture design. Finally, we can consider other architectures from the NLP community such as the Transformer network (Vaswani et al., 2017) and a proof of convergence as potential future directions.

Acknowledgements

This work was financially supported by the National Aeronautics and Space Administration (NASA), National Science Foundation's Cyber Physical Systems (CPS) award # 1932288 and Graduate Research Fellowship DGE-1842487, and USC Annenberg and WiSE Top-Off Fellowships.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL http://tensorflow.org/. Software available from tensorflow.org.

Mario Annunziato and Alfio Borzi. Optimal control of probability density functions of stochastic processes. *Mathematical Modelling and Analysis*, 15(4):393–407, 2010.

Mario Annunziato and Alfio Borzì. A fokker–planck control framework for multidimensional stochastic processes. *Journal of Computational and Applied Mathematics*, 237(1):487–507, 2013.

K. S. Bakshi, D. D. Fan, and E. A. Theodorou. Stochastic control of systems with control multiplicative noise using second order fbsdes. In *2017 American Control Conference (ACC)*, pages 424–431, May 2017. doi: 10.23919/ACC.2017.7962990.

Christian Beck, E Weinan, and Arnulf Jentzen. Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations. *Journal of Nonlinear Science*, 29(4):1563–1619, 2019.

Richard Bellman. *Dynamic Programming*. Dover Publications, March 2003. URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0486428095.

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

I. Exarchos and E. A. Theodorou. Learning optimal control via forward and backward stochastic differential equations. In *American Control Conference (ACC), 2016*, pages 2155–2161. IEEE, 2016.

I. Exarchos and E. A. Theodorou. Stochastic optimal control via forward and backward stochastic differential equations and importance sampling. *Automatica*, 87:159–165, 2018.

I. Exarchos, E. A. Theodorou, and P. Tsiotras. Stochastic L^1-optimal control via forward and backward sampling. *Systems & Control Letters*, 118:101–108, 2018.

Ioannis Exarchos, Evangelos Theodorou, and Panagiotis Tsiotras. Stochastic differential games: A sampling approach via fbsdes. *Dynamic Games and Applications*, 9(2):486–505, Jun 2019. ISSN 2153-0793. doi: 10.1007/s13235-018-0268-4. URL https://doi.org/10.1007/s13235-018-0268-4.

W. H. Fleming and H. M. Soner. *Controlled Markov processes and viscosity solutions*. Applications of mathematics. Springer, New York, 2nd edition, 2006.

Alex Gorodetsky, Sertac Karaman, and Youssef Marzouk. Efficient high-dimensional stochastic optimal motion control using tensor-train decomposition. In *Proceedings of Robotics: Science and Systems*, Rome, Italy, July 2015. doi: 10.15607/RSS.2015.XI.015.

Jiequn Han, Arnulf Jentzen, and Weinan E. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018. ISSN 0027-8424. doi: 10.1073/pnas.1718942115. URL https://www.pnas.org/content/115/34/8505.

V. A. Huynh, S. Karaman, and E. Frazzoli. An incremental sampling-based algorithm for stochastic optimal control. *I. J. Robotic Res.*, 35(4):305–333, 2016. doi: 10.1177/0278364915616866. URL http://dx.doi.org/10.1177/0278364915616866.

H. J. Kappen. Linear theory for control of nonlinear stochastic systems. *Phys Rev Lett*, 95:200201, 2005. Journal Article United States.

I. Karatzas and S. E. Shreve. *Brownian Motion and Stochastic Calculus (Graduate Texts in Mathematics)*. Springer, 2nd edition, August 1991. ISBN 0387976558.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

H. J. Kushner and P. G. Dupuis. *Numerical Methods for Stochastic Control Problems in Continuous Time*. Springer-Verlag, London, UK, UK, 1992. ISBN 0-387-97834-8.

Weiwei Li and Emanuel Todorov. Iterative linearization methods for approximately optimal control and estimation of non-linear stochastic system. *International Journal of Control*, 80(9):1439–1453, 2007.

P. McLane. Optimal stochastic control of linear systems with state- and control-dependent disturbances. *IEEE Transactions on Automatic Control*, 16(6):793–798, December 1971. ISSN 0018-9286. doi: 10.1109/TAC.1971.1099828.

Djordje Mitrovic, Stefan Klanke, Rieko Osu, Mitsuo Kawato, and Sethu Vijayakumar. A computational model of limb impedance control based on principles of internal model uncertainty. *PLOS ONE*, 5(10):1–11, 10 2010. doi: 10.1371/journal.pone.0013601. URL https://doi.org/10.1371/journal.pone.0013601.

Etienne Pardoux and Shige Peng. Adapted solution of a backward stochastic differential equation. *Systems & Control Letters*, 14(1):55–61, 1990.

Etienne Pardoux and Aurel Rascanu. *Stochastic Differential Equations, Backward SDEs, Partial Differential Equations*, volume 69. 07 2014. doi: 10.1007/978-3-319-05714-9.

Marcus A Pereira, Ziyi Wang, Ioannis Exarchos, and Evangelos A Theodorou. Learning deep stochastic optimal control policies using forward-backward sdes. In *Robotics: science and systems*, 2019.

Y. Phillis. Controller design of systems with multiplicative noise. *IEEE Transactions on Automatic Control*, 30(10):1017–1019, October 1985. ISSN 0018-9286. doi: 10.1109/TAC.1985.1103828.

J. A. Primbs. Portfolio optimization applications of stochastic receding horizon control. In *2007 American Control Conference*, pages 1811–1816, July 2007. doi: 10.1109/ACC.2007.4282251.

Maziar Raissi. Forward-backward stochastic neural networks: Deep learning of high-dimensional partial differential equations. *arXiv preprint arXiv:1804.07010*, 2018.

Steven E Shreve. *Stochastic calculus for finance II: Continuous-time models*, volume 11. Springer Science & Business Media, 2004.

R. F. Stengel. *Optimal control and estimation*. Dover books on advanced mathematics. Dover Publications, New York, 1994.

Y. Tassa, T. Erez, and E. Todorov. Synthesis and stabilization of complex behaviors through online trajectory optimization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4906–4913, Oct 2012. doi: 10.1109/IROS.2012.6386025.

E.A. Theodorou, J. Buchli, and S. Schaal. A generalized path integral approach to reinforcement learning. *Journal of Machine Learning Research*, (11):3137–3181, 2010a.

E.A. Theodorou, Y. Tassa, and E. Todorov. Stochastic differential dynamic programming. In *American Control Conference*, pages 1125–1132, 2010b.

E. Todorov. Stochastic optimal control and estimation methods adapted to the noise characteristics of the sensorimotor system. *Neural Computation*, 17(5):1084, 2005.

E. Todorov. Linearly-solvable markov decision problems. In B. Scholkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, Vancouver, BC, 2007. Cambridge, MA: MIT Press.

E. Todorov. Efficient computation of optimal actions. *Proceedings of the national academy of sciences*, 106 (28):11478–11483, 2009.

E. Todorov and W. Li. A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems. pages 300–306, 2005a.

E. Todorov and Weiwei Li. A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 300–306 vol. 1, June 2005b. doi: 10.1109/ACC.2005.1469949.

G. DeLa Torre and E.A. Theodorou. Stochastic variational integrators for system propagation and linearization. Sept 2015.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Ziyi Wang, Marcus A Pereira, Tianrong Chen, Emily A Reed, and Evangelos A Theodorou. Deep 2fbsdes for systems with control multiplicative noise. *arXiv preprint arXiv:1906.04762*, 2019.