2021 60th IEEE Conference on Decision and Control (CDC)
December 13-15, 2021. Austin, Texas

Non-Markovian Reinforcement Learning using Fractional Dynamics

Gaurav Gupta[†] Chenzhong Yin[†] Jyotirmoy V. Deshmukh[‡] Paul Bogdan[†]

2021 60th IEEE Conference on Decision and Control (CDC) | 978-1-6654-3659-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/CDC45484.2021.9683076

***Abstract*— Reinforcement learning (RL) is a technique to learn the control policy for an agent that interacts with a stochastic environment. In any given state, the agent takes some action, and the environment determines the probability distribution over the next state as well as gives the agent some reward. Most RL algorithms typically assume that the environment satisfies Markov assumptions (i.e. the probability distribution over the next state depends only on the current state). In this paper, we propose a model-based RL technique for a system that has non-Markovian dynamics. Such environments are common in many real-world applications such as in human physiology, biological systems, material science, and population dynamics. Model-based RL (MBRL) techniques typically try to simultaneously learn a model of the environment from the data, as well as try to identify an optimal policy for the learned model. We propose a technique where the non-Markovianity of the system is modeled through a fractional dynamical system. We show that we can quantify the difference in the performance of an MBRL algorithm that uses bounded horizon model predictive control from the optimal policy. Finally, we demonstrate our proposed framework on a pharmacokinetic model of human blood glucose dynamics and show that our fractional models can capture distant correlations on real-world datasets.**

I. Introduction

Reinforcement learning (RL) [1] is a technique to synthesize control policies for autonomous agents that interact with a stochastic environment. The RL paradigm now contains a number of different kinds of algorithms, and has been successfully used across a diverse set of applications including autonomous vehicles, resource management in computer clusters [2], traffic light control [3], web system configuration [4], and personalized recommendations [5]. In RL, we assume that in each state, the agent performs some action and the environment picks a probability distribution over the next state and assigns a reward (or negative cost). The reward is typically defined by the user with the help of a state-based (or state-action-based) reward function. The expected payoff that the agent may receive in any state can be defined in a number of different ways; in this paper, we assume that the payoff is an discounted sum of the local rewards (with some discount factor $\gamma \in [0, 1]$) over some time horizon H. The purpose of RL is to find the stochastic policy (i.e. a distribution over actions conditioned on the current state), that optimizes the expected payoff for the agent. Most RL algorithms assume that the environment satisfies Markov assumptions, i.e. the probability distribution over the next state is dependent only on the current state (and not the history). In contrast, here, we investigate an RL procedure for a non-Markovian environment.

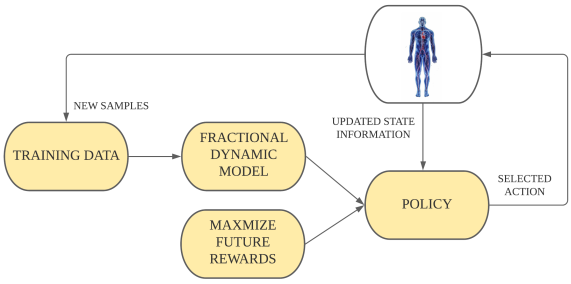


Fig. 1. Non-Markovian Model Based Reinforcement Learning setup. The model based predictions are used to select actions, and then iteratively update the model dynamics.

Broadly speaking, there are two classes of RL algorithms [6]: model-based and model-free algorithms. Most classical RL algorithms are *model-based*; they assume that the environment is explicitly specified as a Markov Decision Process (MDP), and use dynamic programming to compute the expected payoff for each state of the MDP (called its value), as well as the optimal policy [7], [8]. Classical RL algorithms have strong convergence guarantees stemming from the fact that the value of a state can be recursively expressed in terms of the value of the next state (called the Bellman equation), which allows us to define an operator to update the value (or the policy) for a given state across iterations. This operator (also known as the Bellman operator) can be shown to be a contraction mapping [1]. However, obtaining exact symbolic descriptions of models is often infeasible. This led to the development of model-free reinforcement learning (MFRL) approaches that rely on sampling many model behaviors through simulations and eschew building a model altogether. MFRL algorithms can converge to an optimal policy under the right set of assumptions; however, can suffer from high sample complexity (i.e. the number of simulations required to learn an optimal policy). This has led to investigation of a new class of model-based RL (MBRL) algorithms where the purpose is to simultaneously learn the system model as well as the optimal policy [2]. Such algorithms are called *on policy*, as the policy learned during any iteration is used for improving the learned model as well as optimizing the policy further. Most MBRL approaches use function approximators or Bayesian models to efficiently learn from scarce sample sets of system trajectories. MBRL approaches tend to have lower sample complexity than MFRL as the learned model can accelerate

[†]Ming Hsieh Department of Electrical and Computer Engineering, Univ. of Southern California, Los Angeles, CA, USA {ggaurav, chenzhoy, pbogdan}@usc.edu
[‡]Department of Computer Science, Univ. of Southern California, Los Angeles, CA, USA jdeshmuk@usc.edu

the convergence by focusing on actions that are likely to be close to the optimal action. However, MBRL approaches can suffer severely from modeling errors [9], and may converge to less optimal solutions.

In both MFRL and MBRL algorithms, a fundamental assumption is that the environment satisfies Markovian properties, partly to avoid the complexity of dealing with the historical dependence in transitions. To overcome this challenge, we propose a non-Markovian MBRL framework that captures non-Markovian characteristics through a fractional dynamical systems formulation. Fractional dynamical systems can model non-Markovian processes characterized by a single fractal exponent and commonly arise in mathematical models of human physiological processes [10], [11], biological systems, condensed matter and material sciences, and population dynamics [12]–[15]. Such systems can effectively model spatio-temporal properties of physiological signals such as blood oxygenation level dependent (BOLD), electromyogram (EMG), electrocardiogram (ECG), etc. [12], [16]. The advantage of using fractional dynamical models is that they can accurately represent long-range (historical) correlations (memory) through a minimum number of parameters (e.g., using a single fractal exponent to encode a long-range historical dependence rather than memorizing the trajectory itself or modeling it through a large set of autoregressive parameters). Though fractional models can be used to perform predictive control [17], problems such as learning these models effectively or obtaining optimal policies for such models in an RL setting have not been explored.

In this paper, we develop a novel non-Markovian MBRL technique in which our algorithm alternates between incrementally learning the fractional exponent from data and learning the optimal policy on the updated model. We show that the optimal action in a given state can be efficiently computed by solving a quadratic program over a bounded horizon rollout from the state. The overview of our model-based reinforcement learning algorithm is shown in Fig. 1. In this algorithm, we use on-policy simulations to gather additional RL data that is then used to update the model. Our model learning algorithm is based on minimizing the distance between the data's state-action distribution and the next state distribution induced by the controller. The fractional dynamic model is then retrained using the cumulative dataset. The MBRL procedure is run for a finite number of user-specified iterations.

The rest of this paper is constructed as follows. We present our problem statement in Section II. Section III contains our proposed non-Markovian MBRL algorithm. We demonstrate our experimental results in Section IV. In the end, we conclude this paper with discussion and conclusion in Section V.

II. Problem Formulation

The reinforcement learning deals with the design of the controller (or policy) which minimizes the expected total cost. In the setting of a memoryless assumption, the Markov Decision Process (MDP) [18] is used to model the system dynamics such that the future state depends only on the current state and action. For a state $\mathbf{s}_t \in \mathbb{R}^n$ and action $\mathbf{a}_t \in \mathbb{R}^p$, the future state evolve as $\mathbf{s}_{t+1} \sim P(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$, and a cost function $r_t = c(\mathbf{s}_t, \mathbf{a}_t)$. However, the Markov assumption does not work well with the long-range memory processes [19]. In this work, we take the non-Markovian setting, or History Dependent Process (HDP), and hence, the future state depends not only on the current action but also the history of states. The history at time t is the set $\mathcal{H}_t = \{(s_k)_{k\leqslant t}\}$, and for a trajectory $h \in \mathcal{H}_t$, we have $P(\mathbf{s}_{t+1}|h, \mathbf{a}_t)$, or alternatively, $P_h(\mathbf{s}_{t+1}|s_t, \mathbf{a}_t)$, where the terminal state of the trajectory h is written as $h(t) = s_t$. We consider a model-based approach for reinforcement learning in a finite-horizon setting. A non-Markovian policy $\pi(.|h)$ provides a distribution over actions given the history of states until time t as $h \in \mathcal{H}_{\sqcup}$. For a given policy, the value function is defined as $V_h^\pi = \mathbb{E}_{\pi(.|h)} \sum_{t=0}^{T-1} c(s_t, a_t)$, where the expectation is taken over state trajectories using policy π and the HDP, and T is the horizon under consideration. We formally define the non-Markovian MBRL problem in the Section II-B.

A. Fractional Dynamical Model

A linear discrete time fractional-order dynamical model is described as follows:

$$\Delta^\alpha \mathbf{s}[k+1] = \mathbf{A}\mathbf{s}[k] + \mathbf{B}\mathbf{a}[k], \tag{1}$$

where $\mathbf{s} \in \mathbb{R}^n$ is the state, $\mathbf{a} \in \mathbb{R}^p$ is the input action. The difference between a classic linear time-invariant (or Markovian) and the above model is the inclusion of fractional-order derivative whose expansion and discretization for any ith state $(1 \leqslant i \leqslant n)$ can be written as

$$\Delta^{\alpha_i} s_i[k] = \sum_{j=0}^{k} \psi(\alpha_i, j) s_i[k-j], \tag{2}$$

where α_i is the fractional order corresponding to the ith state dimension and $\psi(\alpha_i, j) = \frac{\Gamma(j-\alpha_i)}{\Gamma(-\alpha_i)\Gamma(j+1)}$ with $\Gamma(.)$ denoting the gamma function. The system dynamics can also be written in the probabilistic manner as follows:

$$P_{\boldsymbol{\theta}}(\mathbf{s}[k+1]|\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k]) = \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\theta}}, \boldsymbol{\Sigma}),$$

$$\boldsymbol{\mu}_{\boldsymbol{\theta}} = \begin{bmatrix} \sum_{j=1}^{k} \psi(\alpha_i, j) s_0[k-j] + \mathbf{a}_0^T \mathbf{s}[k] + \mathbf{b}_0^T \mathbf{a}[k] + \mu_0 \\ \sum_{j=1}^{k} \psi(\alpha_i, j) s_1[k-j] + \mathbf{a}_1^T \mathbf{s}[k] + \mathbf{b}_1^T \mathbf{a}[k] + \mu_1 \\ \vdots \\ \sum_{j=1}^{k} \psi(\alpha_i, j) s_{n-1}[k-j] + \mathbf{a}_{n-1}^T \mathbf{s}[k] \\ \qquad + \mathbf{b}_{n-1}^T \mathbf{a}[k] + \mu_{n-1} \end{bmatrix}, \tag{3}$$

where $\boldsymbol{\theta} = \{\alpha, \mathbf{A}, \mathbf{B}, \mu, \boldsymbol{\Sigma}\}$, and $\mathbf{A} = [\mathbf{a}_0, \ldots, \mathbf{a}_{n-1}]$, $\mathbf{B} = [\mathbf{b}_0, \ldots, \mathbf{b}_{n-1}]$. The fractional differencing operator in (3) introduce the non-Markovianity by having long-range filtering operation on the state vectors.

B. Non-Markovian Model Based Reinforcement Learning

The actions in MBRL are preferred on the basis of predictions made by the undertaken model of the system dynamics. For many real-world systems, example blood glucose [17], [20], ECG activities [11], the assumption of Markovian dynamics does not hold and hence the predictions are not accurate, leading to less rewarding actions selected for the system. As we note in the previous section II-A that non-Markovian dynamics can be effectively and compactly modeled as fractional dynamical system, we aim to use this system model for making predictions. The non-Markovian MBRL problem is formally defined as follows.

Problem Statement: Given non-Markovian state transitions, and actions dataset in the time horizon $k \in [0, T-1]$ as $\mathcal{D} = \big\{(\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k]), \mathbf{s}[k+1]\big\}$. Let $P_{\boldsymbol{\theta}}(\mathbf{s}[k+1]|\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k])$ be the non-Markovian system dynamics parameterized by the model parameters $\boldsymbol{\theta}$. Estimate the optimal policy which minimizes the expected future discounted cost

$$\pi^* = \arg\min_{\pi} \mathbb{E} \sum_{k=0}^{T-1} \gamma^k c(\mathbf{s}[k], \mathbf{a}[k]), \tag{4}$$

where γ is the discount factor satisfying $\gamma \in [0, 1]$, and T is the horizon under consideration.

III. Non-Markovian Reinforcement Learning

The MBRL comprises of two key steps, namely (*i*) the estimation of the model dynamics from the given data $\mathcal{D}$, and (*ii*) the design of a policy for optimal action selection which minimizes the total expected cost using estimated dynamics. We discuss the solution to the non-Markovian MBRL as follows.

A. Non-Markovian Model Predictive Control

The Model Predictive Control (MPC) aims at estimating the closed-loop policy by optimizing the future discounted cost under a limited-horizon H using some approximation of the environment dynamics and the cost. In this work, we are concerned with HDP using non-Markovian state dynamics. In MPC, the policy could be a deterministic action, or a distribution over actions, and we sample the action at each time-step in the latter. The MPC problem to estimate the policy at time-step k for a given $h \in \mathcal{H}_k$ can be formally defined as

$$\begin{aligned} \min_{\pi(.|h)} \quad & \textstyle\sum_{l=k}^{k+H-1} \gamma^{l-k} \hat{c}(s[l], a[l]) \\ \text{subject to} \quad & \\ & s[l+1] = f(h, a[l], e[l]), \forall l \geqslant k \end{aligned} \tag{5}$$

The approximation of the environment dynamics f could be non-linear in general, and $e[l]$ is the system perturbation noise following some distribution $e \sim g_e$. The presence of e provides randomness in the action sampling through policy, and the sampled action at each step is $a[k]$. The performance of the non-Markovian MPC based policy is bounded within the optimal policy using the following result.

Theorem 1. *Given an approximate HDP with* $||\hat{P}_{h'}(s'|s,a) - P_h(s'|s,a)||_1 \leqslant \mathcal{O}(t^q)$, $\forall h, h' \in \mathcal{H}_t$ *with* $h(t) = h'(t) = s$, *and* $||c(s,a) - \hat{c}(s,a)||_\infty \leqslant \varepsilon$. *The performance of the non-Markovian MPC based policy* $\hat{\pi}$ *is related to the optimal policy* π^* *as*

$$\begin{aligned} ||V_{h_0}^{\hat{\pi}} - V_{h_0}^{\pi^*}||_\infty \leqslant\ & 2\frac{1-\gamma^H}{1-\gamma}\left(\frac{c_{max} - c_{min}}{2}\right) H\mathcal{O}(T^q) \\ & + 2\varepsilon \frac{1-\gamma^H}{1-\gamma}\frac{1-\gamma^T}{1-\gamma}, \end{aligned} \tag{6}$$

where, $h_0 \in \mathcal{H}_0$ *is the initial history given to the system.*

The proof of Theorem 1 is provided in the full version of the manuscript. The assumption of model approximation is critical here, and the error increases if the exponent q increases. For the MDP setting, the approximation is taken as $q = 0$. However, for a HDP with the history of length t, we scale the approximation gap with t. The MPC horizon also plays a role in the error bound, and the error increases for larger H.

The non-Markovian MPC could be computationally prohibitive (expensive) in the general setting. Consequently, we now discuss the fractional dynamical MPC approach which is non-Markovian but computationally tractable.

B. Fractional Model Predictive Control

The linear discrete fractional dynamical model as discussed in (1) is used as an approximation to the non-Markovian environment dynamics. Formally, for our purpose, the fractional MPC problem using (5) is defined as

$$\begin{aligned} \min_{\mathbf{a}[k]} \quad & \textstyle\sum_{l=k}^{k+H-1} \gamma^{l-k} \hat{c}(\mathbf{s}[l], \mathbf{a}[l]) \\ \text{s.t.} \quad & \\ & \Delta^\alpha \bar{\mathbf{s}}[l+1] = \mathbf{A}\bar{\mathbf{s}}[l] + \mathbf{B}\mathbf{a}[l] + e[l], \\ & \bar{\mathbf{s}}[k'] = \mathbf{s}[k], \forall k' \leqslant k, \mathbf{s}_{min} \leqslant \bar{\mathbf{s}}[l] \leqslant \mathbf{s}_{max}, \forall l, \end{aligned} \tag{7}$$

where $\mathbf{s}_{min}, \mathbf{s}_{max}$ are feasibility bounds on the problem according to the application, and the model noise $e \sim \mathcal{N}(0, \Sigma)$. Note that (7) provides a policy using fractional MPC. The action $a[k]$ is sampled from this policy by first sampling $e \sim \mathcal{N}(0, \Sigma)$, and then solving (7). The non-Markovian fractional dynamics would introduce the computation complexities in optimally solving the problem in (7). However, since the constraints in (7) are linear, for cost approximations $\hat{c}$ that are quadratic, a quadratic programming (QP) solution can be developed to solve the fractional MPC efficiently. We refer the reader to the full version of the paper for the QP version of the fractional MPC. Further, a convex formulation of the costs $\hat{c}$ also enables efficient solution of the fractional MPC using convex programming solvers, for example, CPLEX and Gurobi [21], [22].

Next, we discuss the methodologies required to make an approximation of the non-Markovian environment using fractional dynamics.

C. Model Estimation

The fractional dynamical model as described in the Section II-A is estimated using the approach proposed in [12] by replacing the unknown inputs with known actions at any time-step. For the sake of completeness, we present estimation algorithm as Algorithm 1. We note that in [12] the input data is obtained only once, and hence in this work appropriate modification in Algorithm 1 is performed to work with recursively updated dataset as we see in Section III-D.

Algorithm 1 Fractional_Dynamics_Estimation

Input: $\mathcal{D} = \{(\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k]), \mathbf{s}[k+1]\}$ in the time-horizon $k \in [0, T-1]$

Output: $\boldsymbol{\theta} = \{\alpha, \mathbf{A}, \mathbf{B}, \mu, \boldsymbol{\Sigma}\}$

1: Estimate α using wavelets fitting for each state dimension
2: **for** $i = 1, 2, \ldots, n$ **do**
3: Compute $z_i[k] = \Delta^{\alpha_i} s[k+1]$ using α_i ▷ Eq.2
4: Aggregate $z_i[k], s[k], a[k]$ as Z_i, S, U
5: $[a_i^T, b_i^T, \mu] = \arg\min_{a,b,\mu} ||Z_i - Sa - Ub - \mu||_2^2$ with $\boldsymbol{\Sigma}$ as squared error
6: **end for**

The Markovian model assume memoryless property and hence lacks long-range correlations for further accurate modeling. The existence of long-range correlations can be estimated by computing the Hurst exponent $\bar{H}$. For long-range correlations, the $\bar{H}$ lies in the range of $(0.5, 1]$. The fractional coefficient α in our model is related with $\bar{H}$ as $\alpha = \bar{H} - 0.5$. The Hurst exponent can be estimated from the slope of log-log variations of the variance of wavelets coefficients vs scale as noted in [23]. In the experiments Section IV-B, we show log-log plot to observe the presence of long-range correlations in the real-world data.

D. Model Based Reinforcement Learning

The non-Markovian MPC exploiting the fractional dynamical model formulation in Section III-B utilizes a dataset of the form $\mathcal{D} = \{(\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k]), \mathbf{s}[k+1]\}$ in the time-horizon $k \in [0, T-1]$. We note that the performance of such MPC can be further improved by using reinforcement learning. The selected actions by the MPC $\mathbf{a}[k]$ can be used to gather new transitions $\mathbf{s}[k+1]|\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k]$, or acquiring data using on-policy. The aggregated data is now used to re-estimate the model dynamics, and then perform MPC. Specifically, the MBRL proceeds as follows: Using the seed dataset, a parameterized fractional dynamics model is learned as $P_{\boldsymbol{\theta}}(\mathbf{s}[k+1]|\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k])$. This model is used to minimize the discounted future cost as MPC in eq. (7). The selected action along with the history of states $\mathbf{s}[0], \ldots, \mathbf{s}[k]$ is used to gather the next transition using on-policy as $\mathbf{s}[k+1]|\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k]$. The seed dataset is updated with the gathered on-policy data $\mathcal{D}_{RL}$ to get aggregated dataset. The fractional dynamics are updated using the new dataset, and the aforementioned steps are repeated for a given number of iterations. These steps are summarized as Algorithm 2. The Algorithm 2 utilizes Algorithm 1 iteratively for the fractional model estimation. Next, we discuss the numerical experiments validation in Section IV.

Algorithm 2 Fractional_Reinforcement_Learning

Input: Seed dataset $\mathcal{D}_s = \{(\mathbf{s}[0], \ldots, \mathbf{s}[k], \mathbf{a}[k]), \mathbf{s}[k+1]\}$ in the time-horizon $k \in [0, T-1]$

Output: $\boldsymbol{\theta}$

Initialize: $\mathcal{D}_{RL} \leftarrow \phi$

1: **for** $iter = 1, 2, \ldots, iter\_max$ **do**
2: $\theta \leftarrow$ Fractional_Dynamics_Estimation$(\mathcal{D}_s \cup \mathcal{D}_{RL})$
3: Set initial state $\bar{\mathbf{s}}[0] \leftarrow \mathbf{s}[0]$
4: **for** $k = 0, 1, \ldots, T-1$ **do**
5: Sample action $a[k]$ from the fractional MPC based policy using $\bar{\mathbf{s}}[l], \forall l \leqslant k$ ▷ Eq.7
6: Get $\bar{\mathbf{s}}[k+1]$ by executing $\mathbf{a}[k]$
7: $\mathcal{D}_{RL} \leftarrow \mathcal{D}_{RL} \cup \{(\bar{\mathbf{s}}[0], \ldots, \bar{\mathbf{s}}[k], \mathbf{a}[k]), \bar{\mathbf{s}}[k+1]\}$
8: **end for**
9: **end for**

IV. Experiments

We test the fractional MBRL on a blood glucose (BG) control case study. BG control seeks to maintain the BG within $70 - 180 mg/dL$ range. BG control is crucial in the treatment of T1 diabetes patients which have inability to produce the required insulin amounts. The low levels of glucose in the blood plasma is termed as hypoglycemia, while the high levels is termed as hyperglycemia. For the application of reinforcement learning, the cost function is taken as risk associated with different levels of BG in the system. In [24] a quantified version of risk is proposed as function of BG levels which is written as follows.

$$\begin{aligned} f(b) &= 1.509 \times (\log(b)^{1.084} - 5.381), \\ R(b) &= 10 \times (f(b))^2. \end{aligned} \tag{8}$$

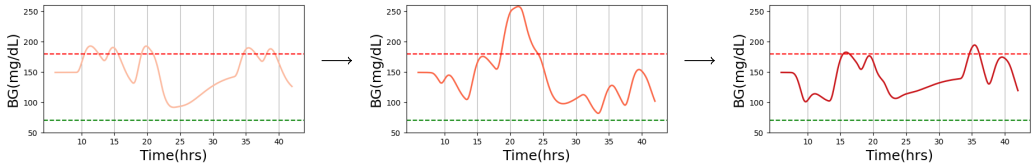
Next, the cost for the transition instance $s[k+1]|s[0], \ldots, s[k], a[k])$ is written as

$$\hat{c}(s[k], a[k]) = R(s[k+1]) - R(s[k]), \tag{9}$$

where the state $s[k] \in \mathbb{R}$ represents the BG level at time instant k, and $a[k]$ represents the insulin dose and $R(.)$ is from (8). In rest of the section, we experiment with simulated and real-world dataset, respectively.

A. UVa T1DM Simulator

The UVa/Padova T1DM [25] is a FDA approved T1 Diabetes simulator which supports multiple virtual subjects(we used an open-source implementation [26]). We take similar simulation setup as in [27]. Each subject is simulated for a total of 36 hours starting from 6 a.m. in the morning. The meal timings/quantity are fixed as $50g$ CHO at 9 a.m., $70g$ at 1 p.m, $90g$ at 5:30 p.m, and $25g$ at 8 p.m. On day 2, $50g$ at 9 a.m., and $70g$ at 1 p.m. The continuous glucose monitor (CGM) sensor measures the BG at every 5 mins.

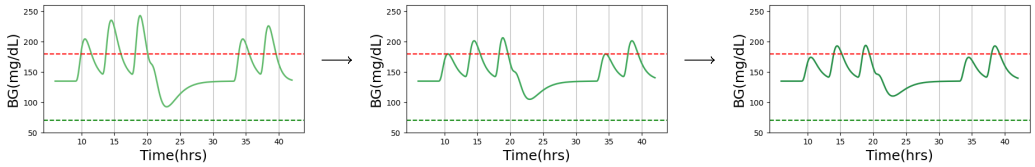


< 70 mg/dL	$70 - 180$ mg/dL	> 180 mg/dL
0%	76.67%	23.32%

< 70 mg/dL	$70 - 180$ mg/dL	> 180 mg/dL
0%	84.52%	15.47%

< 70 mg/dL	$70 - 180$ mg/dL	> 180 mg/dL
0%	93.53%	6.46%

(a)



< 70 mg/dL	$70 - 180$ mg/dL	> 180 mg/dL
0%	74.13%	25.86%

< 70 mg/dL	$70 - 180$ mg/dL	> 180 mg/dL
0%	87.52%	12.47%

< 70 mg/dL	$70 - 180$ mg/dL	> 180 mg/dL
0%	90.30%	9.69%

(b)

Fig. 2. Blood Glucose (BG) level with time, by implementation of fractional Reinforcement Learning Scheme as Controller, of two Adults in (a) and (b). For each subject, the BG level trajectories are shown from left-to-right in the increasing number of RL iterations with leftmost, middle, and rightmost are outputs at 5, 10, and 15 iterations. As RL iterations increase the MBRL scheme learns better policy and the BG level stays more in the desired level of $70 - 180 mg/dL$. The percentage of time spend in different BG level zone is shown in tables below each plot.

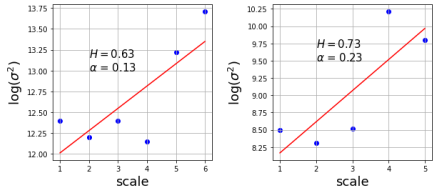


Fig. 3. Log-log plot of wavelet coefficients variance vs scale of two subjects in (a) and (b). Slope α lies in $(0, 0.5]$, indicating long-range memory.

For applying Algorithm 2, we set the horizon length H in MPC be 100 samples, discount factor $\gamma = 0.99$. The s_{min}, s_{max} in MPC problem (7) are set as $70, 180$ respectively. The maximum number of RL iterations $iter\_max$ are set as 30. We show the BG output of the simulator using Algorithm 2 as controller in Fig. 2. We observe that the fraction of time BG stays in the desired zone $70-180 mg/dL$ increase with increasing the learning iterations in the Algorithm 2. The data gathered using on-policy helps the model making better prediction, and with as few iterations as 15 we have more than 90% of time BG stays in the desired levels.

B. Real-World Data

Testing the controllers on real-world systems is difficult because of the health risks associated with the patients. We take the Diabetes dataset from UCI repository [28] which records the BG level and insulin dosage for 70 patients. While testing controller is not possible here, hence we present the analysis regarding the modeling part. The long-range memory in the signals exist if the associated fractional exponent lies in the range of $(0, 0.5]$ as noted in Section III-C. In Fig. 3, we show the log-log plots of the variance of wavelets coefficients at various scales, for two subjects. We observe that the estimated value of α lies in $(0, 0.5]$ which indicates presence of long-range memory, and hence fractional models can be used to make better predictions.

V. Conclusion

There are many important learning control problems that are not naturally formulated as Markov decision processes. For example, if the agent cannot directly observe the environment state, then the use of a partially observable Markov decision process (POMDP) [29] model is more appropriate. Even in presence of full observability, the probability distribution over next states may not depend only on the current state. A more general class can be termed as History Dependent Process (HDP), which can be looked as infinite-state POMDP [30]. Another non-MDP class for model-free is Q-value Uniform Decision Process (QDP) [31]. The non-Markovianity in the rewards structure is explored in [32], [33] which utilize model-free learning, and RL for POMDP is explored in [34] which is also model-free. MBRL is used for various robotics application [35] in the MDP setting. The deep probabilistic networks using MDP is used in [6].

In this work, we constructed a non-Markovian Model Based Reinforcement Learning (MBRL) algorithm consisted with fractional dynamics model and the model predictive control. The current Reinforcement learning (RL) approaches have two kinds of limitations: (i) model-free RL models can achieve a high predict accuracy, but these approaches need a large number of data-points to train the model; (ii) current models don't make latent behavioral patterns into considerations which can affect the prediction accuracy in MBRL. We show that our non-Markovian MBRL model can validly avoid these limitations. Firstly, in our algorithm, we gather additional on-policy data to alternate between gathering the initial data, hence it needs less sample points than

the general model-free RL approaches. Secondly, fractional dynamical model is the key element in our algorithm to improve/guarantee the prediction accuracy. The experiments on the blood glucose (BG) control to dynamically predict the desired insulin amount show that the proposed non-Markovian framework helps in achieving desired levels of BG for longer times with consistency.

The richness of complex systems cannot be always modeled as Markovian dynamics. Previous works have shown that the long-range memory property of fractional differentiation operators can model biological signals efficaciously and accurately. Thus, we have modeled the blood glucose as non-Markovian fractional dynamical system and developed solutions using reinforcement learning approach. Finally, while the application of non-Markovian MBRL open venues for real-world implementation but proper care has to be taken especially when we have to deal with the healthcare systems. The future investigations would involve more personalized modeling capabilities for such systems with utilization of the domain knowledge. Nonetheless, we show that the use of long-range dependence in the biological models is worth exploring and simple models yield benefits of compactness as well as better accuracy of the predictions.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support by the National Science Foundation (NSF) Career award under Grant No. CPS/CNS-1453860, the NSF award under Grant CCF-1837131, MCB-1936775, CNS-1932620, the Okawa Foundation award, and the Defense Advanced Research Projects Agency (DARPA) Young Faculty Award and DARPA Director Award under Grant No. N66001-17-1-4044, and a Northrop Grumman grant. The views, opinions, and/or findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies, either expressed or implied by the Defense Advanced Research Projects Agency, the Department of Defense or the National Science Foundation.

REFERENCES

[1] D. P. Bertsekas, *Reinforcement learning and optimal control*. Athena Scientific Belmont, MA, 2019.

[2] H. Mao, M. Alizadeh, I. Menache, and S. Kandula, "Resource management with deep reinforcement learning," in *Proceedings of the 15th ACM Workshop on Hot Topics in Networks*, 2016, pp. 50–56.

[3] I. Arel, C. Liu, T. Urbanik, and A. G. Kohls, "Reinforcement learning-based multi-agent system for network traffic signal control," *IET Intelligent Transport Systems*, vol. 4, no. 2, pp. 128–135, 2010.

[4] X. Bu, J. Rao, and C.-Z. Xu, "A reinforcement learning approach to online web systems auto-configuration," in *2009 29th IEEE International Conference on Distributed Computing Systems*, 2009, pp. 2–11.

[5] G. Zheng, F. Zhang, Z. Zheng, Y. Xiang, N. J. Yuan, X. Xie, and Z. Li, "Drn: A deep reinforcement learning framework for news recommendation," in *World Wide Web Conference*, 2018, pp. 167–176.

[6] K. Chua, R. Calandra, R. McAllister, and S. Levine, "Deep reinforcement learning in a handful of trials using probabilistic dynamics models," in *Advances in Neural Information Processing Systems*, 2018.

[7] R. S. Sutton, D. Precup, and S. Singh, "Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning," *Artificial intelligence*, vol. 112, no. 1-2, pp. 181–211, 1999.

[8] A. L. Strehl, L. Li, and M. L. Littman, "Reinforcement learning in finite mdps: Pac analysis." *Journal of Machine Learning Research*, vol. 10, no. 11, 2009.

[9] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.

[10] B. West, "Fractal physiology and the fractional calculus: A perspective," *Frontiers in Physiology*, vol. 1, p. 12, 2010. [Online]. Available: https://www.frontiersin.org/article/10.3389/fphys.2010.00012

[11] Y. Xue, S. Pequito, J. R. Coelho, P. Bogdan, and G. J. Pappas, "Minimum number of sensors to ensure observability of physiological systems: a case study," in *Allerton*, 2016.

[12] G. Gupta, S. Pequito, and P. Bogdan, "Dealing with unknown unknowns: Identification and selection of minimal sensing for fractional dynamics with unknown inputs," in *2018 Annual American Control Conference (ACC)*, 2018.

[13] C. Yin, G. Gupta, and P. Bogdan, "Discovering laws from observations: A data-driven approach," in *International Conference on Dynamic Data Driven Application Systems*, 2020.

[14] G. Gupta, S. Pequito, and P. Bogdan, "Re-thinking eeg-based non-invasive brain interfaces: modeling and analysis," in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2018, pp. 275–286.

[15] ——, "Learning latent fractional dynamics with unknown unknowns," in *2019 American Control Conference (ACC)*, 2019, pp. 217–222.

[16] D. Baleanu, J. A. T. Machado, and A. C. Luo, *Fractional dynamics and control*. Springer Science & Business Media, 2011.

[17] M. Ghorbani and P. Bogdan, "Reducing risk of closed loop control of blood glucose in artificial pancreas using fractional calculus," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2014, pp. 4839–4842.

[18] R. BELLMAN, "A markovian decision process," *Journal of Mathematics and Mechanics*, vol. 6, no. 5, pp. 679–684, 1957.

[19] S. Micciche, "Modeling long-range memory with stationary markovian processes," *Physical Review E*, vol. 79, no. 3, p. 031116, 2009.

[20] M. Otoom, H. Alshraideh, H. M. Almasaeid, D. López-de Ipiña, and J. Bravo, "A real-time insulin injection system," in *International Workshop on Ambient Assisted Living*. Springer, 2013, pp. 120–127.

[21] J. Kronqvist, D. E. Bernal, A. Lundell, and I. E. Grossmann, "A review and comparison of solvers for convex minlp," *Optimization and Engineering*, vol. 20, no. 2, pp. 397–455, 2019.

[22] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Automated configuration of mixed integer programming solvers," in *International Conference on Integration of Artificial Intelligence (AI) and Operations Research (OR) Techniques in Constraint Programming*, 2010.

[23] P. Flandrin, "Wavelet analysis and synthesis of fractional brownian motion," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 910–917, March 1992.

[24] W. Clarke and B. Kovatchev, "Statistical tools to analyze continuous glucose monitor data," *Diabetes Technol. Ther.*, pp. 45–54, Jun 2009.

[25] B. P. Kovatchev, M. Breton, C. D. Man, and C. Cobelli, "In silico preclinical trials: a proof of concept in closed-loop control of type 1 diabetes," *J Diabetes Sci Technol*, vol. 3, no. 1, pp. 44–55, Jan 2009.

[26] J. Xie, "Simglucose v0.2.1," 2018. [Online]. Available: https://github.com/jxx123/simglucose

[27] Q. Wang, J. Xie, P. Molenaar, and J. Ulbrecht, "Model predictive control for type 1 diabetes based on personalized linear time-varying subject model consisting of both insulin and meal inputs: In silico evaluation," in *2015 American Control Conference (ACC)*, 2015.

[28] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[29] R. D. Smallwood and E. J. Sondik, "The optimal control of partially observable markov processes over a finite horizon," *Operations Research*, vol. 21, no. 5, pp. 1071–1088, 1973.

[30] J. Leike, "Nonparametric general reinforcement learning," 2016.

[31] S. J. Majeed and M. Hutter, "On q-learning convergence for non-markov decision processes," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI*, 2018.

[32] M. Gaon and R. I. Brafman, "Reinforcement learning with non-markovian rewards," 2019.

[33] M. Agarwal and V. Aggarwal, "Reinforcement learning for joint optimization of multiple rewards," 2021.

[34] J. Perez and T. Silander, "Non-markovian control with gated end-to-end memory policy networks," 2017.

[35] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018.