REINFORCEMENT LEARNING FOR CONTROL OF NON-MARKOVIAN CELLULAR POPULATION DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Many organisms and cell types, from bacteria to cancer cells, exhibit a remarkable ability to adapt to fluctuating environments. Additionally, cells can leverage memory of past environments to better survive previously-encountered stressors. From a control perspective, this adaptability poses significant challenges in driving cell populations toward extinction, and is thus an open question with great clinical significance. In this work, we focus on drug dosing in cell populations exhibiting phenotypic plasticity. For specific dynamical models switching between resistant and susceptible states, exact solutions are known. However, when the underlying system parameters are unknown, and for complex memory-based systems, obtaining the optimal solution is currently intractable. To address this challenge, we apply reinforcement learning (RL) to identify informed dosing strategies to control cell populations evolving under novel non-Markovian dynamics. We find that model-free deep RL is able to recover exact solutions and control cell populations even in the presence of long-range temporal dynamics. To further test our approach in more realistic settings, we demonstrate performant RL-based control strategies in environments with dynamic memory strength.

1 INTRODUCTION

In order to survive, organisms must adapt to unpredictable environmental stressors occurring over diverse timescales. As a result, biological systems display remarkable adaptive capabilities, making it exceptionally challenging to control them through environmental modulation, marking an open and significant question in the physics of living systems. Two examples of adaptive systems with great clinical importance are cancer cell resistance to chemotherapy Vasan et al. (2019), and bacterial resistance to antibiotics Blair et al. (2015). In both settings, the control problem of drug application is complicated by unknown model parameters and non-Markovian dynamics. Notably, the constant application of a drug does not typically result in population extinction, as drug application also drives a certain fraction of the population to alter their phenotypic state to become drug-resistant. This new resistant state can then persist even after drug removal and across cell lineages, thus encoding a memory of the stressful environment which allows populations to more quickly adapt if the drug is reapplied Harmange et al. (2023); Mathis & Ackermann (2017); Banerjee et al. (2021). Thus, it is unclear how to devise a control strategy that can effectively mitigate the population in these scenarios. We will address this open problem in the present work.

Previous literature Padmanabhan et al. (2017); Engelhardt (2020); Gallagher et al. (2024); Fischer & Bluethgen (2024) has studied temporal drug dosing protocols to slow or prevent adaptation in various cancer or bacterial models using different methods. However, these models fail to account for the variety of adaptive timescales present in real biological systems. The presence of such memory effects greatly complicates the control problem, making the study of more realistic models imperative. Thus, to study the control of such memory-based adaptive systems, in this work we introduce a novel population model exhibiting phenotypic plasticity with non-Markovian dynamics. Prior work has shown that learning-based methods are able to form the basis of patient-specific treatment protocols Miotto et al. (2016); Ahn et al. (2021). We show that insights from control theory can be coupled with deep reinforcement learning to discover treatment protocols which successfully prevent proliferation. We highlight our main contributions below:

MAIN CONTRIBUTIONS

1. We develop a novel memory-based model for phenotypic switching relevant for bacterial and cancerous population dynamics.
2. We pose the problem of optimal drug dosing in this model and provide insights on the solution from control theory.
3. We find a robust optimal policy with deep RL that is independent of memory strength, requiring only clinically-accessible observations.

In the following sections, we first introduce the new model, derive bang-bang optimal control, and experimentally show it is required to find a high-performing policy. Because of the non-Markovian dynamics, we introduce a short history to the agent's state which we implement via framestacking. Using ideas from optimal control theory and deep reinforcement learning, we thus simplify the original control problem in a well-principled manner. We then apply standard and more advanced RL techniques to solve different versions of the posed problem.

2 PROPOSED MODEL AND APPROACH

In this section, we develop our novel non-Markovian model for cellular dynamics. These dynamics are based on a simple but general switching model (Figure 1) with switching rates dependent on drug concentration. We then introduce a physically motivated non-linear memory kernel, before discussing general approaches to the optimal control problem.

2.1 NON-MARKOVIAN PHENOTYPIC SWITCHING MODEL

To model the treatment response of an adaptive cell population, we use a general phenotypic switching model which captures the time evolution of a susceptible subpopulation, with size $S(t)$, and a resistant subpopulation, with size $R(t)$. Phenotypic switching models have been successful in describing a wide variety of biological scenarios, including development of persister cells and resistant cells in bacteria, and development of drug resistance in cancer cells Balaban et al. (2004); Fischer & Bluethgen (2024); Witzany et al. (2023); Kumar et al. (2019); Kratz & Banerjee (2024). In our model, susceptible cells with a net growth rate $\kappa_S(u)$ ($\kappa_S(u) < 0$ corresponds to net cell death) switch to a resistant state at a rate $\alpha(u)$, where u is the drug concentration. Similarly, when the drug is removed, resistant cells with a net growth rate $\kappa_R(u)$ switch back to the susceptible state at a concentration-dependent rate $\delta(u)$ (Fig. 1). All growth and switching rates are a function of drug concentration normalized by the maximum allowable dose, thus $u \in [0, 1]$. The time evolution of the size of each subpopulation, $\mathbf{x}(t) = [S(t), R(t)]^T$, is then given by the dynamical system:

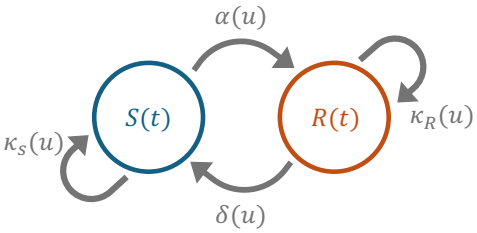

Figure 1: Depiction of the deterministic phenotypic switching model. The susceptible subpopulation S transitions to the resistant state R at a concentration-dependent rate, δ. Similarly, the resistant state switches back to a susceptible state at a rate α. The subpopulations each have a concentration-dependent growth or death rate, κ.

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), u(t)) = \mathbf{A}(u(t))\mathbf{x}(t) \tag{1}$$

where the state-transition matrix is given by:

$$\mathbf{A}(u) = \begin{bmatrix} \kappa_S(u) - \alpha(u) & \delta(u) \\ \alpha(u) & \kappa_R(u) - \delta(u) \end{bmatrix} . \tag{2}$$

To relate the net growth rate to drug concentration for each subpopulation, we assume that both rates take the general form:

$$\kappa_S(u) = \kappa_S^{\max} - (\kappa_S^{\max} - \kappa_S^{\min})g(u) \ , \quad \kappa_R(u) = \kappa_R^{\min} + (\kappa_R^{\max} - \kappa_R^{\min})h(u) \ ,$$

where $g(u), h(u) \in [0,1]$ is a monotonic dose-response function which relates drug dose to net growth rate, $\kappa_S^{\max} > 0$ denotes the maximum growth rate of the susceptible subpopulation in the absence of drug application, and where $\kappa_S^{\min} < 0$ is the maximum death rate of the susceptible subpopulation caused by application of the maximum drug dose ($u = 1$). Importantly, $\kappa_R^{\max} > 0$ denotes the maximum growth rate of the resistant subpopulation, which occurs in the *presence* of the maximum drug dose, and $\kappa_R^{\min} < 0$ corresponds to the maximum death rate of the resistant subpopulation, which occurs when the drug is removed. This parametrization corresponds to "drug addiction" behavior in the resistant subpopulation, a robust phenomenon which has been observed not only in cell culture Suda et al. (2012); Sun et al. (2014); Moriceau et al. (2015), but in animal models Das Thakur et al. (2013) and *in vivo* Seifert et al. (2016); Dooley et al. (2016). Similarly, $\delta(u)$ decreases with u while $\alpha(u)$ increases with u, as drug application drives the population to become more resistant, while reduction in drug concentration causes the system to recover susceptibility. These switching rates can then be defined as:

$$\alpha(u) = \alpha_{\max} j(u) \;\text{ and }\; \delta(u) = \delta_{\max}(1 - k(u)) \ ,$$

where $j(u), k(u) \in [0,1]$ and where $\alpha_{\max}, \delta_{\max} > 0$ denote the phenotypic switching rates of cells switching from the susceptible state to the resistant state, and vice versa.

Recently, cell populations of many types, including human cancer cell lines, yeast, and bacteria, have been shown to maintain a memory of past environments which facilitates adaptation to previously-seen stressors over many timescales Shaffer et al. (2020); Harmange et al. (2023); Larkin et al. (2024); Wolf et al. (2008); Mathis & Ackermann (2017). To better capture this memory dependence on treatment response, we introduce a memory kernel into the previously-described dynamics (equation 1), making them non-local in time. Specifically, we choose a fractional differential equation (FDE) formulation as a phenomenological way to introduce multiple timescales of adaptation, one which has been used successfully to model memory effects in other biological Lundstrom et al. (2008), ecological Khalighi et al. (2022), and physical contexts Bonfanti et al. (2020). With this addition the dynamics now become:

$$\dot{\mathbf{x}}(t) = \boldsymbol{F}(\mathbf{x}(t), u(t)) = \int_0^t \frac{(t-\tau)^{\mu-2}}{|\Gamma(\mu-1)|} \mathbf{f}(\mathbf{x}(\tau), u(\tau)) d\tau \ , \tag{3}$$

where $\Gamma(\cdot)$ denotes the Gamma function, $\mathbf{f}(\cdot)$ is given by equation 1, and here we introduce the parameter $\mu \in (0,1]$ which controls memory strength. A value of $\mu = 1$ corresponds to the memoryless case (first order derivative), whereas smaller values of μ correspond to an increased influence of past states on the current dynamics (lower order fractional derivative).

We seek to obtain a temporal drug concentration protocol $u(t)$ which minimizes the growth of a population and thus choose the final cost as $C := \log N(T)/N(0)$ over the time interval T, where $N(t) = S(t) + R(t)$ represents the total population. Thus, we aim to solve the following control problem:

$$\min_u C(\mathbf{x}(0), \mathbf{x}(T; u(\cdot))) \;\text{ subject to }\; \dot{\mathbf{x}}(t) = \boldsymbol{F}(\mathbf{x}(t), u(t)),\; \mathbf{x}(0) = \mathbf{x}_0,\; u(t) \in [0,1] \ , \tag{4}$$

where $\mathbf{x}(T; u(\cdot))$ denotes the state of $\mathbf{x}$ at terminal time T subject to control u from $0 \leq t \leq T$. Interestingly, in our minimal model (equation 3), as long as $\kappa_S(u)$, $\kappa_R(u)$, $\alpha(u)$, and $\delta(u)$ are monotonic functions, we can show that the optimal control solution follows "bang-bang" control, regardless of model parameters and memory strength.

2.2 Derivation of bang-bang control

Using equation 1 and the fact that $N(t) = S(t) + R(t)$, the model dynamics can be rewritten as a single fractional differential equation in terms of the fraction of resistant cells, $\phi := R/N$, namely:

$$D_0^\mu \phi(t) = f(\phi, u) = (\kappa_S(u) - \kappa_R(u))\phi^2 + (\kappa_R(u) - \kappa_S(u) - \delta(u) - \alpha(u))\phi + \alpha(u) \ , \tag{5}$$

where D_0^μ denotes the Caputo fractional derivative of order μ starting at $t = 0$ Garrappa (2018), and here we drop the explicit time dependence for notational clarity. This can equivalently be written as the continuous delay differential equation (we use this form in equation 3):

$$\dot{\phi}(t) = \int_0^t \frac{(t-\tau)^{\mu-2}}{|\Gamma(\mu-1)|} f(\phi(\tau), u(\tau)) d\tau \ . \tag{6}$$

Given these definitions, the Hamiltonian associated with this control problem is then

$$H(\phi(t), u(t), \lambda(t)) = \lambda(t) f(\phi(t), u(t)) \ , \tag{7}$$

where the trajectory of the Lagrangian multiplier $\lambda(t)$ is the solution to the costate equation Gomoyunov (2023):

$$\lambda(t) = -\frac{\partial_\phi C(\mathbf{x}(0), \mathbf{x}(T; u(\cdot)))}{\Gamma(\mu)(T-t)^{1-\mu}} + \frac{1}{\Gamma(\mu)} \int_t^T \frac{\partial_\phi \lambda(\tau) f(\phi(\tau), u(\tau))}{(\tau - t)^{1-\mu}} \mathrm{d}\tau \ . \tag{8}$$

Applying Pontryagin's minimum principle, we obtain the resulting inequality

$$H(\phi^*(t), u^*(t), \lambda^*(t)) \leq H(\phi^*(t), u(t), \lambda^*(t)) \ , \tag{9}$$

which along with equation 5 and equation 7 can be used to obtain the optimal control policy:

$$u^* = \arg\min_u \lambda^* [\Delta_S(\phi^* - \phi^{*2}) g(u) + \Delta_R(\phi^* - \phi^{*2}) h(u) + \delta_{\max} \phi^* k(u) + \alpha_{\max}(1 - \phi^*) j(u)] \ ,$$

where the following shorthand notation is introduced: $\Delta_S = \kappa_S^{\max} - \kappa_S^{\min}$ and $\Delta_R = \kappa_R^{\max} - \kappa_R^{\min}$. Critically, the fact that $\kappa_S^{\max}, \kappa_R^{\max}, \alpha_{\max}, \delta_{\max} > 0$, $\kappa_S^{\min}, \kappa_R^{\min} < 0$, and $g,\ h,\ j,\ k,\ \phi \in [0, 1]$ ensures that the bracketed sum which multiplies λ^* is positive for $u > 0$. As a result, as long as $g(u),\ h(u),\ j(u)$, and $k(u)$ are monotonically increasing functions of u, then the resulting optimal control is said to be "bang-bang", where $u^*(t)$ only takes on its extreme values. Furthermore, the switching times between the maximum and minimum values of u are determined by $\lambda^*(t)$, yielding the optimal control solution:

$$\begin{aligned} &u^*(t) = 0 \ \text{ if } \ \lambda^*(t) > 0, \\ &u^*(t) = 1 \ \text{ if } \ \lambda^*(t)) < 0, \ \text{ and} \\ &u^*(t) \in [0, 1] \ \text{ otherwise.} \end{aligned}$$

In principle, the optimal control trajectory can be obtained through numerical integration of the model dynamics (equation 5) along with the corresponding costate equation (equation 8). In the context of bang-bang control on non-Markovian systems however, using this approach can be difficult, as it requires careful choice of integration technique and update rule. Thus, we turn to reinforcement learning.

As shown above, the optimal control solution to Problem 4 follows bang-bang control, regardless of model parameters. This allows the continuous model of equation 1 to be simplified to a discrete model with binary controls without altering the optimal solution. This yields:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), u(t)) = \begin{cases} \mathbf{T}\mathbf{x}(t) & \text{for} \quad u = 1 \text{ (Treatment Phase)} \\ \mathbf{P}\mathbf{x}(t) & \text{for} \quad u = 0 \text{ (Pause Phase)} \end{cases} \ , \tag{10}$$

where now there are two state-transition matrices, given by:

$$\mathbf{T} = \begin{bmatrix} \kappa_S^{\min} - \alpha_{\max} & 0 \\ \alpha_{\max} & \kappa_R^{\min} \end{bmatrix}, \qquad \mathbf{P} = \begin{bmatrix} \kappa_S^{\max} & \delta_{\max} \\ 0 & \kappa_R^{\min} - \delta_{\max} \end{bmatrix} \ . \tag{11}$$

We use this formulation of the environment when training the reinforcement learning agent.

2.3 Control Strategies

Despite this simplification in action space, model control remains difficult, as the number of times and duration of drug application must be optimized. Constant application of the drug at the maximum dose ($u = 1$) results in cell adaptation and proliferation (Fig. 3): a highly suboptimal solution to Problem 4 and a catastrophic result in the clinical context.

Previous work Fischer & Bluethgen (2024) has shown that in the memoryless case ($\mu = 1$), the optimal solution for this type of model requires an initial drug application phase, followed by pulsing between treatment and pause phases at a regular interval dependent on the model parameters. However, as seen in Fig. 3, we find that the addition of memory ($\mu < 1$) renders the control strategy of the memoryless case ineffective, as cells which have previously encountered treatment switch to

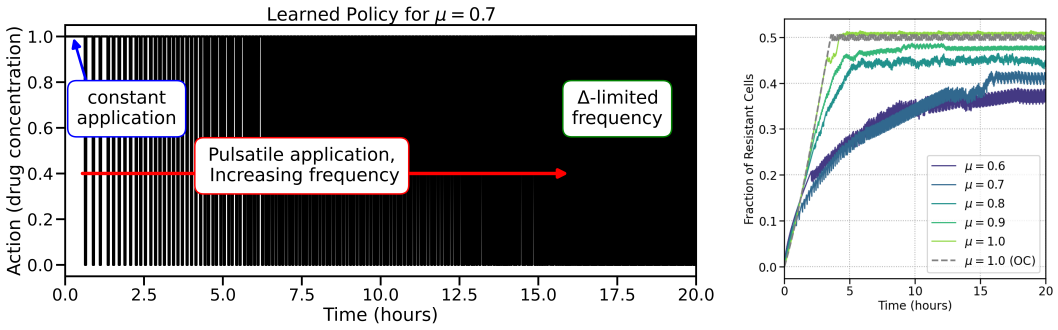

Figure 2: **Left**: The learned policy shows a resemblance to the optimal memoryless strategy, with an initial constant application phase followed by a pulsatile phase. However, in the case of memory-based dynamics, the frequency of pulsing must be increased over time as discussed in Sec. 4. Since the policy is eventually limited by the simulation time, the pulsing frequency becomes bottlenecked by our choice of time discretization Δ after $\approx$ 20 hours. Despite this, the policy is still able to perform well with rapid pulsing. **Right**: Effect of learned policy on resistant fraction. for different memory strengths. The RL agent finds (for distinct μ values) appropriate lower and upper bounds for the fraction of resistant cells. Maintaining the subpopulation in this range ensures the population can be controlled.

the resistant state faster upon subsequent applications. Furthermore, in the clinical or experimental setting, obtaining the values which parameterize equation 3 is usually not feasible, and one can only rely on direct and more macroscopic measurements from the cell populations. Thus, obtaining the appropriate switching frequencies through direct computation via optimal control (OC) theory becomes impossible. One may propose to learn these values directly before feeding them into an OC solution, but this may be a brittle pipeline, especially as the model discussed may not fully describe the underlying biological dynamics. As a result, we seek to learn the optimal policy end-to-end, directly through experience using deep reinforcement learning (Sec. 3).

2.4 Obtaining the optimal solution in the memoryless case

Previous work Fischer & Bluethgen (2024) has shown that the optimal pulsing protocol for the memoryless case requires an initial drug application phase until the resistant fraction reaches some upper threshold ϕ_h, followed by a pause phase ("drug holiday") in which the resistant fraction naturally relaxes back to some lower bound ϕ_l. This is followed by similarly repeated cycles of treatment and pause phases where the switching time occurs when the resistant fraction reaches ϕ_h and ϕ_l, respectively. To obtain the optimal values of ϕ_h and ϕ_l for our specific model parameterization, we swept over values of ϕ_h and ϕ_l between 0.1 and 0.9, with increments of 0.011, selecting the values which yielded the highest net death rate: $\phi_l = 0.494$ and $\phi_h = 0.505$.

To compare as baselines against our learned policy in non-Markovian environments, we repeated the same procedure for different values of μ. The optimal switching fractions are given in Table 1.

3 Reinforcement Learning

Reinforcement learning (RL) allows an agent to learn an optimal control strategy (policy) directly from the experience it collects. This data may be from simulation Akkaya et al. (2019), or from real-world data Haarnoja et al. (2024), or even from fully offline datasets Levine et al. (2020). In RL, the interactive agent receives observations from the environment. As a result of this input, the agent outputs a control or action (denoted u throughout), yielding a transition to a new state and a scalar reward (negative cost). The goal of the agent, analogous to OC, is to act in a way that maximizes the expected long-term accumulated rewards. In our problem formulation, this will correspond to the use of drug to minimize the total population count.

We consider the online setting in simulation, where a drug protocol is learned through experience (Fig. 2, left) by continually improving a policy as it is learned. Despite the non-Markovian dynamics

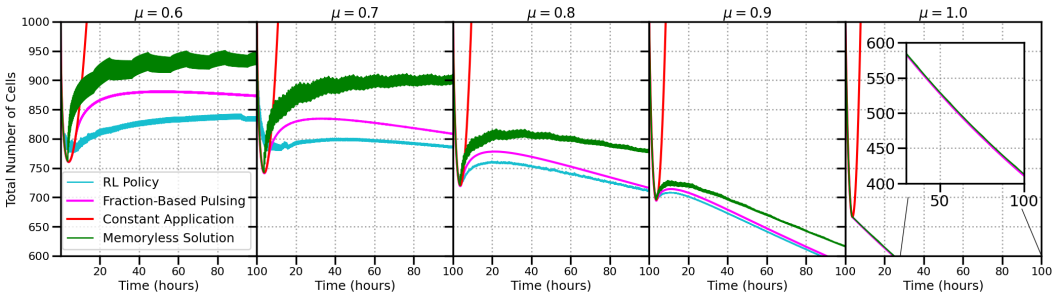

Figure 3: Performance comparison of constant drug application, solution for the memoryless case, resistant fraction-based pulsing technique, and policy learned by RL. For the fraction-based policy, an optimal lower and upper bound for resistant fractions are found through sweeping (Appendix 2.4). The RL policy is capable of controlling the cell population better than any other scheme.

and even without access to the underlying environment-specific model parameters, we show that RL is capable of providing high-performing policies. To formulate the RL problem, we define the relevant characteristics as follows (in the following, let Δ denote the simulation time between actions):

μ	ϕ_l	ϕ_h
1.0	0.494	0.505
0.9	0.483	0.494
0.8	0.466	0.477
0.7	0.448	0.459
0.6	0.428	0.439

Table 1: Optimal switching fraction values for the baseline policy for each memory strength μ.

State: The state (s_t) of the agent's environment is a list of the last $K = 5$ estimates of the instantaneous growth rate and one-hot encoded actions, $s_t = [c_{t-K,..t}, a_{t-K,..t}]$ where $c_t = \Delta^{-1} \log N_t/N_{t-\Delta}$ is the cost defined above. Thus, a history of past observations is encoded in the state vector, a crucial design choice if the agent is to learn control without a recurrent hidden state. **Action**: As motivated in Section 2.1, we choose a binary action space $u \in \{0, 1\}$ representing whether the drug is applied or not, as this is suitable to recover optimal control. **Reward**: As we seek to solve Problem 4, the reward is simply the negative growth rate, $r_t = -c_t$. Notice that with this choice of reward function, the sum of rewards across a trajectory simplifies as $R_{0:T} = \Delta^{-1} \log N_0/N_T$, ensuring the agent's objective is aligned with a reduction in total cell population. **Dynamics:** The dynamics of the total cell population N_t are governed by the dynamical system described in 2.1, initialized to be fully susceptible ($\mathbf{x}_0 = [1000, 0]$). After an action is executed, the simulation (a numerical solution [1] of equation 3) is computed with the action (dose) fixed for $\Delta = 0.01$ hours to compute the next state ($s_{t+\Delta}$). To encourage the agent to reduce the cell population while decreasing computational runtime and allowing for exploration, we terminate the episode if the number of cells ever exceeds 150% of its initial amount. The goal in RL is to learn the policy π^* which maximizes the expected cumulative discounted sum of rewards across a trajectory $\tau = (s_1, a_1, s_2, a_2, \dots)$:

$$\pi^*(a|s) = \arg\max_{\pi} \mathbb{E}_{\tau \sim p, \pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]. \tag{12}$$

To solve this optimization problem, we consider discrete-action value-based methods. Within this framework, the optimal action-value function $Q^*(s, a)$ is learned based on off-policy data observed during exploration. The value function satisfies the Bellman optimality equation (written to match the control notation introduced above for actions):

$$Q^*(s_t, u_t; \theta) = r(s_t, u_t) + \gamma \max_{u'} Q^*(s_{t+\Delta}, u'; \theta) \,. \tag{13}$$

Within this class of methods, DQN Mnih et al. (2015) learns Q^* by exploring with an annealed greedy exploration strategy. After an action is taken, the experience tuple (s, a, r, s') is stored in an

[1] The numerical solution of fractional differential equations requires some care; cf. Garrappa (2018) for details.

experience replay buffer for later use. In deep RL, the action-value function is parameterized with a neural network (denoted by the trainable parameters θ), and trained via SGD by sampling mini-batches uniformly at random from the buffer. The temporal-difference loss function is defined as the Bellman residual loss – the squared difference between left and right hand sides of equation 13:

$$\mathcal{L}_\theta = \sum_{\{s,u,r,s'\}\in\mathcal{B}} \left| Q^*(s_t, u_t; \theta) - \left(r + \gamma \max_{u'} Q^*(s_{t+\Delta}, u'; \bar{\theta}) \right) \right|^2 . \tag{14}$$

As common in value-based algorithms, we use an additional target network to reduce the effect of bootstrapping during training. The target network (denoted with parameters $\bar{\theta}$) has its parameters frozen during training, and updated periodically directly by copying the online network parameters $\theta \to \bar{\theta}$. We additionally use the double DQN approach to mitigate the over-estimation bias in standard DQN Van Hasselt et al. (2016). Further details on reinforcement learning can be found e.g. in Sutton & Barto (2018); Hessel et al. (2018). We tune over several hyperparameters (whose values we list in the Appendix) to optimize performance. We implement double DQN Van Hasselt et al. (2016) (and later FQF) based on open-source code from Stable-Baselines3 Raffin et al. (2021). Each agent is trained for 3×10^5 environment steps. All code to reproduce our experimental results can be found at `https://anonymous.4open.science/r/iclr-7B12/README.md`.

4 Case Study of Fixed Memory Environments

We first test DQN in the memoryless case ($\mu = 1$), for which an optimal controller is known Fischer & Bluethgen (2024). In this case, the optimal policy can be derived based only on two consecutive resistant fractions. However, recall that the RL agent is only given access to the instantaneous growth rates, not the resistant fractions. Nevertheless, the RL agent is surprisingly able to reliably recover the optimal policy with only two frames and without any access to underlying model parameters. The control strategy found by the agent involves first applying the drug and then pulsing regularly, in agreement with the OC solution, as seen in Fig. 3 (rightmost plot). Indeed, Fig. 2 (right panel) shows that the agent discovers an internal representation based on growth rates that is compatible with fraction-based switching.

With confirmation that RL can find the optimal dosing strategy in the memoryless case, we turn to the more difficult memory-based dynamics ($\mu < 1$). We do not have a solution to Problem 4 in this regime, so we compare to two baselines: The memoryless protocol and a modified version in which switching times occur when the resistant fraction reaches a threshold value. We note that these baselines can have arbitrarily small switching times, which is practically infeasible. Remarkably, we find that using a small but bounded Δ, our learned policy outperforms both of these baselines (Fig. 3). In addition, our experiments show that decreasing Δ beyond a certain threshold does not considerably increase performance (Fig. 6).

Interestingly, the learned policy reveals that the agent initially maintains constant drug application before transitioning to a memory-dependent pulsing protocol. The agent increases the frequency of pulsing throughout the trajectory until saturation at the maximum rate (Δ^{-1}) (Fig. 2). For cases with memory, this increase in dosing frequency can be understood as the result of faster cellular adaptation back to the resistant state after multiple drug encounters. Thus, to maintain a susceptible population, the policy's pulse frequency must continuously be increased to compensate for the memory-based adaptability. We also find that the agent maintains a lower average resistant fraction at higher memory strengths (Fig. 2, right). This qualitative trend is in agreement with the optimal switching fractions found by optimal control.

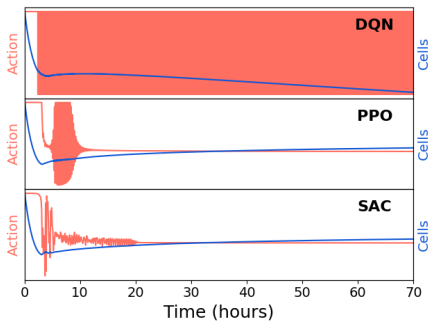

Figure 4: PPO and SAC fail to find a bang-bang control policy and have a lower performance than DQN; highlighting the need for discrete action algorithms, as informed by optimal control.

To independently show the benefit of using discrete actions, the original problem can instead be approached with continuous action algorithms, such as PPO Schulman et al. (2017) or SAC Haarnoja et al. (2018). As shown in Fig. 4, these algorithms considerably underperform relative to DQN.

Qualitatively, we find that these continuous action algorithms successfully find the constant application period at the beginning of the episode. As the episode continues, though, both PPO and SAC attempt a short-lived bang-bang policy before defaulting to the average action for the remainder of the episode. These results were obtained from an equal-compute hyperparameter sweep in a fixed $\mu = 0.9$ environment. We conjecture that continuous action algorithms struggle because of the saturating gradient effect at the limits of action space, which correspond to bang-bang control. Additionally, continuous action algorithms that give bonuses for policy entropy (e.g. SAC) make learning these deterministic policies even more difficult.

5 Model successfully generalizes to memory and observation noise perturbations

In the previous sections, we construct RL agents that are each trained on specific dynamics determined by a fixed μ value. However, this can be problematic in practice, as the memory strength μ is difficult to measure (it would require fitting equation 5 to entire episodes of data), which could correspond to adverse costs in the clinical setting. Thus, a useful policy should be able to interact immediately without requiring additional samples, and with no access to the value of μ. The unknown value of μ thus represents "side information" that is not given to the agent but provides a *context* for the agent to understand the current environment. This places the new problem setting in the framework of contextual MDPs Hallak et al. (2015); Sodhani et al. (2022). To approach this problem while maintaining clinical applicability, we train a general agent over a range of memory strengths. This approach of "domain randomization" (DR) has been applied with great success in other RL problems Tobin et al. (2017); Akkaya et al. (2019).

To tackle this more challenging environment, we use FQF Yang et al. (2019), a recent algorithm for distributional RL, where the distribution (rather than only the mean) of future returns is learned. FQF and its variants have shown success in discrete control environments. In particular, we use a noisy variant that combines the approach of NoisyNets Fortunato et al. (2018) for improved exploration. Further, we find it useful to supplement the RL state with the one-hot encoded actions used in the last K steps. This allows the agent to observe the response of the environment to the presence of drug: At lower μ, application of drug has a drastically different effect than at higher μ values at similar growth rates. With these changes, we sweep over hyperparameters (cf. Appendix) to find a model capable of reducing cell populations in any memory strength.

Although the μ-specific models outperform the generalist agent in their own finetuned environments, the generalist agent can achieve good performance across a range of memory strengths 5. It is worth noting that this model now enables use in settings where memory strength is entirely unknown and possibly dynamical or noisy in nature. Experimentally, we find this agent is very robust to perturbations of various types: namely large changes in μ and large amounts of observation noise. In both cases, the learned policy employed by the agent is able to successfully adapt and continue to minimize population growth 5.

6 Related Work

The use of reinforcement learning in control of biological systems, especially in dosage control, has been investigated by others. We will briefly highlight the most relevant prior work in this section. (Padmanabhan et al., 2017) studies a deterministic memory-less model of cancer dynamics with additional "immune" and "normal" cell types included. The number of tumor cells is used as a discrete RL state, and hence a tabular method is used. In Engelhardt (2020), a coupled SDE governs dynamics of d-many phenotypic subpopulations and the RL state is composed of measurements of individual phenotype populations. Although the specific dynamical model varies, prior work often requires unrealistic inputs to define the RL observation. An important contribution of the current work is in advancing the model class itself while maintaining clinical relevance by using easily measured quantities and providing a policy that generalizes across memory strengths.

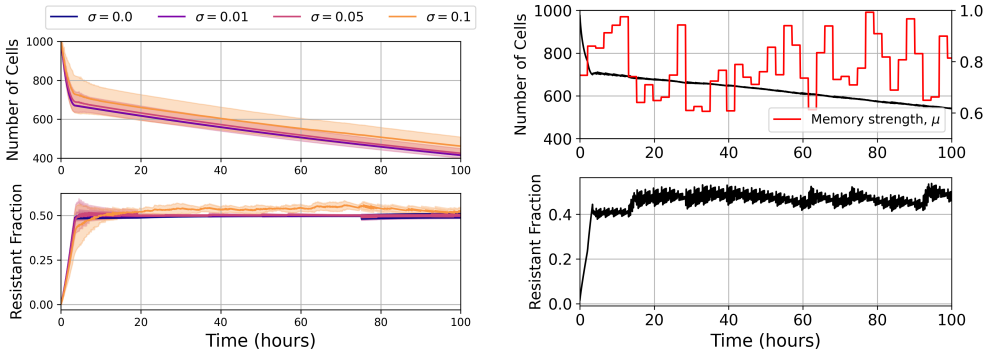

Figure 5: Left: Policy is robust to observation noise, as tested in memoryless environment. Noise draw from a normal distribution with standard deviation σ was added to each state before given to the agent. Despite large amounts of observation noise, the agent was able to drive population reduction and maintain a similar resistant fraction. Each trace represents the mean over 10 trajectories, with the standard error represented by the shaded region. Right: Learned policy is general and robust to changes in memory strength. Every 20 decision steps the memory strength is reset to a new value, draw from a uniform distribution from $[0.6, 1]$. Agent is able to quickly adapt to mitigate population growth in the new environment.

7 Future Work

We utilize frame-stacking Mnih et al. (2015) to encapsulate the history of the agent's trajectory, but in future work we will test the use of recurrent policies to capture more nuanced long-term effects and perhaps further improve performance. We also plan to extend our deterministic framework to the stochastic setting, as population heterogeneity is known to further complicate the control process. This work demonstrates the benefits of combining OC and RL, showing how their frameworks can be effectively integrated. This connection can be further developed by using the memoryless OC solution to enhance RL training through reward shaping Ng et al. (1999); Wiewiora (2003); Adamczyk et al. (2023) or pre-training Uchendu et al. (2023) techniques.

8 Discussion

In this work, we study the control of a highly non-Markovian model of adaptive cellular growth dynamics using deep reinforcement learning. Although we focus here on a specific parameterization most relevant to cancer, we expect this methodology to be applied successfully to other scenarios, including resistance development in bacteria. We find that deep RL is capable of recovering the known optimal policy for the memoryless case and can successfully find a policy for memory-based systems which prevents proliferation, all without access to the underlying model parameters. Furthermore, we obtain a policy that is robust to observation noise and memory strength perturbations.

Exact methods from OC require direct access to the population's resistant fraction and underlying model parameters to determine switching rates. In practice, experimentally measuring the growth rate of a pathogenic population is simpler than determining its corresponding resistant fraction. Surprisingly, we find that model-free RL can provide successful policies directly from the growth rate itself, making it a promising method for use in clinical settings.

References

Jacob Adamczyk, Argenis Arriojas, Stas Tiomkin, and Rahul V Kulkarni. Utilizing prior solutions for reward shaping and composition in entropy-regularized reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6658–6665, 2023.

Sang Hee Ahn, EunSook Kim, Chankyu Kim, Wonjoong Cheon, Myeongsoo Kim, Se Byeong Lee, Young Kyung Lim, Haksoo Kim, Dongho Shin, Dae Yong Kim, et al. Deep learning method for

prediction of patient-specific dose distribution in breast cancer. *Radiation Oncology*, 16:1–13, 2021.

Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.

Nathalie Q. Balaban, Jack Merrin, Remy Chait, Lukasz Kowalik, and Stanislas Leibler. Bacterial persistence as a phenotypic switch. *Science*, 305(5690):1622–1625, 9 2004. ISSN 00368075.

Shiladitya Banerjee, Klevin Lo, Nikola Ojkic, Roisin Stephens, Norbert F. Scherer, and Aaron R. Dinner. Mechanical feedback promotes bacterial adaptation to antibiotics. *Nature Physics*, 17(3): 403–409, 1 2021. ISSN 1745-2481.

Jessica M.A. Blair, Mark A. Webber, Alison J. Baylay, David O. Ogbolu, and Laura J.V. Piddock. Molecular mechanisms of antibiotic resistance. *Nature Reviews Microbiology*, 13(1):42–51, 2015. ISSN 17401534.

Alessandra Bonfanti, Jonathan Louis Kaplan, Guillaume Charras, and Alexandre Kabla. Fractional viscoelastic models for power-law materials. *Soft Matter*, 16(26):6002–6020, 2020.

Meghna Das Thakur, Fernando Salangsang, Allison S. Landman, William R. Sellers, Nancy K. Pryer, Mitchell P. Levesque, Reinhard Dummer, Martin McMahon, and Darrin D. Stuart. Modelling vemurafenib resistance in melanoma reveals a strategy to forestall drug resistance. *Nature*, 494(7436):251–255, 1 2013. ISSN 1476-4687.

Andrew J Dooley, Avinash Gupta, and Mark R Middleton. Ongoing Response in BRAF V600E-Mutant Melanoma After Cessation of Intermittent Vemurafenib Therapy: A Case Report. *Targeted Oncology*, 11:557–563, 2016.

Dalit Engelhardt. Dynamic control of stochastic evolution: a deep reinforcement learning approach to adaptively targeting emergent drug resistance. *Journal of Machine Learning Research*, 21 (203):1–30, 2020.

Matthias M Fischer and Nils Bluethgen. On minimising tumoural growth under treatment resistance. *Journal of Theoretical Biology*, 579:111716, 2024.

Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Matteo Hessel, Ian Osband, Alex Graves, Volodymyr Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, Charles Blundell, and Shane Legg. Noisy networks for exploration. In *International Conference on Learning Representations*, 2018.

Kit Gallagher, Maximilian A.R. Strobl, Derek S. Park, Fabian C. Spoendlin, Robert A. Gatenby, Philip K. Maini, and Alexander R.A. Anderson. Mathematical Model-Driven Deep Learning Enables Personalized Adaptive Therapy. *Cancer Research*, 84(11):1929–1941, 2024. ISSN 15387445.

Roberto Garrappa. Numerical solution of fractional differential equations: A survey and a software tutorial. *Mathematics*, 6(2):16, 2018.

M. I. Gomoyunov. On the Relationship Between the Pontryagin Maximum Principle and the Hamilton–Jacobi–Bellman Equation in Optimal Control Problems for Fractional-Order Systems. *Differential Equations*, 59(11):1520–1526, 2023. ISSN 16083083.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

Tuomas Haarnoja, Ben Moran, Guy Lever, Sandy H Huang, Dhruva Tirumala, Jan Humplik, Markus Wulfmeier, Saran Tunyasuvunakool, Noah Y Siegel, Roland Hafner, et al. Learning agile soccer skills for a bipedal robot with deep reinforcement learning. *Science Robotics*, 9(89):eadi8022, 2024.

Assaf Hallak, Dotan Di Castro, and Shie Mannor. Contextual markov decision processes. *arXiv preprint arXiv:1502.02259*, 2015.

Guillaume Harmange, Raúl A Reyes Hueros, Dylan L Schaff, Benjamin Emert, Michael Saint-Antoine, Laura C Kim, Zijian Niu, Shivani Nellore, Mitchell E Fane, Gretchen M Alicea, Ashani T Weeraratna, M Celeste Simon, Abhyudai Singh, and Sydney M Shaffer. Disrupting cellular memory to overcome drug resistance. *Nature Communications*, 14, 2023.

Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Moein Khalighi, Guilhem Sommeria-Klein, Didier Gonze, Karoline Faust, and Leo Lahti. Quantifying the impact of ecological memory on the dynamics of interacting communities. *PLoS Computational Biology*, 18(6):1–21, 2022. ISSN 15537358.

Josiah C Kratz and Shiladitya Banerjee. Gene Expression Tradeoffs Determine Bacterial Survival and Adaptation to Antibiotic Stress. *PRX Life*, 2:1–15, 2024.

Niraj Kumar, Gwendolyn M Cramer, Seyed Alireza Zamani Dahaj, Bala Sundaram, Jonathan P Celli, and Rahul V Kulkarni. Stochastic modeling of phenotypic switching and chemoresistance in cancer cell populations. *Scientific reports*, 9(1):10845, 2019.

Ajay Larkin, Colin Kunze, Melissa Seman, Alexander Levashkevich, Justin Curran, Dionysus Morris-Evans, Sophia Lemieux, Ahmad S Khalil, and Kaushik Ragunathan. Mapping the dynamics of epigenetic adaptation in s. pombe during heterochromatin misregulation. *Developmental cell*, 59(16):2222–2238, 2024.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Brian N. Lundstrom, Matthew H. Higgs, William J. Spain, and Adrienne L. Fairhall. Fractional differentiation by neocortical pyramidal neurons. *Nature Neuroscience*, 11(11):1335–1342, 2008. ISSN 15461726.

Roland Mathis and Martin Ackermann. Asymmetric cellular memory in bacteria exposed to antibiotics. *BMC Evolutionary Biology*, 17(1):1–14, 2017. ISSN 14712148.

Riccardo Miotto, Li Li, Brian A Kidd, and Joel T Dudley. Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Scientific reports*, 6(1): 1–10, 2016.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Gatien Moriceau, Willy Hugo, Aayoung Hong, Hubing Shi, Xiangju Kong, Clarissa C. Yu, Richard C. Koya, Ahmed A. Samatar, Negar Khanlou, Jonathan Braun, Kathleen Ruchalski, Heike Seifert, James Larkin, Kimberly B. Dahlman, Douglas B. Johnson, Alain Algazi, Jeffrey A. Sosman, Antoni Ribas, and Roger S. Lo. Tunable-Combinatorial Mechanisms of Acquired Resistance Limit the Efficacy of BRAF/MEK Cotargeting but Result in Melanoma Drug Addiction. *Cancer Cell*, 27(2):240–256, 2 2015. ISSN 1535-6108.

Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *International Conference on Machine Learning*, volume 99, pp. 278–287, 1999.

Regina Padmanabhan, Nader Meskin, and Wassim M Haddad. Reinforcement learning-based control of drug dosing for cancer chemotherapy treatment. *Mathematical biosciences*, 293:11–20, 2017.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Heike Seifert, Rosalie Fisher, Juan Martin-Liberal, Kim Edmonds, Peta Hughes, Komel Khabra, Martin Gore, and James Larkin. Prognostic markers and tumour growth kinetics in melanoma patients progressing on vemurafenib. *Melanoma Research*, 26(2):138 – 144, 2016.

Sydney M. Shaffer, Benjamin L. Emert, Raúl A. Reyes Hueros, Christopher Cote, Guillaume Harmange, Dylan L. Schaff, Ann E. Sizemore, Rohit Gupte, Eduardo Torre, Abhyudai Singh, Danielle S. Bassett, and Arjun Raj. Memory Sequencing Reveals Heritable Single-Cell Gene Expression Programs Associated with Distinct Cellular Behaviors. *Cell*, 182(4):947–959, 2020. ISSN 10974172.

Shagun Sodhani, Franziska Meier, Joelle Pineau, and Amy Zhang. Block contextual mdps for continual learning. In *Learning for Dynamics and Control Conference*, pp. 608–623. PMLR, 2022.

Kenichi Suda, Kenji Tomizawa, Hirotaka Osada, Yoshihiko Maehara, Yasushi Yatabe, Yoshitaka Sekido, and Tetsuya Mitsudomi. Conversion from the "oncogene addiction" to "drug addiction" by intensive inhibition of the egfr and met in lung cancer with activating egfr mutation. *Lung Cancer*, 76(3):292–299, 2012. ISSN 0169-5002.

Chong Sun, Liqin Wang, Sidong Huang, Guus J.J.E. Heynen, Anirudh Prahallad, Caroline Robert, John Haanen, Christian Blank, Jelle Wesseling, Stefan M. Willems, Davide Zecchin, Sebastijan Hobor, Prashanth K. Bajpe, Cor Lieftink, Christina Mateus, Stephan Vagner, Wipawadee Grernrum, Ingrid Hofland, Andreas Schlicker, Lodewyk F.A. Wessels, Roderick L. Beijersbergen, Alberto Bardelli, Federica Di Nicolantonio, Alexander M.M. Eggermont, and Rene Bernards. Reversible and adaptive resistance to BRAF(V600E) inhibition in melanoma. *Nature*, 508(7494): 118–122, 3 2014. ISSN 1476-4687.

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

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 23–30. IEEE, 2017.

Ikechukwu Uchendu, Ted Xiao, Yao Lu, Banghua Zhu, Mengyuan Yan, Joséphine Simon, Matthew Bennice, Chuyuan Fu, Cong Ma, Jiantao Jiao, et al. Jump-start reinforcement learning. In *International Conference on Machine Learning*, pp. 34556–34583. PMLR, 2023.

Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Neil Vasan, José Baselga, and David M. Hyman. A view on drug resistance in cancer. *Nature*, 575 (7782):299–309, 2019. ISSN 14764687.

Eric Wiewiora. Potential-based shaping and Q-value initialization are equivalent. *Journal of Artificial Intelligence Research*, 19:205–208, 2003.

Christopher Witzany, Jens Rolff, Roland Regoes, and Claudia Igler. The pharmacokinetic-pharmacodynamic modeling framework as a tool to predict drug resistance evolution. *Microbiology*, pp. 1–37, 2023.

Denise M. Wolf, Lisa Fontaine-Bodin, Ilka Bischofs, Gavin Price, Jay Keasling, and Adam P. Arkin. Memory in microbes: Quantifying history-dependent behavior in a bacterium. *PLoS ONE*, 3(2), 2008. ISSN 19326203.

Derek Yang, Li Zhao, Zichuan Lin, Tao Qin, Jiang Bian, and Tie-Yan Liu. Fully parameterized quantile function for distributional reinforcement learning. *Advances in neural information processing systems*, 32, 2019.

A Reinforcement Learning Details

We adapted DQN from Stable-Baselines3 Raffin et al. (2021) with a Double DQN action selection rule Van Hasselt et al. (2016). We train the RL agent for 3×10^5 total environment steps. We limited the episodes to be of length 10^4 steps (corresponding to 100 hours in simulation). An MLP of fixed size (2 hidden layers with 64 dimensions each and ReLU activation) was used to parameterize the Q-function.

Regarding the environment and training, we list several key implementation choices: At the beginning of an episode, the state is zero-padded to always be of length $K = 5$. We experimented with adding a penalty for allowing the number of cells to increase beyond the initial amount upon termination, but found this was not necessary for successful training. Borrowing terminology from the literature on Atari environments Mnih et al. (2015), we stack K frames (previous cost values) to form the RL agent's state vector. Although initially we let K be a μ-dependent hyperparameter, we found a constant choice of $K = 5$ to work well across the values of μ studied.

We find exponentially decaying the exploration parameter ε to work better than the typical linear annealing (with constant, positive final ε) scheduling. We conjecture that this is because non-greedy actions can be quite detrimental (causing the environment to terminate), and exponentially decaying ε ensures some exploration continues to occur but with increasingly fewer random actions. To ensure the agent is not overly myopic (especially for such long episodes) we found a large discount factor of $\gamma = 0.999$ (corresponding to an effective horizon of $H = (1-\gamma)^{-1} = 10^3$) to be helpful. When training the agent, we wait until the completion of one rollout episode, and take as many gradient steps as environment steps have occurred.

A.1 Hyperparameters

We find that sweeping over a range of hyperparameters (as shown in Table 5) did not have a significant effect on performance, though for reproducibility we list the final hyperparameters used (for $\mu = 0.7$) below in Table 4.

Table 2: Finetuned Hyperparameter Values for Double DQN

Hyperparameter	Finetuned Value
batch size	32
buffer size	100,000
exploration rate	0.05
frames stacked	5
gradient steps (UTD / RR)	1
learning rate	3.60×10^{-4}
target update interval	$1,000$
discount factor	0.999
learning starts	$10,000$

Table 3: Hyperparameter Sweep Ranges for Double DQN

Hyperparameter	Sweep Values
batch size	16, 32, 64
exploration rate	$0.01 - 0.2$
learning rate	$10^{-5} - 10^{-3}$
target update interval	$1{,}000, 5{,}000, 10{,}000, 30{,}000$
target Polyak averaging	$0.95, 0.99, 0.995, 1.0$

Table 4: Finetuned Hyperparameter Values for (NoisyNet) FQF

Hyperparameter	Finetuned Value
batch size	8
buffer size	100,000
exploration rate	0.2
learning rate	2.7×10^{-3}
target Polyak averaging	0.005
n-step TD	1
entropy coeff.	0.003

Table 5: Hyperparameter Sweep Ranges for (NoisyNet) FQF

Hyperparameter	Sweep Values
batch size	8, 64
exploration rate	$0.02 - 0.2$
learning rate	$5 \times 10^{-4} - 5 \times 10^{-3}$
target Polyak averaging	$0.001, 0.005, 0.01$
n-step TD	$1, 2, 4, 32$
entropy coeff.	$0, 0.0005, 0.001, 0.003$

B FURTHER EXPERIMENTAL RESULTS

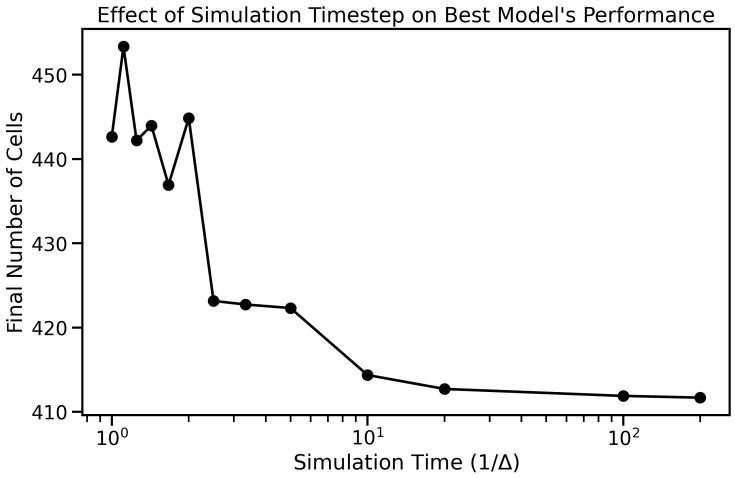

Figure 6: We find that the simulation time can have a significant effect on RL performance. For each choice of Δ, we run a random sweep of size 30 over various hyperparameters, selecting the highest-performing run for each Δ. Since smaller values of Δ require longer compute-times for simulations, there is a tradeoff between the amount of time (also, the inverse of max pulsing rate, which may be more relevant in clinical settings) and the best performance. We have chosen to use $\Delta = 0.01$ throughout.

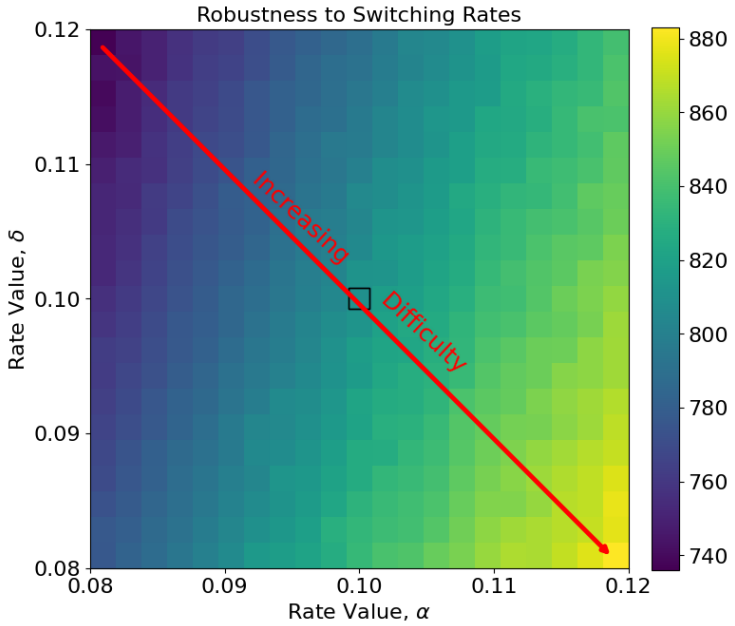

Figure 7: Robustness experiment on changing switching rates α, δ. The black box in the center highlights the rates at which the agent was trained. On each axis, the rate is varied by $\pm 20\%$, representing a large range of biologically plausible parameters. As α increases (or δ decreases), the cells are more often in the resistant state (cf. Fig. 1), making the control problem more difficult for the agent.