This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

Reinforcement Learning with Safe Exploration for Adaptive Plasma Cancer Treatment

Zichao Hou, Taeyoung Lee, and Michael Keidar

***Abstract*—Cold Atmospheric Plasma (CAP) jet is an ionized gas with a rich combination of reactive oxygen/nitrogen species, charged particles, and photons. By both *in vitro* and *in vivo* experiments, it has been demonstrated that CAP can be successfully utilized in cancer treatments. However, the therapeutic effects of CAP depend on various factors such as plasma discharge voltage, gas composition, treatment duration, and type of cancer cells. This paper presents an adaptive plasma system, where the CAP treatment conditions are adjusted online depending on the actual cancer cell response. In particular, we utilize safe Q-learning to schedule CAP cancer treatment autonomously while prohibiting excessive treatments caused by large uncertainties. As it is capable of learning the dynamic characteristics of the particular cancer cell under treatments in situ, we can treat cancer successfully without the complete prior knowledge of cancer characteristics, where the uncertainties of the learned dynamics are carefully accounted. The efficacy of the proposed algorithm is illustrated by numerical examples with an empirical cancer dynamic model constructed from *in vitro* experiments.**

I. Introduction

Cold atmospheric plasma (CAP) jet is generated by ionization that is initialized when noble gas, such as helium and argon, pass through an electric field. In particular, CAP or non-thermal plasma jet refers to the case when the ion temperature is close to the room temperature [1]. There has been increasing interests in CAP specifically for potential application in cancer treatments. It is studied that the rich environments of reactive species, charged particles, photons, and UV, provided by CAP jet trigger cell death pathway selectively for cancer cells while leaving healthy cells unharmed. This has been illustrated by both *in vitro* and *in vivo* under various conditions [2]. The CAP jet is capable of eliminating cancer cells *in vitro* and reducing the size of tumor *in vivo* [3], [4].

However, there are several challenges remaining to achieve safe and reliable CAP cancer treatments. First, the therapeutic effectiveness of CAP treatment depends on various parameters affecting plasma generation, such as plasma discharge voltage, ionized gas composition, and gas flow rate. They are further affected by the environment, including the ambient temperature and composition of atmosphere. Next, cancer cells may exhibit different responses depending on their type or status even when exposed to the same CAP treatments. Finally, the underlying biochemical mechanism behind the interaction between living cells and CAP is not completely understood yet. As such, there is no clear guideline regarding how to schedule CAP cancer treatments. It is impractical and inefficient to develop an optimal treatment plan via exhaustive trial-and-errors.

Zichao Hou, Taeyoung Lee, and Michael Keidar are with the Department of Mechanical and Aerospace Engineering, The George Washington University, Washington DC 20052 USA (e-mail: {zichao,tylee,keidar}@gwu.edu)

This research has been supported by NSF under the grant IIP-1747760

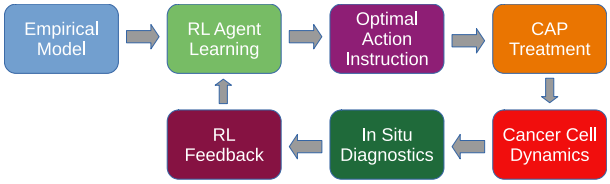


Fig. 1: Reinforcement learning (RL) adaptive CAP cancer treatment schematic

To address these, the concept of adaptive plasma has been proposed in [5]. One of the desirable feature is that the composition and the intensity of reactive species generated by CAP can be changed promptly. Therefore, it is possible to control the CAP parameters such that the therapeutic effects are customized in real-time according to the actual response of the cancer cell under treatments. This is to introduce feedback control mechanism into CAP cancer therapy so that the desired outcome of treatment can be reached even under uncertain modeling in cancer responses to CAP and potential perturbation caused by the environment.

On the one hand, in [6], [7], model predictive control (MPC) is introduced to an atmospheric pressure plasma jet (APPJ) testbed to control the plasma dose delivery where the interaction between the plasma jet and the substrate is studied. Recently, in [8], the linear parameter-varying (LPV) framework incorporated with the model predictive control is presented to provide a data-driven method of controlling the nonlinear APPJ thermal plasma dose. In [9], a learning-based stochastic model predictive control strategy is proposed for reference tracking of stochastic linear systems with additive state-dependent uncertainty, where the state-dependent uncertainty model is adjusted online to reduce plant-model mismatch of APPJ.

For CAP cancer treatments, an empirical dynamic model is constructed to represent the evolution of cancer cell viability under several treatment conditions. Next, MPC is applied to address the discrepancy between the actual cancer cell viability and the corresponding value predicted by the mathematical model [5], [10]. While it is illustrated that the presented MPC for adaptive plasma can handle a modest level of uncertainties, its performance is directly affected by the accuracy of the model, and it is not capable of adjusting the mathematical model online. In other words, there is no improvements in adaptivity or performance based on the prior experience.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

The main objective of this paper is to address such issues. More specifically, we aim to develop adaptive plasma framework that is continuously learning about the dynamic characteristics of the particular cancer cell under treatments, so that the treatment is adapted to the prior cancer cell responses. Machine learning (ML) provides advanced computational tools to recognize patterns with given data and to generalize them [11]. In particular, reinforcement learning (RL) is a goal-direct approach that allows an agent to learn how to perform a task through maximizing a numerical reward signal for a dynamic system modeled as a Markov decision process [12]. The agent can construct the knowledge of reaching the designed goal through multiple interactions with the environment, thereby eliminating the need to developing an exact mathematical model in prior.

These two features of reinforcement leaning, namely gaining performance through experience and avoiding the need for exact models, are particularly advantageous in adaptive plasma for cancer treatments, where it is infeasible to construct an accurate dynamic model from first principles and there are greater variabilities in dynamic characteristics. However, successful implementation of reinforcement learning often requires numerous training episodes, and during the learning process the treatments are planned randomly, which are appropriate neither in *in vitro* nor *in vivo* experiments for CAP cancer treatments where the safety is utmost. Due to the stochastic nature of the cell dynamics, the results of the actual CAP treatments might not be well aligned with the best prediction, thereby causing safety concerns.

To address these, we propose to synergistically integrate an empirical dynamic model with safe reinforcement learning. First, an empirical model for cancer cell response is constructed according to a set of *in vitro* experiments [13], which provides the temporal response of cancer cell viability for several CAP treatments. These data are incorporated into a Gaussian process [14] that can generalize the data beyond the particular treatment conditions chosen in the experiments while accounting the level of uncertainties. Next, CAP cancer treatment is modeled as a Markov decision process, to which safe Q-learning is applied. Initially, the action-value function in Q-learning is pre-trained with a large number of data generated by the empirical model. Later, it is updated *in situ* with the actual response in a simulated environment. This procedure is illustrated in Figure 1. This approach inherits the desirable properties of Q-learning while taking the full advantage of the prior knowledge represented by the empirical model.

Further, the issues of safety are addressed as follows. To avoid the potential harmful outcomes, reinforcement learning has been extended with various formulations of risks [15], [16]. For instance, in [17], [18] the cost function of reinforcement learning is augmented with additional risk-related terms to prevent adverse consequences. In [19], [20], a teacher-learner framework is implemented where the teacher can provide advices for potential risky situation, and the learner interacts with the actual environment while taking advices. Furthermore, [21], [22] utilize the uncertainty provided by the Gaussian process to safely explore the environment.

In this paper, the prior knowledge of the cancer dynamics constructed from the empirical model is constantly updated based on the actual response. Specifically, while updating the action-value function during the interaction with the cancer, we update the Gaussian process model with the newly acquired data so that the predicted mean of the Gaussian process model may accurately represent the actual dynamics. The desirable feature is that the confidence level of the current model is also adjusted through the framework of Gaussian process. Consequently, we can assess the risk of each planned treatment in probabilistic sense. Utilizing these, we present one of safe reinforcement learning techniques in CAP. In particular, we consider the safety in th exploration process, where risky treatments that have a higher probability of excessive outcomes are excluded. Interestingly, as more data become available through the course of treatments, the uncertainties in the model reduce. Since the risk is formulated by accounting such uncertainties, the safety concerns diminish and more aggressive treatments can be planned through the treatments. In short, the proposed approach takes the full advantage of reinforcement learning where in the adaptive CAP treatment while addressing the safety issues of the exploration process.

Our approach should be distinguished from a series of work [6], [7], [8], [9], [23], [24], [25], [26], where the MPC strategy and ML algorithms are applied to regulate various parameters of a device generating CAP, such that substrate temperature, plasma current, and power. Instead we focus on the cellular response to CAP and the control of cancer cell viabilities.

This paper is organized as follows. In Section II, an empirical model for cancer dynamics is formulated using Gaussian process, in Section III, reinforcement learning for adaptive plasma is introduced with numerical examples, and in Section IV, the safe reinforcement learning is discussed with numerical simulations, followed by conclusions.

II. Empirical Cancer Dynamics with Gaussian Process

A. In Vitro Experiments

The temporal evolution of cancer cell viability after CAP treatment has been presented in [13]. Two types of cancer cells, namely U87 (glioblastoma) and MDA-MB-231 (breast adenocarcinoma) are treated with CAP for varying conditions, and the corresponding viability is measured repeatedly over 48 hours using RealTime-Glo MT Cell Viability Assay to evaluate the effectiveness of each treatment. In particular, the plasma treatment duration is varied from 0 to 180 seconds, and the discharge voltage is changed between $3.16\,\mathrm{kV}$ and $3.71\,\mathrm{kV}$. The cell viability is measured at every 10 minutes up to the first hour, and later, it is measured at $6, 12, 24, 48$ hours.

This paper utilizes the data set for U87 with the discharge voltage of $3.16\,\mathrm{kV}$, where the cell viabilities at the above time instances are given for five treatment durations Δt in $\{0, 30, 60, 90, 180\}$ seconds.

B. Gaussian Process

Throughout this paper, we focus on controlling the plasma treatment duration to reduce the cancer cell viability to a prescribed desired level. The experimental data provide valuable

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

insight into cancer cell response to CAP for varying treatment conditions. However, to utilize it for adaptive plasma, the data should be generalized such that the cancer cell viability can be predicted for an arbitrary treatment duration that is not in the data set.

In [10], the viability is assumed to evolve according to a particular ordinary differential equation,

$$\dot{v} = vF(t, v; c),$$

where $v \in \mathbb{R}$ is the viability, $F : \mathbb{R}^{2+p}$ is a prescribed real-valued function that is dependent on the time t, the current viability v, and free parameters $c \in \mathbb{R}^p$. The parameters are chosen such that the discrepancy between the experimental data and the numerical results from the above model is minimized for each treatment duration in the data set. Then, they are linearly interpolated for a given arbitrary treatment duration. While this successfully models the experimental results in [10], its reliability, especially in generalization through interpolation of parameters, greatly depends on how the form of F is selected in a heuristic manner.

In this paper, we utilize Gaussian process to formulate a dynamic model [14]. A Gaussian process is a stochastic process, defined such that any finite number of collection is jointly Gaussian. More specifically, consider a real-valued function $g(x) : \mathbb{R}^n \to \mathbb{R}$ dependent on the input vector $x \in \mathbb{R}^n$. We do not have an analytic expression of $g(x)$. Instead, the sample values g_i of $g(x_i)$ can be measured at a set of inputs $x_i \in \{x_1, x_2, \ldots, x_N\}$ up to an additive, independent noise, as given by

$$g_i \sim g(x_i) + \epsilon_{g_i}, \tag{1}$$

with $\epsilon_{g_i} \sim \mathcal{N}(0, \sigma_{g_i}^2)$, which denotes the Gaussian distribution with the mean 0, and the covariance $\sigma_{g_i}^2$. The objective is to model $g(x)$ using the given data set $\mathcal{D} = \{(x_i, g_i, \sigma_{g_i})\}_{i \in 1,\ldots N}$.

A Gaussian process is completely described by its second-order statistics. By defining a mean function $\mathbf{m}(x) : \mathbb{R}^n \to \mathbb{R}$ and a positive-definite covariance function $\mathbf{K}(x, x') : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, which is referred to as kernel, the corresponding Gaussian process is denoted by

$$g(x) \sim \mathcal{GP}(\mathbf{m}(x), \mathbf{K}(x, x')). \tag{2}$$

For the given mean function, kernel, and data, the regression to evaluate the function g for an arbitrary input is completed as follows. Define $\mathbf{g}, \mathbf{x}$, and $\mathbf{m}(\mathbf{x}) \in \mathbb{R}^N$ be the concatenation of g_i, x_i and $\mathbf{m}(x_i)$ for $i \in \{1, \ldots, N\}$, respectively. Also, let the matrix $\mathbf{K}(\mathbf{x}, \mathbf{x}) \in \mathbb{R}^{N \times N}$ be defined such that its i, j-th element is $\mathbf{K}(x_i, x_j)$, and let $\Sigma_{\mathbf{g}} = \mathrm{diag}[\sigma_{g_1}^2, \ldots, \sigma_{g_N}^2] \in \mathbb{R}^{N \times N}$. From the definition of the Gaussian process, $\mathbf{g}$ is distributed according to

$$\mathbf{g} \sim \mathcal{N}(\mathbf{m}(\mathbf{x}), \mathbf{K}(\mathbf{x}, \mathbf{x}) + \Sigma_{\mathbf{g}}). \tag{3}$$

Let $g_* \in \mathbb{R}$ be a sample value when $x = x_*$. It is jointly Gaussian with $\mathbf{g}$ as

$$\begin{bmatrix} \mathbf{g} \\ g_* \end{bmatrix} = \mathcal{N}\left(\begin{bmatrix} \mathbf{m}(\mathbf{x}) \\ \mathbf{m}(x_*) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\mathbf{x}, \mathbf{x}) + \Sigma_{\mathbf{g}} & \mathbf{K}(\mathbf{x}, x_*) \\ \mathbf{K}(x_*, \mathbf{x}) & \mathbf{K}(x_*, x_*) \end{bmatrix} \right). \tag{4}$$

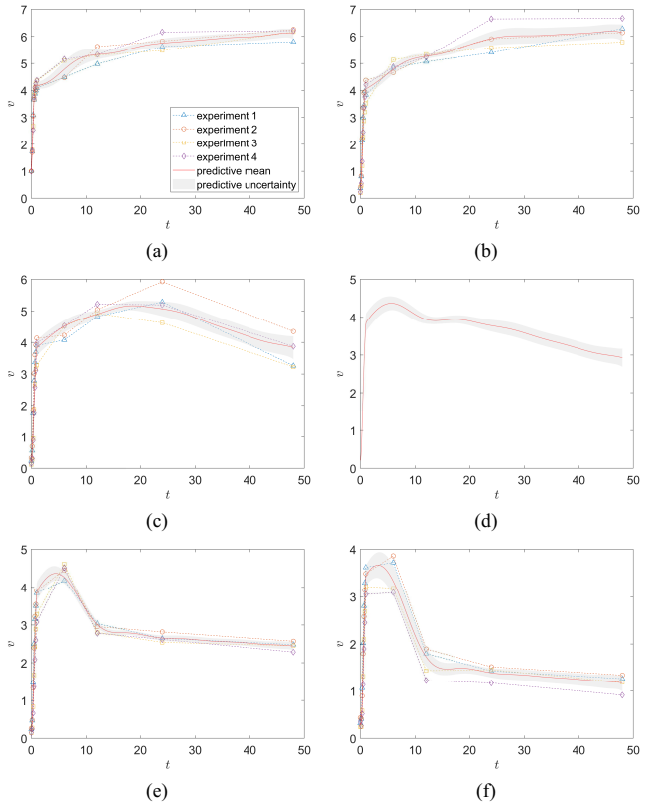


Fig. 2: Gaussian process regression of U87 for discharge voltage $U = 3.16$ kV with varying treatment durations Δt in seconds: (a) $\Delta t = 0.0$ (b) $\Delta t = 30.0$ (c) $\Delta t = 60.0$ (d) $\Delta t = 75.0$ (e) $\Delta t = 90.0$ (f) $\Delta t = 180.0$. The variable t in the horizontal axis represents the time after treatment in hours, and v in the vertical axis indicates the normalized cell viability.

Therefore, from the conditional distribution of joint Gaussian distributions, the regression equation for g_* is

$$\begin{aligned} g_* | \mathcal{D}, x_* \sim \mathcal{N}(&\mathbf{m}_* + \mathbf{K}_{*\mathbf{x}}(\mathbf{K}_{\mathbf{xx}} + \Sigma_{\mathbf{g}})^{-1}(\mathbf{g} - \mathbf{m}_{\mathbf{x}}), \\ &\mathbf{K}_{**} - \mathbf{K}_{*\mathbf{x}}(\mathbf{K}_{\mathbf{xx}} + \Sigma_{\mathbf{g}})^{-1}\mathbf{K}_{\mathbf{x}*}), \end{aligned} \tag{5}$$

where the subscripts for $\mathbf{m}$ and $\mathbf{K}$ denote the input arguments, e.g., $\mathbf{K}_{*\mathbf{x}} = \mathbf{K}(x_*, \mathbf{x}) \in \mathbb{R}^{1 \times N}$.

In short, for a given input and output data of an unknown function, its output for an arbitrary input is constructed by (5). The desirable feature is that a Gaussian process may represent an arbitrary function explicitly without the need for training or numerical optimization required for common multi-layer neural networks. The uncertainties are represented by Gaussian distributions that are provided by various properties, which can be utilized to simplify the required mathematical analysis.

C. Empirical Cancer Dynamics

We utilize a Gaussian process to formulate an empirical dynamic model, which generalizes the experimental data beyond particular treatment durations considered in [13]. First, a Gaussian process is formulated, where the input is composed of the treatment duration Δt and the time t,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

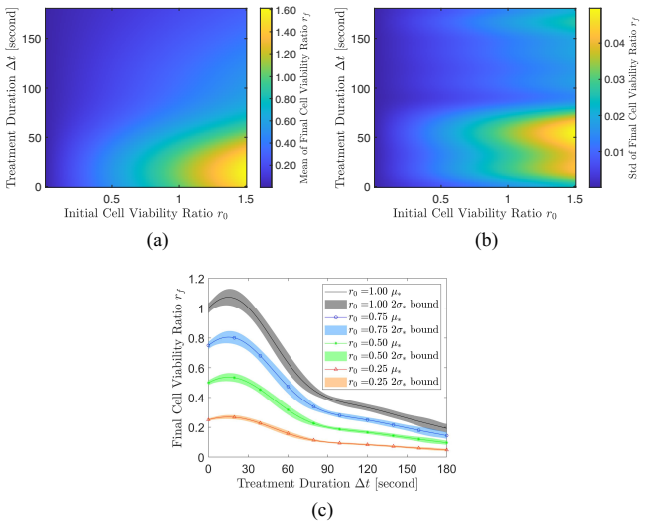


Fig. 3: Empirical Gaussian process model for the viability 48 hours after treatment with various initial cell viability r_0 and treatment duration Δt. Figure (a) and (b) provide the predicted mean values μ_* and predicted standard deviation values σ_* for all pairs of $(r_0, \Delta t)$. Figure (c) illustrates the predicted viability 48 hours after the treatment with respect to treatment duration Δt for the cases $r_0 = \{1.00, 0.75, 0.50, 0.25\}$.

and the output is the corresponding cell viability v. The data set is composed of the treatment durations $\Delta t \in \{0, 30, 60, 90, 180\}$ in seconds and the cell viability v measured at $\{0, \frac{1}{6}, \frac{2}{6}, \ldots, \frac{5}{6}, 1, 6, 12, 24, 48\}$ in hours, where four measurements are available for each Δt.

The prior mean function is $\mathbf{m}(x) = 0$, and the kernel is chosen as squared-exponential function, given by

$$\mathbf{K}(x_i, x_j) = \sigma_f^2 \exp(-\frac{1}{2}(x_i - x_j)^T M (x_i - x_j)) \quad (6)$$

where $M = \mathrm{diag}([l_1, l_2])^{-2} \in \mathbb{R}^{2\times 2}$, and $\theta = [l_1, l_2, \sigma_f] \in \mathbb{R}^3$ is the hyper-parameter vector defined by the length-scale l_1, l_2 and the variance σ_f. The hyper-parameters are estimated by the training data as follows. For the given data $\mathbf{x}$ and the hyper-parameter θ, the marginal distribution of $\mathbf{g}$ is given by $\mathbf{g} \sim \mathcal{N}(0, \mathbf{K}(\mathbf{x}, \mathbf{x}) + \Sigma_{\mathbf{g}})$, and its log likelihood is

$$\log p(\mathbf{g}; \mathbf{x}, \theta) = -\frac{n}{2}\log 2\pi - \frac{1}{2}\log\det(\mathbf{K} + \Sigma_{\mathbf{g}}) - \frac{1}{2}\mathbf{g}^T(\mathbf{K} + \Sigma_{\mathbf{g}})^{-1}\mathbf{g}. \quad (7)$$

The optimal hyper-parameters are obtained by maximizing the above log likelihood. This can be interpreted as selecting the most probable hyper-parameters for the given data set.

The corresponding cell viability predicted by the Gaussian process is illustrated in Figure 2, which includes five treatment durations in the data set along with the experimental results, and one treatment duration $\Delta t = 75$ that is not in the data set to illustrate the capability of generalization.

Next, we take the experimental results with no treatment, i.e., $\Delta t = 0.0$ as a control group, to evaluate the effectiveness of CAP treatment. We introduce the following cell viability ratio r as the ratio of cell viability v of a treatment group to the control group.

$$r(t; \Delta t) = \frac{v_{treatment}(t; \Delta t)}{v_{control}(t)}, \quad (8)$$

for $t \in [0, 48]$.

In the adaptive plasma treatment problem formulated in the next section, it is assumed that the treatment is repeated at every 48 hours. Since the initial cell viability is normalized to $v(0) = 1.0$ for all the experimental data, the initial cell viability ratio is $r_0 = r(t = 0) = 1.0$ for all experimental results. However, for multiple treatments, the initial cell viability ratio at the second or the later treatments may be less than one. We assume that the effects of the subsequent CAP treatments are identical to the first treatment such that the viability ratio is reduced in the same manner as the first treatment. More explicitly, it is assumed that for any $c > 0$,

$$r(t; \Delta t, r_0 = c) = c\, r(t; \Delta t, r_0 = 1).$$

Let $r_f \in \mathbb{R}$ be the cell viability ratio 48 hours after the treatment. Through the Gaussian process and the above generalization, now we have the following probabilistic empirical model:

$$p(r_f | r_0, \Delta t) \sim \mathcal{N}(\mu_*, \sigma_*^2), \quad (9)$$

which describes the distribution of the final cell viability ratio, for a given initial cell viability ratio and a treatment duration. This is given as a Gaussian distribution with the mean μ_* and the variance σ_* computed by (3). It is further illustrated in Figure 3.

III. Reinforcement Learning for Adaptive Plasma

In this section, CAP cancer treatment is formulated as a Markov decision process, to which reinforcement learning is applied for adaptive plasma cancer treatments. Furthermore, we consider a realistic case where the actual cancer cell response does not exactly follow the empirical model of the preceding section, and we present how the reinforcement learning mitigates such modeling errors and disturbances. Additionally, to prevent potentially risky treatments, the safe action selection is studied with Gaussian process model learning.

A. Markov Decision Process Formulation

The adaptive CAP treatment problem considered in this paper is formulated as follows. Assuming that CAP treatment is repeated at every 48 hours, and the objective is to determine the optimal treatment duration Δt such that the cancer cell viability is reduced to a prescribed desired level, namely $r_d \in \mathbb{R}$.

Since CAP cancer treatments are performed on a discrete time step, the process of treatment can be formulated into a Markov decision process (MDP). In a discrete-time MDP, at given time step t_k, an agent at the state S_k takes an action A_k so that it is transferred to another state S_{k+1} at the next time step t_{k+1} while receiving a reward R_{k+1}. The accumulated reward is referred to as a goal G_k, and the transition between states are governed by the state transition probability

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

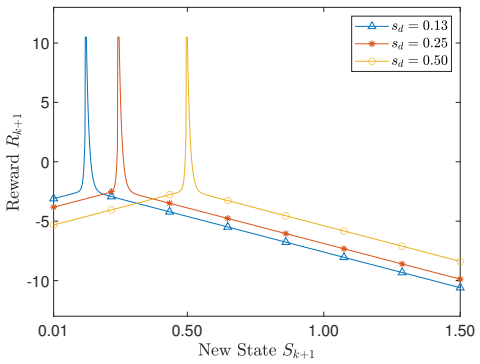


Fig. 4: Reward signals for various target states

$p(S_{k+1}|S_k, A_k)$ that describes the probability distribution of the state at the next step, given the current state and the chosen action.

For adaptive CAP treatments, the components of MDP can be chosen as follow:

- *State* S: cell viability ratio r, where the desired target r_d can be expressed as s_d
- *Action* A: treatment duration Δt for each treatment
- *Transition Probability* $p(S_{k+1}|S_k, A_k)$: constructed by $p(r_f|r_0, \Delta t)$ in (3)
- *Reward* R: it is designed such that the reward is increased as S_{k+1} is closer to s_d

More specifically, the reward is chosen as

$$R_{k+1} = R(s_d, S_{k+1}) = \begin{cases} 13 \times e^{(-\frac{|D_r|}{0.005})} - 2.5 \\ \quad - 6 \times (|D_r| - 0.02), & \text{if } D_r \in (-\infty, -0.02] \\ 13 \times e^{(-\frac{|D_r|}{0.005})} - 2.5, & \text{if } D_r \in (-0.02, -0.005] \\ 11.5, & \text{if } D_r \in (-0.005, 0] \\ 13 \times e^{(-\frac{|D_r|}{0.01})} - 2.5, & \text{if } D_r \in (0, 0.02] \\ 13 \times e^{(-\frac{|D_r|}{0.01})} - 2.5 \\ \quad - 6 \times (|D_r| - 0.02), & \text{otherwise} \end{cases}$$

where $D_r = S_{k+1} - s_d$ is the difference value between state at time step t_{k+1} and desired target state. As illustrated in Figure 4 for several values of desired target states s_d, the reward has the peak value when the state, i.e. cell viability ratio, after the current treatment, namely S_{k+1}, is equal to its desired value s_d, and it gradually decreases as the discrepancy between S_{k+1} and s_d increases.

The accumulated reward is referred to as the goal, which is defined as

$$G_k = \sum_{i=0}^{\infty} \gamma^i R_{i+k+1},$$

where $0 < \gamma < 1$ is a discount rate. While it is formulated for $i \to \infty$, here we consider a finite time execution of MDP, where the treatment stops as the cell viability ratio is sufficiently close to the target.

For a given current cell viability ratio or the state S_k, the treatment duration or the action A_k is selected by a policy

$$\pi(S_k, A_k) = p(A_k|S_k),$$

TABLE I: Procedure for Q-learning

procedure Q-LEARNING UPDATE ITERATION
  Initialize $Q(S, A)$ randomly for all (S, A) pairs
  Let $Q(S, A) = 0.0$ for $S \le s_d$
  repeat
    $S_k = S_0$ where S_0 is chosen randomly with $S_0 > s_d$
    repeat
      Choose action A_k from $Q(S_k, :)$ using ϵ-greedy method
      Obtain new state S_{k+1} and reward R_{k+1} by performing action A_k
      $Q(S_k, A_k) = Q(S_k, A_k) + \alpha[R_{k+1} + \gamma \max_a Q(S_{k+1}, a) - Q(S_k, A_k)]$
      $S_k = S_{k+1}$
    until $S_k \le s_d$
  until Episode number reaches designed maximum
end procedure

which specifies the probability distribution of the action when in the state S_k, from which the actual action is sampled. Under the presented MDP formulation of the adaptive CAP treatment, the objective is to find the optimal policy, namely π^* that maximizes the expected goal of the treatment.

B. Q-learning

In Q-learning [12], the above problem is addressed by introducing the action-value function, or the Q-function:

$$Q^\pi(S_k, A_k) = \mathbb{E}[\sum_{i=0}^{\infty} \gamma^i R_{i+k+1} | S = S_k, A = A_k],$$

which describes the expected prospective goal, when the current state is $S = S_k$ and the current action is $A = A_k$, assuming that all of the prospective actions are chosen according to the given policy π.

The core idea of Q-learning is that the action-value function for the optimal policy can be *learned* online by experiences. Suppose that at the current state S_k, an action A_k is chosen, which makes the state transferred to S_{k+1} while generating a reward R_{k+1}. From this experience, the action-value function is updated according to

$$Q(S_k, A_k) = Q(S_k, A_k) + \alpha[R_{k+1} + \gamma \max_a Q(S_{k+1}, a) - Q(S_k, A_k)], \tag{10}$$

where $\alpha > 0$ is learning rate indicating learning speed. As shown in [27], any random action-value function asymptotically converges to the optimal action-value function, denoted by Q^* through the above iteration.

Once the action-value function is optimized, the optimal policy can be formulated. For example, for the deterministic greedy method, we have

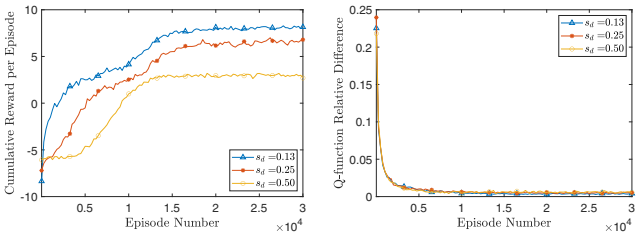
$$A_k^* = \arg\max_a Q^*(S_k, a).$$

These procedures are summarized at Table I.

To implement the Q-learning for the CAP treatment, the state space and the action space are discretized as follows:

- *State Space* $\mathcal{S}$ ranges from 0.01 to 1.50 with 150 grids
- *Action Space* $\mathcal{A}$ ranges from 0.0 to 180.0 with 121 grids

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences



(a) (b)

Fig. 5: Q-learning evaluation with respect to the episode number with varying target states s_d: (a) cumulative reward per episode (b) Q-functions relative difference

We consider three desired cancer viability ratio, or desired target states $s_d \in [0.13, 0.25, 0.50]$. To numerically simulate each episode for training, the new state S_{k+1} is sampled from (3) for a given S_k and A_k.

The progress of Q-learning iteration is presented in Figure 5. As shown in Figure 5(a), the cumulative reward per episode increases as Q converges to the optimal action-value function Q^*. Also in Figure 5(b), the relative difference of Q between the current Q-functions and the Q-functions from previous episode decreases, implying the convergence.

The resulting optimal action-value function and the corresponding deterministic greedy optimal policies are illustrated in Figure 6. This can serve as the baseline treatment plan to reduce the cancer cell viability ratio to a desired level for the given current viability ratio.

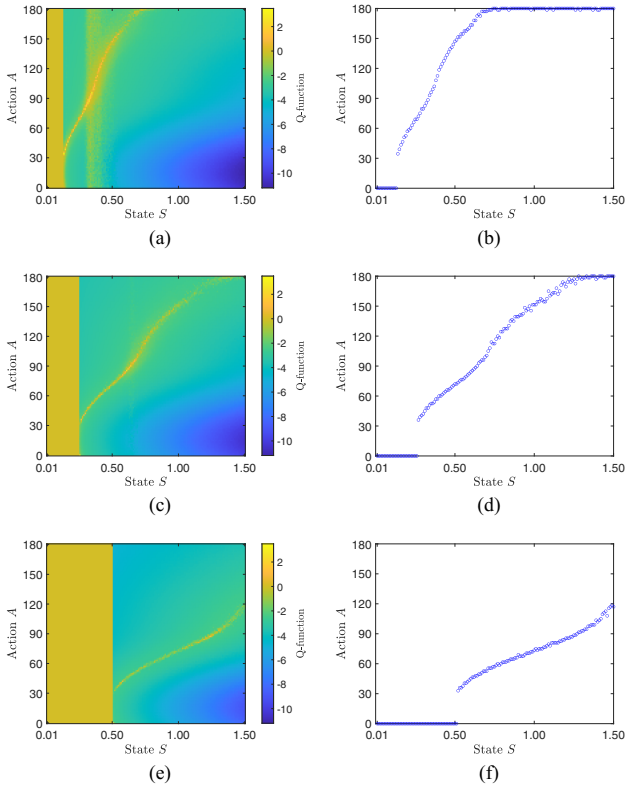


Fig. 6: Optimal Q-functions (left column) and optimal policies (right column) for the empirical dynamics with varying target states s_d: (a)(b) $s_d = 0.13$ (c)(d) $s_d = 0.25$ (e)(f) $s_d = 0.50$

C. Adaptive Learning with Modeling Errors

Although the above reinforcement learning with the empirical dynamics can provide a guide on how to administer CAP treatments, it is unlikely that the actual responses of cancer cells under treatments behave exactly same as the empirical model. While Q-learning will eventually adapted to the actual dynamics, it may take plenty of episodes until convergences. Here, we address it by utilizing the Q-function optimized for the empirical dynamics to initialize the Q-function for the actual dynamics. Instead of choosing the action deterministically using the greedy policy, the action is sampled through the softmax function such that the reinforcement learning can explore the actions beyond the specific optimal action for the empirical model.

Here, we assume that the actual cell dynamics is considered as the empirical dynamics with a random unknown perturbation Δ. Since the empirical dynamics is modeled with a Gaussian process, it is natural to model the perturbation with a normal distribution. More specifically, for any state-action pair (s, a),

$$\Delta(s,a) \sim \mathcal{N}(\mu_\Delta(s,a), \sigma^2_\Delta(s,a)), \tag{11}$$

where the mean and the variance of the perturbation are chosen as a function of the current state and the action as follows.

$$\mu_\Delta(s,a) = 3\sigma_*(s,a),$$
$$\sigma_\Delta(s,a) = 0.$$

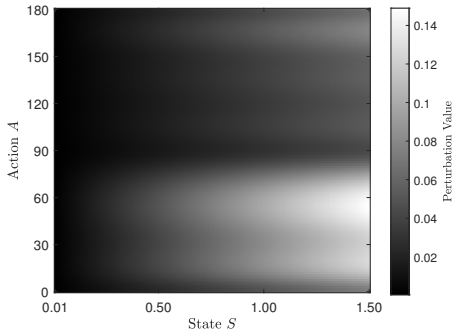


Fig. 7: Perturbation table for all state-action pairs

In other words, the mean value is shifted by the threefold of the standard deviation of the empirical model.

Now the actual cell dynamics is given by

$$\mathcal{GP}_{actual}(s,a) = \mathcal{GP}_{empirical}(s,a) + \Delta(s,a) \tag{12}$$

The sample values of the perturbation are illustrated in Figure 7, and when the current state is $S_k = 1$, the distribution of the new state S_{k+1} for varying actions is presented in Figure 8.

We apply the Q-learning to the actual dynamics with three desired state of $s_d \in [0.13, 0.25, 0.50]$. During the training, for every time step t_k, the action A_k is chosen from a probability distribution that is numerically computed through the softmax function. For any action $a \in \mathcal{A}$, with Q-function,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

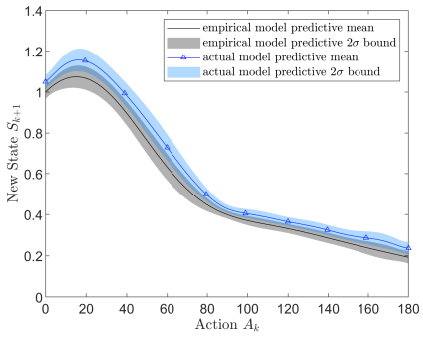


Fig. 8: Comparison between empirical dynamics and actual dynamics with $S_k = 1.00$

the probability of being selected can be seen as follows.

$$p(a|S_k) = \frac{\exp\left(\frac{Q(S_k,a)-\max Q(S_k,:)}{\tau}\right)}{\sum_{\text{all } a_i \in \mathcal{A}} \exp\left(\frac{Q(S_k,a_i)-\max Q(S_k,:)}{\tau}\right)}, \qquad (13)$$

where the temperature parameter τ controls the sensitivity of the probability distribution to Q-function. The resulting optimal Q-function and the deterministic greedy policy for the actual dynamics are given in Figure 9. It is shown that both of the Q-functions and the optimal policies are adapted to the actual dynamics by interacting with them.

As discussed above, a more important question is how fast the Q-function is adjusted for the actual environment. To examine the adaptive learning speed, an independent Q-learning, i.e. the control group, is carried out for the actual dynamics, after choosing the initial Q-function randomly. The learning speed of both cases are illustrated in Figure 10, where the reward and the discrepancy of the Q-function from its optimal values are given with respect to the number of episodes, for three cases of the desired state. It is shown that initializing the Q-function with the value optimized for the empirical dynamics along with the softmax policy improved the reward and the convergence rate substantially. In particular, the reward is quickly increased to the optimal range with a small number of episodes compared against the random initialization which requires numerous iterations. These verify that the knowledge gained from the empirical model can be strategically utilized in the actual dynamics through reinforcement learning.

IV. Reinforcement Learning with Safe Exploration

As is discussed in Section III-C, during the adaptive learning progress, the reinforcement learning agent would select actions through the softmax function that provides the probabilities of each action to be selected in the action space. However, in the actual CAP treatment, due to safety concerns, actions that may cause undesirable and unrecoverable results or that have large uncertainties in the outcome should be avoided. For instance, while treatments with large durations Δt may cause the cell viability ratio r reach the desired level r_d quickly, the actual resulting r may descend to a lower level than r_d due to the stochastic nature of the cell dynamics, especially when there is large uncertainties in the prediction

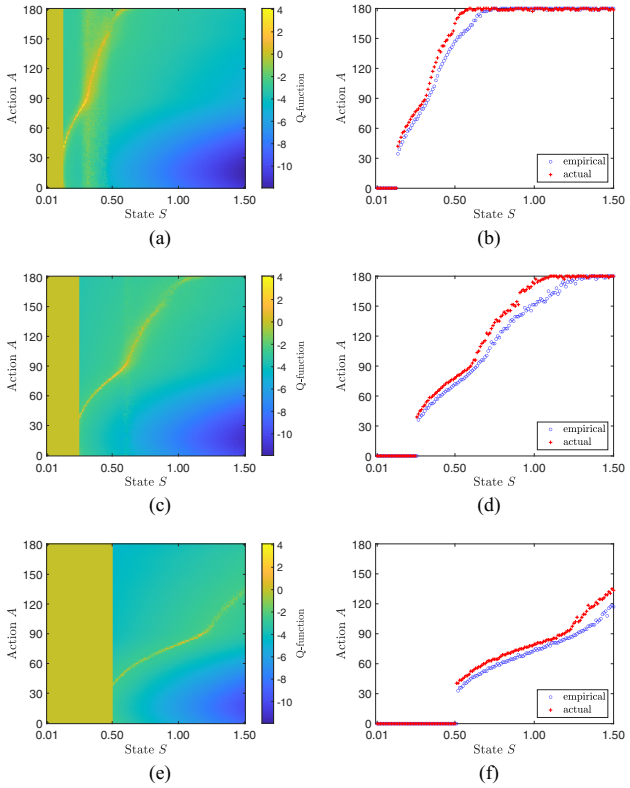


Fig. 9: Optimal Q-functions (left column) and optimal policies (right column) for the actual cancer dynamics with varying target states s_d: (a)(b) $s_d = 0.13$ (c)(d) $s_d = 0.25$ (e)(f) $s_d = 0.50$. In the right column of figures, the optimal policies for the empirical dynamics (blue) are compared with those for the actual cancer dynamics (red).

of the cell viability ratio after the treatment. Such excessive actions can be considered risky as there is a non-trivial chance for unsafe results. In particular, this is problematic especially during the initial phase of the Q-learning, while the initial Q-function optimized for the empirical dynamics is adapted to the actual cancer dynamics. In this section, we present safe reinforcement learning strategies where the cancer dynamics model is also updated to the actual dynamics in the probabilistic framework, and unsafe actions are excluded by accounting the uncertainties in the learned dynamic model.

A. Safe Action Selection

Throughout this paper, the cancer dynamics in response to CAP treatments is represented by a Gaussian process, as illustrated in Figure 3 for the empirical dynamics and in Figure 8 for the actual dynamics. To account the uncertainties of the Q-learning in safe action selection, the empirical dynamics is also updated throughout the cancer treatment. More specifically, the Gaussian process representing the actual cancer dynamics is initialized with the empirical Gaussian process, and at each treatment, the corresponding pair of the selected action and the resulting state is added to the data set of the Gaussian process. This is desirable as the data set is expanded the learned model converges toward the actual dynamics, and we

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

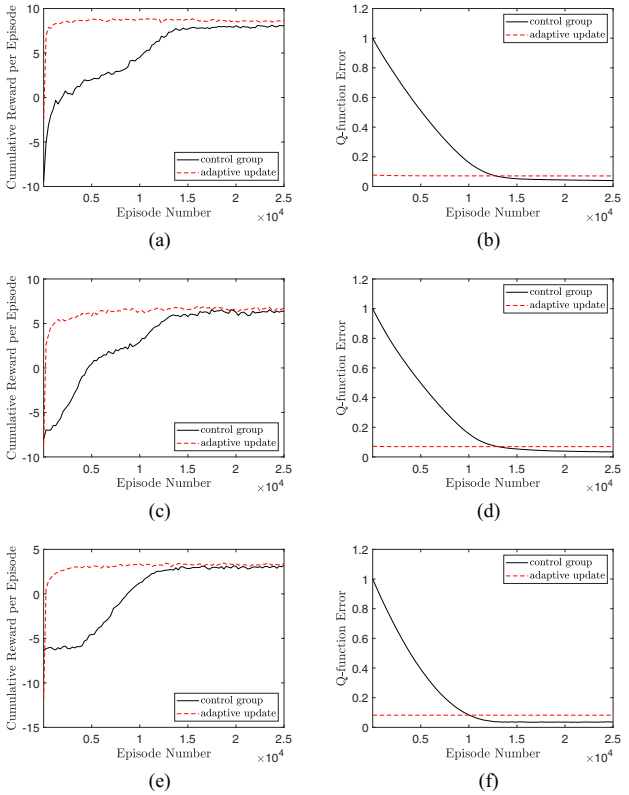


Fig. 10: Convergence of Q-learning for the actual cancer dynamics represented by the reward (left column) and the Q-function error (right column) with respect to the number of episodes with varying target states s_d: (a)(b) $s_d = 0.13$ (c)(d) $s_d = 0.25$ (e)(f) $s_d = 0.50$. The red dash curve that was initialized by the optimal Q-function of the empirical dynamics exhibits substantially faster convergence against the random initialization denoted by the blue curves.

can further access the confidence in the learned model as the Gaussian process provides the standard deviation of the prediction. For example, over the range of states and actions where new data points are acquired, the standard deviation will be small, or in the visualization of Figure 8, the shaded region will be thinner. Therefore, for each treatment duration selection, we can predict the resulting viability ratio for the actual cancer dynamics with an expected level of confidence, and the prediction becomes more accurate as the treatment is repeated.

This learned dynamic model is utilized in safe action selection as follows. The safe treatment is declared as the treatment where the ratio of the state after the treatment S_{k+1} to the state before the treatment S_k is not too excessive. In the preceding numerical simulation, the safe range of the ratio S_{k+1}/S_k is chosen as $[0.3, 1.1]$, i.e., the cell viability ratio after treatment should not exceed 110% of the current cell viability ratio and it should not become lower than 30%. The specific bound of the ratio can be adjusted as desired. For the current state S_k, the above dynamic model represented by a Gaussian process provides the predictive mean $\mu(S_k, a)$ and the standard deviation $\sigma(S_k, a)$ for the new state S_{k+1} at each

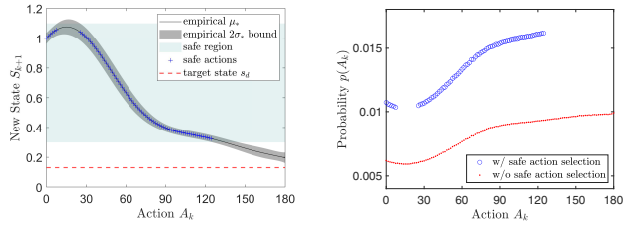


Fig. 11: At $S_k = 1.00$, with target state $s_d = 0.13$, the safe region and all possible safe actions are given on the left. On the right figure, the blue circle markers represent probabilities of actions being selected when the safe action selection strategy is implemented, where the red dot markers represent the regular softmax probabilities of action selection.

action a. The corresponding set of feasible action is defined as the Safe Action Space as follows.

$$\mathcal{A}_{safe}(S_k) = \{a \in \mathcal{A} \,|\, \mu(S_k, a) - 2\sigma(S_k, a) \geq 0.3S_k \\ \mu(S_k, a) + 2\sigma(S_k, a) \leq 1.1S_k\}.$$

In other words, for any action a in $\mathcal{A}_{safe}$, the prescribed ratio of the state S_{k+1}/S_k is guaranteed to be satisfied with the probability of 0.95. When integrated with the above dynamics learning, the safe action set is enlarged over the course of the treatment as the standard deviation of the prediction $\sigma(s, a)$ is reduced.

During the Q-learning, the action is selected using the softmax function as presented in (13), but using the safe action set $\mathcal{A}_{safe}$ instead of all of the possible actions. For example, when the current state is $S_k = 1.00$ and the target state is $s_d = 0.13$, the safe actions and the safe action selection probabilities are presented in Figure 11. In particular, Figure 11(a) illustrates how the safe action space is formulated, and Figure 11(b) shows the resulting action selection probability compared with the regular Q-learning, where it is observed that risky actions are properly disregarded.

The procedure for the Q-learning with the dynamics learning and the safe action selection is summarized in Table II.

B. Safe Reinforcement Learning for CAP

The proposed safe Q-learning is compared against the regular Q-learning presented in Section III-B, using the simulated actual dynamics shown in Figure 8. For both cases, the Q-function is initialized with the Q-function optimized for the empirical dynamics, and the desired target states are varied in $s_d \in [0.13, 0.25, 0.50]$.

After a 201-episode learning, the change of the Q-function, i.e., the difference between the empirical optimal Q-function and the updated Q-functions, are illustrated in Figure 12, where the figures on the left column are for the Q-learning with safe action selection and the figures on the right column are for the regular Q-learning.

It is observed that the safe Q-learning updates the Q-function within the safe region, while the regular Q-learning updates Q-functions in the whole action space. As shown in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

TABLE II: Procedure for Q-learning with dynamics learning and safe action selection

procedure SAFE Q-LEARNING UPDATE ITERATION
  Initialize $Q(S, A)$ with empirical optimal Q-function for all (S, A) pairs
  Initialize Gaussian process model with empirical Gaussian process model
  repeat
    $S_k = S_0$ where S_0 is chosen randomly with $S_0 > s_d$
    repeat
      Determine safe action set $\mathcal{A}_{safe}(S_k)$ through the Gaussian process model
      Choose action A_k from $Q(S_k, \mathcal{A}_{safe}(S_k))$ using softmax method
      Obtain new state S_{k+1} and reward R_{k+1} by performing action A_k
      $Q(S_k, A_k) = Q(S_k, A_k) + \alpha[R_{k+1} + \gamma \max_a Q(S_{k+1}, a) - Q(S_k, A_k)]$
      Update the Gaussian process model with data $< S_k, A_k, S_{k+1} >$
      $S_k = S_{k+1}$
    until $S_k \leq s_d$
  until Episode number reaches designed maximum
end procedure

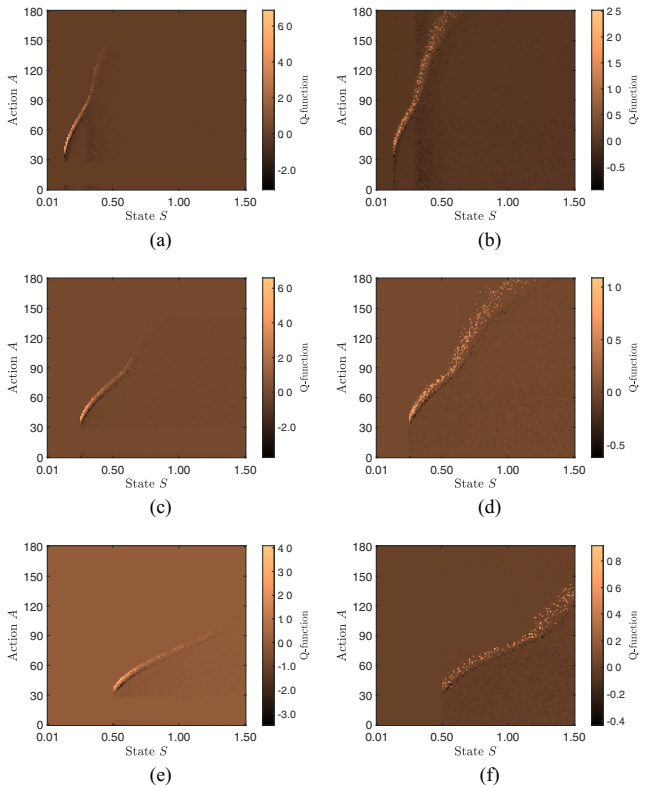


Fig. 12: Q-function changes after 201 episodes of learning for Q-learning with safe action selection (left column) and with regular Q-learning (right column) for varying target states s_d: (a)(b) $s_d = 0.13$ (c)(d) $s_d = 0.25$ (e)(f) $s_d = 0.50$.

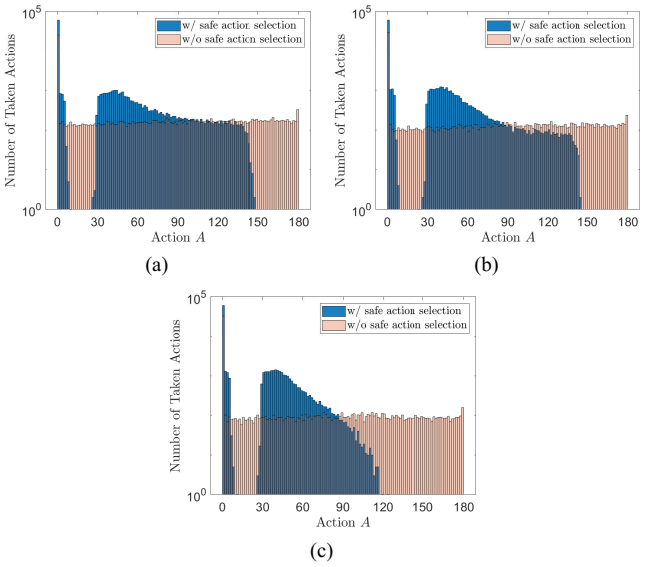


Fig. 13: Histogram of actions taken during learning with varying target states s_d: (a) $s_d = 0.13$ (b) $s_d = 0.25$ (c) $s_d = 0.50$. The blue histograms represent the actions taken with safe Q-learning and the orange histograms represent the actions taken with regular Q-learning.

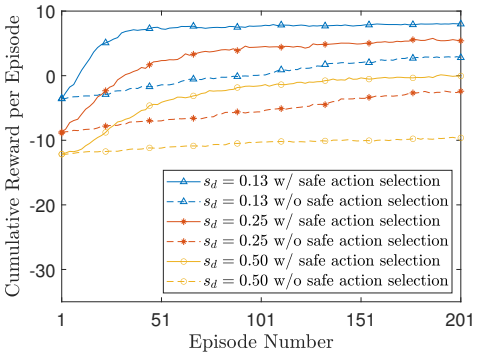


Fig. 14: Progression of the cumulative reward per episode for the safe Q-learning (solid lines with markers), and the regular Q-learning (dash lines with markers)

Figure 13 for the histogram of all actions taken, the regular Q-learning implemented all of possible actions. Whereas no excessive action is selected for the safe Q-learning.

More importantly, the magnitude of Q-function changes in the safe Q-learning is greater than the regular Q-learning cases, since by limiting the number of actions that can be selected, the Q-function is updated more often in the safe region for the same number of episodes, thereby accelerating the learning. Consequently, the cumulative reward per episode, representing the evaluation of the learning progress, is consistently greater for the proposed safe Q-learning, as illustrated in Figure 14.

Next, we evaluate the accuracy of the learned dynamic model as follows. First, the prediction error in the safe region, i.e., the differences between the mean of the actual dynamics and the mean value predicted by the Gaussian process decreases as is shown in Figure 15. Next, for target state $s_d = 0.13$, the learned dynamics at episode number

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TRPMS.2021.3094874, IEEE Transactions on Radiation and Plasma Medical Sciences

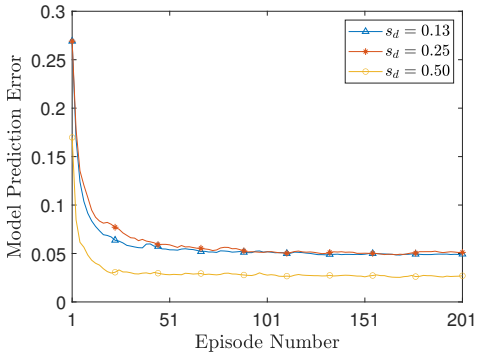


Fig. 15: Safe-region Gaussian process model prediction error for various target states s_d

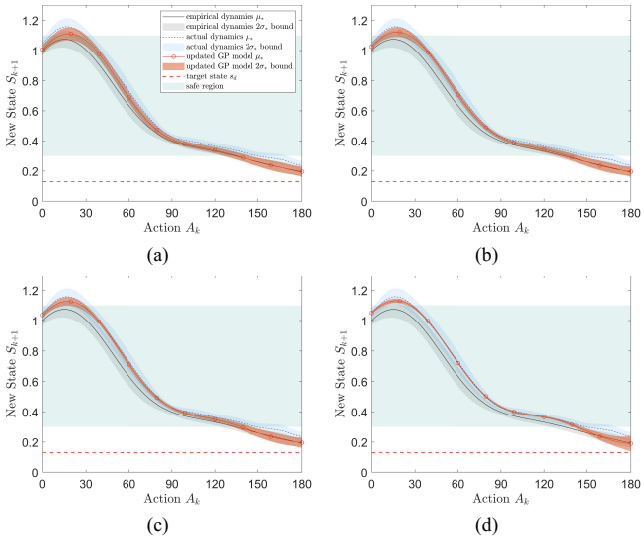


Fig. 16: Progress of learned model with target state $s_d = 0.13$, when evaluating at $S_k = 1.00$. Figure (a)-(d) represent the progress at episode number $[1, 3, 5, 61]$. The predicted mean (orange circle curve) and predicted 2σ bound (orange shaded area) are compared with the empirical dynamics (black solid curve and black shaded area) and the actual dynamics (blue dash curve and blue shaded area), where the safe region (gray green shaded area) and target state (red dash line) are included in the figures.

$[1, 3, 5, 61]$ are illustrated in Figure 16. It is shown that as the treatments progress, the learned dynamic model converges to the actual dynamics, and also the uncertainties in the prediction visualized by the 2σ bounds decrease over time.

Finally, we present 16 random treatment scenarios for the target cell viability ratio of $r_d = 0.13$, as illustrated in Figure 17 where the evolution of the viability ratio over time and the viability ratio after the period of 48 hour are given for both of the regular Q-learning and the safe Q-learning. As shown in Figure 17(a), the regular Q-learning may cause the cancer cell viability ratio to be unexpectedly increased over the first two treatment periods, thereby causing larger treatment errors. Whereas the safe Q-learning exhibits more regularized results consistent with the treatment goal as given in Figure 17(b).

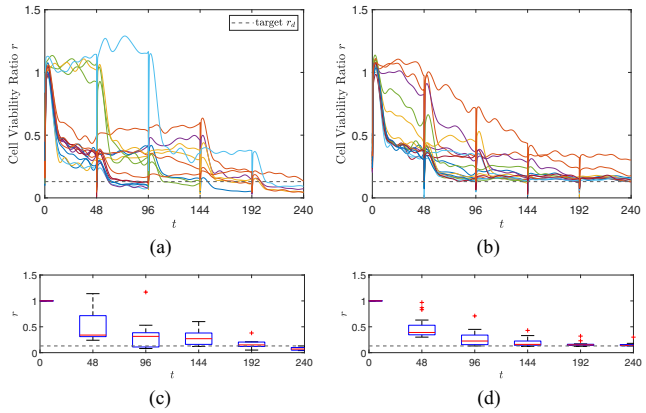


Fig. 17: Figure (a) and (b) represent the time-evolution of cell viability ratio, where the regular Q-learning (left column) is compared with the safe Q-learning (right column) for the target cell viability ratio $r_d = 0.13$. Figure (c) and (d) are the time-evolution of the box plot for cell viability ratio error.

In short, the proposed safe Q-learning is composed of the synergistic integration of learning the actual cancer dynamics with a Gaussian process and selecting the safe actions with a probabilistic guarantee. It is illustrated that the proposed approach avoids excessive risky treatments while accelerating the learning process.

V. Conclusions

This paper studies safe reinforcement learning for adaptive cancer treatments with cold atmospheric plasma (CAP). First, an empirical model is constructed to represent the cancer cell response to various CAP treatment conditions. A set of data constructed through *in vitro* experiments is learned through a Gaussian process, which is capable of generalizing the cancer response for arbitrary conditions beyond the specific treatments considered in the experiment. Next, the CAP cancer treatment is formulated as a Markov decision process, to which a reinforcement learning is applied to find the optimal treatment plan to reduce the cell viability ratio of cancer to a desired level. It is shown that Q-learning is capable of generating an optimal policy for the empirical dynamics, and it can be quickly adapted to the actual dynamics. Finally, by utilizing a Gaussian process, a safe Q-learning is proposed to prevent exceedingly aggressive treatments with greater uncertainties.

Future directions include incorporating various treatment conditions beyond the treatment durations, such as the discharge voltage or the gas composition of the plasma jet. However, directly adopting a Gaussian process for such higher-dimensional inputs is challenging due to the potential difficulties associated with repeating experiments with live cancer cells. Also, safety in reinforcement learning can be addressed beyond the exploration process considered in this paper. Another interesting direction will be utilizing real-time diagnostics, such as electrochemical impedance measurement in [28], for *in vivo* experiments. Also, we are investigating the proposed adaptive plasma framework with the explicit consideration of selectivity to avoid any harm to healthy cells.

REFERENCES

[1] M. Keidar and I. Beilis, *Plasma engineering: applications from aerospace to bio and nanotechnology*. Academic Press, 2013.

[2] J. Schlegel, J. Köritzer, and V. Boxhammer, "Plasma in cancer treatment," *Clinical Plasma Medicine*, vol. 1, no. 2, pp. 2–7, 2013.

[3] D. Yan, J. H. Sherman, and M. Keidar, "Cold atmospheric plasma, a novel promising anti-cancer treatment modality," *Oncotarget*, vol. 8, no. 9, p. 15977, 2017.

[4] M. Keidar, "Plasma for cancer treatment," *Plasma Sources Science and Technology*, vol. 24, no. 3, p. 033001, 2015.

[5] M. Keidar, D. Yan, I. I. Beilis, B. Trink, and J. H. Sherman, "Plasmas for treating cancer: opportunities for adaptive and self-adaptive approaches," *Trends in biotechnology*, vol. 36, no. 6, pp. 586–593, 2018.

[6] D. Gidon, D. B. Graves, and A. Mesbah, "Effective dose delivery in atmospheric pressure plasma jets for plasma medicine: a model predictive control approach," *Plasma Sources Science and Technology*, vol. 26, no. 8, p. 085005, 2017.

[7] ——, "Predictive control of 2d spatial thermal dose delivery in atmospheric pressure plasma jets," *Plasma Sources Science and Technology*, vol. 28, no. 8, p. 085001, 2019.

[8] D. Gidon, H. S. Abbas, A. D. Bonzanini, D. B. Graves, J. M. Velni, and A. Mesbah, "Data-driven lpv model predictive control of a cold atmospheric plasma jet for biomaterials processing," *Control Engineering Practice*, vol. 109, p. 104725, 2021.

[9] A. D. Bonzanini, D. B. Graves, and A. Mesbah, "Learning-based smpc for reference tracking under state-dependent uncertainty: An application to atmospheric pressure plasma jets for plasma medicine," *IEEE Transactions on Control Systems Technology*, 2021.

[10] Y. Lyu, L. Lin, E. Gjika, T. Lee, and M. Keidar, "Mathematical modeling and control for cancer treatment with cold atmospheric plasma jet," *Journal of Physics D: Applied Physics*, vol. 52, no. 18, p. 185202, 2019.

[11] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[12] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[13] E. Gjika, S. Pal-Ghosh, A. Tang, M. Kirschner, G. Tadvalkar, J. Canady, M. A. Stepp, and M. Keidar, "Adaptation of operational parameters of cold atmospheric plasma for in vitro treatment of cancer cells," *ACS applied materials & interfaces*, vol. 10, no. 11, pp. 9269–9279, 2018.

[14] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 11 2005. [Online]. Available: https://doi.org/10.7551/mitpress/3206.001.0001

[15] J. Garcıa and F. Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 1437–1480, 2015.

[16] M. Pecka and T. Svoboda, "Safe exploration techniques for reinforcement learning–an overview," in *International Workshop on Modelling and Simulation for Autonomous Systems*. Springer, 2014, pp. 357–375.

[17] D. Di Castro, A. Tamar, and S. Mannor, "Policy gradients with variance related risk criteria," *arXiv preprint arXiv:1206.6404*, 2012.

[18] P. Geibel and F. Wysotzki, "Risk-sensitive reinforcement learning applied to control under constraints," *Journal of Artificial Intelligence Research*, vol. 24, pp. 81–108, 2005.

[19] A. Geramifard, J. Redding, and J. P. How, "Intelligent cooperative control architecture: a framework for performance improvement using safe learning," *Journal of Intelligent & Robotic Systems*, vol. 72, no. 1, pp. 83–103, 2013.

[20] P. Quintía Vidal, R. Iglesias Rodríguez, M. Á. Rodríguez González, and C. Vázquez Regueiro, "Learning on real robots from experience and simple user feedback," *Journal of Physical Agents (JoPha)*, vol. 7, no. 1, p. 57–65, 2013.

[21] M. Turchetta, F. Berkenkamp, and A. Krause, "Safe exploration in finite markov decision processes with gaussian processes," *arXiv preprint arXiv:1606.04753*, 2016.

[22] Y. Sui, A. Gotovos, J. Burdick, and A. Krause, "Safe exploration for optimization with gaussian processes," in *International Conference on Machine Learning*. PMLR, 2015, pp. 997–1005.

[23] M. Witman, D. Gidon, D. B. Graves, B. Smit, and A. Mesbah, "Sim-to-real transfer reinforcement learning for control of thermal effects of an atmospheric pressure plasma jet," *Plasma Sources Science and Technology*, vol. 28, no. 9, p. 095019, 2019.

[24] D. Gidon, B. Curtis, J. A. Paulson, D. B. Graves, and A. Mesbah, "Model-based feedback control of a khz-excited atmospheric pressure plasma jet," *IEEE Transactions on Radiation and Plasma Medical Sciences*, vol. 2, no. 2, pp. 129–137, 2017.

[25] A. Mesbah and D. B. Graves, "Machine learning for modeling, diagnostics, and control of non-equilibrium plasmas," *Journal of Physics D: Applied Physics*, vol. 52, no. 30, p. 30LT02, 2019.

[26] A. D. Bonzanini, J. A. Paulson, D. B. Graves, and A. Mesbah, "Safe dose delivery in fast sampling atmospheric plasmas using projection-based approximate economic mpc," in *Proc. IFAC World Congr.*, 2020.

[27] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[28] L. Lin, Z. Hou, X. Yao, Y. Liu, J. R. Sirigiri, T. Lee, and M. Keidar, "Introducing adaptive cold atmospheric plasma: The perspective of adaptive cold plasma cancer treatments based on real-time electrochemical impedance spectroscopy," *Physics of Plasmas*, vol. 27, no. 6, p. 063501, 2020.