2022 IEEE International Conference on Big Data (Big Data) | 978-1-6654-8045-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/BigData55660.2022.10020647

Deep Reinforcement Learning for Bandit Arm Localization

Wenbin Du
College of Computer Science and Software Engineering
Shenzhen University
Shenzhen, China
wbdu@szu.edu.cn

Huaqing Jin
Department of Radiology and Biomedical Imaging
University of California, San Francisco
California, U.S.A.
huaqing.jin@ucsf.edu

Chao Yu
School of Computer Science and Engineering
Sun Yat-Sen University
Guangzhou, China
yuchao3@mail.sysu.edu.cn

Guosheng Yin
Department of Statistics and Actuarial Science
University of Hong Kong
Hong Kong, China
gyin@hku.hk

***Abstract*—In the multi-armed bandit (MAB) framework, we investigate the problem of learning the means of distributions that are associated with a finite number of arms under a monotonic constraint. Different from the traditional MAB, our problem involves a parameter constraint and a limited trial budget (i.e., the number of arm pulls is small). However, the number of training samples can be as large as possible through (infinite) simulations, while each training sample is of limited size. This situation arises when some additional information is provided before the trial starts and each arm pull (or testing) could be of extraordinary cost. For example, in cancer dose-finding clinical trials, higher toxicity probabilities are typically associated with higher dose levels (i.e., the monotonic dose–toxicity constraint), and the loss due to the drug's toxicity, side-effects or death of patients can be enormous. We formulate this problem in the reinforcement learning (RL) paradigm, which is referred to as a bandit arm localization problem. We propose a novel approach in a double deep Q-learning framework, which is integrated with a state-of-the-art statistical model to preserve the parameter constraint and develop a more effective learning strategy. The double deep Q-learning model can be trained with a large (can be as large as infinite) number of simulated trials, which is the first time to cast dose finding in the RL framework. We evaluate the performance of our approach through extensive simulation studies in realistic settings of phase I clinical trials. The proposed double deep Q-learning is shown to outperform the baseline methods in cancer dose-finding trials.**

***Index Terms*—deep Q-learning, dose finding, multi-armed bandit, phase I clinical trial, reinforcement learning**

I. Introduction

Multi-armed bandit (MAB) problems have been studied extensively on the basis of combining exploitation and exploration schemes [1]–[3], which witness important applications in clinical trials. The reinforcement learning (RL) has been widely used in various health care problems [4]. We reformulate the dose-finding problem in cancer clinical trials as the first step of development of a new treatment in the RL paradigm. In a phase I cancer trial, patients may develop negative responses (toxicities or side-effects) depending on the amount of treatment (dosage) they receive [5]. The probability for patients to develop the dose-limiting toxicity (DLT) increases with the amount of dosage that is administrated. The goal of such clinical trials is to identify the maximum tolerated dose (MTD), which corresponds to the amount of drug that would cause a certain proportion (e.g., the target toxicity rate is 30%) of patients to experience the DLT outcomes. In practice, a limited number of patients are enrolled into a phase I clinical trial, where only several prespecified (typically fewer than 10) dose levels are tested to find the MTD. In this context, the dose assignment is carried out sequentially, i.e., the decision on which dose level to choose for the next patient or the next cohort of patients (the cohort size is typically less than 3) is based on the choices and outcomes of the previously treated patients in the trial. After all the patients in the trial are completed with treatment and outcome evaluation, the MTD is estimated using all the information accumulated in the trial (e.g., the number of patients experiencing DLT at each dose level). The essential problem is to identify the target arm (MTD) using limited samples that are sequentially drawn from the distributions associated with the arms.

Viewing each dose level as one arm in the bandit, our setup is similar to the MAB problem [6]. However, there are two key features of the dose-finding problem that are distinct from the typical MAB. First, there is additional information about the underlying distributions of the arms because higher dose levels naturally imply higher toxicity probabilities. More specifically, we know whether one arm's expected "reward" (or response) is larger than another based on the dose levels. This additional information is referred to as the monotonic relationship among the toxicity probabilities of the investigated doses, i.e., the DLT rate increases as the dose increases. Second, the goal is not to

This work was partially supported by the Research Grants Council of Hong Kong (17308321) and the HKU-TCL Joint Research Center for Artificial Intelligence sponsored by TCL Corporate Research (Hong Kong).

maximize the accumulated "reward"; instead, the objective of the above problem is to identify the arm with the total "reward" closest to a predefined target. These distinctive features of the dose-finding design make it essentially different from the conventional MAB problem and its various extensions [7]–[9].

Although the aforementioned two problems (MAB versus dose finding) are distinct, they share the same exploration–exploitation dilemma. More specifically, the more patients are tested at one dose level, the better estimation of the toxicity probability we can obtain for that dose level. However, because the total number of patients is limited for one trial, we need to develop an effective strategy to identify the target arm as accurately and efficiently as possible, by taking advantage of the monotonic relationship. We define this new framework as a bandit arm localization (BAL) problem, which is common in cancer clinical trials but has never been studied before under the RL paradigm [10]. The BAL problem is well motivated by many real applications. Another example is dynamic pricing in auctions. Suppose there are a total of T rounds of auction, and at each round t, the algorithm chooses a price p_t and offers one item for sale at that price. The absentee bidder, who cannot show up in person at the auction, may have in mind some value v_t for that item, where v_t is drawn independently from some fixed but unknown distribution. The customer buys the item if and only if $p_t < v_t$. It is natural to assume that the higher the price, the fewer the customers would engage in bidding on the item. The goal of the BAL algorithm is to determine the proper price that has the largest potential to attract a proportion (e.g., 80%) of customers to participate in the auction. Similar problems also occur when one needs to decide how much resource should be invested in a project to achieve an acceptable output with a certain level of confidence by a trial procedure.

To properly incorporate the monotonic relationship in dose finding, many statistical methods, e.g., the Bayesian approaches [11]–[14], have been developed. However, these methods have major issues that may hamper the optimality of the design. In most of these methods, the choice of the next arm (dose level) is completely based on the estimation of the target arm, which could be wrong due to bias or model misspecification, especially at the early stage of a trial when the data are very sparse. As the goal of a trial is to achieve the best estimation based on all the observed responses at the end of the trial, a new strategy can be devised to account for future rewards rather than a myopic design using a greedy approach such as the continual reassessment method (CRM) [11]. To our knowledge, all methods for dose finding adopt a greedy approach, which cannot guarantee the optimality in the long term [15].

We propose a deep Q-learning framework, which considers the need for exploration and models future rewards explicitly. The main contributions of our work are threefold: First, we formalize the bandit arm localization problem under the RL paradigm and adopt a Q-learning model to solve or approximate it. Second, to take advantage of the monotonic relationship, we incorporate the state-of-the-art statistical model into the RL framework, so that the regression model can be seamlessly integrated in the RL framework. Third, we evaluate our approach under both fixed and random scenarios and the proposed Q-learning model outperforms the existing baseline methods.

II. Related Works

Multi-Armed Bandit The MAB problems [8], [16] are built upon the popular theoretical models of exploration–exploitation trade-offs in machine learning. There are various extensions of the traditional MAB problem, such as bandits with knapsacks [17], which deals with the case where several constrained resources are consumed by the algorithm, such as the inventory of products in the dynamic pricing problem. There are similar problem settings in the budget-limited MAB [18], [19] or the best-arm identification problem [20]–[23]. Many works focused on solving the dose-finding problem by utilizing theoretical tools developed for the MAB. For example, in [24], [25], the patient allocation problem in clinical trials is addressed based on Gittins index [26], which is a method originally designed for MAB. In [2], a Thompson sampling-based algorithm is proposed for dose-finding clinical trials. However, the dose-finding problem has never been studied under the RL framework, because all existing methods take a greedy approach that misses the exploration mechanism.

Bayesian Approaches Many statistical methods have been proposed to solve similar problems [11], [15], [27], [28]. The most popular one is called the CRM [11], which takes a Bayesian regression approach to handling the additional monotonic constraint and the uncertainty of the toxicity probability at each dose level. However, statistical methods, such as the CRM and its variants, are all greedy algorithms in essence. They make the best guess of the target arm (e.g., the dose level whose toxicity probability closest to the target) based on all the available information, while no exploration of the arms or strategies is considered for potentially better future estimation. In other words, all the existing methods are typically myopic and often cannot deliver an optimal solution at the end of the trial.

To achieve the optimality in the long term, we propose a deep RL framework to solve the BAL problem. Furthermore, we utilize the monotonic relationship and model the uncertainty of the environment. We also integrate a Bayesian regression model into the RL framework when we define the agent's environment. Although the sample size of each trial is limited, we can generate as many as we wish simulation scenarios to train the double deep Q-learning model. In that sense of big data, the number of simulations for model training can be infinite, while the parameter learning would reach saturation after a large number of simulations are conducted.

III. Problem Formulation

The BAL problem itself is new and distinct from the MAB, as formulated below. There is a machine with a sequence of finite K arms indexed by $k = 1, \ldots, K$, and the response of each arm follows some distribution denoted by $\mathcal{F}_k$. The

means of these distributions are denoted as μ_k's, which are unknown but satisfy a monotonic relationship, $\mu_j < \mu_k$, for $1 \leqslant j < k \leqslant K$. The arms are pulled (or experimented) sequentially, and the total number of arm pulls or the horizon is set as T. When the kth arm is pulled at round or time step t, a response x_t is observed, which is generated from the underlying distribution $\mathcal{F}_k$. Given the information collected up to time step t, an algorithm determines which arm to choose next. We are interested in identifying the arm k^* that has the response rate closest to a given target ϕ,

$$k^* = \arg \min_{1 \leqslant k \leqslant K} |\mu_k - \phi|. \tag{1}$$

Due to its novel definition which is related to but different from the traditional MAB problems, the BAL problem is summarized in Definition 1.

IV. New Approach

Considering the sequential arm pulling (or dose testing) procedure as a Markov chain [29], we can formulate the BAL problem as a Markov decision process [30] and further solve the problem in the RL paradigm.

A. Reinforcement Learning

For the BAL problem, we define the four conventional components of the RL framework as follows.

State Suppose that arm k has been experimented for n_k times, and we observe z_k responses and $n_k - z_k$ non-responses. For each arm, the information can be represented as a 2-dimensional vector $(z_k, n_k - z_k)$. The state at time step t is designated to be $s_t = \{(z_1, n_1 - z_1), \ldots, (z_K, n_K - z_K)\}$, which is a $2K$-dimensional vector.

Action The choice that the agent makes at time step $t \in \{1, \ldots, T\}$ corresponds to which is the next arm to pull (or test).

Environment Let μ_k be the unknown probability parameter (mean) for the distribution associated with the kth arm. The environment of the RL paradigm can be defined as the large number of various scenarios that the algorithm aims to explore and estimate the mean μ_k. We utilize a scenario-generating process to mimic this environment, where one scenario corresponds to a sequence of arms with their means μ_k's satisfying the monotonic constraint, i.e., $\mu_1 < \cdots < \mu_K$. Note that each scenario corresponds to one episode (or one trial). It is natural to assume the response from each arm follows a Bernoulli distribution, $x_t^{(k)} \sim \text{Bernoulli}(\mu_k)$, where $x_t^{(k)}$ represents the response if arm k is pulled (or experimented) at time step t. As a result, let k_t denote the arm pulled at time step t, $k_t \in \{1, \ldots, K\}$, and then $z_k = \sum_{t=1}^{T} x_t^{(k)} I(k_t = k)$, where $I(\cdot)$ is the indicator function. In the training phase, a sequence of K arms with increasing response probabilities would be randomly generated as part of the environment to train the agent. Furthermore, a statistical state-of-the-art model is integrated into the environment. By adopting the statistical model into the RL framework, we can incorporate the monotonic constraint explicitly to enhance the estimation accuracy. More specifically, at the last time step T, we estimate the response rate $\hat{\mu}_k$ for each arm using the statistical model.

Reward The aim of the RL algorithm is to identify the arm k^* whose response rate is closest to a pre-specified target ϕ. At each time step t, an action a_t is taken, which means that the arm $a_t \in \{1, \ldots, K\}$ is pulled. If the current time step has not reached T, i.e., $t < T$, the intermediate reward function (which can improve the next arm selection) is defined as

$$R_t = \begin{cases} 1 & \text{if } a_t = k^*, \\ -\lambda |a_t - k^*| & \text{otherwise.} \end{cases} \tag{2}$$

where λ is a hyper-parameter controlling the degree of penalty for inferior selections. In the experiments, λ is set to be 0.5, and the distance of the pulled arm from the optimal arm k^*, $|a_t - k^*|$, imposes more punishment on those selections that are far away from the target arm.

At the last time step T, the CRM model is used to estimate the response rate $\hat{\mu}_k$ for each arm, and the estimated target arm is $\hat{k}^* = \arg\min_{1 \leqslant k \leqslant K} |\hat{\mu}_k - \phi|$. To make the reward mechanism consistent with the BAL's primary goal, the final reward function for the action taken at time step T is defined as

$$R_T = \begin{cases} T/2 & \text{if } \hat{k}^* = k^*, \\ -T/2 & \text{otherwise.} \end{cases} \tag{3}$$

With specification of these key components in RL, we can solve the problem by learning a value function (e.g., Q-function). The Q-value of taking action a_t at state s_t under policy π is denoted as $Q_\pi(s_t, a_t)$, which represents the expected return at time step t,

$$Q_\pi(s_t, a_t) = \mathbb{E}_\pi \left(\sum_{k=0}^{T-t} \gamma^k R_{t+k} \big| S_t = s_t, A_t = a_t \right),$$

where γ is the discount rate and policy π represents a probability distribution over the set of actions a_t, given the current state s_t. In the deep Q-learning framework, this Q-function can be modelled by a deep neural network (DNN), called the Q-network,

$$Q_\pi(s_t, a_t) = \text{DNN}(s_t, a_t).$$

Given the historical data up to time step t, i.e., $S_t = s_t$, the Q-network aims to estimate $Q_\pi(s_t, a_t)$, so that the agent can choose the arm that maximizes $Q_\pi(s_t, a_t)$. As detailed in Algorithm 1, we adopt a double deep Q-learning network (DDQN) [31], [32] as the backbone algorithm to learn the unknown parameters in the Q-network. Following the setting in the DDQN, a replay memory is used to store the agent's all past experience at each time step under different scenarios (or episodes). The DDQN includes a target Q-network and an action Q-network, which are updated under different frequencies and are synchronized periodically.

B. Continual Reassessment Method (CRM)

Among many statistical methods, we select one of the most popular methods, called the CRM [11], as another component

Definition 1: Bandit Arm Localization (BAL) Problem

Design Parameters: K arms with each indexed by k, the total number of pulls T, and the target response rate ϕ.
Goal: Identify the optimal arm $k^* = \arg\min_{1\leqslant k\leqslant K} |\mu_k - \phi|$, where μ_k is the unknown mean of the distribution associated with arm k.
Monotonic Constraint of Bandit Arms: $\mu_1 < \cdots < \mu_K$.
At each time step, $t = 1, \ldots, T$:
(i) An algorithm selects an arm $k_t \in \{1, \ldots, K\}$.
(ii) Arm k_t is pulled and then a response x_t is observed.
Once the total consumption of the pulling resource reaches its budget or horizon T, the optimal arm $\hat{k}^* = \arg\min_{1\leqslant k\leqslant K} |\hat{\mu}_k - \phi|$ is selected, where $\hat{\mu}_k$ is the estimate of μ_k based on all the responses $X = \{x_1, \ldots, x_T\}$.

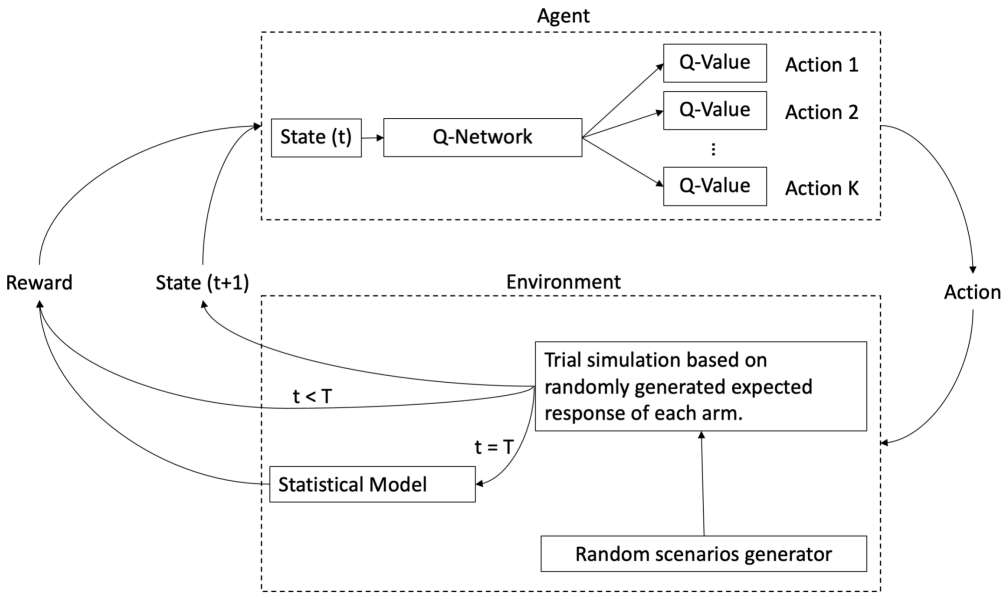


Fig. 1. The proposed reinforcement learning framework for the bandit arm localization problem.

of our approach. The CRM takes a Bayesian approach to dealing with the monotonic relationship and the uncertainty of the response rate for each arm. The CRM assumes a prior dose–response (or arm–response) curve, and then continuously updates this curve based on the observed accumulative outcomes from the pulled (or tested) arms in the trial. At each time step t, the next arm is chosen to be the one with an estimated response rate closest to the target. For the CRM, a single-parameter model is typically adopted, i.e., for the response rate $\mu_k(\cdot)$ of arm k,

$$\mu_k(\alpha) = p_k^{\exp(\alpha)},$$

where α is an unknown parameter and $0 < p_1 < \cdots < p_K < 1$ are prespecified toxicity probabilities at the K dose levels. Suppose that arm k has been pulled n_k times, and z_k responses have been observed. Let $\mathcal{D}$ denote the observed data in the trial, $\mathcal{D} = \{(z_k, n_k), k = 1, \ldots, K\}$. Based on the binomial distribution, the likelihood function is

$$L(\alpha|\mathcal{D}) = \prod_{k=1}^{K} \{p_k^{\exp(\alpha)}\}^{z_k} \{1 - p_k^{\exp(\alpha)}\}^{n_k - z_k}.$$

In the Bayesian paradigm, the response rate of arm k can be estimated by the corresponding posterior means of $\mu_k(\alpha)$,

$$\hat{\mu}_k = \int p_k^{\exp(\alpha)} \frac{L(\alpha|\mathcal{D}) f(\alpha)}{\int L(\alpha|\mathcal{D}) f(\alpha) d\alpha} d\alpha,$$

where $f(\alpha)$ is a prior distribution for the parameter α and typically a normal prior distribution $N(0, \sigma^2)$ is adopted.

During the training stage, a reward is calculated at each time step t based on the correctness of the arm selection following (2) and (3). This strategy encourages the agent to learn more effectively how to localize the target arm by taking the monotonic relationship into consideration, which coincides with the overall goal of the BAL problem.

At the end of the trial when the sample size is exhausted, the CRM model is used at time step T to estimate the response rate for each arm based on all the observed data. Finally, the best arm is selected as the one with the estimated response rate closest to the target.

V. Experiments

We apply the proposed BAL approach to a real dose-finding problem for phase I clinical trials in oncology. This application is of paramount importance due to cancer treatment development that may potentially save millions of lives. We consider five dose levels (arms) and assume that the toxicity probabilities increase monotonically with respect to the doses, i.e., the monotonic constraint.

The target toxic probability is ϕ = 0.3. Following the CRM setting in [12], at each time step we treat a cohort of three patients at the selected dose level and then observe the

Algorithm 1 Double deep Q-learning for bandit arm localization

Initialize the updating frequencies for the target Q-network as C_{tgt} and for the action Q-network as C_{act}, the capacity of the replay memory as C_{mem};
Initialize the action Q-network Q with random weights θ and the target Q-network Q' with weights θ', set $\theta' = \theta$;
for $n \leftarrow 1$ to N **do**
  Randomly generate the response rate (mean) for all arms $\{\mu_1, \ldots, \mu_K\}$
  for $t \leftarrow 1$ to T **do**
    if $t = 1$ **then**
      Select the first arm
    else
      With probability ε select a random neighbor arm a_t
      With probability $1 - \varepsilon$ set $a_t = \arg\max_a Q(s_t, a; \theta)$
      Observe response x_t from the selected arm a_t and update the state s_{t+1}
      if $t = T$ **then**
        Estimate the response rate by the CRM model and obtain reward r_T by (3)
      else
        Obtain reward r_t by (2)
      end if
      Store transition (s_t, a_t, r_t, s_{t+1}) in the replay memory
    end if
    if $n \bmod C_{\mathrm{tgt}} = 0$ **then**
      $\theta' = \theta$
    end if
  end for
  if $n \bmod C_{\mathrm{act}} = 0$ **then**
    Sample a random mini-batch of transitions (s_j, a_j, r_j, s_{j+1}) from the replay memory
    Set $y_j = \begin{cases} r_j & \text{for terminal } s_{j+1} \\ r_j + \gamma \max_{a'} Q'(s_{j+1}, \arg\max_{a'} Q(s_{j+1}, a; \theta); \theta') & \text{for non-terminal } s_{j+1} \end{cases}$
    Perform a gradient descent step on $\{y_j - Q(s_j, a_j; \theta)\}^2$
  end if
end for

corresponding responses. Because there is little information about the distribution associated with each arm, for the sake of safety, the first cohort is treated at the lowest dose and only the neighbors of the previous selected dose level are considered for the next time step in the trial, i.e., no dose skipping. The maximum sample size is set as 30, but the number of simulated scenarios/trials for training our model can be as large as possible (i.e., we can simulated as many trials as possible). For the CRM model, the initial guesses of the toxicity probabilities are chosen by the model calibration method of [33]: $(p_1, p_2, p_3, p_4, p_5) = (0.01, 0.09, 0.30, 0.54, 0.73)$. We take the prior distribution of α as a normal distribution with mean 0 and variance σ^2 = 2. Following Algorithm 1, we train our Q-network using the Adam algorithm [34]. The output of the Q-network is a K-dimensional vector representing the Q-value $Q_\pi(s_t, a_t)$ for $a_t \in \{1, \ldots, K\}$. The Q-network consists of six fully connected layers with {2048, 2048, 1024, 256, 256, 5} neurons respectively. Each fully connected layer is followed by a ReLU function except for the last layer. The replay memory capacity is set to be $C_{\mathrm{mem}} = 18000$. In each mini-batch, 50 transitions (s_j, a_j, r_j, s_{j+1}) are randomly selected from the replay memory. The learning rate is set to be 0.001, and our model is implemented by Pytorch [35].

A. Evaluation

In a typical RL problem, an environment with certain rules or dynamics is defined so that the algorithm can be tested with real performance on the system. However, the BAL problem deals with an uncertain environment (i.e., the distributions associated with arms are unknown and the target can be any one of the arms under investigation), and therefore we can only test the performance of our algorithm via simulation studies. As the underlying true response rates of the arms are unknown, any form of the assumption on these unknown response rates may deviate from the truth and thus introduce bias. We first introduce several baseline approaches and then evaluate different methods under both fixed scenarios and randomly generated scenarios (to avoid cherry picking scenarios).

B. Baseline approaches

Although the BAL problem is different from the MAB, it still can be formulated in the MAB paradigm in an approximate sense. For comparison, we adapt two best-arm identification algorithms to address this problem because the settings are similar. One is the upper confidence bound exploration

TABLE I
SIMULATION RESULTS WITH A TOXICITY TARGET ϕ = 30% UNDER FIXED SCENARIOS 1–5.

Model	Recommendation percentage at each dose level					DLT (%)
	1	2	3	4	5	
Scenario 1	0.30	0.40	0.55	0.60	0.65	
UCB-E	**47.8**	29.7	11.2	7.2	4.1	46.4
APT	**47.7**	31.4	11.1	6.4	3.3	46.2
CRM	**70.2**	28.2	1.5	0.1	0.0	33.8
DDQN	**55.3**	39.6	5.0	0.2	0.0	38.6
Scenario 2	0.20	0.30	0.60	0.70	0.75	
UCB-E	53.6	**33.9**	7.9	2.9	1.7	42.1
APT	43.9	**44.8**	7.3	2.7	1.3	42.7
CRM	29.5	**66.8**	3.7	0.0	0.0	28.5
DDQN	18.9	**76.1**	5.0	0.0	0.0	35.1
Scenario 3	0.06	0.15	0.30	0.55	0.60	
UCB-E	30.2	20.9	**30.2**	11.1	7.7	30.2
APT	12.2	29.5	**40.6**	11.4	6.3	30.2
CRM	0.2	27.1	**66.7**	5.8	0.1	24.0
DDQN	0.1	20.1	**72.6**	7.0	0.2	35.4
Scenario 4	0.06	0.08	0.10	0.30	0.50	
UCB-E	28.8	10.8	13.8	**31.5**	15.1	21.1
APT	14.5	16.5	17.2	**37.6**	14.2	20.3
CRM	0.2	6.2	26.4	**60.3**	6.9	18.3
DDQN	0.0	0.7	27.6	**64.1**	7.5	27.3
Scenario 5	0.02	0.06	0.10	0.20	0.30	
UCB-E	23.6	7.1	12.9	26.1	**30.3**	14.6
APT	5.7	13.4	17.3	33.8	**29.9**	14.4
CRM	0.0	1.1	15.2	48.1	**35.6**	15.5
DDQN	0.0	0.1	5.8	46.6	**47.5**	19.7

UCB-E: upper confidence bound exploration, APT: anytime parameter-free thresholding, CRM: continual reassessment method, DDQN: the proposed double deep Q-learning network. Correct selection percentages are highlighted in boldface and dose-limiting toxicity (DLT) percentages show the aggressiveness of the methods.

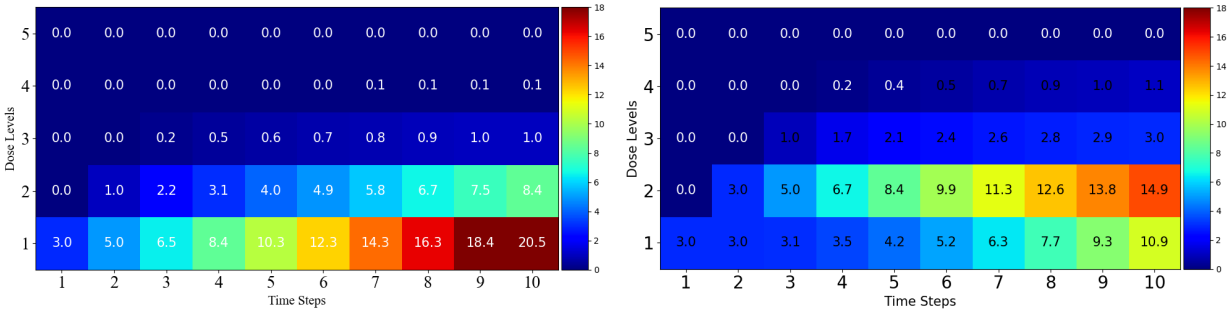
(UCB-E) algorithm [20] and the other is the thresholding algorithm [21]. However, the monotonic constraint, which is the unique feature of BAL, is not taken into consideration in the adaptation of both algorithms.

UCB-E algorithm: Following the adaptation in [21], at each time step t, the UCB-E algorithm pulls the arm that minimizes $B_k(t)$, where $B_k(t) = |\hat{\mu}_k(t) - \phi| - \sqrt{a/T_k(t)}$, and $T_k(t)$ denotes the number of pulls assigned to arm k till time step t and a is a hyper-parameter controlling the degree of exploration. We choose $a = (T - K)/H$, where $H = \sum_{i=1}^{K} \Delta_k^{-2}$ characterizes the degree of the problem difficulty and $\Delta_k = \phi - \mu_k$ is the gap between the target response rate and the true response rate of each arm. This algorithm requires the knowledge of H, and the calculation of H depends on μ_k, which is a strong assumption [21].

Thresholding algorithm: The anytime parameter-free thresholding (APT) algorithm is introduced to solve the thresholding bandit problem in [21]. Given a fixed time horizon T, the goal of the thresholding bandit problem is to find the set of arms whose response rates are above the threshold, up to a given precision ϵ, i.e., to correctly discriminate arms with $\mu_k > \phi + \epsilon$ from those with $\mu_k < \phi - \epsilon$. We adapt the APT algorithm by simply changing the output or the recommendation as the arm with the index $\hat{k}^* = \arg\min_{1 \le k \le K} |\hat{\mu}_k - \phi|$. Following the setting in [21], the precision ϵ is set to be 0.1.

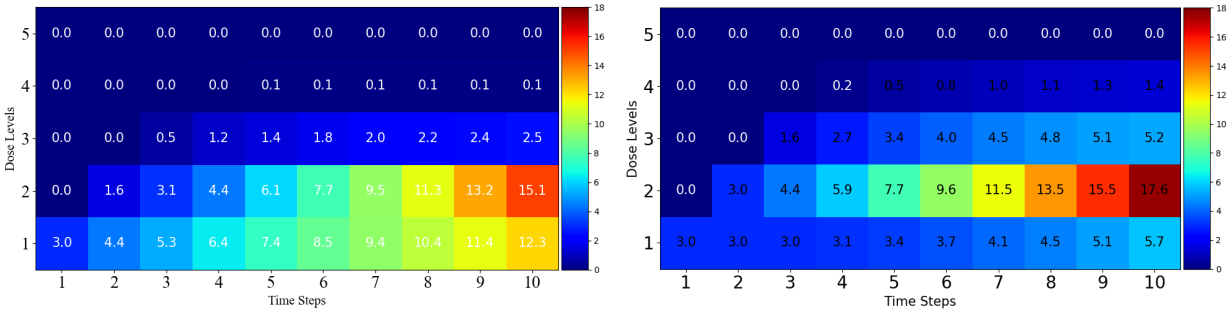
C. Fixed scenarios

We first evaluate our model with five representative fixed scenarios (i.e., the toxicity probabilities of all arms are prefixed), where the MTD is located at different dose levels. For each scenario, we carry out 10000 simulated trials. The result is shown in Table I, where we can see that our DDQN approach yields the best performance in four out of the five scenarios and it achieves the best overall performance. Another observation is that both the CRM and DDQN approaches avoid selecting a dose level whose toxicity probability is much higher or much lower than the toxicity target. This indicates that by taking the monotonic relationship into consideration, the statistical model can reduce the risk of making extremely poor decisions. As shown in Table I, the overall performances of both UCB-E and APT algorithms are rather poor compared with others, possibly due to the following two reasons. First, the total number of arm pulls is small in the BAL problem, i.e., the trial budget is rather limited. For example, in the dose-finding problem, a typical phase I trial has a small sample size, which is often as low as 30 to 50 patients [12]. With such a limited budget, it requires that the algorithm should be able to locate the target with high efficiency and accuracy. However, typical MAB algorithms are designed to converge after a large number of pulling rounds, and their capacity is limited to dealing with such low-budget situations. Second, for the traditional MAB algorithms, there is no mechanism to account for the monotonic relationship specified in the BAL



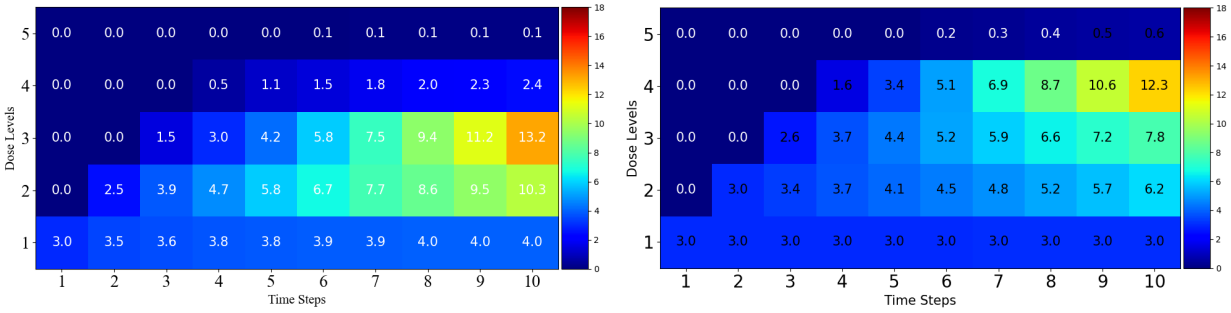
(a) CRM with MTD selection 70.2% ($k^* = 1$)

(b) DDQN with MTD selection 55.3% ($k^* = 1$)



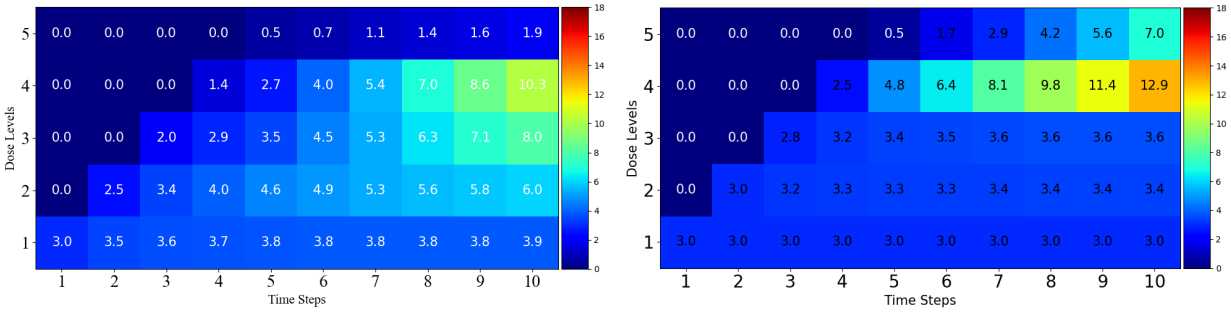
(c) CRM with MTD selection 66.8% ($k^* = 2$)

(d) DDQN with MTD selection 76.1% ($k^* = 2$)



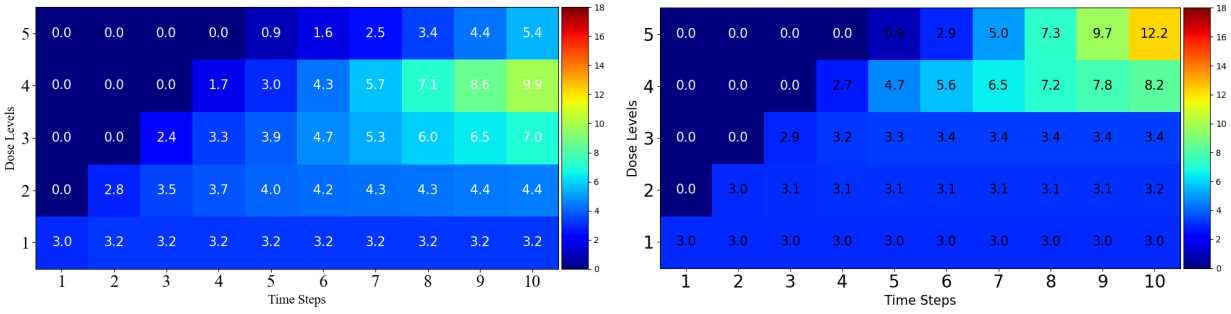
(e) CRM with MTD selection 66.7% ($k^* = 3$)

(f) DDQN with MTD selection 72.6% ($k^* = 3$)



(g) CRM with MTD selection 60.3% ($k^* = 4$)

(h) DDQN with MTD selection 64.1% ($k^* = 4$)



(i) CRM with MTD selection 35.6% ($k^* = 5$)

(j) DDQN with MTD selection 47.5% ($k^* = 5$)

Fig. 2. Average number of patients allocated at each dose level over 10000 simulations under the five fixed dose–toxicity scenarios. Sub-figures (a, c, e, g, i) are results from CRM and sub-figures (b, d, f, h, j) are results from DDQN, corresponding to fixed scenarios 1–5.

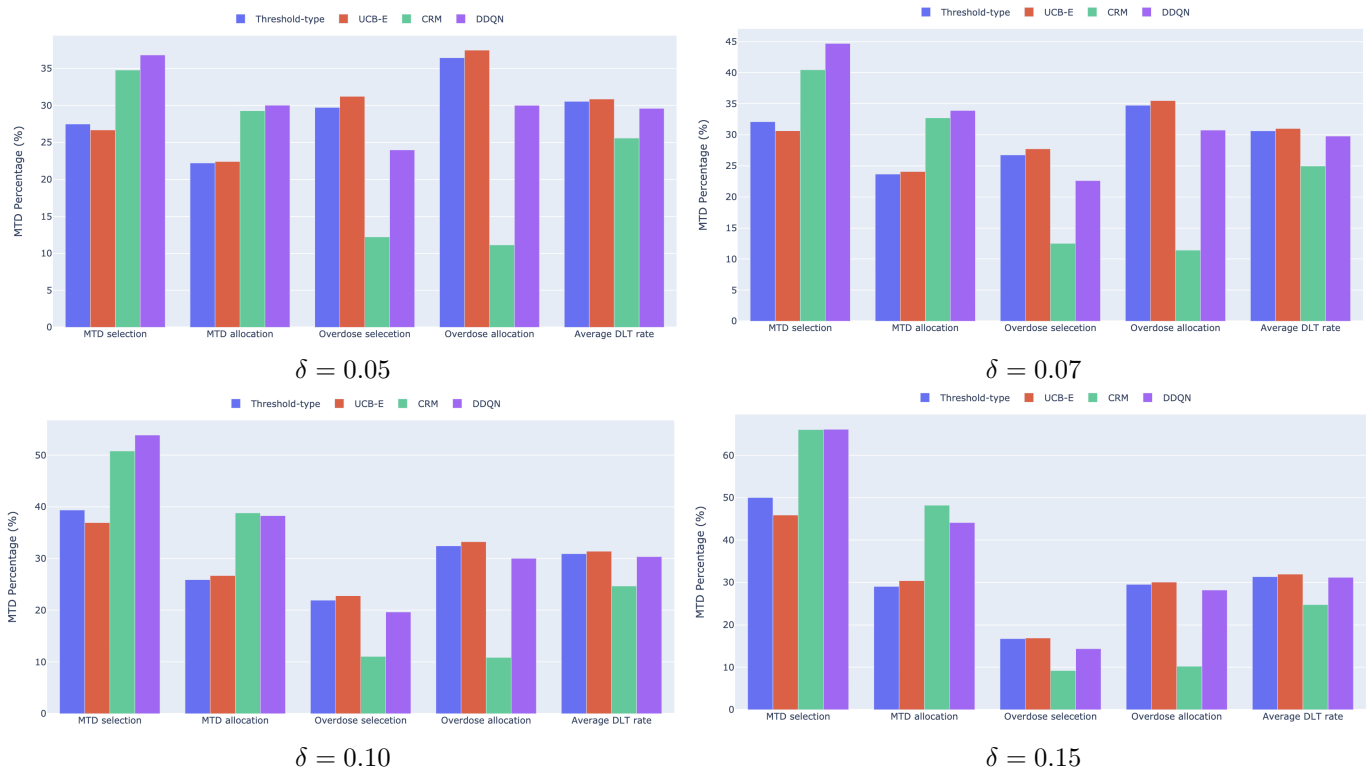


Fig. 3. Simulation results for the MTD identification based on 5000 randomly generated dose–toxicity scenarios with the average probability difference of $\delta = 0.05$, 0.07, 0.10 and 0.15 around the target toxicity probability $\phi = 0.3$, respectively.

problem, which also leads to inefficiency in identifying the target arm.

We also show the average number of patients allocated at each dose level over 10000 simulations under the fixed dose–toxicity scenarios in Figure 2. Overall, the CRM is more conservative compared with the DDQN as it tends to assign more patients to the dose levels lower than the target level. This is understandable because the CRM is a myopic design which only focuses on the best estimation so far with no mechanism to explore new arms (i.e., CRM is less adventurous). While the DDQN is more aggressive due to its exploitation and exploration features, it typically assigns more patients to the target dose level (referring to scenarios 2, 3, 5), which is more desirable in practice. As expected, the DLT percentage under the DDQN is slightly higher than that under the CRM, because the DDQN explicitly employs exploration of untried arms and thus it can locate the MTD more accurately but is slightly riskier.

D. Random scenarios

To avoid cherry-picking scenarios, we further evaluate our model under randomly generated scenarios following the approach of [36], where the target response rates are located at various positions. To evaluate the performances of different approaches under various degrees of the problem difficulty, we set the average probability difference between the target dose and the adjacent doses to be $\delta = \{0.05, 0.07, 0.10, 0.15\}$, respectively. Clearly, the case with $\delta = 0.05$ is the most difficult situation due to the small gaps of the adjacent doses from the target, and the one with $\delta = 0.15$ is the easiest situation. For each setting, we generate 5000 random scenarios. The simulation results in Figure 3 show the percentages of MTD selection and MTD allocation under different methods. We can see that the DDQN approach outperforms the other three methods under the first three settings with small δ, where it is relatively difficult to identify the target arm. Similar to the results under the fixed scenarios, the performances of the adapted bandit approaches are rather poor, which implies the importance of taking the monotonic constraint

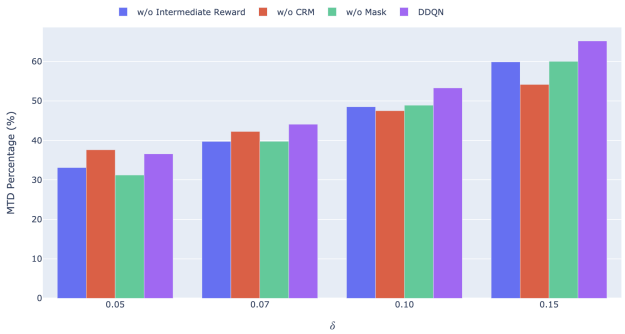


Fig. 4. MTD selection percentages in the ablation studies when ablating the intermediate rewards, CRM and mask operation modules based on 20000 randomly generated dose–toxicity scenarios with different δ's. Our DDQN performs the best with the highest MTD selection percentages.

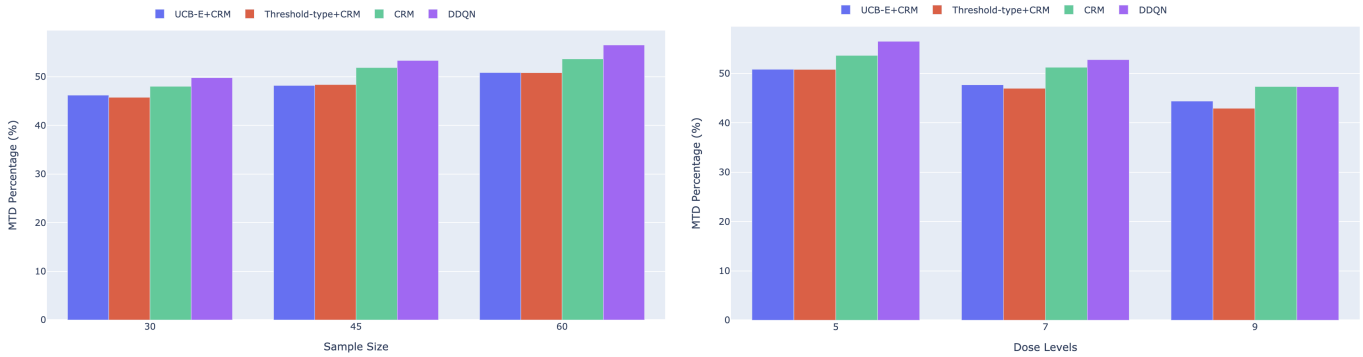


Fig. 5. MTD selection percentages based on 20000 randomly generated dose–toxicity scenarios with different sample sizes and different numbers of dose levels.

Fig. 6. MTD selection percentages based on 20000 randomly generated dose–toxicity scenarios with different target toxicity rates.

into consideration in the design. We also show other three measurements related to the safety aspects of a trial in Figure 3, including the percentage of trials that select overdoses as the MTD, the percentage of patients allocated to overdoses, and the percentage of patients experiencing DLT. Obviously, the CRM leads to the lowest risk compared with other methods, while the DDQN method also yields satisfactory performance in terms of the three safety measurements.

E. Ablation study

To further evaluate different components' contributions to the performance in our DDQN, we carry out the ablation study as follows. We evaluate the importance of each component by removing it from our method and rerun the experiments with the same setting as the original one. We choose three key components for the ablation study: the intermediate rewards, the CRM component, and the mask operation.

Intermediate rewards: We set the intermediate reward to be zero while keeping the final reward the same as the original setting. As seen from Figure 4, experiments without intermediate rewards yield inferior results under all four cases with different δ's. This phenomenon implies that intermediate rewards can encourage the model to learn the parameters in DDQN more effectively during the dose escalation.

CRM: We also run experiments without the CRM module. In this setting, we adopt the same Q-network to determine the dose level for the patient assignment as well as estimating the final MTD for the trial. Except for the case with $\delta = 0.05$, i.e, the most difficult scenario, the performance without the CRM module is also worse than that of the original DDQN method, which implies the importance of the CRM module under general scenarios. When the scenario is extremely difficult (e.g., $\delta = 0.05$ so that the neighboring doses are clustered near the target), the CRM module undermines the performance due to the model misspecification.

Mask operation: In dose-finding clinical trials, dose skipping is typically not allowed during dose escalation or de-escalation for safety reasons. When assigning a dose to each new cohort of patients, we change the dose level by at most one level only. To achieve such constrained dose movement, we introduce a mask operation for the training algorithm. More specifically, we only compare the Q-values of the neighbors of the previously selected dose level. We introduce a binary vector, containing the information about which actions are available. For example, if the previously selected dose level is the third one, then the binary vector would be [0, 1, **1**, 1, 0], which indicates that only the second, third and fourth dose levels are available to be selected for the next Q-network. In Line 10 of Algorithm 1, we can instead use the following formula with the mask operation to select the best dose level,

$$a_t = \arg \max_{1 \leqslant a \leqslant K} \frac{M_a \exp\{Q(s_t, a)\}}{\sum_{a=1}^{K} \exp\{Q(s_t, a)\}}.$$

where M_a is a binary mask for the a-th dose level. A similar mask operation is utilized in Line 25 of Algorithm 1 as well. We run experiments without the mask operation to validate its effectiveness. For a fair comparison, we achieve the same purpose in the experiments without the mask operation by an alternative implementation as follows. We allow the Q-network to select the best action (dose level) but only move one dose level at most when deciding the dose level for the next cohort of patients. The results in Figure 4 demonstrate the effectiveness of the mask operation across the four settings with different δ's.

F. Generalization

To validate the effectiveness of our approach under different settings, we consider different sample sizes, different toxicity rates, and different numbers of dose levels. We run experiments for three strong baseline methods, i.e., CRM, UCB-E+CRM, and Threshold-type+CRM. To take advantage of the monotonic relationship, we use the CRM for the final MTD estimation under UCB-E+CRM and Threshold-type+CRM.

For the experiments with different numbers of dose levels, we set the target toxicity rate as 0.3 with sample size 60. To explore different target toxicity rates, we fix the sample size as 30 and the number of dose levels as 5. Under different sample sizes, the target rate is set as 0.3 with 5 dose levels. Figures 5–6 show that our DDQN approach outperforms the three baseline methods across all the settings. As the sample size increases, the gaps between our DDQN and other methods become larger, which indicates our method is more efficient in utilizing the available information.

VI. Discussion

In traditional RL, it is relatively easier to train an agent because the reward and environment are well defined in comparison with BAL. The environment of BAL is more uncertain because there are an infinite number of dose–toxicity random scenarios with MTD located at different positions or sometimes the MTD does not even exist. Our work is the first attempt to solving dose finding problems using RL, which shows promising results in phase I clinical trials. Due to the large parameter space of the deep Q-learning and the limited sample size of each trial, our method may not be robust enough to handle all possible scenarios in dose finding.

References

[1] W. H. Press, "Bandit solutions provide unified ethical models for randomized clinical trials and comparative effectiveness research," *Proceedings of the National Academy of Sciences*, vol. 106, no. 52, pp. 22 387–22 392, 2009.

[2] M. Aziz, E. Kaufmann, and M.-K. Riviere, "On multi-armed bandit designs for dose-finding clinical trials," *Journal of Machine Learning Research*, vol. 22, no. 14, pp. 1–38, 2021.

[3] T. Chen, A. Gangrade, and V. Saligrama, "Strategies for safe multi-armed bandits with logarithmic regret and risk," *arXiv preprint arXiv:2204.00706*, 2022.

[4] C. Yu, J. Liu, S. Nemati, and G. Yin, "Reinforcement learning in healthcare: A survey," *ACM Computing Surveys (CSUR)*, vol. 55, no. 1, pp. 1–36, 2021.

[5] G. Yin, *Clinical trial design: Bayesian and frequentist adaptive methods*. John Wiley & Sons, 2012, vol. 876.

[6] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[7] O. Besbes and A. Zeevi, "Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms," *Operations Research*, vol. 57, no. 6, pp. 1407–1420, 2009.

[8] A. Slivkins, "Introduction to multi-armed bandits," *arXiv:1904.07272*, 2019.

[9] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, pp. 285–294, 1933.

[10] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT press, 2018.

[11] J. O'Quigley, M. Pepe, and L. Fisher, "Continual reassessment method: A practical design for phase 1 clinical trials in cancer," *Biometrics*, vol. 46, no. 1, pp. 33–48, 1990.

[12] G. Yin and Y. Yuan, "Bayesian model averaging continual reassessment method in phase i clinical trials," *Journal of the American Statistical Association*, vol. 104, no. 487, pp. 954–968, 2009.

[13] H. Jin and G. Yin, "Cfo: Calibration-free odds design for phase i/ii clinical trials," *Statistical Methods in Medical Research*, p. 09622802221079353, 2022.

[14] H. Jin, W. Du, and G. Yin, "Approximate bayesian computation design for phase i clinical trials," *Statistical Methods in Medical Research*, p. 09622802221122402, 2022.

[15] B. E. Storer, "Design and analysis of phase i clinical trials," *Biometrics*, pp. 925–937, 1989.

[16] H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol. 58, no. 5, pp. 527–535, 1952.

[17] A. Badanidiyuru, R. Kleinberg, and A. Slivkins, "Bandits with knapsacks," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. IEEE, 2013, pp. 207–216.

[18] L. Tran-Thanh, A. Chapman, E. M. de Cote, A. Rogers, and N. R. Jennings, "Epsilon–first policies for budget-limited multi-armed bandits," in *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[19] L. Tran-Thanh, A. Chapman, A. Rogers, and N. R. Jennings, "Knapsack based optimal policies for budget-limited multi-armed bandits," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[20] J.-Y. Audibert, S. Bubeck, and R. Munos, "Best arm identification in multi-armed bandits." in *COLT*, 2010, pp. 41–53.

[21] A. Locatelli, M. Gutzeit, and A. Carpentier, "An optimal algorithm for the thresholding bandit problem," in *International Conference on Machine Learning*. PMLR, 2016, pp. 1690–1698.

[22] E. Kaufmann, O. Cappé, and A. Garivier, "On the complexity of best-arm identification in multi-armed bandit models," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1–42, 2016.

[23] V. Gabillon, M. Ghavamzadeh, A. Lazaric, and S. Bubeck, "Multi-bandit best arm identification," in *Proceedings of the 24th International Conference on Neural Information Processing Systems*, 2011, pp. 2222–2230.

[24] S. S. Villar, J. Bowden, and J. Wason, "Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges," *Statistical Science*, vol. 30, no. 2, p. 199, 2015.

[25] S. S. Villar and W. F. Rosenberger, "Covariate-adjusted response-adaptive randomization for multi-arm clinical trials using a modified forward looking gittins index rule," *Biometrics*, vol. 74, no. 1, pp. 49–57, 2018.

[26] P. Whittle, "Multi-armed bandits and the gittins index," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 42, no. 2, pp. 143–149, 1980.

[27] D. Faries, "Practical modifications of the continual reassessment method for phase i cancer clinical trials," *Journal of Biopharmaceutical Statistics*, vol. 4, no. 2, pp. 147–164, 1994.

[28] S. Piantadosi, J. Fisher, and S. Grossman, "Practical implementation of a modified continual reassessment method for dose-finding trials," *Cancer Chemotherapy and Pharmacology*, vol. 41, no. 6, pp. 429–436, 1998.

[29] J. R. Norris and J. R. Norris, *Markov chains*. Cambridge University Press, 1998, no. 2.

[30] R. Bellman, "A markovian decision process," *Indiana Univ. Math. J.*, vol. 6, pp. 679–684, 1957.

[31] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv:1312.5602*, 2013.

[32] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[33] S. M. Lee and Y. K. Cheung, "Calibration of prior variance in the bayesian continual reassessment method," *Statistics in Medicine*, vol. 30, no. 17, pp. 2081–2089, 2011.

[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[35] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[36] X. Paoletti, J. O'Quigley, and J. Maccario, "Design efficiency in dose finding studies," *Computational Statistics & Data Analysis*, vol. 45, no. 2, pp. 197–214, 2004.