applied sciences

MDPI

Article

Multi-Objective Deep Reinforcement Learning for Personalized Dose Optimization Based on Multi-Indicator Experience Replay

Lin Huo [1,*,†] and Yuepeng Tang [2,†]

[1] International College, Guangxi University, Nanning 530000, China
[2] School of Computer and Electronic Information, Guangxi University, Nanning 530000, China
* Correspondence: lhuo@gxu.edu.cn; Tel.: +86-1867-719-0090
† These authors contributed equally to this work.

Abstract: Chemotherapy as an effective method is now widely used to treat various types of malignant tumors. With advances in medicine and drug dosimetry, the precise dose adjustment of chemotherapy drugs has become a significant challenge. Several academics have investigated this problem in depth. However, these studies have concentrated on the efficiency of cancer treatment while ignoring other significant bodily indicators in the patient, which could cause other complications. Therefore, to handle the above problem, this research creatively proposes a multi-objective deep reinforcement learning. First, in order to balance the competing indications inside the optimization process and to give each indicator a better outcome, we propose a multi-criteria decision-making strategy based on the integration concept. In addition, we provide a novel multi-indicator experience replay for multi-objective deep reinforcement learning, which significantly speeds up learning compared to conventional approaches. By modeling various indications in the body of the patient, our approach is used to simulate the treatment of tumors. The experimental results demonstrate that the treatment plan generated by our method can better balance the contradiction between the tumor's treatment effect and other biochemical indicators than other treatment plans, and its treatment time is only one-third that of multi-objective deep reinforcement learning, which is now in use.

Keywords: chemotherapy; multi-objective reinforcement learning; multi-criteria decision making; personalized dosing

check for updates

Citation: Huo, L.; Tang, Y. Multi-Objective Deep Reinforcement Learning for Personalized Dose Optimization Based on Multi-Indicator Experience Replay. *Appl. Sci.* **2023**, *13*, 325. https://doi.org/10.3390/app13010325

Academic Editor: Jose-Maria Buades-Rubio

Received: 5 November 2022
Revised: 20 December 2022
Accepted: 22 December 2022
Published: 27 December 2022

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Cancer is a broad term for a group of malignant tumors, and the number of tumor types included has surpassed 100. Malignant tumors are prone to recurrence and metastasis, and many malignant tumors develop quickly and can cause serious diseases or death by spreading throughout the body. Cancer will be the second most common cause of death worldwide, with an estimated 18.1 million new cases and 9.6 million deaths in 2018 [1]. Every year, the complex and widespread disease of cancer places a tremendous economic burden on nations all over the world, and many patients struggle to pay for costly treatment. For example, one out of every two cancer patients in China borrowed money for treatment. In total, 52% of cancer patients surveyed said that they were in financial difficulties as a result of the disease, 44% said they only had financial difficulties in borrowing money for cancer treatment, 8% said they not only experienced borrowing money for cancer treatment, but they also had to discontinue treatment due to a lack of funds, and 1 out of every 5.5 patients who borrowed money for medical treatment borrowed more than 50,000 RMB [2]. The economic burden of malignancies has cost China $221.4 billion in 2015, accounting for 17.7% of government public health care spending [3]. The majority of cancers that are detected early can be cured with treatments such as surgery [4], radiotherapy [5], or chemotherapy [6]. Based on these considerations, individualized dosing is required for both patients and countries.

Designing drug dosing regimens using mathematical models of cancer is one way to address the problem of personalized dosing. Scholars have been studying mathematical models of cancer since their inception. Scholars strive for accurate mathematical models of tumors because mathematical modeling can be used to understand the changing processes of many complex problems. Mathematical models of cancer propose specific indicators to evaluate the progression of the disease, such as the quantity of tumor cells [7] or the size of the tumor [8], and they also model additional biological indicators that can affect the tumor's progression or that are pertinent to the patient. The statistics have shown that the results of mathematical models are frequently congruent with the results of clinical trials. In order to create the best medicine dosing regimens, we will use these mathematical models, which have been tested and have high economic value. Patients may be provided various doses of chemotherapy drugs based on the equipment and knowledge of the local medical institution. On the basis of the Riccati Equation, Cimen et al. [9] proposed an ideal optimization method for optimizing the medicine dose. Johanna et al. [10] proposed a controller based on linear state feedback, in which the tumor volume can be constantly decreased up to the final stable state. Batmani et al. [11] introduced a patient state feedback-based particular controller that determines the ideal dosage of chemotherapy medications by choosing the proper states and weights in the cost function. Valle et al. [12] devised individual-specific treatment protocols based on LaSalle's invariance principle, and used the localization method of compact invariant sets to determine the upper and lower bounds of cell populations. In order to stop the growth of tumors, Sharifi M. et al. [13] proposed a compound adaptive controller. Shindi et al. [14] proposed a method that combines swarm intelligence algorithms and optimal control to eliminate tumors. Singha [15] proposed to apply the Adomian decomposition method to solve the optimal control problem, and numerically analyzed the solution of the mathematical model. Das et al. [16] established a theory based on quadratic control to optimize medicine dose in treating patients in a way that minimizes toxic damage. Dhanalakshmi et al. [17] used the Gronwall inequality and Lyapunov stability to develop an optimal controller for cancer that can effectively cope with fuzzy mathematical models.

However, open-loop control methods, which are frequently used in treatment protocols developed using mathematical theory, demand exact mathematical models to produce outstanding outcomes [18,19]. In order to generate treatment protocols, the mathematical models must satisfy a number of theorems. These problems determine that using these methods in a clinical situation diminishes their effectiveness [8]. Moreover, many treatment regimens require patients to take chemotherapy medications on a daily basis without taking into account a particular amount of rest time, i.e., intermittent dosing; and, finally, the minimal dose of chemotherapy drugs is impacted by the equipment of the local health facilities, and may not achieve the best results for some dosing regimens. We therefore need a method that can generate a treatment plan with acceptable efficacy for multiple discrete doses.

Artificial intelligence is becoming a part of everyday life thanks to the information era. Among these, reinforcement learning (RL) is a well-liked technology with applications in a variety of industries, including game playing [20], autonomous vehicles [21], and network deployment optimization [22], among others. A group of methodological frameworks for learning, predicting, and making decisions are referred to as RL. If a problem can be formulated as a sequential choice problem or transformed into one, RL can either solve it or offer a better suitable solution. Because it continuously seeks better feedback or lessens negative input, RL is well suited to solving sequential challenge [23]. Eventually, this results in a pattern of activities that optimizes the cumulative reward. Other machine learning algorithms tend to evaluate only the current best solution, whereas RL considers the long-term reward and is not limited to the current optimal solution, making it more suitable for real-world applications. The personalized dosage is a sequential decision problem, where RL chooses various actions depending on the environment to treat patients with various physical states and biological indicators in order to maximize the long-term benefits.

With the use of this characteristic, dynamic treatment regimens may be created that accurately adjust drug dosages in response to the patient's many biological indicators, the patient's underlying disease, and the change of the patient's cancer status. For individualized medicine dosing, RL has been deeply researched. Zhao et al. [24] first applied RL to chemotherapeutic drug dose optimization based on mathematical models of tumors, and placed restrictions on drug toxicity. Padmanabhan et al. [25] defined the state of the tumor based on the number of tumor cells, developed the appropriate treatment plan for the patient using the Q-learning algorithm, and reduced the toxicity caused by the drug by adjusting the maximum dose of the drug. In order to optimize the use of temozolomide, Yauney et al. [26] added an extra reward to the RL framework, using the extra reward to constrain the toxicity of the treatment protocol. Yazdjerdi et al. [8] used the Q-learning algorithm to administer patients' daily doses of anti-vascular treatment medication, which lowered the tumor volume and the endothelial volume of the tumor. Zade et al. [27] validated the high efficacy of the treatment regimen produced by the algorithm by using the Q-learning algorithm to optimize the timing and dose of temozolomide, comparing it with state-of-the-art traditional treatments. Ebrahimi et al. [28] used the RL framework to design the optimal strategy for radiation therapy, and validated the effectiveness of the controller using the case of non-small cell lung cancer. Adeyiola et al. [29] proposed to model tumors using Markov decision-making to optimize drug doses; this method of modeling has some subjective elements and may not objectively evaluate the patient's status. Chamani et al. [30] proposed conservative Q-Learning (CQL) for determining the initial drug dose and evaluating the final drug dosage using clinical experience of professionals.

However, RL, as a popular approach to cope with the sequence problem, is particularly good at optimizing specific goals, but when optimizing medicine dosage, a multi-objective problem occurs. If simply tumor suppression or elimination is desired, the medication concentration in the patient's body will be high, causing the normal cell population to be similarly removed and the immune system to be compromised, perhaps leading to fungal or viral infections. Severe organ damage in the patient's body is another risk associated with high medication concentrations. In order to achieve a balance between multiple objectives in optimization, we should not only focus on eliminating tumors as the end goal, but also consider the variables that may have an impact on the patient's quality of life while undergoing treatment [31,32]. Patients should therefore receive a multi-objective optimized treatment plan [33].

In this paper, we proposed a Multi-Objective Deep Q-Network based on Multi-Indicator Experience Replay (MIER-MO-DQN) to address the problem of exact medicine dose regulation. First, we provide a mixed evaluation model based on the integration principle, which, by adding two different nonlinear decision techniques on the conventional multi-objective decision methods, enables multi-objective deep reinforcement learning to make more logical decisions. Second, in contrast to the conventional experience replay, we design a novel experience replay for multi-objective deep reinforcement learning that takes into account more indications and establishes a multi-objective format in order to increase the network's stability and speed of convergence. Third, while still preserving the level of health that the patient considers acceptable, our algorithm is capable of choosing the most effective treatment plan out of a variety of drug doses for patients with various physical conditions. Our method performs better when compared to existing multi-objective deep reinforcement learning methods.

This paper is organized as follows, with Section 2 detailing the mathematical model of cancer chemotherapy. In Section 3, the detailed MIER-MO-DQN implementation and modeling problem are presented. The experimental results are presented in Section 4, along with a discussion of the suggested model's robustness. Section 5 concludes our study.

2. Mathematical Model of the Tumor

A mathematical model of cancer was put forth by Pillis et al. [7]. Tumor cells, effector-immune cells, circulating lymphocytes, and the concentration of chemotherapeutic drugs

are the primary elements in this mathematical model. Once the initial values of each element have been established, the model uses a series of ordinary differential equations to simulate the changes that each modeled element produces over time, as shown in (1)–(4).

$$\frac{dX_1}{dt} = aX_1(1 - bX_1) - cX_1X_2 - K_xX_1X_4 \tag{1}$$

$$\frac{dX_2}{dt} = d - eX_2 + g\frac{X_1}{h + X_1}X_2 - fX_1X_2 - K_uX_2X_4 \tag{2}$$

$$\frac{dX_3}{dt} = k - lX_3 - K_zX_3X_4 \tag{3}$$

$$\frac{dX_4}{dt} = -\vartheta X_4 + I(t) \tag{4}$$

where X_1, X_2, X_3, and X_4 stand for the quantity of tumor cell populations, the quantity of effector-immune cells, the quantity of circulating lymphocytes, and the chemotherapeutic drug concentration, respectively. t stands for time. The effector-immune cells and drug concentration both reduce X_1 in (1) by two polynomials, cX_1X_2 and $K_xX_1X_4$, while X_1 is rising at $aX_1(1 - bX_1)$. In (2), X_2 increases at the rate of d and decreases at the rate of eX_2, while being created and eliminated by $g\frac{X_1}{h + X_1}X_2$ and fX_1X_2, respectively. The drug concentration decreases X_2 at the rate of $K_uX_2X_4$. The variations in the circulating lymphocyte population are described by (3), where X_3 proliferates at a rate of k, naturally dies at a rate of lX_3, and is damaged by the medication at a rate of $K_zX_3X_4$. In (4), ϑ denotes the rate at which a medication degrades in the body, and $I(t)$ is the dose of the drug consumed at time t. The initial values of X_1, X_2, X_3, and X_4 were X_{10}, X_{20}, X_{30}, and X_{40}. Table 1 displays the values of the parameters in (1)–(4) and their associated descriptions [7,12,15,16,34].

Table 1. Parameters of the mathematical equations of the tumor.

Parameters	Description	Values and Units
a	The growth rate of tumor cells	4.31×10^{-1} day^{-1}
b	1/b is the maximum tumor cell carrying capacity	1.02×10^{-14} day^{-1}
c	The killing rate of tumor cells by effector-immune cells	3.41×10^{-10} day^{-1}cells^{-1}
K_x	The killing rate of tumor cells by drug concentration	8×10^{-1} day^{-1}
d	Constant source of effector-immune cells	1.2×10^{4} day^{-1}cells
e	Mortality of effector-immune cells	4.12×10^{-2} day^{-1}
g	Generation rate of effector-immune cells under the influence of tumor cells	1.50×10^{-2} day^{-1}
h	Steepness coefficient of the recruitment of effector-immune cells	2.02×10^{1} cells^{2}
f	Mortality of effector-immune cells under the influence of tumor cells	2×10^{-11} day^{-1}cells^{-1}
K_u	The killing rate of effector-immune cells according to drug concentration	6.00×10^{-1} day^{-1}
k	Constant production of circulating lymphocytes	7.50×10^{8} day^{-1}cells
l	Mortality of circulating lymphocytes	1.20×10^{-2} day^{-1}
K_z	The killing rate of circulating lymphocytes by drugs	6.00×10^{-1} day^{-1}
ϑ	Decay rate of chemotherapy drug	9.00×10^{-1} day^{-1}
$I(t)$	Chemotherapy drug dosage	$0 \leq I(t) \leq 1$

3. Methods and Model

The principles and implementation details of our proposed solution will be described in this section. We first explain Q-Learning, a traditional reinforcement learning approach, and then we propose the MIER-MO-DQN, which is based on Q-Learning. The DQN and network architecture of MIER-MO-DQN are explained in Section 3.2, the multi-objective action decision mechanism of MIER-MO-DQN and the multi-indicator experience replay method applicable to MIER-MO-DQN are detailed in the next two sections, respectively. The final subsection describes the modeling process and results for the personalized dose problem using reinforcement learning as a guide.

3.1. The Q-Learning Algorithm

RL is a significant subfield of machine learning. An agent can earn rewards and improve its own action strategy; the ultimate goal is to select the best actions in order to maximize cumulative rewards by interacting with the environment. The agent is originally given a state s by the environment. The agent decides to take action a in order to maximize the cumulative reward. After completing action a, the environment returns the current state s' and the reward r to the agent. Figure 1 depicts a schematic of RL.

Watkins et al. [35] first proposed the Q-Learning algorithm. The four components of the algorithm are as follows: (S, A, R, Q). S stands for a collection of state spaces, where $S = \{s_1, s_2, \cdots, s_n\}$, and n is the number of states in the state space. The actions that may be performed at a particular state are represented by the action space A. R stands for the reward attained following the completion of an action, also known as r. The action-value function $Q(s, a)$ expresses the potential future benefit that can be attained by taking action a in state s. The total action-value pair is abbreviated as Q. Q can be updated, as shown in (5).

$$Q(s,a) = Q(s,a) + \eta \cdot [r + \gamma \cdot \max_{a' \in A(s')} Q(s',a') - Q(s,a)] \tag{5}$$

where the value of η, the learning rate, ranges from 0 to 1. The importance of r is determined by the value of η. $\gamma \in [0, 1]$ stands for the reward decay rate, and the significance of the anticipated future reward depends on the size of γ. The state that the environment returns to after state s executes action a is called s'.

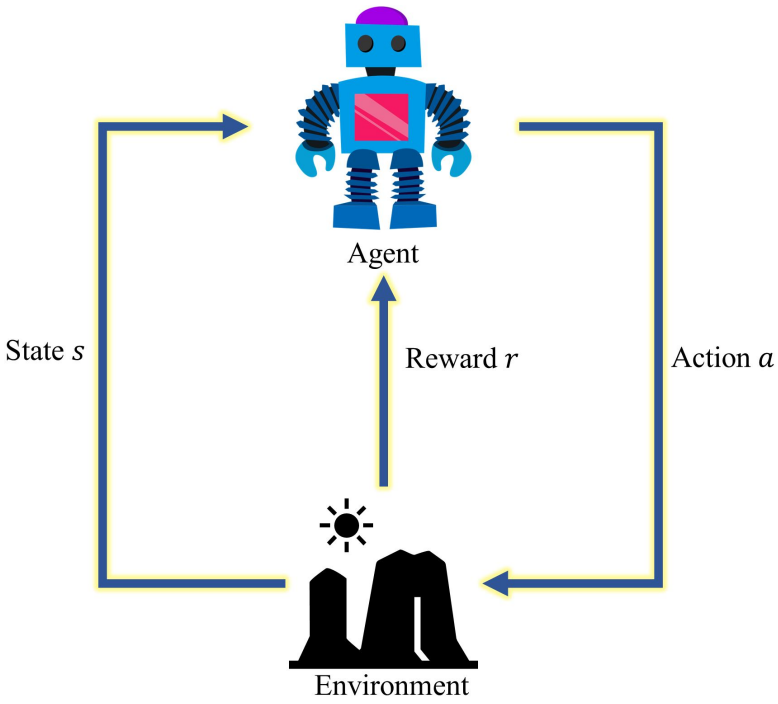

Figure 1. Reinforcement learning schematic.

3.2. The Architecture of MIER-MO-DQN

The Deep Q-Network (DQN) was proposed by Mnih V et al. [36]. It shares many similarities with the Q-Learning algorithm, but differs in that DQN simulates the action-value function $Q(s, a)$ using a neural network. Real-world continuous control problems usually have enormous state spaces and action spaces, making it challenging to find solutions using the Q-Learning algorithm. However, the advent of DQN offers a fresh approach to dealing with such problems. Whether the action-value function $Q(s, a)$ is a scalar or vector determines the difference between the the single-objective DQN method

and the multi-objective DQN algorithm. When $Q(s,a)$ is a scalar, the action with the largest action-value function is carried out by the greedy operator. Selecting the best action becomes a challenge when there are several objects to be optimized. Q will take on the form of a vector, and the action-value function is specified as $\bar{Q}(s,a)$, $\bar{Q}(s,a) = \{Q(s,a,o_1), Q(s,a,o_2), \cdots, Q(s,a,o_m)\}$, where o_i and m stand for object i and the number of objectives, respectively.

The paper creatively proposes a Multi-Objective Deep Q-Network based on multi-indicator experience replay to better resolve the problem of DQN applied to multi-objective optimization, whose network architecture is depicted in Figure 2. To simulate the action-value functions of various objects, we employ a set of DQNs, and each DQN only corresponds to one object. The environment sends the current state s to this group of DQNs, and each of them feeds the Mixed Evaluation Model with the action-value function, in order to determine which action to carry out and to add the experience to the Replay buffer. To reduce the discrepancy between the output of DQN and the true value, the batch of experiences is taken from the Replay buffer, and the Adam optimizer modifies the weights of DQNs.

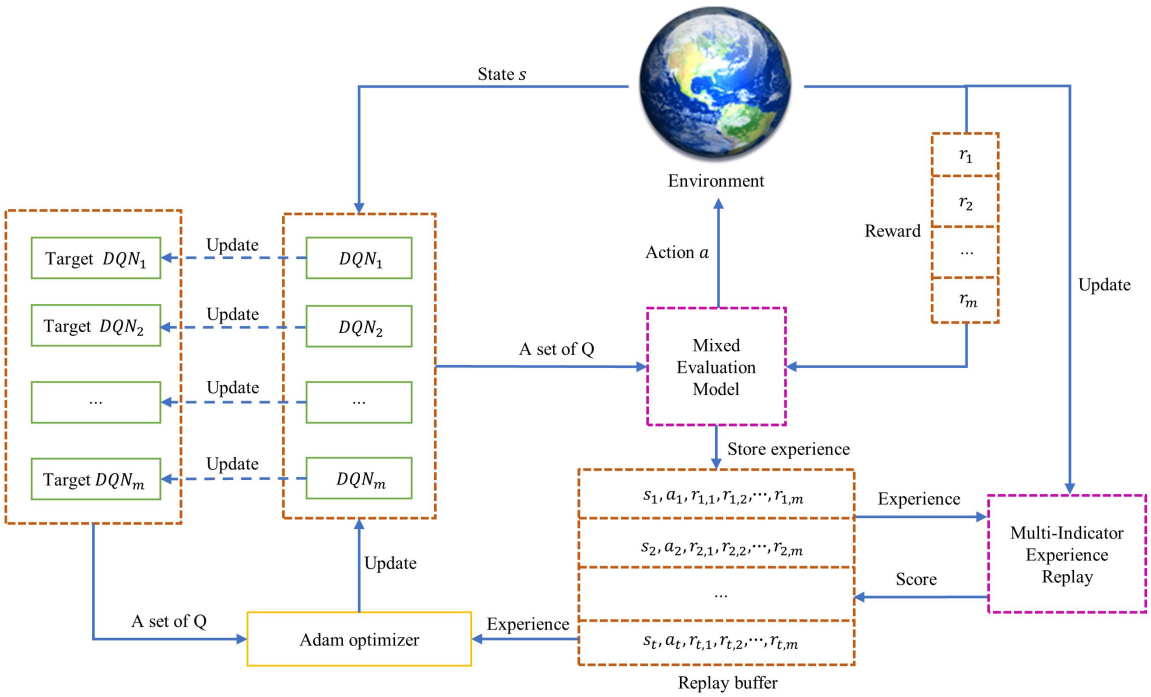

Figure 2. Architecture of MIER-MO-DQN.

3.3. A Mixed Evaluation Model

Figure 3 depicts the action selection procedure with several action-value functions. Each Q-Network is in charge of fitting the action-value functions of the corresponding objects. First off, each Q-Network receives an identical state s from the environment, and outputs to the Mixed Evaluation Model the corresponding action-value function vector $\{Q(s,a_1), Q(s,a_2), \cdots, Q(s,a_z)\}$, where z represents the number of actions and the Mixed Evaluation Model decides the final action to be executed.The two nonlinear ranking methods that are most widely used are TOPSIS and VIKOR. While VIKOR looks for the solution with the greatest group utility and the lowest individual regret value, TOPSIS seeks the compromise solution that is most closest to the ideal option. Given that both approaches have advantages, our model combines them.

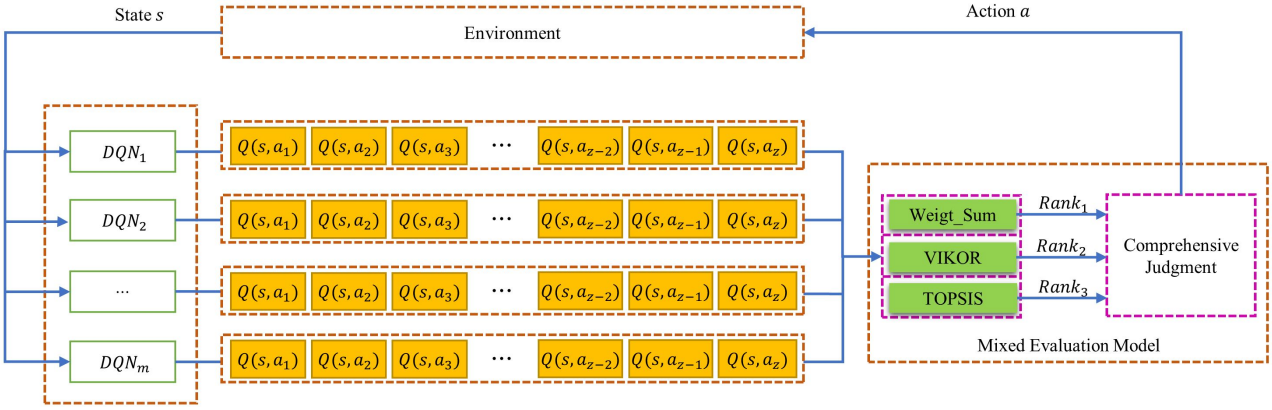

Figure 3. Action selection process.

3.3.1. The Linear Weighted Sum Function

The most popular multi-criteria decision-making approach is the linear weighted sum function [37,38], and its formula f is displayed in (6), where m is the number of objectives, and w_i is the weight of objective o_i. The sum of the weights is 1.

$$f(\bar{Q}(s,a)) = \sum_{i=1}^{m} w_i \cdot Q(s,a,o_i), \sum_{i=1}^{m} w_i = 1. \tag{6}$$

Its simplicity of usage and the capacity to acquire convergence guarantees supported by mathematical theory are its main advantages, but it also has certain drawbacks, and in some complicated conditions, this decision-making process may not produce the best results [39,40]. Because nonlinear functions are more capable than linear weighted sum functions to provide a higher quality solution, they are used to address multi-objective problems. However, nonlinear functions also have the obvious drawback of being susceptible to non-convergence. In this research, a novel multi-objective deep reinforcement learning method based on the integration notion is proposed to address the aforementioned problems. This method integrates linear weighted sum functions and other nonlinear functions.

3.3.2. VIKOR

The first nonlinear evaluation method, the VIKOR, was proposed by Opricovic et al. [41,42]. As indicated in (7), the method first normalizes the Q that is supplied to the model, where z and m stand for the size of the action space and the number of objects, respectively. The greatest and minimum values in the action-value function are calculated by (8)–(11), commonly known as the positive ideal solution (PIS) and the negative ideal solution (NIS). Equations (12) and (13) determine the collective utility value C and the individual regret value I. Lastly, the trade-off index value Y is calculated using (14), and all actions are ranked from smallest to largest according to Y. v defines the weight of the group utility value C and the individual regret value I, which is often set to 0.5.

$$r_{i,j} = \frac{Q(s,a_i,o_j) - \min(Q(s,*,o_j))}{\max(Q(s,*,o_j)) - \min(Q(s,*,o_j))}, i = 1,2,3,\cdots,z, j = 1,2,3,\cdots,m \tag{7}$$

$$PIS = u_1^+, u_2^+, u_3^+, \cdots, u_m^+ \tag{8}$$

$$NIS = u_1^-, u_2^-, u_3^-, \cdots, u_m^- \tag{9}$$

$$u_j^+ = \max(r_{*,j}), j = 1,2,3,\cdots,m. \tag{10}$$

$$u_j^- = \min(r_{*,j}), j = 1,2,3,\cdots,m. \tag{11}$$

$$C_i = \sum_{j=1}^{z} w_j \cdot \frac{u_j^+ - r_{i,j}}{u_j^+ - u_j^-}, i = 1, 2, 3, \cdots, z \tag{12}$$

$$I_i = \max_{1 \le j \le m} w_j \cdot \frac{z_j^+ - r_{i,j}}{z_j^+ - z_j^-}, i = 1, 2, 3, \cdots, z \tag{13}$$

$$Y_i = v \cdot \frac{C_i - \min(C)}{\max(C) - \min(C)} + (1 - v) \cdot \frac{I_i - \min(I)}{\max(I) - \min(I)}, i = 1, 2, 3, \cdots, z \tag{14}$$

3.3.3. TOPSIS

Another multi-criteria decision-making method called TOPSIS can effectively choose the best compromise solution from a group of options [43]. The VIKOR takes into account each action's specific regret value. In comparison, the TOPSIS takes a greater account of group effects and employs the Euclidean distance between the current solution and the positive and negative ideal solutions as a measure to decide the best action. The initial steps of the method are consistent with VIKOR, because TOPSIS also calls for the computation of PIS and NIS. Following the acquisition of PIS and NIS, TOPSIS calculates the Euclidean distance between each solution and the ideal solution according to (15) and (16). Each action is ranked from largest to smallest according to U_i. U_i is obtained from (17).

$$e_i^+ = \sqrt{\sum_{j=1}^{m} w_j \cdot (r_{i,j} - u_j^+)^2}, i = 1, 2, 3, \cdots, z \tag{15}$$

$$e_i^- = \sqrt{\sum_{j=1}^{m} w_j \cdot (r_{i,j} - u_j^-)^2}, i = 1, 2, 3, \cdots, z \tag{16}$$

$$U_i = \frac{e_i^-}{e_i^- + e_i^+}, i = 1, 2, 3, \cdots, z \tag{17}$$

3.3.4. Comprehensive Judgement

The ranks from the linear weighted sum function, VIKOR, and TOPSIS, are set as $rank_1$, $rank_2$, and $rank_3$, correspondingly. The synthetic rank SR_i and the variance P_i of action i are likewise computed, as illustrated in (18) and (19), respectively. Finally, the decision was made to choose the action with the lowest SR among those whose P_i was lower than the mean variance.

$$SR_i = \frac{1}{3} \cdot (rank_{1,i} + rank_{2,i} + rank_{3,i}), i = 1, 2, 3, \cdots, z \tag{18}$$

$$P_i = \frac{1}{3} \cdot ((rank_{1,i} - SR_i)^2 + (rank_{2,i} - SR_i)^2 + (rank_{3,i} - SR_i)^2), i = 1, 2, 3, \cdots, z \tag{19}$$

3.4. Multi-Indicator Experience Replay

The DQN is difficult to converge due to the correlation between decisions and the instability of training labels as the Q-Network is updated, and so experience replay is required to improve the Q-Network's stability. The effect of experience replay is significantly influenced by the experience extraction method, and so we propose an experience replay method that is multi-indicator and suitable for multi-objective optimization. Moreover, the target DQN is used to copy the parameters of the corresponding DQN into their own network after each G round, in order to increase the network's stability.

The main idea of experience replay is to extract a certain number of experiences and to calculate the error between the output value of DQN and the estimated value as the TD error, and then to back propagate it to change the parameters of the neural network to bring it closer to the real action-value function. In general, there are two popular ways to replay memories. One is to choose experiences at random [36]. The another is prioritized experience replay [44], which increases the likelihood of replaying experiences with larger TD errors and gives them more weight. It quantifies each experience's value by using the absolute value of the TD error. However, more elements should be taken into account during the experience replay process, and the criteria for assessment are described below.

3.4.1. TD Error

We should first take the TD error into account. Experiences with greater TD error should be replayed with a higher probability, and as a result, should be given a higher priority. The TD error reflects the difference between the estimated value and the output value of DQN. Each single-objective experience's TD error was computed by (20), and the priority of experience x is $\rho_{TD}(x)$. However, since there are now multiple targets to be optimized, the multi-objective priority of the TD error is calculated by (21), where θ_i and θ_i^- represent the parameters of the DQN network and the target DQN network corresponding to target o_i, respectively.

$$\rho_{TD}(x) = |r_x + \gamma \cdot \max_{a' \in A(s')} Q(s', a'|\theta^-) - Q(s_x, a_x|\theta)| \tag{20}$$

$$\rho_{TD}(x) = \max_{1 \le i \le z} |r_i + \gamma \cdot \max_{a' \in A(s')} Q(s', a'|\theta_i^-) - Q(s_x, a_x|\theta_i)| \tag{21}$$

3.4.2. The Information Entropy of Q

Hannon et al. [45] introduced the concept of information entropy. Information entropy can be used to measure the uncertainty of a random variable or the entire system. A higher entropy indicates a higher level of system uncertainty. The larger the variation of the action-value function for the DQN output, the more uncertain the system is. As a result, one of the indicators for choosing an experience is the information entropy of the action-value function, so that an action-value function with greater information entropy should have a higher chance of replay and a higher priority. Equation (22) calculates the information entropy of the action-value function for object i. Following that, $\rho_{IE}(x)$ is the replay priority of experience x in the multi-objective situation by (23).

$$\rho_{IEi}(x) = -\sum_{n=1}^{z} \frac{|Q(s_x, a_n, o_i|\theta)|}{\sum_{j=1}^{z} |Q(s_x, a_j, o_i|\theta)|} \cdot \log \frac{|Q(s_x, a_n, o_i|\theta)|}{\sum_{j=1}^{z} |Q(s_x, a_j, o_i|\theta)} \tag{22}$$

$$\rho_{IE}(x) = \frac{1}{m} \sum_{i=1}^{m} \rho_{IEi}(x) \tag{23}$$

3.4.3. Number of Replays

To avoid the Replay buffer repeatedly selecting experiences with certain characteristics, the priority of those experiences should be adjusted to a lower position so that the experiences that have not been replayed can be sampled. We define the number of replays as φ and use the Sigmoid function as the evaluation function. The sigmoid function is shown in (24). The priority of the number of replays for experience x is calculated by (25).

$$Sigmoid(\varphi) = \frac{1}{1 + \exp(-\varphi)} \tag{24}$$

$$\rho_{NR}(x) = Sigmoid(\varphi) \tag{25}$$

3.4.4. Number of Experience Storage

The training process of deep reinforcement learning involves constantly adding new experiences and deleting old ones from the replay buffer. More and more repeated experiences will be continuously replayed as the training time increases, especially when these same experiences also rank well in some indicators. Finally, experience replay wastes a lot of time while producing poor outcomes. The quantity of experience x repetitions stored in the replay buffer was set to τ. Rare experiences need greater consideration because they will seldom be uncommonly learned. Equation (26) defines the experience repetition priority $\rho_{NS}(x)$ of the experience x.

$$\rho_{NS}(x) = Sigmoid(\tau) \tag{26}$$

3.4.5. Composite Score

In this study, the TD error, the information entropy of the action-value function, the number of replays and repetitions are used to quantify the worth of each experience. To combine the evaluation of the four indicators, we use (27)–(29) as the integrated evaluation function, and we denote $Score(x)$ as the result of the integrated evaluation function, or the composite score of experience x.

$$Score_1(x) = \frac{\rho_{TD}(x)}{\sum_{i=1}^{MS} \rho_{TD}(i)} + \frac{\rho_{IE}(x)}{\sum_{i=1}^{MS} \rho_{IE}(i)} \tag{27}$$

$$Score_2(x) = \rho_{NR}(x) + \rho_{NS}(x) \tag{28}$$

$$Score(x) = Score_1(x) \cdot Score_2(x) \tag{29}$$

The two components of $Score(x)$ are $Score_1(x)$ and $Score_2(x)$, where MS is the capacity of Replay buffer. A Sum Tree is built by adding the score from each experience to each leaf node of the Sum Tree [44].

3.4.6. Sample Weights

After picking the experiences to be replayed, each experience does not have the same value, and so there may be some departure from the original value distribution. We need to introduce the notion of sample weights to give various weights to the selected samples so that the divergence is corrected. The sample weights are also known as importance weights, and the formula is shown in (30).

$$w_x = (\frac{1}{bs} \cdot \frac{1}{Score(x) / \sum_{i=1}^{MS} Score(i)})^{\beta} \tag{30}$$

The number of experiences extracted from replay buffer is called the batch size, where bs stands for the batch size, and w_x stands for the weight of sample x. β is a hyperparameter that can be used to entirely correct the bias caused by sample priority when it equals 1. To more effectively rectify the bias, the value of β should be raised continually over the training period.

The process of multi-indicator experience replay is shown in Figure 4. It chooses the particular experience to pick based on the score of each experience, and gives weight to these experiences to correct for the bias introduced using this sampling method.

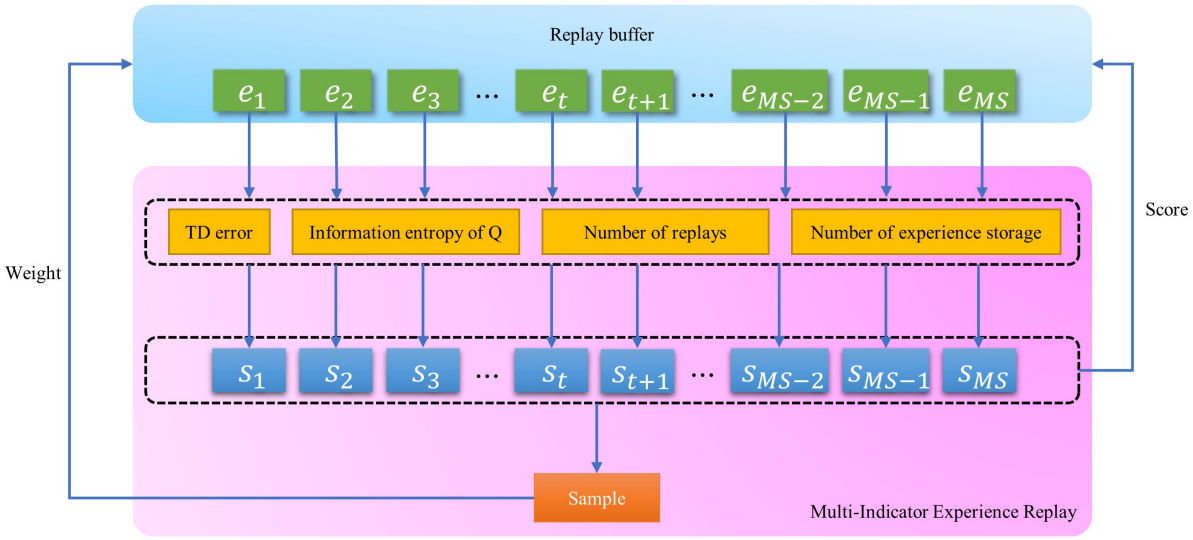

Figure 4. Process of multi-indicator experience replay.

3.5. Model

We utilize a multi-objective DQN as the RL controller to more effectively address the challenge of personalized dose. Since DQN has the ability to handle a number of state spaces and action spaces, its outcome should be improved by more accurately defining the state space. The state of the system is defined as $s(t) = \{s_1(t), s_2(t), s_3(t)\}$, where t denotes time. $s_1(t)$, $s_2(t)$, and $s_3(t)$ are given by (31)–(33).

$$s_1(t) = X_1(tP) - X_{1d} \tag{31}$$

$$s_2(t) = X_2(tP) \tag{32}$$

$$s_3(t) = X_4(tP) \tag{33}$$

where P is the representation of the sample period. Tumor cells, effector-immune cells, and drug concentration are represented by the symbols $s_1(t)$, $s_2(t)$, and $s_3(t)$, respectively. The tumor cells' ideal value, X_{1d}, can also be thought of as the treatment's final state. In accordance with the [7], X_{1d} is set to 1, so the ultimate state at the end of treatment stated to be $s_1(t)$ is less than 0 in this paper. For the drug dose $I(t)$, the restriction was given in Table 1, and the drug dose should be kept within the range of [0,1]. The setting of the action space should be subdivided as much as possible in order to discover the appropriate dose for patients in different states. Hence, in this work, action space $A(*) = [0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1]$, which has satisfied the patient's medicine needs. The adjustment of the drug dose is important because the RL controller's ultimate purpose is to keep the number of effector-immune cells at a high level while treating tumor cells. Because the only external factor that can affect the number of circulating lymphocytes is the drug dose, the number of circulating lymphocytes and the number of effector-immune cells can be regarded as the same goal. The goal of maintaining the number of effector-immune cells in the system is the same as maintaining the number of circulating lymphocytes. The setting for the reward function R is depicted in (34) and (35).

$$r_1 = \frac{X_1(tP) - X_1((t+1)P)}{\max X_1} \tag{34}$$

$$r_2 = \frac{X_2((t+1)P) - X_2(tP)}{\max X_2} + loss(t+1) \tag{35}$$

The rewards from a change in the tumor cell cells and effector-immune cells are represented by r_1 and r_2. The two reward functions are normalized using parameters $\max X_1$

and max X_2, where max X_1 represents the maximum value of tumor cells [7] and max X_2 represents the maximum value of effector-immune cells [12], respectively. A constraint $loss(t)$ is added, since our purpose is to prevent large variations of effector-immune cells in neighboring cycles and guarantee that the patient's effector-immune cell count is greater than the lowest threshold that is acceptable to the patient. The $loss(t)$ was calculated by (36)–(38).

$$loss(t) = \begin{cases} z, & X_2(tP) \leq X_{2tsh} \text{ or } gap \geq 20\% \\ 0, & otherwise \end{cases} \tag{36}$$

$$gap = \frac{X_2((t-1)P) - X_2(tP)}{X_2((t-1)P)} \tag{37}$$

$$z = \begin{cases} -10 \cdot I(t-1), & X_2(tP) \leq X_2((t-1)P) \\ 0, & otherwise \end{cases} \tag{38}$$

where X_{2tsh} is lowest threshold. The agent must make difficult choices from a range of rewards based on the problem modeling in this work, and make sure that the patient's cancer therapy is successful while assuring that the number of effector-immune cells declines within a specific level.

4. Results

In order to demonstrate the performance of the MIER-MO-DQN algorithm used to solve the problem of personalized drug dosing, we treat patients individually using treatment plans obtained from other deep reinforcement learning methods.

4.1. Design of Experimental Parameters

This paper assigns X_{10}, X_{20}, and X_{30} in each iteration to be random values between the upper and lower bounds of the corresponding cell populations, X_{40} is set to zero to simulate more realistic scenarios. MIER-MO-DQN performs better, since the initial states are generated at random. The two distinct therapy scenarios for patients are presented. The more aggressive and milder treatment plans, respectively, are depicted in Scenario 1 and Scenario 2. In Scenario 1, the patient is assumed to be in good health, and X_{2tsh} is set to 50% of the effector-immune cells' initial value. Scenario 2 supposes the patient is in poor health and its X_{2tsh} is set to 70% of the original value of the effector-immune cells. To demonstrate the algorithm's robustness, we performed our proposed algorithm in two scenarios with parameter deviations and illustrated the experimental results. Table 2 displays the parameter values for the experiment. The MIER-MO-DQN is compared with the conventional DQN approach [36] and the linear weighted sum function-based DQN (W_DQN) [46], to confirm the superiority of the treatment plan developed by MIER-MO-DQN based on the patient's biochemical indicators.

The same network architecture is employed by the DQNs to simulate each objective's action-value function, and each DQN comprises two hidden layers with 100 and 120 neurons each. After each hidden layer, the ReLU function is added as the activation function, and the MSE is utilized as the loss function. The $\varepsilon - greedy$ exploration, identical problem modeling, and the same architecture of DQN were applied in all three solutions.

The indicators for the experimental results are based on [7].

Table 2. Design of experimental parameters.

Parameters	Description	Values
Epoch	Number of iterations	5000
Step	Maximum number of steps per round	1000
γ	Reward decay rate	0.7
m	Number of goals	2
ε	Exploration rate	0.1
MS	Replay buffer capacity	10,000
bs	Batch size	64
G	Update frequency of Target network	20
a	Learning rate	0.001
w_i	Weight of each target	$\frac{1}{m}$
max X_1	Maximum value of tumor cells	1×10^{14}
max X_2	Maximum value of effector-immune cells	1×10^{7}

4.2. Scenario 1

The changes in several biological indicators during the course of the treatment according to the chemotherapy regimens created using MIER-MO-DQN and other deep reinforcement learning algorithms are shown in Figure 5a–d. Each cell number in Figure 5 is normalized. Each method adheres to the restrictions of Scenario 1.

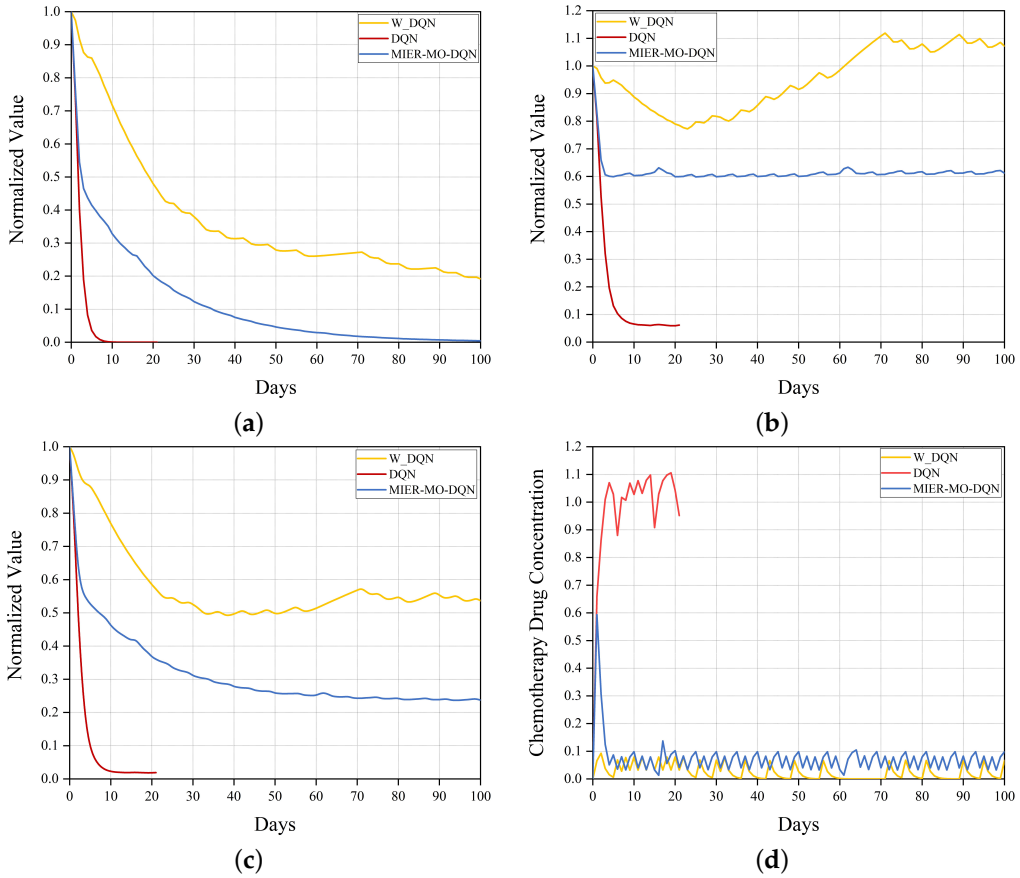

Figure 5. The change of (**a**) tumor cells (**b**) effector-immune cells (**c**) circulating lymphocytes (**d**) drug concentration using different methods in Scenario 1.

The change in the patient's tumor cell count is shown in Figure 5a. The treatment was finished in 22 days due to the DQN algorithm's rapid reduction in tumor cells. In comparison to W_DQN, the MIER-MO-DQN algorithm had a more favorable treatment outcome, with a roughly 18.7% gap in tumor cells at 100 days. Figure 5b,c shows the trends of

effector-immune cell numbers and circulating lymphocyte in patients, respectively. The restrictions of Scenario 1 are violated by the DQN algorithm's treatment plan, which calls for the continuous use of large dosages of chemotherapeutic drugs during therapy, leading to a rapid reduction in effector-immune cells and circulating lymphocytes as well. Though the W_DQN algorithm provided patients with larger quantities of effector-immune cells and circulating lymphocytes than MIER-MO-DQN, it does not pay enough attention to the efficiency of tumor cell reduction, and the poor efficiency of tumor cell decline results in an almost 300% longer treatment time than MIER-MO-DQN. We can observe that both the W_DQN algorithm and MIER-MO-DQN use low drug doses as the main usage regimen from Figure 5d, but MIER-MO-DQN will pursue the tumor treatment effect with high drug doses first and then maintain the patient's other cell counts with low dosages. The drug use regimen for the first 100 days generated by the MIER-MO-DQN algorithm is shown in Figure 6. Table 3 shows that the MIER-MO-DQN algorithm outperforms other deep reinforcement learning algorithms in balancing the treatment of tumor cells and the maintenance of various other cell populations in the body.

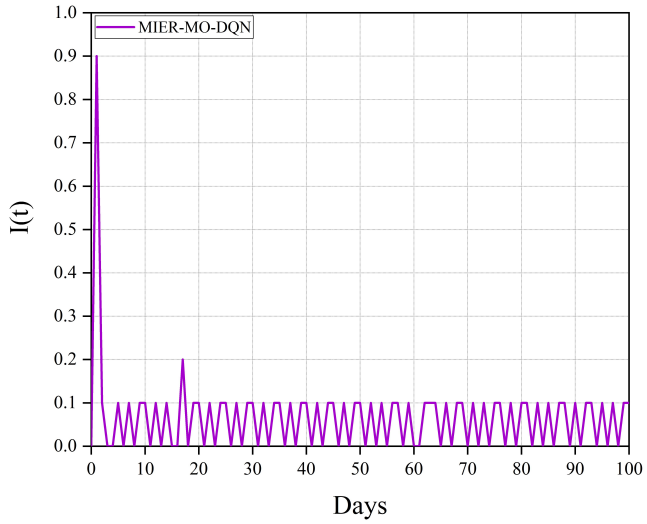

Figure 6. Drug dose of MIER-MO-DQN algorithm in Scenario 1.

Table 3. Comparison of experimental indicators in different methods.

Indicators	MIER-MO-DQN	W_DQN	DQN
Time to completion of treatment	336	1204	22
Value of tumor cells at 50 days	4.6%	27.9%	0.8
Value of tumor cells at 100 days	0.4%	19.1%	0.8
Average value of effector-immune cells	61.4%	98%	18.4%
Minimum value of effector-immune cells	59.7%	78.1%	5.9%
Average value of circulating lymphocytes	26.3%	48%	14.7%
Average value of drug concentration	0.065	0.022	0.96

4.3. Robustness Experiments in Scenario 1

In this experiment, treatment plans are created using the MIER-MO-DQN algorithm by changing all of the parameters in Table 1 while taking Scenario 1's limitations into consideration. The effect of the created treatment regimen on the patient's numerous cell populations is shown in Figure 7.

Figure 7a shows the patient's tumor cell drop trend over time, and the three curves nearly overlap to show the same therapy effect. The three tumor cell counts are 0.4%, 0.7%, and 0.3% of the original levels at 100 days, respectively. The changes in the patient's effector-immune cell and circulating lymphocyte counts are seen in Figure 7b,c; the three curves' trends also show consistency. With very similar results, the gap of variation in

the mean numbers of effector-immune cells and circulating lymphocytes for the treatment regimens produced by the various parameters did not surpass 1%. Figure 7d,e shows that the three regimens treated the patients with almost the same drug dosage. Table 4 displays the experiment's indicators.

4.4. Scenario 2

The drug use plan produced using the MIER-MO-DQN algorithm is displayed in Figure 8; Figure 9a–c depicts the cell populations as they change during the course of treatment using chemotherapy regimens created by MIER-MO-DQN and other deep reinforcement learning algorithms over a period of 100 days. Each regimen adheres to the restrictions of Scenario 2.

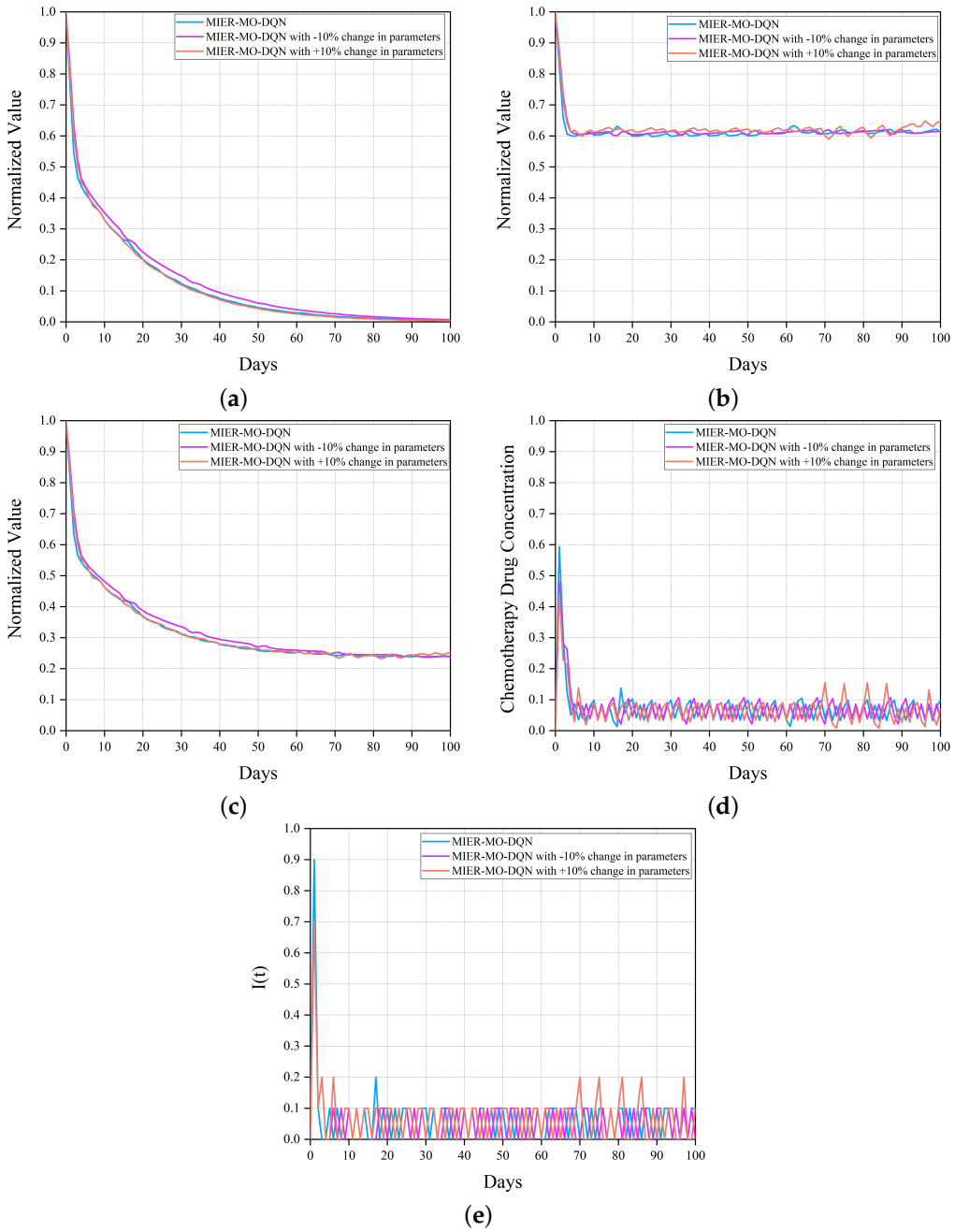

Figure 7. The change of (**a**) tumor cells, (**b**) effector-immune cells, (**c**) circulating lymphocytes, (**d**) drug concentration, (**e**) drug dose, according to different parameters on cells in patients.

The treatment regimens produced by the DQN algorithm are just about identical to those in Scenario 1, since only one objective is optimized, which ignored the effector-immune cells and circulating lymphocytes. Figure 9a shows that there is a higher disparity between MIER-MO-DQN and W_DQN than Scenario 1, with a gap of 31.6% of the initial tumor cells at 100 days. The reason for the less successful tumor treatment is that the number of effector-immune cells in the W_DQN algorithm in Figure 9b is occasionally even higher than the initial value during the treatment. In comparison, even if MIER-MO-DQN's mean value of effector-immune cells is lower than W_DQN, it still meets the minimal requirement of 70%. Figure 9c shows identical results, and the low tumor treatment rate of W_DQN maintains a high mean value of circulating lymphocytes. Figure 9d shows that the W_DQN and MIER-MO-DQN both exhibit nearly the same trend in the amplitude of drug concentration changes, with a difference between the mean values of the drug concentration of 0.028, but the MIER-MO-DQN algorithm is more effective at treating tumors. The comparison of a few experiment-related indicators is shown in Table 5. According to experimental findings, the MIER-MO-DQN algorithm completes patient treatment faster and without breaking any restrictions by achieving a superior trade-off between the effectiveness of tumor treatment and the maintenance of circulating lymphocyte and effector-immune cell populations than other methods.

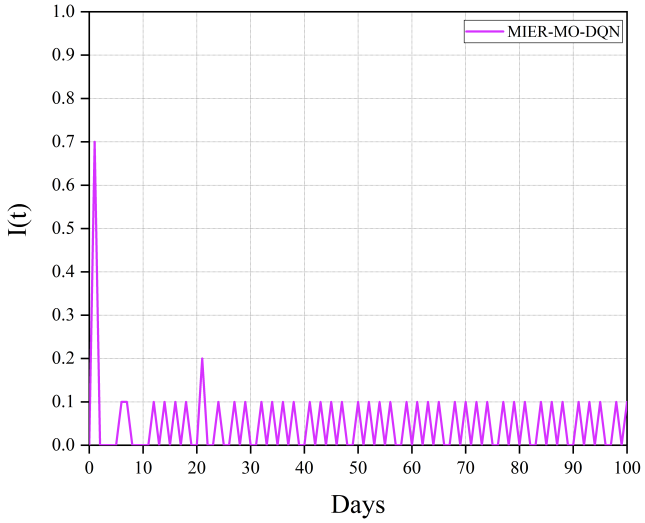

Figure 8. Drug dose of MIER-MO-DQN algorithm in Scenario 2.

Table 4. Comparison of experimental indicators in different parameters.

Indicators	Original Parameter	Parameter with +10%	Parameter with −10%
Time to completion of treatment	336	315	363
Value of tumor cells at 50 days	4.6%	4.3%	5.6%
Value of tumor cells at 100 days	0.4%	0.3%	0.7%
Average value of effector-immune cells	61.4%	61.9%	61.3%
Minimum value of effector-immune cells	59.7%	59.9%	60.1%
Average value of circulating lymphocytes	26.3%	26.6%	26.3%
Average value of drug concentration	0.065	0.062	0.065

Table 5. Comparison of experimental indicators in different methods.

Indicators	MIER-MO-DQN	W_DQN	DQN
Time to completion of treatment	481	1635	22
Value of tumor cells at 50 days	11.9%	55.4%	0.8
Value of tumor cells at 100 days	2.1%	33.7%	0.8
Average value of effector-immune cells	72.6%	104%	18.7%
Minimum value of effector-immune cells	70.3%	93%	6%
Average value of circulating lymphocytes	32.3%	68.4%	15%
Average value of drug concentration	0.047	0.019	0.95

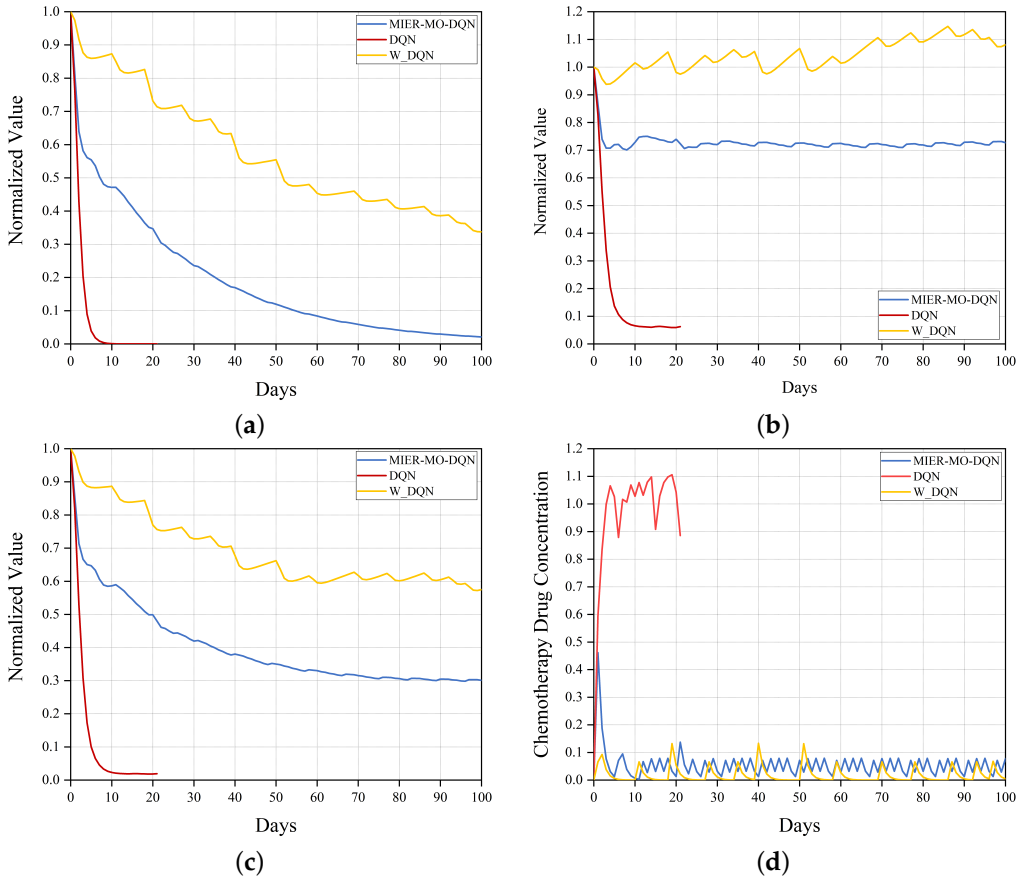

Figure 9. The change of (**a**) tumor cells, (**b**) effector-immune cells, (**c**) circulating lymphocytes, (**d**) drug concentration, according to different methods in Scenario 2.

4.5. Robustness Experiments in Scenario 2

Figure 10 displays the impact on the patient's cells based on the treatment plans created by the three parameters under Scenario 2's limit. The three treatment regimens both used MIER-MO-DQN.

According to the slope of the curve in Figure 10a, all three treatment regimens had virtually the same rate of tumor cell inhibition. Figure 10b shows the patient's change in effector-immune cell count, and the mean value of effector-immune cells differs by 0.9% and 0.3% between the treatment regimen with the original parameters and the other two regimens with changed parameters. The changes in the patient's circulation lymphocyte counts are shown in Figure 10c, and the difference between the three regimens' average circulating lymphocyte counts does not surpass 1%. Figure 10d,e show the drug concentrations and doses given for each of the three dosing regimens, and it can be seen that the doses and usage times for each regimen almost follow the same pattern. Table 6 displays the various experimental indices.

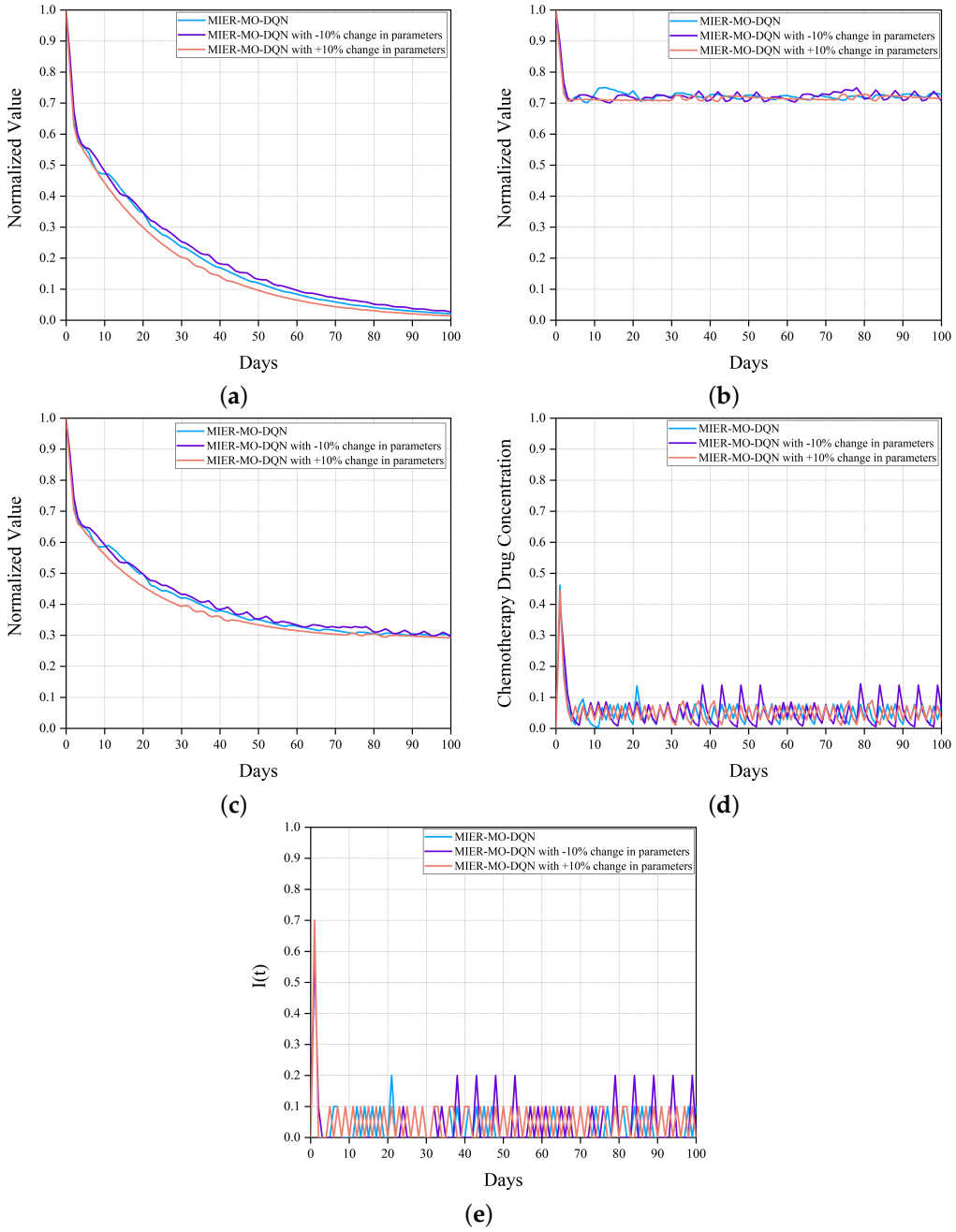

Figure 10. The changes of (**a**) tumor cells, (**b**) effector-immune cells, (**c**) circulating lymphocytes, (**d**) drug concentration, (**e**) drug dose according to different parameters on cells in patients.

Table 6. Comparison of experimental indicators using different parameters.

Indicators	Original Parameter	Parameter with +10%	Parameter with −10%
Time to completion of treatment	481	450	508
Value of tumor cells at 50 days	11.9%	11.1%	13.5%
Value of tumor cells at 100 days	2.1%	1.7%	2.7%
Average value of effector-immune cells	72.6%	71.7%	72.9%
Minimum value of effector-immune cells	70.3%	70.2%	70.1%
Average value of circulating lymphocytes	32.3%	31.8%	31.9%
Average value of drug concentration	0.047	0.048	0.048

4.6. Performance Comparison

In this research, a multi-objective experience replay algorithm based on multiple indicators is proposed to address the problem of the slow convergence of the multi-objective deep reinforcement learning algorithm. We compare the multi-indicator experience replay with the random replay (Random_Replay) [36] under the constraint of Scenario 1 to illustrate the effectiveness of the multi-indicator experience replay applied to multiple objectives. The error between the current value and the predicted value needs to be extracted when adjusting the weights for various objectives of the DQN network, called Loss. Due to the fact that this article has two optimization targets, it will also produce two Losses. We can observe the changing trend of the losses throughout the experiment, and so we are able to roughly predict whether the method is close to convergence and the speed of convergence. We define the Loss of DQN for tumor cells as Loss1, and the loss of DQN for effector-immune cells as Loss2. The changes in Loss1 and Loss2 are shown in Figure 11a,b. It is clear that the MIER-MO-DQN approach exhibits greater fluctuations in the early iterations. This is because the size of the loss is mostly driven by the TD error, and the larger the TD error, the larger the loss. In fact, the multi-indicator experience replay mainly learns from the experience with the larger TD error and the information entropy of Q in the early iterations in order to speed up model convergence. The loss of the MIER-MO-DQN method is therefore initially much larger than the loss of the random method, but as the number of iterations rises, the loss of the MIER-MO-DQN method begins to become significantly lower than the loss of the random replay, and the loss1 and loss2 of the MIER-MO-DQN method always fluctuates within a particular range after 200 iterations without any greater change. Therefore, MIER-MO-DQN's multi-indicator experience replay is more effective in this problem.

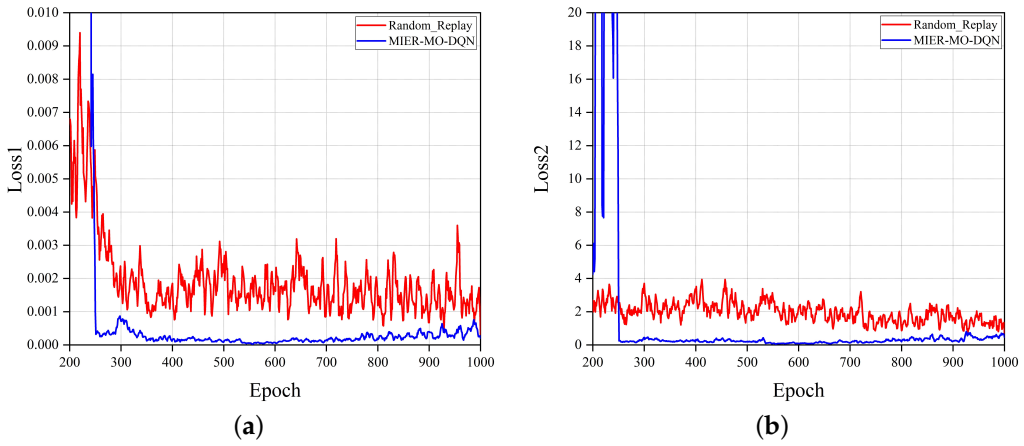

Figure 11. Change in (**a**) Loss1, (**b**) Loss2.

Another crucial indices of the deep reinforcement learning algorithm's performance is the average reward. The average reward of the reward function r_1 during each iteration is shown in Figure 12a, and the average reward for r_2 throughout each iteration is also shown in Figure 12b. Because the effects of random strategy and action decision mechanisms are almost identical in the initial time, the average reward of r_1 is higher and the average reward of r_2 is lower in the initial time. The time of treatment gradually increases as the iteration rises, and as the action-value function becomes closer to the true value, the average reward of r_1 starts to decline while the average reward of r_2 exhibits a general increasing trend. After about 4600 iterations, the average reward of both approaches converges near zero. The MIER-MO-DQN algorithm has strong convergence in personalized dosing on the basis of the aforementioned experimental data.

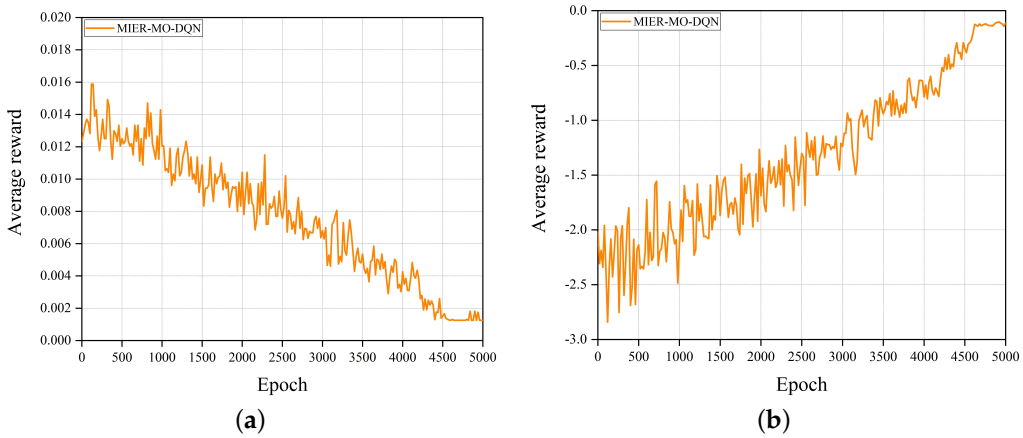

Figure 12. Average reward of (**a**) r_1, (**b**) r_2.

5. Conclusions

In order to reduce tumor cells as soon as possible during tumor treatment while effectively maintaining the patient's effector-immune cells, this study proposes a multi-objective deep reinforcement learning algorithm—MIER-MO-DQN, based on ensemble principles. First, MIER-MO-DQN offers patients a more personalized and effective treatment plan. It takes the number of tumor cells and effector-immune cells in the patient as input, constructs the reward function for each object, and restricts the great fluctuation of the patient's effector-immune cells. By combining three different decision-making approaches, MIER-MO-DQN solves the problem that a weighted sum function may make it difficult to choose the best drug dose in complex situations. The experiment shows that MIER-MO-DQN has a treatment duration of just one-third that of other multi-objective deep reinforcement learning methods, and reduces the number of tumor cells by about 30% at day 100 of treatment when compared to other regimens, and the number of effector-immune cells was always higher than the lower limit that we set. Second, our method is somewhat adaptable, enabling patients with varying parameters to achieve the same outstanding treatment outcomes. These parameters include the growth rate of tumor cells and the constant source of effector-immune cells, etc. For different parameters of the tumor model, we individually ran MIER-MO-DQN to produce therapy protocols. The results reveal that modifications of the tumor model's parameters have negligible effects on MIER-MO-DQN, and that MIER-MO-DQN consistently generates similar values for the experimental metrics. Finally, we present an approach that can accelerate MIER-MO-DQN's convergence, which can reduce the time to generate treatment plans. When conducting experience replay, we give each experience's features additional thought and choose the experiences for learning based on these features. The results demonstrated that although MIER-MO-DQN exhibits considerable oscillations in the early iterations, as the number of iterations rises, it can converge more rapidly than through random replay.

Author Contributions: Conceptualization, L.H. and Y.T.; methodology, L.H. and Y.T.; software, Y.T.; validation, L.H. and Y.T.; formal analysis, L.H. and Y.T.; investigation, L.H. and Y.T.; resources, L.H. and Y.T.; data curation, L.H. and Y.T.; writing—original draft preparation, Y.T.; writing—review and editing, L.H. and Y.T.; visualization, L.H. and Y.T.; supervision, L.H.; project administration, L.H. and Y.T.; funding acquisition, L.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 61962005.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We thank all of the authors of the primary studies included in this article. We are looking forward to valuable advice from the anonymous reviewers.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RL	Reinforcement learning
DQN	Deep Q-Network
MIER-MO-DQN	Multi-Objective Deep Q-Network based on Multi-Indicator Experience Replay

References

1. Bray, F.; Ferlay, J.; Soerjomataram, I.; Siegel, R.L.; Torre, L.A.; Jemal, J. A. Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA-Cancer J. Clin.* **2018**, *68*, 394–424. [CrossRef] [PubMed]
2. Su, M.Z.; Lao, J.H.; Zhang, N.; Wang, J.L.; Anderson, R.T.; Sun, X.J.; Yao, N.L. Financial hardship in Chinese cancer survivors. *Cancer-Am. Cancer Soc.* **2020**, *126*, 3312–3321. [CrossRef] [PubMed]
3. Cai, Y.; Xue, M.; Chen, W.Q.; Hu, M.G.; Miao, Z.W.; Lan, L.; Zheng, R.S.; Meng, Q. Expenditure of hospital care on cancer in China, from 2011 to 2015. *Chin. J Cancer Res.* **2017**, *29*, 253–262. [CrossRef] [PubMed]
4. Sanai, N.; Berger, M.S. Surgical oncology for gliomas: The state of the art. *Nat. Rev. Clin. Oncol.* **2018**, *15*, 112–125. [CrossRef]
5. Barton, M.B.; Jacob, S.; Shafiq, J.; Wong, K.R.; Thompson, S.R.; Hanna, T.P.; Delaney, G.P. Estimating the demand for radiotherapy from the evidence: A review of changes from 2003 to 2012. *Radiother. Oncol.* **2014**, *112*, 140–144. [CrossRef]
6. Bazrafshan, N.; Lotfi, M.M. A multi-objective multi-drug model for cancer chemotherapy treatment planning: A cost-effective approach to designing clinical trials. *Comput. Chem. Eng.* **2016**, *87*, 226–233. [CrossRef]
7. de Pillis, L.G.; Gu, W.; Fister, K.R.; Head, T.; Maples, K.; Murugan, A.; Neal, T.; Yoshida, K. Chemotherapy for tumors: An analysis of the dynamics and a study of quadratic and linear optimal controls. *Math. Biosci.* **2007**, *209*, 292–315. [CrossRef]
8. Yazdjerdi, P.; Meskin, N.; Al-Naemi, M.; Al Moustafa, A.E.; Kovacs, L. Reinforcement learning-based control of tumor growth under anti-angiogenic therapy. *Comput. Meth. Prog. Biomed.* **2019**, *173*, 15–26. [CrossRef]
9. Cimen, T. Systematic and effective design of nonlinear feedback controllers via the state-dependent Riccati equation (SDRE) method. *Annu. Rev. Control* **2010**, *34*, 32–51. [CrossRef]
10. Sápi, J.; Drexler, D.A.; Harmati, I.; Sápi, Z.; Kovács, L. Linear state-feedback control synthesis of tumor growth control in antiangiogenic therapy. In Proceedings of the 2012 IEEE 10th International Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 26–28 January 2012; pp.143–148.
11. Batmani, Y.; Khaloozadeh, H. Optimal chemotherapy in cancer treatment: State dependent Riccati equation control and extended Kalman filter. *Optim. Control Appl. Met.* **2013**, *34*, 562–577. [CrossRef]
12. Valle, P.A.; Starkov, K.E.; Coria, L.N. Global stability and tumor clearance conditions for a cancer chemotherapy system. *Commun. Nonlinear Sci.* **2016**, *40*, 206–215. [CrossRef]
13. Sharifi, M.; Moradi, H. Nonlinear composite adaptive control of cancer chemotherapy with online identification of uncertain parameters. *Biomed. Signal Process.* **2019**, *49*, 360–374. [CrossRef]
14. Shindi, O.; Kanesan, J.; Kendall, G.; Ramanathan, A. The combined effect of optimal control and swarm intelligence on optimization of cancer chemotherapy. *Comput. Meth. Programs Biomed.* **2020**, *189*, 105327. [CrossRef] [PubMed]
15. Singha, N. Implementation of fractional optimal control problems in real-world applications. *Fract. Calc. Appl. Anal.* **2020**, *23*, 1783–1796. [CrossRef]
16. Das, P.; Das, S.; Das, P.; Rihan, F.A.; Uzuntarla, M.; Ghosh, D. Optimal control strategy for cancer remission using combinatorial therapy: A mathematical model-based approach. *Chaos Soliton Fract.* **2021**, *145*, 110789. [CrossRef]
17. Dhanalakshmi, P.; Senpagam, S.; Mohanapriya, R. Finite-time fuzzy reliable controller design for fractional-order tumor system under chemotherapy. *Fuzzy Sets Syst.* **2022**, *432*, 168–181. [CrossRef]
18. Doruk, R.O. Angiogenic inhibition therapy, a sliding mode control adventure. *Comput. Meth. Programs Biomed.* **2020**, *190*, 105358. . [CrossRef]
19. Khalili, P.; Vatankhah, R. Derivation of an optimal trajectory and nonlinear adaptive controller design for drug delivery in cancerous tumor chemotherapy. *Comput. Biol. Med.* **2019**, *109*, 195–206. [CrossRef]
20. Jeerige, A.; Bein, D.; Verma, A. Comparison of Deep Reinforcement Learning Approaches for Intelligent Game Playing. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, LV, USA, 7–9 January 2019; pp. 366–371.
21. Kiran, B.R.; Sobh, I.; Talpaert, V.; Mannion, P.; Al Sallab, A.A.; Yogamani, S.; Perez, P. Deep Reinforcement Learning for Autonomous Driving: A Survey. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4909–4926. [CrossRef]
22. Pei, J.N.; Hong, P.L.; Pan, M.; Liu, J.Q.; Zhou, J.S. Optimal VNF Placement via Deep Reinforcement Learning in SDN/NFV-Enabled Networks. *IEEE J. Sel. Area Common.* **2020**, *38*, 263–278. [CrossRef]

23. Yang, C.; Shiranthika, C.; Wang, C.; Chen, K.; Sumathipala, S. Reinforcement learning strategies in cancer chemotherapy treatments: A review. *Comput. Meth. Programs Biomed.* **2022**, *229*, 107280. [CrossRef] [PubMed]
24. Zhao, Y.; Kosorok, M.R.; Zeng, D. Reinforcement learning design for cancer clinical trials. *Stat. Med.* **2009**, *28*, 3294–3315. [CrossRef] [PubMed]
25. Padmanabhan, R.; Meskin, N.; Haddad, W.M. Reinforcement learning-based control of drug dosing for cancer chemotherapy treatment. *Math. Biosci.* **2017**, *293*, 11–20. [CrossRef]
26. Yauney, G.; Shah, P. Reinforcement Learning with Action-Derived Rewards for Chemotherapy and Clinical Trial Dosing Regimen Selection. In Proceedings of the 3rd Machine Learning for Healthcare Conference(MLHC), California, CA, USA, 16–18 August 2018; pp. 161–226.
27. Zade, A.E.; Haghighi, S.S.; Soltani, M. Reinforcement learning for optimal scheduling of Glioblastoma treatment with Temozolomide. *Cancer-Am. Cancer Soc.* **2020**, *193*, 105443. [CrossRef]
28. Ebrahimi, S.; Lim, G.J. A reinforcement learning approach for finding optimal policy of adaptive radiation therapy considering uncertain tumor biological response. *Artif. Intell. Med.* **2021**, *121*, 102193 . [CrossRef] [PubMed]
29. Adeyiola, A.O.; Rabia, S.I.; Elsaid, A.; Fadel, S.; Zakaria, A. A Markov Decision Process Framework for Optimal Cancer Chemotherapy Dose Selection. In *Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2022; Volume 193, p. 12002.
30. Shiranthika, C.; Chen, K.W.; Wang, C.Y.; Yang, C.Y.; Sudantha, B.H.; Li, W.F. Supervised Optimal Chemotherapy Regimen Based on Offline Reinforcement Learning. *IEEE J. Biomed. Health Inform.* **2022**, *26*, 4763–4772. [CrossRef]
31. Gottesman, O.; Johansson, F.; Komorowski, M.; Faisal, A.; Sontag, D.; Doshi-Velez, F.; Celi, L.A. Guidelines for reinforcement learning in healthcare. *Nat. Med.* **2019**, *25*, 16–18. [CrossRef]
32. Yue, R.; Dutta, A. Computational systems biology in disease modeling and control, review and perspectives. *NPJ Syst. Biol. Appl.* **2022**, *8*, 37. [CrossRef]
33. Eckardt, J.N.; Wendt, K.; Bornhauser, M.; Middeke, J.M. Reinforcement Learning for Precision Oncology. *Cancers* **2021**, *13*, 4624. [CrossRef]
34. Dhieb, N.; Abdulrashid, I.; Ghazzai, H.; Massoud, Y. Optimized drug regimen and chemotherapy scheduling for cancer treatment using swarm intelligence. *Ann. Oper. Res.* **2021** . [CrossRef]
35. Watkins, C.J.C.H.; Dayan, P. Q-learning. *Mach. Learn.* **1998**, *8*, 279–292. [CrossRef]
36. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef] [PubMed]
37. Roijers, D.M.; Vamplew, P.; Whiteson, S.; Dazeley, R. A Survey of Multi-Objective Sequential Decision-Making. *J Artif. Intell. Res.* **2013**, *48*, 67–113. [CrossRef]
38. Oliveira, T.H.F.D.; Medeiros, L.P.D.S.; Neto, A.D.D.; Melo, J.D. Q-Managed: A new algorithm for a multiobjective reinforcement learning. *Expert Syst. Appl.* **2021**, *168*, 114228. [CrossRef]
39. Vamplew, P.; Dazeley, R.; Foale, C. Softmax exploration strategies for multiobjective reinforcement learning. *Neurocomputing* **2017**, *263*, 74–86. [CrossRef]
40. Hayes, C.F.; Rădulescu, R.; Bargiacchi, E.; Källström, J.; Macfarlane, M.; Reymond, M.; Verstraeten, T.; Zintgraf, L.M.; Dazeley, R.; Heintz, F.; et al. A practical guide to multi-objective reinforcement learning and planning. *Auton Agents Multi-Agent Syst.* **2022**, *36*, 26. [CrossRef]
41. Opricovic, S. Multicriteria optimization of civil engineering systems. *Fac. Civ. Eng. Belgrade* **1998**, *2*, 5–21.
42. Hashemi, A.; Dowlatshahi, M.B.; Nezamabadi-Pour, H. VMFS: A VIKOR-based multi-target feature selection. *Expert Syst. Appl.* **2021**, *182*, 11522. [CrossRef]
43. Li, L.L.; Xiong, J.L.; Tseng, M.L.; Yan, Z.; Lim, M.K. Using multi-objective sparrow search algorithm to establish active distribution network dynamic reconfiguration integrated optimization. *Expert Syst. Appl.* **2022**, *193*, 116445. [CrossRef]
44. Schaul, T.; Quan, J.; Antonoglou, I.; Silver, D. Prioritized experience replay. In Proceedings of the International Conference on Learning Representations (ICLR), San Juan, PR, USA, 2–4 May 2016.
45. C., E.S. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. [CrossRef]
46. Kaur, G.; Chanak, P.; Bhattacharya, M. Energy-Efficient Intelligent Routing Scheme for IoT-Enabled WSNs. *IEEE Internet Things.* **2021**, *8*, 11440–11449. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.