EKSPLOATACJA I NIEZAWODNOŚĆ MAINTENANCE AND RELIABILITY

Eksploatacja i Niezawodnosc – Maintenance and Reliability

Volume 24 (2022), Issue 4

journal homepage: http://www.ein.org.pl

Article citation info:
Liu Y, Qian X. Selective maintenance optimization with stochastic break duration based on reinforcement learning. Eksploatacja i Niezawodnosc – Maintenance and Reliability 2022; 24 (4): 771–784, http://doi.org/10.17531/ein.2022.4.17

Selective maintenance optimization with stochastic break duration based on reinforcement learning

Indexed by: Web of Science Group

Yilai Liu[a,b], Xinbo Qian[a,c,*]

[a] Key Laboratory of Metallurgical Equipment and Control Technology, Ministry of Education, Wuhan University of Science and Technology, Wuhan, China
[b] Hubei Key Laboratory of Mechanical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan , China
[c] Precision Manufacturing Institute, Wuhan University of Science and Technology, Wuhan, China

Highlights

- Selective maintenance model with stochastic break duration is proposed.
- Reinforcement learning(RL) method is applied to selective maintenance model.
- The advantages of considering stochastic break duration and RL are analysed.

Abstract

For industrial and military applications, a sequence of missions would be performed with a limited break between two adjacent missions. To improve the system reliability, selective maintenance may be performed on components during the break. Most studies on selective maintenance generally use minimal repair and replacement as maintenance actions while break duration is assumed to be deterministic. However, in practical engineering, many maintenance actions are imperfect maintenance, and the break duration is stochastic due to environmental and other factors. Therefore, a selective maintenance optimization model is proposed with imperfect maintenance for stochastic break duration. The model is aimed to maximize the reliability of system successfully completing the next mission. The reinforcement learning(RL) method is applied to optimally select maintenance actions for selected components. The proposed model and the advantages of the RL are verified by three case studies verify.

Keywords

selective maintenance; stochastic break duration; imperfect maintenance; reinforcement learning.

1. Introduction

Maintenance can restore aging systems to better condition and extend the system's life and is a crucial factor affecting industrial, military, and aerospace development. In many industrial and military applications, systems usually perform a sequence of missions with a finite break between two adjacent missions. Maintenance of the equipment is essential [39]. Maintenance actions can be performed during the break to guarantee the reliability of system successfully completing the next mission during subsequent production or missions. However, due to limited maintenance resources (time, manpower, spare parts, etc.), it may be impossible to perform maintenance on all components. Therefore, only some of the system components can be maintained during the limited break so that the reliability of the system meets the requirements or is maximized to complete subsequent production or missions successfully. In this case, managers need to decide which components to maintain based on the actual situation, rather than always following a fixed schedule for all components [4]. This maintenance strategy is known as selective maintenance.

Selective maintenance is vital in balancing limited maintenance resources with system performance. Rice et al [37] first introduced the selective maintenance problem by considering only one maintenance action to replace the failed components, assuming that all components are identical and that the lifetime follows an exponential distribution. Since 1998, many researchers have studied selective maintenance. Cassady et al [7] extended the model in Rice et al [37], assuming that the component life obeys Weibull distribution and considers three maintenance actions: minimal repair, preventive replacement and corrective replacement, and takes the total maintenance time as the constraint to maximize the reliability of the system successfully completing the next mission. Rajagopalan et al [36], an improved enumeration method was used to solve the selective maintenance problem with the constraints of total maintenance time and cost and the objective function of maximizing the next mission reliability of the system, which improves computational efficiency. Xu et al [44] further improved the enumeration method based on Rajagopalan et al [36], significantly reducing the number of candidate solutions and improving computational efficiency. When the scale of the system is large, the number of different components of the system and the number of maintenance actions increase. The enumeration method does not apply to selective maintenance problems with large and complex solution spaces when the number of feasible solutions grows exponentially. Lust et al [27]

(*) Corresponding author.
E-mail addresses: Y. Liu (ORCID: 0000-0003-2370-9250): 1454786628@qq.com, X. Qian: xinboqian@wust.edu.cn

studied a multi-component system with a series-parallel general structure and proposed a selective maintenance optimization method based on a heuristic algorithm, which has a better solution efficiency and promotes the optimization of the selective maintenance model. For the time being, only three maintenance actions were considered in the above study, and imperfect maintenance was not considered. However, in reality, imperfect maintenance is more realistic in engineering. Therefore, some researchers gradually considered imperfect maintenance [11, 19, 33]. Pandey et al [33], it was proposed that introducing imperfect maintenance can describe the decision problem more accurately and was more in line with practical applications. Among other works, Diallo et al [11] was first to propose a selective maintenance model for large k-out-of-n systems and an improved two-stage approach to improve computational efficiency, and Khatab et al [19] considered the stochastic of maintenance action quality.

Various uncertainties are inevitable in maintenance decisions of engineering systems, and ignoring these potential uncertainties may lead to inefficient optimization decisions, and the system may face the risk of not completing the mission [49]. Current studies on selective maintenance problems assume mainly deterministic values for break duration. In practice, unexpected events may lead to early termination or continuation of the mission, resulting in an increase or decrease in the break duration. For example, delays in flight departures or ship departures due to weather can lead to increased break duration. In the military, the time of the next mission start cannot be accurately determined, so the break between two adjacent missions is also uncertain. In similar situations, the break duration should be a random variable that obeys an appropriate distribution. Other literature [17, 18, 20, 25] considered the stochastic break duration with the decision goal of reducing maintenance resources. Zhao et al [48] considered stochastic mission time and multiple maintenance workers with different capacities. However, in many engineering practices, when maintenance resources cost, time, and manpower are limited, selective maintenance problems are often aimed at maximizing the reliability of system successfully completing the next mission rather than minimizing maintenance resources [6].

In recent years, selective maintenance optimization problems have been intensively studied. With the increasing complexity of selective maintenance optimization models, some advanced intelligent optimization algorithms, such as particle swarm algorithm [28], artificial bee colony [10], ant colony algorithm [25, 40], and genetic algorithm [5, 13, 43] have been widely adopted. As the scale of the system becomes larger, the factors considered become more comprehensive. Therefore the solution of large-scale selective maintenance decision-making problems poses new challenges, and the efficiency of optimization algorithms and global optimization capabilities need to be further improved [8]. Reinforcement learning belongs to machine learning methods, which have attracted more and more attention from researchers in solving decision problems [22]. Some reinforcement learning algorithms can be explored to obtain immediate payoffs and then select appropriate strategies to obtain the optimal solution of the model [14]. In recent years RL is effective in decision performance and computational efficiency. Other heuristic solution methods continuously iterate the algorithm randomly on the feasible solution space until the best solution is obtained or the number of iterations reaches the maximum. It may lead to problems such as complex model solving and limited computational efficiency [39]. In contrast, in RL, the agent continuously learns from each iteration and, in return, improves the result of the next iteration based on the previous one, and the optimal solution converges faster, thus improving the computational efficiency [31]. Although RL methods have been successfully applied to different problems and have significant advantages, they have not yet attracted sufficient attention in selective maintenance optimization.

In summary, this paper proposes a new selective maintenance model that considers the stochastic break duration. To maximize the reliability of the system successfully completing the next mission, each component has multiple optional maintenance actions, including minimal repair, imperfect maintenance, and replacement. The selective maintenance decision problem is modeled as a Markov decision process (MDP), and a RL approach is proposed to solve the model.

The rest of this paper is presented as follows. Section 2 is the related work about RL in other maintenance areas. Section 3 is the problem description and basic assumptions and describes the evaluation of imperfect maintenance and system reliability based on the Kijima type II model. Section 4 presents the selective maintenance model and the solution method of this paper. Three case studies are given in Section 5 to verify the accuracy of the model and the validity of the method. Finally, a summary and an outlook for future works are given in Section 6.

2. Related work

The main objective of selective maintenance optimization is to maximize the reliability of the system successfully completing the next mission. As the number of components and optional maintenance actions increases, traditional solution methods may have the problems of difficult model solving and limited solving efficiency. In recent years RL has become an effective method for solving complex decision problems. RL has been applied to solve various decision problems such as scheduling, manufacturing and maintenance. In this section, we briefly review the work of RL in other maintenance areas and selective maintenance.

Nooshin et al [47] proposed a dynamic condition-based maintenance (CBM) model that considers components subject to degradation and random shocks. Instead of discretizing the degradation state, the exact degradation level was considered as the state of the system, and finally deep reinforcement learning (DRL) was used to derive the optimal maintenance action for each degradation level. Mahmoodzadeh et al [29] studied CBM of dry gas pipeline and proposed a test bench to simulate pipeline corrosion while interacting with the RL to adjust the maintenance action and minimize maintenance costs. Peng et al [35] considered that RL can be effective in solving MDP problems with large state spaces, and models the CBM problem as a discrete-time continuous-state MDP rather than a discrete system with deterioration conditions. An RL algorithm was proposed to minimize the long-run average cost, and a Gaussian process regression function was used to model the state transfer and the value functions of the states in RL. Stephane et al [2] used MDP to model preventive maintenance for equipment consisting of multi-non-identical components with different probability distributions of failure times, which has the advantage of not requiring to estimate the main parameters of the model. Finally, the optimal strategy was solved using Monte Carlo reinforcement learning, which was not restricted by mathematical formulas. Huang et al [15] formulated the preventive maintenance(PM) decision for serial production lines as an MDP framework, considered the system production loss, and used DRL to solve the optimization model.

In addition to the above maintenance optimization, there are also some applications of RL for decision optimization problems. Andriotis et al [1] considered that in engineering systems management decisions can be made with MDPs or partially observable MDPs. For large multi-component systems, the number of system states and actions grows exponentially with the number of components, and it is difficult to characterize the environmental dynamics of the whole system, which can only be obtained by expensive numerical simulators. Therefore, a DRL algorithm was proposed to obtain an effective life cycle strategy. Ruan et al [38] studied the aircraft maintenance routing problem, where the objective was to generate maintenance feasible optimal routes for each aircraft under the constraints of maximum flight time, limitation on the number of takeoffs between two consecutive maintenance checks, and labor capacity maintenance. An RL approach was developed to solve the problem, by comparing with common optimization software, RL can solve the problem quickly and efficiently. Panagiotis et al [34] studied the maintenance

problem of a stochastic production/inventory system producing a single type of product, maximizing the total profit of the system when maintenance and repair duration as random variables. The commonly used dynamic programming methods were not suitable for solving the problem discussed in this paper, so a RL approach was proposed. Hu et al [16] proposed an RL framework with extreme learning machine optimization algorithm for aircraft life cycle maintenance, considering engine lifetime, performance degradation and random failures. It was found that the RL-driven maintenance strategy have a advantage compare to the CM, schedule Maintenance and prognostics and health driven strategies.

According to the above reviews, RL is effective in decision performance and computational efficiency. To the best of our knowledge, the proposed approach is novel in dealing with the single-mission selective maintenance problem.

3. Problem statements

3.1. Selective maintenance problem description for multi-component systems

In many military and industrial environments, systems are scheduled to perform multiple sequential missions with a finite break between two adjacent missions. Maintenance actions can be performed during the break to restore the aging system to a better condition for subsequent missions. However, due to the constraints of maintenance resources such as time and manpower, it may not be possible to perform maintenance on all components and select only some for maintenance depending on the situation. The basic process of selective maintenance decisions with stochastic break duration is shown in Fig. 1. As shown in Fig. 1, scenario 2 has a longer break compared to scenario 1, and only maintenance action 3 is not completed. And in scenario 1, both maintenance actions 2 and 3 are not completed.

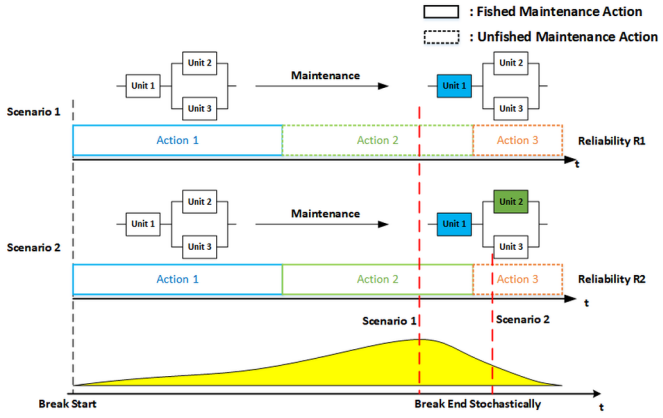

Fig. 1. Schematic diagram of selective maintenance decisions with stochastic break duration

To describe the selective maintenance problem, the basic assumptions are as follows:

(1) Assume a series-parallel system, and the system consists of i (i = 1, 2, ... , m) independent subsystems in series, and each subsystem i consists of j (j = 1, 2, ... , n) independent components C_{ij} in parallel, i and j denote the location of the components in the system. It is assumed that the components have only one failure mode, and the component's states are either failure or functioning. Here the variables $X_{\text{break,s}}(k)$ and $X_{\text{break,e}}(k)$ are used to denote the state of component C_{ij} at the beginning of kth break and the end of kth break, respectively, i.e., the state of component C_{ij} at the beginning of kth break can be expressed as:

$$X_{\text{break,s}}(k)=\begin{cases}1, \text{ if } C_{ij} \text{ functioning at the beginning of } k\text{th break}\\ 0, \text{otherwise}\end{cases} \quad (1)$$

The state of the component C_{ij} at the end of the kth break can be expressed as:

$$X_{\text{break,e}}(k)=\begin{cases}1, \text{ if } C_{ij} \text{ functioning at the end of } k\text{th break}\\ 0, \text{otherwise}\end{cases} \quad (2)$$

(1) Assume that during the break, the set of optional maintenance actions for the component is {do nothing(DN), minimal repair(MR), imperfect maintenance(IM), preventive replacement(PR), corrective replacement(CR)}, and the corresponding codes of maintenance actions are shown in Table 1. No maintenance means doing nothing, and no maintenance resources are consumed. The minimal repair can only be performed on failed components, consumes fewer resources, and can restore the failed components to functioning, but it does not change the reliability. The imperfect maintenance effect is between minimal repair and replacement. Preventive replacement can only be performed on functioning components, and corrective replacement can only be performed on failed components. When $X_{\text{break,s}}(k)$=0, the C_{ij} optional maintenance actions are minimal repair, imperfect maintenance, and corrective replacement. When $X_{\text{break,s}}(k)$=1, the C_{ij} optional maintenance actions are imperfect maintenance and preventive replacement. Fig. 2 shows the correspondence between maintenance action and component state.

Table 1. Codes of different maintenance action l

Maintenance Action	Do Nothing	Minimal Repair	Imperfect Maintenance	Preventive Replacement	Corrective Replacement
Corresponding code l	0	1	2,..., $L_{ij}-2$	$L_{ij}-1$	L_{ij}

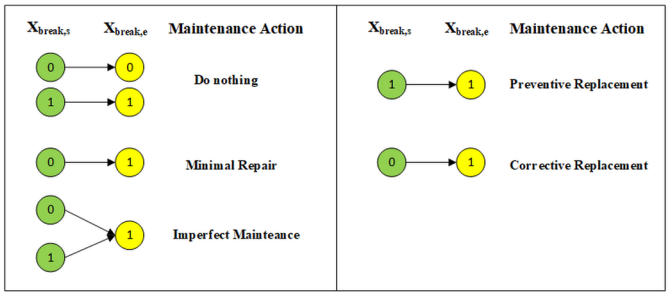

Fig. 2. Component state changes under different maintenance actions of components

(2) It is assumed that all maintenance actions can only be performed during the break. If the current maintenance action is not completed by the beginning of next mission, then it is assumed that the maintenance action has no repair effect on the component.

(3) Assume that only two types of maintenance resource constraints, maintenance time and manpower are considered in this paper.

(4) Assume that failure time of the component C_{ij} in the system obeys a two-parameter Weibull distribution.

3.2. Stochastic break duration

The break duration is stochastic because unexpected events may lead to early termination or continuation of production or mission such that the break duration decreases or increases randomly. In this study, the break duration Z_k is a random variable that obeys $f(Z_k)$. Therefore, the number of maintenance actions that can be completed during the

break is also uncertain. A binary decision variable $W_{ij}(l)$ is used to indicate whether the component C_{ij} is maintained during the break, which is defined as follows:

$$W_{ij}(l)=\begin{cases}1, \text{ if the maintenance action } l \text{ for component } C_{ij}\\ 0, \text{otherwise}\end{cases} \quad (3)$$

The maintenance time consumed during the break can be expressed as:

$$T=\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=0}^{L}t_{ij}(l)w_{ij}(l) \quad (4)$$

where $t_{ij}(l)$ is the maintenance time of completing maintenance action l.

The break duration Z_k as a random variable obeying $f(Z_k)$, it is required that the probability of completing the maintenance action during the break should be greater than or equal to a predetermined critical value τ , the range of τ values is (0,1], which is expressed as follows:

$$\Pr\left(T \le Z_k\right) \ge \tau \quad (5)$$

3.3. Evaluating the reliability of system successfully completing the next mission

There are many imperfect maintenance models about imperfect maintenance action [3, 23, 32, 30, 41, 42]. In this paper, we use the Kijima type II model to represent the maintenance effect of maintenance action by age reduction. The effective age of the component can be expressed as:

$$A_{ij}(k+1)=b_{ij}(l)B_{ij}(k) \quad (6)$$

where $A_{ij}(k+1)$ is the effective age of component C_{ij} after taking maintenance action l during the kth break. $B_{ij}(k)$ is the effective age of component C_{ij} at the beginning of the kth break. $b_{ij}(l)$ $(0 \le b_{ij}(l) \le 1)$ is the age reduction factor, which is influenced by the number of maintenance resources invested, the more maintenance time required for the executed maintenance actions, the smaller $b_{ij}(l)$ is, the better the maintenance effect.

Fig. 3 shows the relationship between the maintenance time of the component and effective age after the component is maintained during the break. The age reduction factor $b_{ij}(l)$ can be expressed as:

$$b_{ij}(l)=1-\left(\frac{t_{ij}(l)}{t_{ij}(L)}\right)^{1/\zeta_l} \quad (7)$$

where $t_{ij}(l)$ is the maintenance time for component C_{ij} to complete maintenance action l within the break. When the state of component C_{ij} is 1, $t_{ij}(L)$ is the time consumed for the preventive replacement of component C_{ij}. When the state of component C_{ij} is 0, $t_{ij}(L)$ is the time consumed for corrective replacement of component C_{ij}. ζ_l is a characteristic constant reflecting the relationship between maintenance time and age reduction factor function. When the maintenance action consumes the same time, the larger the ζ_l , the more obvious the maintenance effect.

According to the above effective age model, the conditional survival probability of a component after maintenance can be expressed as [21]:

$$r_{ij}(x)=1-\Pr\left\{Y-A_{ij}\le x|Y>A_{ij}\right\}=\frac{\Pr\left\{Y>x+A_{ij}\right\}}{\Pr\left\{Y>A_{ij}\right\}} \quad (8)$$

where the random variable Y represents the failure time. If the component is functional at the beginning of the kth mission and has an effective age of A_{ij}, then $r_{ij}(x)$ represents the probability that the component does not fail at any moment x. Since the failure time of component C_{ij} obeys the Weibull distribution, it is functioning at the beginning of the kth mission and has an effective age of $A_{ij}(k)$. The conditional survival probability of component C_{ij} at the end of kth mission is:

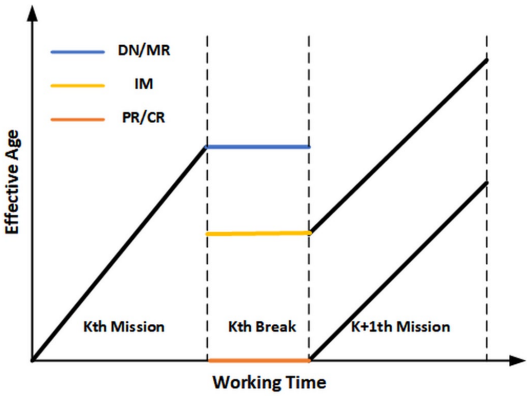

Fig. 3. Relationship between component working time and effective age in the Kijima II model

$$r_{ij}(k)=\exp\left[-\left(\frac{U(k)+A_{ij}(k)}{\eta_{ij}}\right)^{\beta_{ij}}+\left(\frac{A_{ij}(k)}{\eta_{ij}}\right)^{\beta_{ij}}\right] \quad (9)$$

where $U(k)$ is the duration of the kth mission. η_{ij} is the scale parameter in the Weibull distribution of component C_{ij}. β_{ij} is the shape parameter in the Weibull distribution of component C_{ij}. The reliability $R_{ij}(k)$ of component C_{ij} in the kth mission depends on the conditional survival probability $r_{ij}(k)$ and the component state $X_{\text{break,e}}(k-1)$ at the end of the kth break. The expression for the reliability $R_{ij}(k)$ of component C_{ij} as:

$$R_{ij}(k)=r_{ij}(k)X_{\text{break,e}}(k-1) \quad (10)$$

The study in this paper is a complex series-parallel system, i.e. the system consists of subsystems in series and subsystems comprised of components in parallel. The reliability $R_i(k)$ of subsystem i in kth mission can be expressed as:

$$R_i(k)=1-\prod_{j=1}^{n}\left(1-R_{ij}(k)\right) \quad (11)$$

The reliability $R_{\text{sys}}(k)$ of the system in kth mission as:

$$R_{sys}(k)=\prod_{i=1}^{m}R_i(k)=\prod_{i=1}^{m}\left(1-\prod_{j=1}^{n}(1-R_{ij}(k))\right) \quad (12)$$

where m is the number of subsystems in the system and n is the number of components in the subsystem.

4. Selective maintenance model and optimization based on the stochastic break duration

4.1. Selective maintenance optimization model

For a system performing sequential missions, using limited maintenance resources in a finite break to maximize the reliability of the system to complete the next mission is the key to maintenance decisions. Assume that the states $X_{\text{break},s}(k)$ and effective age $B_{ij}(k)$ of each component in the system are known at the beginning of the kth break. Given the optional maintenance actions of each component, the selective maintenance problem can be described as follows: with limited maintenance time and manpower, select the components to be maintained and their corresponding maintenance action so that the reliability of the system to complete the next mission is maximized. When the break duration Z_k is a random variable and the probability distribution function is known, the selective maintenance decision model can be expressed as:

$$\max_{[w_1,w_2,\cdots,w_n]\in A} R_{sys} = \prod_{i=1}^{m}\left(1-\prod_{j=1}^{n}\left(1-R_{ij}\left(k+1\right)\right)\right) \tag{13}$$

Subject to:

$$p\left(T=\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=0}^{L}t_{ij}(l)W_{ij}(l)\le Z_k\right)\ge\tau \tag{14}$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=0}^{L}W_{ij}(l)\le 1 \tag{15}$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{l=0}^{L}X_{break,e}(k)W_{ij}(l)\le 1 \tag{16}$$

$$W_{ij}(l)\le 1-Y_{break,s}(k) \tag{17}$$

$$X_{break,e}(k)=Y_{break,s}(k)+(1-Y_{break,s}(k))\cdot W_{ij}(l) \tag{18}$$

$$A_{ij}(k+1)=[b_{ij}(l)\cdot W_{ij}(l)+(1-W_{ij}(l))]\cdot B_{ij}(k) \tag{19}$$

$$W_{ij}(l),X_{break,e}(k),Y_{break,s}(k)\in\{0,1\};b_{ij}(l)\in[0,1] \tag{20}$$

In the above selective maintenance decision model, Eq. (13) is the decision objective to maximize the reliability of system successfully completing the next mission. Eq. (14) is the chance constraint, when the break duration is a random variable, the probability of completing the selected maintenance action is required to be greater than or equal to τ, the range of τ values is [0,1]. Eq. (15) illustrates that in each break, each maintenance action is selected at most once and can only be performed on one component. Eq. (16) shows that in each break, a component that is selected for maintenance can perform at most one maintenance action. Eq. (17) shows that minimal repair can only be performed on the failed component. Eq. (18) is used to update the state of the component C_{ij}, for example, when the component C_{ij} state $X_{\text{break},s}(k)=0$ at the beginning of the kth break, after maintenance i.e. $W_{ij}(l)=1$, the component C_{ij} state $X_{\text{break},e}(k)=1$. Eq. (19) is used to update the effective age of the component C_{ij}, for example, when the component C_{ij} after maintenance i.e. $W_{ij}(l)=1$, then $A_{ij}(k+1)=b_{ij}(l)$-$B_{ij}(k)$. When the component C_{ij} does not maintenance $W_{ij}(l)=0$, then $A_{ij}(k+1)=B_{ij}(k)$.

4.2. The reinforcement learning solution method for selective maintenance optimization

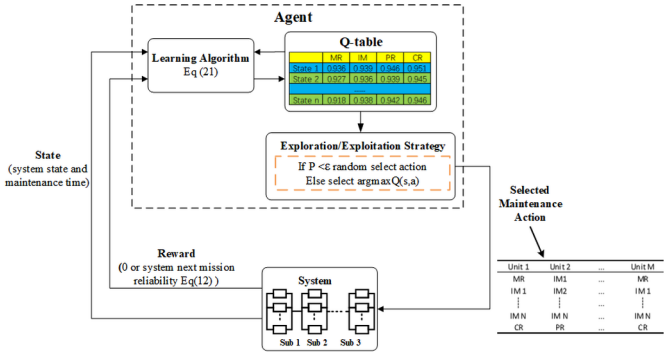

Fig. 4. Reinforcement learning framework for selective maintenance optimization

In this study, a reinforcement learning(RL) based framework is used to describe the selective maintenance decision process using MDP and solved using the Q-learning algorithm. According to this framework, the decision agent interacts with the system and selects a maintenance action at a specific time (decision period) to maximize the decision goal. The described framework is shown in Fig. 4. In MDP, the main factors that determine the decision process include the transfer law of states in the system and the maintenance action scheme. The interaction of these two factors leads to a particular reward for the decision-maker, usually represented by an objective function. MDP is an extension of the Markov chain, and its state space, action space, and reward are described as follows:

State space $\boldsymbol{S}$: It defines a finite two-dimensional state space, each state represents the state of the system at a decision moment and the total maintenance time. The state space can be expressed as $\boldsymbol{S}=\{X_{ij};\ T\}$, where X_{ij} consist of the states of all the components in the system, the component state is binary variables, and T is the total maintenance time. If the system consists of 5 components, the state space at a decision moment can be expressed as $\boldsymbol{S}=\{0_{1,1},1_{1,2},0_{2,1},0_{2,2},1_{2,3};\ 0.5\}$, where $0_{1,1}$ represents component $C_{1,1}$ in failed , $1_{1,2}$ represents component $C_{1,2}$ in functioning, and 0.5 represents the total maintenance time. The RL agent moves from the initial state to the terminated state and assigns an ordinal number to each state.

Action space $\boldsymbol{A}$: The action space consists of optional maintenance actions for all components, which can be expressed as $A=\{l_{ij}\}$, $l=\{DN_{ij}, MR_{ij}, IM1_{ij},..., IMn_{ij}, PR_{ij}, CR_{ij}\}$. Given the current state, the agent can select an action from the action space. By judging whether the selected action meets the constraint, the punishment or reward is obtained in turn. It indicates which actions the agent can choose for each observed state. Given the current state of the system, if the agent is not terminated state, any action in action space can be selected.

Reward $\boldsymbol{R}$: Rewards reflect the aptness of the RL agent for the current maintenance action, so here the reward function is defined as the objective function. The objective function of this paper is to maximize the reliability of the system successfully completing the next mission. In this paper, a negative reward is used when the maintenance action selected by the agent does not meet the constraints. When the maintenance action selected by the agent satisfies the constraints and is not the terminated state, 0 is used as a reward. When the maintenance action selected by the agent satisfies the constraints and is the terminated state, the Eq. (13) is used as a reward.

RL is a simulation-based dynamic programming algorithm mainly used to solve Markov decision problems and is an intelligent agent learning optimal control strategy. Compared with traditional dynamic programming, the RL approach does not require a state transfer probability matrix and avoids dynamic programming modeling dimen-

sional disaster [46]. The state space size of this problem is 2^N, and the action space size is L^N, where N is the total number of components in the system. The Q-learning algorithm is one of the more commonly used RL algorithms. The optimal policy is derived by constructing a table of state-maintenance action Q. The Q-learning algorithms have been shown to eventually reach a convergence condition for each state through continuous learning in a stochastic environment [45]. In this study, the selective maintenance decision optimization problem is modeled as an MDP. The Q-learning algorithm in RL is used to solve it to obtain the optimal maintenance policy, as follows:

Step1: Initialize Q(s,a)=0, Q value table is a list of rows, the value of the mth row mth column represents the value of the action of m maintenance action in the state of S_n, set the maximum number of cycles max_episode.

Step2: Initialize the state $\boldsymbol{S}$ at the beginning of each cycle, the state of each component after the end of the kth mission and the current total maintenance time $\boldsymbol{T}$=0.

Step3: Select the maintenance action w_n according to the ε-greedy policy and get the reward r. Update the Q-value table using the above reward according to Eq. (21) .

Step4: Update the state $\boldsymbol{S}$. Use Eq. (14) to determine whether the state reaches the termination state. If not, repeat the above steps from step 2.

Step5: When the number of cycles equals max_episode, stop the cycle to get the final Q-value table.

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left(r + \gamma \max Q(s',a') - Q(s,a)\right) \tag{21}$$

where α ($0 < \alpha < 1$) is the learning rate and γ ($0 < \gamma < 1$) is the discount factor. The flow chart of the algorithm is shown in Fig. 5.

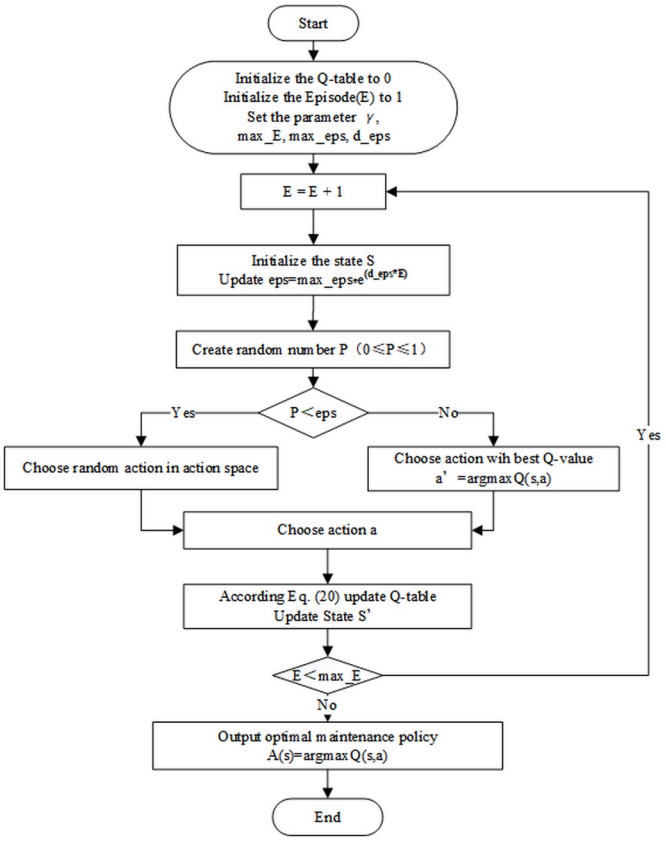

Fig. 5. Q-learning algorithm flowchart

In RL, exploration and exploitation are the two core problems. The decaying ε-greedy policy is used here for the agent to learn a better policy. The ε-greedy policy is to select the current optimal action(a'=argmax$Q$(s,a)) with probability 1 - ε , and randomly select the action among all available actions with probability ε. The conventional ε-greedy strategy constantly explores the action space with the same probability of ε. When ε is small, the exploration is not thorough enough and may obtain the optimal local strategy. When ε is large, the agent may have long explored the optimal strategy but will continue to explore it, resulting in slow convergence. Therefore, the decaying ε-greedy strategy is used here, respectively, the agent starts exploring with a larger ε and gradually decreases ε as the number of iterations increases, and the iteration formula is:

$$\varepsilon = \min\_eps + (\max\_eps - \min\_eps) \cdot e^{d\_eps \cdot E} \tag{22}$$

5. Case study

Three cases are given to test performance of the model and proposed method. The first case is a hydraulic system that is more typical of a real application, in which the key components are analyzed. The superiority of RL and the difference between the stochastic and deterministic break duration are analyzed. The second case is a two-stage 5-component system in which the superiority of RL is verified by comparing the results with other literature. Then, the difference between the stochastic and deterministic break duration is analyzed to illustrate the impact of the stochastic break duration on the system reliability. Due to the redundancy of this case system compared to the first case, the sensitivity analysis of the component parameters is performed here. The third example is a five-stage 14-component coal transportation system, where the performance of the algorithm is compared and the difference between stochastic and deterministic break duration is analyzed. The impact of stochastic on system reliability and the effectiveness of RL for larger scale complex systems are further verified.

5.1. Case 1: Hydraulic tension systems

A hydraulic tension system is known to consist of 16 components, which can be divided into two categories of components. The first category is critical components, and the second category is non-critical components. In this paper, the key components pump, solenoid valve, accumulator and cylinder are analyzed, and these four components are connected in series. The parameters of each component are shown in Table 2, where the Weibull distribution shape and scale parameters are derived from the literature [12]. In Table 2, ζ denotes the characteristic constant of the age regression factor. β, η denote the shape and scale parameters of the Weibull distribution. B(k) denotes the effective age of the component at the beginning of the kth break. X(k) denotes the state of each component of the system at the beginning of the kth break. The maintenance actions that can be adopted for each component and their corresponding maintenance times are shown in Table 3, where 0~4 represents the codes of different maintenance actions in order, where the fix is the fixed maintenance time.

5.1.1. Algorithm performance analysis

To further verify the effectiveness of the RL algorithm, a comparison with the genetic algorithm(GA) algorithm used in most of the literature is conducted. Assuming that the kth mission is just completed now, the duration Z_k of the break obeys a normal distribution of N(0.5, 0.04) with a range of [0.35, 0.65], τ =0.8, and the duration of the k+1th mission U=1500 days, all other component parameters are shown in Table 2. Among them, the parameters related to the GA algorithm, the number of populations NP=80, the crossover rate pc=0.8, the variation rate pm=0.05, and the maximum number of iterations iter=1000. The parameters related to Q-learning, the learning rate α=0.02, the discount rate γ=0.5, and the maximum number of iterations iter=10000. Due to the stochastic of the algorithm, 10 sets of simulations were performed for each method to find its optimal

Table 2. Component parameters

	Characteristic constant of age reduction factor	Shape and scale parameters of Weibull distribution		Effective age of component	Initial state of component
ID	ζ	β	η	$B(k)$	$X(k)$
1	2.5	2.36	1850	3500	1
2	2.0	1.853	3657	2400	1
3	3.0	1.46	3304	4500	1
4	3.2	2.023	3501	3500	1

Table 3. The maintenance time $t_{ij}(l)$ of different maintenance actions l for components (time is in days)

ID	Maintenance actions l code					Fix
	0	1	2	3	4	
1	0	0.0186	0.0371	0.0557	0.0743	0.03
2	0	0.0443	0.0886	0.1330	0.1770	0.03
3	0	0.0471	0.0943	0.1410	0.1890	0.03
4	0	0.0457	0.0914	0.1370	0.1830	0.04

strategy A_{best}, the maintenance time for the optimal strategy T_{best}, the average reliability R_{mean}, the maximum reliability R_{best}, the variance R_{std} and the average running time $\overline{S}$ as the comparison results. To compare the quality of RL and GA solutions, a parameter $\%QOS$ is introduced here as a performance metric, $\%QOS=(R_{best}-R_{mean})/R_{best}$. The comparison results and performance metric results are shown in Table 4.

Table 4. Comparison results of the two algorithms(time is in days)

Method	A_{best}	T_{best}	R_{best}	R_{mean}	R_{std}	$\overline{S}$	$\%QOS$
Q-learning	[4,2,0,4]	0.446	0.9878	0.9869	0.0016	1.15	0.09
GA	[3,2,2,2]	0.460	0.9792	0.9752	0.0032	10	0.41

From Table 4, we can see that the maximum reliability R_{best}=0.9878 solved by Q-learning and the total maintenance time T_{best}=0.446 days. The the maximum reliability R_{best}=0.9792 solved by GA and the total maintenance time T_{best}=0.46 days. From the analysis of the results, we can see that the optimal strategy solved by Q learning is better than GA and the reliability is 0.86% higher. The mean and variance of Q-learning results are better than GA in 10 solving results, which indicates that the stability of the Q-learning algorithm is better than GA. Regarding the computation time, running on a computer configured with Intel (R) Core (TM) i5 -6200U CPU @ 2.30GHz, 12G RAM. Although RL has more iterations than GA, the average time spent by GA is 8.85s more than that of RL. Regarding the quality of the obtained solutions, the optimal solution of RL is better than GA, and the average solution of RL deviates from the optimal solution by only 0.09%, while the average solution of GA deviates from the optimal solution by 0.41%. Therefore, the RL algorithm can find higher quality solutions, which verifies the superiority of RL. In order to verify the superiority of this RL, tests on small-scale systems are not sufficient. Section 5.2.1 will further verify the superiority of RL by making comparisons with other literature, and Section 5.3.1 is a comparison of RL with GA in large-scale complex systems.

5.1.2. Comparison between stochastic and deterministic of break duration

The difference between the stochastic and deterministic break duration is clarified by substituting the strategy derived from the RL-based deterministic model into the uncertainty model to obtain the reliability $R1$. Then comparing the analysis with the reliability $R2$ obtained from the strategy derived from the RL-based uncertainty model, ΔR calculation schematic is shown in Fig. 6. When the break duration is stochastic, the optimal maintenance policy $A1$=[4,2,0,4] solved by RL is known from section 5.1.1, and the reliability $R2$=0.9878, and the maintenance time is 0.446 days. When the break duration Z=0.5 is a fixed value with all other parameters held constant, the optimal maintenance

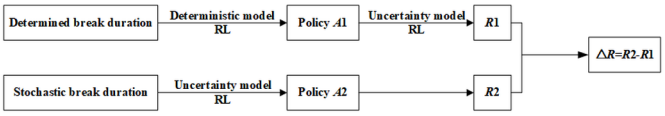

Fig. 6. Calculation schematic for system next mission reliability improvement ΔR of the proposed stochastic model compared to the determined model

policy $A2$=[4,3,0,4] solved by RL, the reliability R=0.9903 and maintenance time is 0.49 days. In order to compare the difference between the stochastic and deterministic break duration, the strategy $A2$ solved for the deterministic case is substituted into the uncertainty model to find the reliability $R1$=0.9795. Therefore, the difference between the deterministic strategy and the strategy substituted into the uncertainty model is 1.08%. As seen in Table 5 the maintenance policy considering uncertainty is better and system next mission reliability improvement ΔR=0.83%. Based on the above observations, the reliability of the system successfully complete the next mission in the deterministic case will be overestimated if the uncertainty of the break duration is ignored.

Table 5. Difference between stochastic and determined break duration(time is in days)

Case	Policy	Reliability	Maintenance time
Stochastic	[4,2,0,4]	0.9878	0.446
Deterministic strategy substitution in the uncertainty model	[4,3,0,4]	0.9795	0.327

5.2. Case 2:Two-stage 5-component system

The two-stage 5-component system is studied with the structure diagram shown in Fig. 7. The relevant parameters of each component are derived from the Chen et al [9], as shown in Table 6. In Table 6, ζ denotes the characteristic constant of the age reduction factor. β and η denote the shape and scale parameters of the Weibull distribution. $B(k)$ denotes the effective age of component at the beginning of the kth break. $X(k)$ denotes the state of each system component at the beginning of the kth break. The different maintenance actions of various components consume different time, as shown in Table 7, and 0~5 represent different codes of maintenance actions in order.

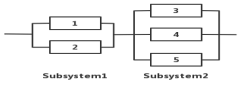

Fig. 7. Two-stage 5-component system structure diagram

Table 6. Component parameters

	Characteristic constant of age reduction factor	Shape and scale parameters of Weibull distribution		Effective age of component	Initial state of component
ID	ζ	β	η	$B(k)$	$X(k)$
1	2.2	2.0	20	20	1
2	2.3	2.1	19	25	0
3	2.1	2.0	21	25	0
4	2.4	2.2	22	25	1
5	2.0	1.9	21	20	0

Table 7. The maintenance time $t_{ij}(l)$ of different maintenance actions l for components (time is in days)

ID	Maintenance actions l code					
	0	1	2	3	4	5
1	0	0.12	0.21	0.35	0.43	0.51
2	0	0.15	0.25	0.30	0.42	0.58
3	0	0.14	0.24	0.32	0.41	0.53
4	0	0.16	0.23	0.38	0.42	0.56
5	0	0.13	0.18	0.35	0.43	0.48

Table 8. RL vs. GA result(time is in days)

Method	Maintenance policy	R_{sys}	Maintenance time
RL	[2,5,5,0,2]	0.983	1.5
GA	[3,5,1,2,2]	0.982	1.48

5.2.1. Comparison of Q-learning and GA

The studied case is introduced by the Chen et al [9], where the break duration Z_k is a fixed 1.5 days and the k+1th mission duration U=5 days, which is solved using the GA. Under the condition of the same other parameters, the Q-learning algorithm is used to solve the problem, which is compared with the results of Chen et al [9], where the related Q-learning parameters α=0.02 and γ=0.5. Using the PYTHON programming solution and obtained the maintenance policy A(s)=[2,5,5,0,2] . The reliability R=0.983 compared to the result solved by the Chen et al [9] using GA is 0.1% larger, and both results are shown in Table 8.

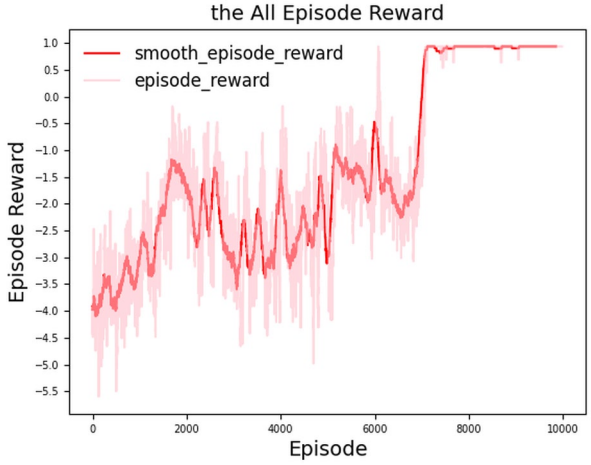

Fig. 8. The training process of the proposed Q-learning algorithm

5.2.2. RL Results with stochastic break duration

Assume that the kth mission is just completed now and the duration of the kth break Z_k obeys a truncated normal distribution of N(1.5, 0.0225) with the range of [1.35, 1.65], τ =0.8, and the duration of the k+1th mission U=5 days. The maintenance policy A(s)=[2,5,2,2,2] solved using Q-learning, the reliability R=0.98 after maintenance. It can be seen that considering the determined break duration leads to an overestimation of the reliability of the system successfully complete next mission. The Q-learning parameters are the same as section 5.2.1, and the Q-learning process is shown in Fig. 8. In the first 7000 iterations, the agent randomly explores the possible maintenance actions, and the Q matrix converges relatively slowly, after which the value of the Q matrix gradually converges and eventually reaches the convergence state.

5.2.3. Comparison between stochastic and deterministic of break duration

When the break duration is a deterministic value of 1.5 days, the strategy solved by RL is $A1$=[2, 5, 5, 0, 2]. Bringing this strategy into the uncertainty model, i.e., the break duration Z_k is a truncated normal distribution N(1.5, 0.0225) with the range of [1.35, 1.65], the optimal reliability $R1$=0.969 under the constraint $P(T \le Z) \ge \tau$ (τ= 0.8). The optimal maintenance policy $A2$=[2, 5, 2, 2, 2] solved by RL under the above uncertainty model has a reliability $R2$=0.98. Therefore, system next mission reliability improvement ΔR=0.011 shows that the maintenance policy considering stochastic is better than the deterministic one with 1.1% higher reliability.

Table 9. System next mission reliability improvement ΔR (%) at the different mean and standard deviation of the distribution of break duration

Distribution mean	Distribution standard deviation				
	0.01	0.05	0.1	0.15	0.2
	ΔR (%) between stochastic and deterministic break duration				
1	0.0	0.0	0.0	0.5	0.5
1.2	0.0	0.6	0.5	0.5	0.5
1.4	0.1	1.2	1.2	0.6	0.5
1.6	0.3	1.2	1.2	0.9	0.8
1.8	0.3	1.4	1.4	1.2	1.2
2	0.1	2.3	2.2	2.1	2.1

The impact of stochastic break duration on the maintenance strategy is illustrated by comparing the system reliability between stochastic and determined break duration. Under the same chance constraint and other parameters, the sequential simulations obtain the system successfully completing the next mission reliability $R2$ with different mean and standard deviation by varying the break duration obeying distribution in the uncertainty model. Mean M={1, 1.2, 1.4, 1.6, 1.8, 2}, standard deviation STD={0.01, 0.05, 0.1, 0.15, 0.2}, 30 combinations exist, and 30 sets of simulation experiments were implemented. The determined break duration T={1, 1.2, 1.4, 1.6, 1.8, 2}, respectively, are derived from the corresponding maintenance policy A_T in the deterministic model by RL, and the system reliability $R1$ is derived by substituting the maintenance policy A_T into the uncertain model with M=T. The results of system next mission reliability improvement ΔR are shown in Table 9 and Fig. 9 below.

As seen in Fig. 9, the overall trend of system next mission reliability improvement ΔR increases with the mean value, indicating

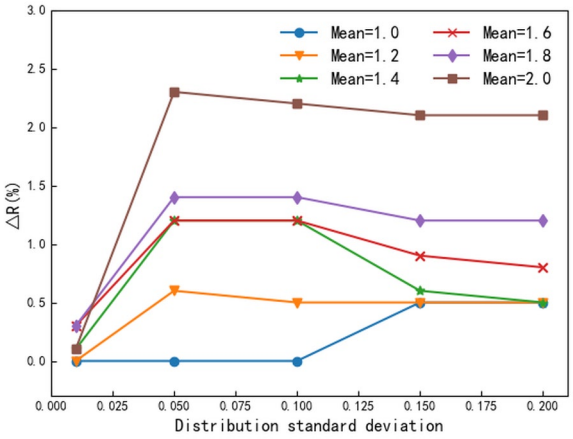

Fig. 9. System next mission reliability improvement ΔR at the different mean and standard deviation of the distribution of break duration

that the difference between uncertainty and certainty is more evident with larger mean values. It is because the increase of the mean value leads to a relatively long break time, allowing to select some maintenance actions with higher code. And the higher-code maintenance action requires longer maintenance time and better maintenance effect. Since the strategy is derived when the break is deterministic, there is sufficient time to complete all maintenance actions. However, in the case of uncertainty, there may not be enough time to complete all maintenance actions due to the constraint of insufficient time. As a result, a maintenance action is performed only partially and not fully completed, in which case the component reliability is unchanged. The relatively high code of the selected maintenance action when the mean value is large leads to lower system reliability under the inability to complete all maintenance actions, resulting in a larger ΔR.

In addition, when the mean value is 1, and the standard deviation is less than 0.1, the ΔR is equal to 0 in the first period and increases with the standard deviation. When the mean value is 1, it is at the left end of the distribution range, and the break duration does not change much with the standard deviation increase in the first period. Then it increases more obviously so that there are more optional maintenance actions, and the final system reliability increases. When the mean value is other values, ΔR increases with the standard deviation increase and gradually becomes smaller. It is because, in the beginning, the standard deviation is small, the uncertainty case is close to the deterministic case, and ΔR is small and close to 0. As the standard deviation increases, the duration of the break decreases relatively gradually, and the gap is the largest at the initial stage, leading to the largest ΔR, and then ΔR gradually decreases. The decrease in the break duration causes it as the standard deviation increases. It can be seen from the above figure that ΔR is greater than or equal to 0, and the maximum difference value reaches 2.3%. It shows that the model considering uncertainty is significantly better than the deterministic model. Considering a deterministic break duration can lead to an overestimation of reliability. In case of uncertainty encountered, it may lead to the inability of the system to complete subsequent mission.

5.2.4. Sensitivity analysis of component parameters

The optimization objective of this paper is to maximize the reliability of the system successfully completing the next mission. The mission duration U, the characteristic parameter ζ, and the Weibull distribution parameter β, η directly affect the optimization results. Sensitivity analysis is performed on the above parameters to verify the validity of the model, the feasibility of the method, and the influence of stochastic on the maintenance policy. For the selective maintenance decision model, the parameter U determines the mission duration, and the larger U is, the lower the reliability R. The characteristic parameter ζ reflects the relationship between the maintenance time and the age reduction factor. The larger ζ is, the more pronounced the maintenance effect of the same maintenance time is, i.e., the larger reliability R is. The shape and scale parameters β and η of the Weibull distribution obeyed by the component failure time, respectively, and the larger β and η are, the larger reliability R is. The following experiments were conducted to verify the effects of U, ζ, β, and η on maintenance decisions.

Simulation tests are performed in three categories, U and ζ, U and β, and U and η. 25 combinations exist in each category, respectively. U={5, 6, 7, 8, 9}, ζ={1.8, 2, baseline(2.2, 2.3, 2.1, 2.4, 2.0), 2.4, 2.6}, β={1.7, 1.9, baseline(2.0, 2.1, 2.0, 2.2, 1.9), 2.2, 2.4}, and η={17, 19, baseline(20, 19, 21, 22, 21), 22, 24}. Except for the baseline parameter value in the table 6, the parameters of the remaining components are taken as shown in the above set and are the same, and all other model parameters and algorithm parameters are the same as in section 5.2.2. Firstly, maintenance policy A is derived in the deterministic case. Then the reliability $R1$ is obtained by substituting maintenance policy A from the deterministic model into the uncertainty model. The reliability $R1$ is compared with the reliability $R2$ obtained in the uncertainty case. The results of system next mission reliability improvement ΔR for each type of experiment are shown in the following Table 10-12 and Figs. 10-12.

Table 10. System next mission reliability improvement ΔR(%) for different mission duration U and component characteristic constants ζ

characteristic constant ζ	Mission duration U				
	5	6	7	8	9
1.8	0.8	1.6	2.8	4.2	6.2
2	0.9	1.7	2.8	4.4	6.5
baseline	1.1	1.8	3.0	4.6	6.5
2.4	1.2	1.9	3.0	4.5	6.5
2.6	1.3	2.1	3.1	4.7	6.6

Table 11. System next mission reliability improvement ΔR(%) for different mission duration U and Weibull distribution shape parameter β

Shape parameter β	Mission duration U				
	5	6	7	8	9
1.7	1.3	1.8	2.5	3.7	5.2
1.9	1.3	1.9	2.9	5.8	7.9
baseline	1.1	1.8	3.0	4.6	9.0
2.2	1.1	2.0	3.2	6.5	9.0
2.4	1.0	1.9	3.2	6.6	9.2

Table 12. System next mission reliability improvement ΔR(%) for different mission duration U and Weibull distribution scale parameter η

Scale parameter η	Mission duration U				
	5	6	7	8	9
17	2.1	3.9	6.2	10.6	14.8
19	1.2	2.4	4.0	6.0	8.8
baseline	1.1	1.8	3.0	4.6	9.5
22	0.7	1.2	2.0	3.2	4.7
24	0.6	0.9	1.3	2.0	3.2

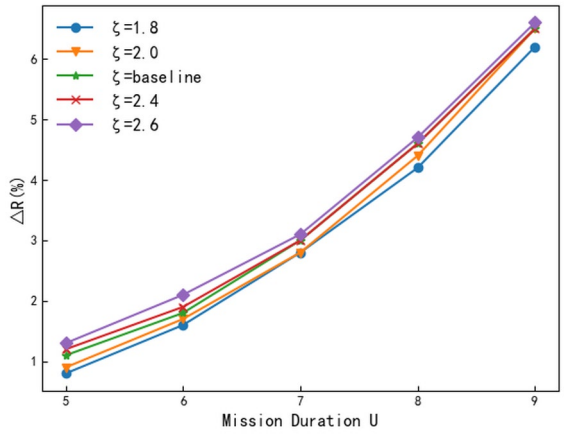

Fig. 10. System next mission reliability improvement ΔR with different component characteristic constants ζ

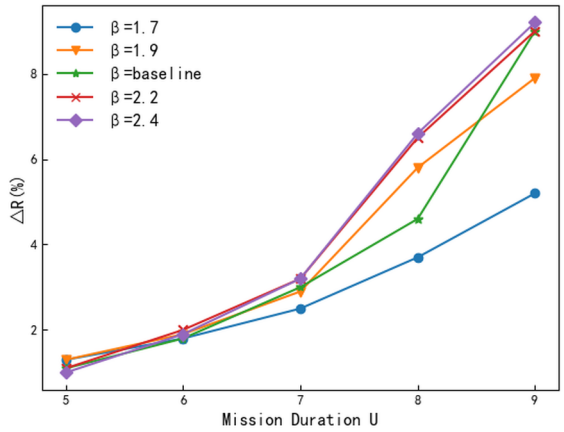

Fig. 11. System next mission reliability improvement ΔR with different weibull distribution shape parameter β

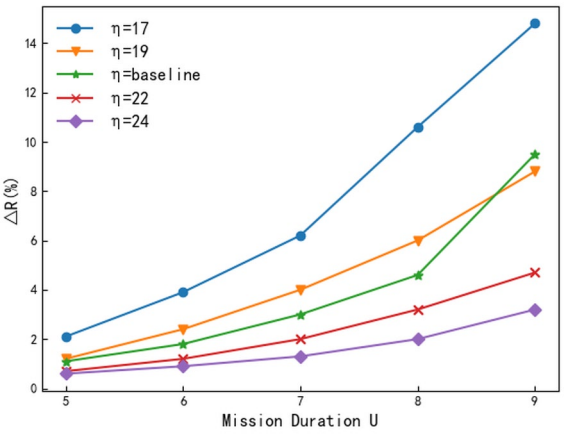

Fig. 12. System next mission reliability improvement ΔR with different weibull distribution scale parameter η

From Figs. 10-12, it can see that next mission reliability improvement ΔR changes less with parameter ζ, and the overall ΔR gradually increases with the increase of parameter ζ. When parameter U is less than 7, ΔR is less affected by parameter β and almost unchanged, and ΔR is gradually increased by parameter β with the increase of parameter U. The larger parameter β is, the larger ΔR is. ΔR changes more obviously with the increase of parameter η, and ΔR gradually decreases with the increase of parameter η. The above figure shows that ΔR is influenced by parameter U the most, followed by parameter η, and parameter ζ has the least influence on ΔR. Among them, ΔR reaches a maximum of 14.8% when analyzing the effect of parameter η. Therefore, the superiority of uncertainty is mainly influenced by the component parameters β, η, and the mission duration U, relative to the deterministic break. And in the case of larger mission duration U, considering the superiority of stochastic break duration is more prominent. Indicating that the larger the parameter U, the greater the uncertainty influence is also.

In summary, the model and algorithm accurately reflect the difference between uncertainty and certainty under each parameter, verifying the validity of the model and the feasibility of the method. This analysis also shows that the model and method apply to other systems. Through the above analysis, ignoring the uncertainty of the break duration can significantly impact the reliability of system to complete the next mission. In the case of large relevant parameters, ignoring the uncertainty of the mission can lead to an overestimation of the system reliability. It can result in a high risk of not being able to complete the next mission.

5.3. Case 3: A complex multi-component coal transportation system

To further verify the the validity of the model and the method, which is also valid for large-scale systems, the coal transmission system of literature [24] is used here as an example. The system consists of 5 subsystems connected in series and 14 components connected in parallels, and its structural sketch is shown in Fig. 13. The relevant parameters of each component are shown in Table 13, derived from the literature [24, 26]. In table 13, m_l^p , m_l^f denotes the characteristic constants of the age reduction factor for preventive maintenance action and corrective maintenance action, respectively. t_l^0 , t_l^p , t_l^f denotes the fixed maintenance time, preventive maintenance time, and corrective maintenance time, respectively. β_l and η_l denote the shape and scale parameters of the Weibull distribution. $B(k)$ denotes the effective age of the component at the beginning of the kth break. $X(k)$ denotes the state of each system component at the beginning of the kth break.

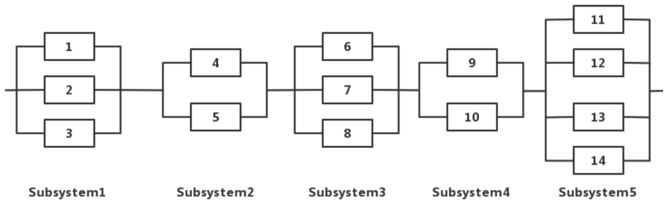

Fig. 13. Five-stage 14-element system structure sketch

Each component has 8 different of maintenance action l. $L(l=7)$ represents the highest maintenance level, where $l=0$ and $l=7$ denote no maintenance and replacement, respectively. For functioning components, $l=1$ ~6 indicates imperfect maintenance. For failed components, $l=1$ indicates minimal repair, and $l=2$~6 indicates imperfect maintenance. When $l>1$, the time for maintenance action l is $t=t_{ij,l}+t_l^0$, where $t_{ij,l}$ is expressed as follows:

$$t_{ij,l} = \begin{cases} \dfrac{(l_{ij}-1)t_l^f}{L-1} & X_{\text{break,s}} = 0 \\ \dfrac{l_{ij}\cdot t_l^f}{L} & X_{\text{break,s}} = 1 \end{cases} \tag{23}$$

Table 13. Relevant parameter values for each component (time is in days)

ID	β_l	η_l	m_l^p	m_l^f	t_l^p	t_l^f	t_l^0	$B(k)$	$X(k)$
1	1.5	25	2.5	2.5	0.13	0.25	0.03	35	1
2	2.4	38	2.2	2.0	0.2	0.31	0.03	24	0
3	1.6	28	2.6	3.0	0.2	0.33	0.03	45	0
4	2.6	40	2.2	3.2	0.12	0.32	0.04	35	0
5	1.8	28	1.8	4.0	0.21	0.34	0.02	28	1
6	2.4	34	2.4	3.2	0.14	0.19	0.03	36	1
7	2.5	26	2.8	3.0	0.2	0.27	0.05	44	0
8	2.0	28	2.3	2.8	0.17	0.31	0.05	28	0
9	1.2	26	2.0	2.5	0.18	0.26	0.04	38	1
10	1.4	35	2.5	2.8	0.2	0.32	0.05	15	0
11	2.8	40	3.2	3.0	0.21	0.31	0.07	30	0
12	1.5	35	2.6	2.2	0.23	0.33	0.04	22	1
13	2.4	30	2.8	2.8	0.16	0.35	0.06	38	1
14	2.2	45	2.2	2.6	0.14	0.35	0.05	35	0

where $t_{ij,l}$ denotes the maintenance time to perform action l on component C_{ij}. l_{ij} denotes the selected maintenance action for component C_{ij}.

The age reduction factor $b_{ij,l}$ is calculated as follows:

$$b_{ij,l} = \begin{cases} 1-\left(\frac{t_{ij,l}}{t_l^f}\right)^{\frac{1}{m_l^f}} & X_{\text{break,s}} = 0 \\ 1-\left(\frac{t_{ij,l}}{t_l^p}\right)^{\frac{1}{m_l^p}} & X_{\text{break,s}} = 1 \end{cases} \tag{24}$$

5.3.1. Algorithm performance analysis

Assuming that the kth mission has just been completed now, the duration Z_k of the kth break obeys a truncated normal distribution of $N(3, 0.0625)$ with range of [2.5, 3.5], $\tau = 0.8$, and the duration of the k+1th mission U=10 days. All other component parameters are shown in Table 13. Among the parameters related to the GA algorithm, the number of populations NP=150, the crossover rate pc=0.8, the variation rate pm=0.05, and the maximum number of iterations equal to 4000. The parameters related to Q-learning, the learning rate α=0.02, the discount rate γ=0.5, and the maximum number iterations equal to 25000. Due to the stochastic of the algorithm, 10 sets of simulations are performed for each method to find its optimal maintenance policy A_{best}, the maintenance time for the optimal maintenance policy T_{best}, the average reliability R_{mean}, the maximum reliability R_{best}, the variance R_{std} and the average running time $\bar{S}$ as the comparison results. In addition, a parameter *%QOS* is introduced here as a performance metric to compare the quality of RL and GA solution, *%QOS*=(R_{best}-R_{mean})/R_{best}. The comparison results and performance metric results are shown in Table 14.

From Table 14, we can see that the optimal maintenance policy solved by Q-learning is better than GA, and the maximum reliability R_{best} is 1.23% higher. Furthermore, the mean and variance of Q-learning results are better than GA in 10 solving results, indicating that the Q-learning algorithm's stability is better than GA. Combined with the experimental results in previous section, the Q-learning algorithm effectively solves the selective maintenance problem and can obtain better values than the GA algorithm. Regarding the computation time, the average time taken by GA is more than twice of RL. Regarding the quality of the obtained solutions, the optimal solution of RL is better than that of GA, and the average solution of RL deviates from the optimal solution by only 1.1%, while the average solution of GA deviates from the optimal solution by 1.38%. Therefore, the RL algorithm can find higher quality solutions and further verifies the effectiveness of the algorithm.This case also illustrates that the advantages of RL are more pronounced for more complex systems.

The iterative evolution of the proposed RL algorithm is shown in Fig. 14. During the initial 15000 iterations, the agent randomly explores all possible maintenance actions, and the Q matrix's value converges slowly. After the first 15,000 iterations of random exploration learning, the Q matrix gradually converges and can eventually reach the convergence state.

Table 14. Comparison results of the two algorithms(time is in days)

Method	A_{best}	T_{best}	R_{best}	R_{mean}	R_{std}	$\bar{S}$	*%QOS*
Q-learning	[0,7,3,7,6,7,6,4,7,7,3,0,0,6]	2.795	0.9414	0.9314	0.0031	71.3	1.1
GA	[0,7,3,7,4,5,6,3,5,7,3,7,5,1]	2.745	0.9291	0.9171	0.0061	148.3	1.38

Table 15. Difference between stochastic and determined break duration(time is in days)

Case	Maintenance policy	Reliability	Maintenance time
Stochastic	[0,7,3,7,6,7,6,4,7,7,3,0,0,6]	0.9414	2.795
Deterministic strategy substitution in the uncertainty model	[0,6,7,7,6,6,7,4,0,7,2,0,7,2]	0.924	2.703

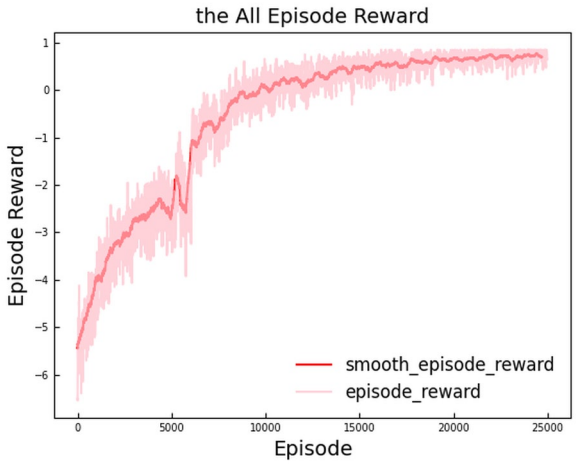

Fig. 14. The training process of the proposed Q-learning algorithm

5.3.2. Comparison between stochastic and determinism of break duration

When the duration of the break is stochastic, it can be seen from 5.3.1 that the optimal maintenance policy $A1$=[0,7,3,7,6,7,6,4,7,7,3,0,0,6], the reliability of the system successfully completing next mission $R2$=0.9414, and the maintenance time is 2.795 days. When the break duration Z_k=3 is a fixed value and other parameters remain unchanged, the optimal maintenance policy $A2$=[0,6,7,7,6,6,7,4, 7,7,2,0,7,2], the system reliability R=0.944, and the maintenance time is 2.923 days. To compare the difference between stochastic and determinism, the maintenance policy $A2$ solved for the deterministic case is substituted into the uncertainty model $R1$=0.924. As seen in Table 15, the uncertainty is not negligible, and the difference between the deterministic maintenance policy and the policy substituted into the uncertainty model is 2%. It can also be seen that the strategy considering uncertainty is better than the deterministic one with a reliability 1.74% higher. Based on the above observations, it can be concluded that the optimal solution in the deterministic case does not guarantee the maximum reliability for successful completion of the next mission in the uncertainty case if the uncertainty in the break duration is ignored.

6. Conclusions and future works

This paper presents a new selective maintenance model for a multi-component system with the decision to maximize the system's reliability to complete the next mission. The components can be maintained during the break between two adjacent missions, each with several optional maintenance actions from minimal repair and imperfect maintenance to replacement. At the same time, this selective maintenance optimization model considers the break duration stochastic, represented by an appropriate probability distribution. The selective maintenance optimization problem is modeled as a Markov Decision Process (MDP). Based on the framework of the MDP, a RL approach is proposed to overcome the problems of complexity and low computational efficiency in solving the model by traditional methods. By analyzing three cases, the accuracy of the model and the RL method are demonstrated to be effective in finding the optimal maintenance strategy. By comparing with the GA method, the more complex the system the more obvious the advantage of RL. The RL can obtain a better maintenance policy making the system more reliable to successfully complete the next mission, and the computation takes much less time than GA. It is also demonstrated that the stochastic of the break duration affects the maintenance policy and the reliability of the system successfully complete the next mission. Ignoring the stochastic of the break duration the reliability of the system successfully complete the next mission will be overestimated and may prevent the system from completing the next mission. Therefore, it is necessary to investigate the optimization of selective maintenance under uncertainty.

In future works, we will explore several questions. Here we study systems consisting of two-state multiple components, where the break is the only uncertain maintenance resource. The process has several intermediate states in practical engineering from function to failure. In addition, the maintenance time required for different maintenance actions may be stochastic due to the different skill levels of different technicians. In the future, we will conduct research for multi-state multi-component systems and other uncertain maintenance resources.

Acknowledgments

This research was supported by the National Key Research and Development Program of China "Manufacturing Basic Technology and Key Components" key project (Grant No. 2021YFB2011200). This work was also supported by the National Natural Science Foundation of China (Grant No.71501148).

References

1. Andriotis C P, Papakonstantinou K G. Managing engineering systems with large state and action spaces through deep reinforcement learning. Reliability Engineering & System Safety, 2019; 191: 106483, https://doi.org/10.1016/j.ress.2019.04.036.
2. Barde S R A, Yacout S, Shin H. Optimal preventive maintenance policy based on reinforcement learning of a fleet of military trucks. Journal of Intelligent Manufacturing, 2019; 30(1): 147-161, https://doi.org/10.1007/s10845-016-1237-7.
3. Baxter L A, Kijima M, Tortorella M. A point process model for the reliability of a maintained system subject to general repair. Stochastic models, 1996; 12(1): 12-1, https://doi.org/10.1080/15326349608807372.
4. Cao W, Jia X, Hu Q, et al. A literature review on selective maintenance for multi-unit systems. Quality and Reliability Engineering International 2018; 34(5): 824-845, https://doi.org/10.1002/qre.2293.
5. Cao W, Jia X, Hu Q, et al. Selective maintenance for maximising system availability: a simulation approach. International Journal of Innovative Computing and Applications 2017; 8(1): 12-20, https://doi.org/10.1504/ijica.2017.082493.
6. Cao W, Li F, Ran Q. Study on selective maintenance optimization for multi-State systems confronting random missions. Journal of Ordnance Engineering College 2017; 29(2): 17-22.
7. Cassady C R, Murdock Jr W P, Pohl E A. Selective maintenance for support equipment involving multiple maintenance actions. European Journal of Operational Research 2001; 129(2): 252-258, https://doi.org/10.1016/s0377-2217(00)00222-8.
8. Chen Y, Jiang T, Liu Y. Selective maintenance optimization: research advances and challenges. Operations Research Transactions, 2019, 23(3): 27-46, 10.15960/j.cnki.issn.1007-6093.2019.03.003.
9. Chen Y, Ma Y, Liu Q, et al. Research on selective maintenance decision-making of equipment considering imperfect maintenance under sequential mission. AERO WEAPONRY 2019.
10. Chen Z, Zhang L, Tian G, et al. Economic maintenance planning of complex systems based on discrete artificial bee colony algorithm. IEEE Access 2020; 8: 108062-108071, https://doi.org/10.1109/ACCESS.2020.2999601.

11. Diallo C, Venkatadri U, Khatab A, et al. Optimal selective maintenance decisions for large serial k-out-of-n: G systems under imperfect maintenance. Reliability Engineering & System Safety 2018; 175: 234-245, https://doi.org/10.1016/j.ress.2018.03.023.
12. Dui H, Zheng X, Zhao QQ, Fang Y. Preventive maintenance of multiple components for hydraulic tension systems. Eksploatacja i Niezawodnosc – Maintenance and Reliability 2021; 23 (3): 489–497, https://doi.org/10.17531/ein.2021.3.9.
13. Gao H, Zhang X, Yang X, et al. Optimal selective maintenance decision-making for consecutive-mission systems with variable durations and limited maintenance Time. Mathematical Problems in Engineering, 2021; (2021), https://doi.org/10.1155/2021/5534659.
14. Hu M. Research on maintenance decision of wind turbine components based on reinforcement learning. School of Energy Power and Mechanical Engineer.
15. Huang J, Chang Q, Arinez J. Deep reinforcement learning based preventive maintenance policy for serial production lines. Expert Systems with Applications 2020; 160: 113701, https://doi.org/10.1016/j.eswa.2020.113701.
16. Hu Y, Miao X, Zhang J, et al. Reinforcement learning-driven maintenance strategy: A novel solution for long-term aircraft maintenance decision optimization. Computers & Industrial Engineering, 2021; 153: 107056, https://doi.org/10.1016/j.cie.2020.107056.
17. Khatab A, Aghezzaf E H, Djelloul I, et al. Selective maintenance for series-parallel systems when durations of missions and planned breaks are stochastic. IFAC-PapersOnLine 2016; 49(12): 1222-1227, https://doi.org/10.1016/j.ifacol.2016.07.677.
18. Khatab A, Aghezzaf E H, Djelloul I, et al. Selective maintenance optimization for systems operating missions and scheduled breaks with stochastic durations. Journal of manufacturing systems 2017; 43: 168-177, https://doi.org/10.1016/j.jmsy.2017.03.005.
19. Khatab A, Aghezzaf E H. Selective maintenance optimization when quality of imperfect maintenance actions are stochastic. Reliability engineering & system safety 2016; 150: 182-189, https://doi.org/10.1016/j.ress.2016.01.026.
20. Khatab A, Aghezzaf E L H, Diallo C, et al. Selective maintenance optimisation for series-parallel systems alternating missions and scheduled breaks with stochastic durations. International Journal of Production Research 2017; 55(10): 3008-3024, https://doi.org/10.1080/00207543.2017.1290295.
21. Li Z, Xu Y, Gao S, et al. Reliability modeling for repairable multi-state Elements based on Markov process. AERO WEAPONRY, 2018, 10.19297/j.cnki.41-1228/tj.2018.05.012.
22. Li Z, Zhong S, Lin L. An aero-engine life-cycle maintenance policy optimization algorithm: Reinforcement learning approach. Chinese Journal of Aeronautics 2019; 32(9): 2133-2150.
23. Lin D, Zuo M J, Yam R C M. General sequential imperfect preventive maintenance models. International Journal of reliability, Quality and safety Engineering 2000; 7(03): 253-266, https://doi.org/10.1142/S0218539300000213.
24. Liu Y, Chen Y, Jiang T. Dynamic selective maintenance optimization for multi-state systems over a finite horizon: A deep reinforcement learning approach. European Journal of Operational Research 2020; 283(1): 166-181, https://doi.org/10.1016/j.ejor.2019.10.049.
25. Liu Y, Chen Y, Jiang T. On sequence planning for selective maintenance of multi-state systems under stochastic maintenance durations. European Journal of Operational Research 2018; 268(1): 113-127, https://doi.org/10.1016/j.ejor.2017.12.036.
26. Liu Y, Huang H Z. Optimal selective maintenance strategy for multi-state systems under imperfect maintenance. IEEE Transactions on Reliability 2010; 59(2): 356-367, https://doi.org/10.1109/TR.2010.2046798.
27. Lust T, Roux O, Riane F. Exact and heuristic methods for the selective maintenance problem. European journal of operational research 2009; 197(3): 1166-1177, https://doi.org/10.1016/j.ejor.2008.03.047.
28. Lv X Z, Yu Y L, Zhang L, et al. Stochastic program for selective maintenance decision considering diagnostics uncertainty of built-in test equipment. IEEE 2011; 584-589, https://doi.org/10.1109/ICQR2MSE.2011.5976681.
29. Mahmoodzadeh Z, Wu K Y, Lopez Droguett E, et al. Condition-based maintenance with reinforcement learning for dry gas pipeline subject to internal corrosion. Sensors 2020; 20(19): 5708, https://doi.org/10.3390/s20195708.
30. Malik M A K. Reliable preventive maintenance scheduling.AIIE transactions, 1979; 11(3):221-228, https://doi.org/10.1080/05695557908974463.
31. Martínez-Tenor A, Fernández-Madrigal J A, Cruz-Martín A, et al. Towards a common implementation of reinforcement learning for multiple robotic tasks. Expert Systems with Applications 2018; 100: 246-259, https://doi.org/10.1016/j.eswa.2017.11.011.
32. Nakagawa T. Optimum policies when preventive maintenance is imperfect. IEEE Transactions on Reliability 1979; 28(4): 331-332, https://doi.org/10.1109/TR.1979.5220624.
33. Pandey M, Zuo M J, Moghaddass R, et al. Selective maintenance for binary systems under imperfect repair. Reliability Engineering & System Safety 2013; 113: 42-51, https://doi.org/10.1016/j.ress.2012.12.009.
34. Paraschos P D, Koulinas G K, Koulouriotis D E. Reinforcement learning for combined production-maintenance and quality control of a manufacturing system with deterioration failures. Journal of Manufacturing Systems, 2020; 56: 470-483, https://doi.org/10.1016/j.jmsy.2020.07.004.
35. Peng S. Reinforcement learning with Gaussian processes for condition-based maintenance. Computers & Industrial Engineering 2021; 158: 107321, https://doi.org/10.1016/j.cie.2021.107321.
36. Rajagopalan R, Cassady C R. An improved selective maintenance solution approach. Journal of Quality in Maintenance Engineering 2006; 12(2):172-185, https://doi.org/10.1108/13552510610667183.
37. Rice W F, Cassady C R, Nachlas J A. Optimal maintenance plans under limited maintenance time. Proceedings of the seventh industrial engineering research conference: 1998: 1-3.
38. Ruan J H, Wang Z X, Chan F T S, et al. A reinforcement learning-based algorithm for the aircraft maintenance routing problem. Expert Systems with Applications 2021; 169: 114399, https://doi.org/10.1016/j.eswa.2020.114399.
39. Su Y, Meng L, Kong X, et al. Generative adversarial networks for gearbox of wind turbine with unbalanced data sets in fault diagnosis. IEEE Sensors Journal, 2022; https://doi.org/10.1109/JSEN.2022.3178137.
40. Sun Y, Sun Z. Selective Maintenance on a Multi-State Transportation System Considering Maintenance Sequence Arrangement. IEEE Access 2021; 9: 70048-70060, https://doi.org/10.1109/ACCESS.2021.3078140.
41. Tanwar M, Rai R N, Bolia N. Imperfect repair modeling using Kijima type generalized renewal process. Reliability Engineering & System Safety 2014; 124: 24-31, https://doi.org/10.1016/j.ress.2013.10.007.
42. Wang H, Pham H. A quasi renewal process and its applications in imperfect maintenance. International journal of systems science 1996; 27(10): 1055-1062, https://doi.org/10.1080/00207729608929311.
43. Wang S, Zhang S, Li Y, et al. Selective maintenance decision-making of complex systems considering imperfect maintenance. International

Journal of Performability Engineering 2018; 14(12): 2960, https://doi.org/10.23940/IJPE.18.12.P6.29602970.
44. Xu Q Z, LM Guo. Method for solving the selective maintenance problem for series-parallel system. Machinery Design & Manufacture 2016; 0(1): 61-65, https://doi.org/10.3969/j.issn.1001-3997.2016.01.017.
45. Yan J, Zhang Q, Hu X. Review of path planning techniques based on reinforcement learning. Computer Engineering 2021; 47(10):10, 10.19678/j.issn.1000-3428.0060683.
46. Yang Z, Qi C. Preventive maintenance of a multi-yield deteriorating machine:Using reinforcement learning . Systems Engineering-Theory & Practice, 2013, 33(7): 1647-1653.
47. Yousefi N, Tsianikas S, Coit D W. Dynamic maintenance model for a repairable multi-component system using deep reinforcement learning. Quality Engineering, 2022; 34(1):16-35, https://doi.org/10.1080/08982112.2021.1977950.
48. Zhao J, Liu J, Zhao Z, et al. A high-performance maintenance strategy for stochastic selective maintenance. Concurr Comp-Pract E 2019; 31(12): e4840, https://doi.org/10.1002/cpe.4840.
49. Zhao X, Al-Khalifa K N, Hamouda A M, et al. Age replacement models: A summary with new perspectives and methods. Reliability Engineering & System Safety 2017; 161: 95-105, https://doi.org/10.1016/j.ress.2017.01.011.