International Journal of Network Dynamics and Intelligence Scilight

Survey/review study

Reinforcement Learning for Disassembly System Optimization Problems: A Survey

Xiwang Guo [1,2,*], Zhiliang Bi [2], Jiacun Wang [1], Shujin Qin [3], Shixin Liu [4], and Liang Qi [5]

[1] Department of Computer Science and Software Engineering, Monmouth University, New Jersey 07710, USA
[2] Department of Information Control, Liaoning Petrochemical University, Fushun 113005, China
[3] Department of Economic Management, Shangqiu Normal University, Shangqiu 476000, China
[4] Department of Information Science and Engineering, Northeast University, Shenyang 110819, China
[5] Department of Computer Intelligence Science and Technology, Shandong University of Science and Technology, Qingdao 266590, China
[*] Correspondence: x.w.guo@163.com

Received: 16 October 2022
Accepted: 28 November 2022
Published: 27 March 2023

Abstract: The disassembly complexity of end-of-life products increases continuously. Traditional methods are facing difficulties in solving the decision-making and control problems of disassembly operations. On the other hand, the latest development in reinforcement learning makes it more feasible to solve such kind of complex problems. Inspired by behaviorism psychology, reinforcement learning is considered as one of the most promising directions to achieve universal artificial intelligence (AI). In this context, we first review the basic ideas, mathematical models, and various algorithms of reinforcement learning. Then, we introduce the research progress and application subjects in the field of disassembly and recycling, such as disassembly sequencing, disassembly line balancing, product transportation, disassembly layout, etc. In addition, the prospects, challenges and applications of reinforcement learning based disassembly and recycling are also comprehensively analyzed and discussed.

Keywords: disassembly lines; reinforcement learning; Q-learning; deep Q-learning

1. Introduction

With the development of economy and improvement of people's consumption level, people upgrade consumer products more frequently than ever, and this causes a lot of resource waste. As a consequence, the recycling of waste products has aroused widespread concerns in society. To meet the needs of consumers, enterprises have to constantly use the required resources to update their products, resulting in the trash of a large number of out-of-date products. According to research, there are 1.3 billion tons of metal waste in the world alone, which is expected to grow up to 27 billion tons by 2050 [1,2]. To solve the severe resource and environmental problems faced by the manufacturing industry, enterprises need to consider the environmental impact and resource efficiency [3,4], reduce the environmental hazard of end-of-life (EOL) products, and maximize the synergy between economic benefits and social benefits of enterprises. Therefore, the recycling of waste products has become an important support point for new industry [5–7].

As the first step of remanufacturing, disassembly lines strengthen the utilization of products from the perspective of manufacturing industry, reduce the cost of remanufacturing, and effectively decrease the average carbon consumption of products [8]. At the same time, as the interface between recycling and remanufacturing, disassembly lines are affected by many factors [9,10].

The main characteristics are as follows. (1) The different recovery rates of disassembled products bring great uncertainty to disassembly [11]. (2) The disassembly sequences and technologies required by different products during disassembly are greatly different [12]. (3) The layout of different disassembly lines may lead to different work efficiency and characteristics [13]. (4) The balance problem of the disassembly line and the different structures of the workstation lead to differences in the model [3,14]. (5) The different pursuit of objective functions, such as maximum profit and minimum carbon emission, will lead to different objectives [15].

In the face of the above characteristics, previous traditional algorithms, such as pure mathematical models [16,17], Bellman optimization formulas or heuristic algorithms [18,19], such as the genetic algorithm [20], ant colony algorithm [21], artificial bee colony algorithm need to code or establish complex problem models and need a lot of iterations to find an excellent feasible solution. In addition, most traditional methods that only consider static environments also face certain challenges in dealing with environment and condition changes [22].

In order to deal with these problems, more effective methods are needed for the prediction, detection, decision planning, and parameter optimization of disassembly lines [23,24]. The characteristics of disassembly lines determine that a solution is needed with fast learning speed, high intelligence, strong expandability, to quickly adjust the constraints. For example, reinforcement learning (RL) can transform most problems in disassembly lines into Markov decision-making processes that can be solved by RL algorithms, and simulated through environment settings.

RL is a machine learning method based on the interaction between the agents and environment, and is widely used in information theory, game theory, automatic control, artificial intelligence (AI), and other fields. The RL is capable of maintaining a balance between exploration and utilization and has achieved great success in disassembly sequencing, disassembly layout design, and disassembly line balancing.

Compared with traditional AI methods, RL makes decisions according to the current state and evaluates the behavior with rewards [25,26], and in contrast to heuristic methods, RL is more stable in terms of model and convergence. In the face of large-scale data problems, RL introduces deep neural networks to evaluate the value function, which effectively improves the processing ability of complex problems and high-dimensional data. At the same time, experience playback and asynchronous concurrent methods are introduced in some deep reinforcement learning (DRL) to accelerate the convergence speed.

In this paper, we aim to discuss various research branches of disassembly lines, analyze the latest problems and solutions of RL (that have been applied to disassembly lines) in a systematically way, and introduce various algorithms and internal mechanisms of RL applications, advantages, disadvantages and environment settings.

The rest of the paper is as follows. In Section 2, we first describe the problems with disassembly lines and several research directions for disassembly lines. In Section 3, the basic model of RL and several typical algorithms are introduced and classified. In Section 4, some application cases and corresponding analyses of RL in disassembly lines are presented. In Section 5, the prospects and challenges of RL for disassembly are discussed.

2. Problem Description

After years of exploration, many papers have been published in the field of disassembly lines [27]. These papers cover a wide range of decision-making, control, and optimization problems in disassembly systems, mainly including disassembly sequences, disassembly line balance, disassembly line layout, product transportation, etc. This section reviews some typical disassembly research subjects (Figure 1).

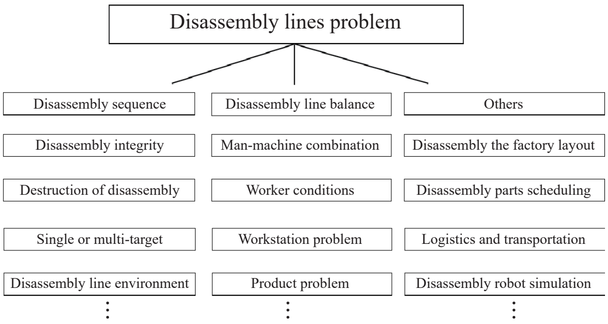
Disassembly lines problem
Disassembly sequence
Disassembly line balance
Others
Disassembly integrity
Man-machine combination
Disassembly the factory layout
Destruction of disassembly
Worker conditions
Disassembly parts scheduling
Single or multi-target
Workstation problem
Logistics and transportation
Disassembly line environment
Product problem
Disassembly robot simulation
⋮
⋮
⋮

Figure 1. Problem classification of disassembly lines.

2.1. Disassembly Sequence

To disassemble a product, we need to figure out the disassembly sequence, select the disassembly method, and perform the actual disassembly operations. Finding the disassembly sequence is the first step of disassembly process and also the key part of remanufacturing and waste recycling industry [28]. It can be divided into several categories according to the factors and conditions to be considered [29].

(1) According to the integrity, disassembly can be divided into complete disassembly and selective disassembly. Complete disassembly refers to the separation of all parts of a whole EOL product, while selective disassembly only disassembles the target parts and high-value parts.

(2) These potential possibilities will influence the decision of decomposition sequences. Some operations in the disassembly process may cause damage to some parts, resulting in changes in the value of the parts. This damage operation sometimes requires consideration of destructiveness in the disassembly sequence.

(3) The objectives of disassembly. For example, the single objective model with the maximum profit or the

minimum carbon emission, or the multi-objective model with multiple-objective Pareto solutions.

RL can play a great role in the process of constructing the optimal decision-making strategy for disassembly sequences. Figure 2 illustrates the process of choosing the existing conditions to obtain the optimal solution [30]. In [30], RL was used to conceive a disassembly system model (DSM) based on numerical values and precedence conditions, which can quickly learn in the face of high-dimensional data, improve plant efficiency, and provide rapid feedback for the disassembly process [31]. In [32], a maintenance and disassembly sequence planning based on DRL was proposed, which combines VR and RL to provide simulation training of disassembly and maintenance, and this helps greatly reduce both costs of personnel training and disassembly cases.

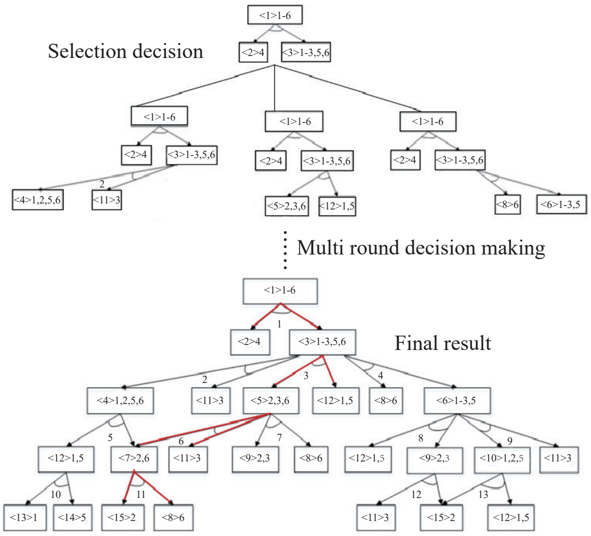

Figure 2. A case of disassembly sequence.

2.2. Disassembly Line Balancing

The disassembly line balancing problem (DLBP) was first proposed by Gungor and Gupta [33], and has evolved to be more and more complex. To date, DLBP has become one of the most important topics in the remanufacturing field attracting extensive attention from the both academic and industrial communities [34,35].

According to the combination of the BLBP case shown in Figure 3 [33] and the current main research, the main concerns of BLBP are given as follows.

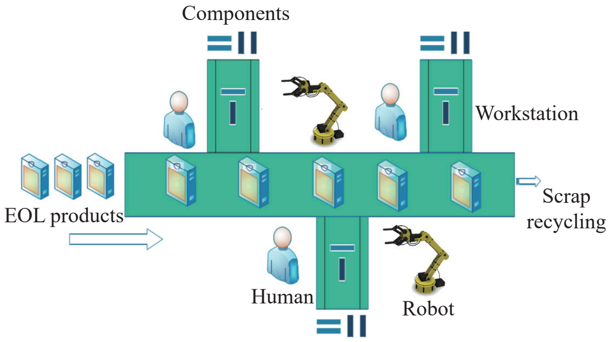

Figure 3. A case of disassembly line balance.

(1) Man-machine combination: the combination of workers and disassembly equipments in the disassembly line, including the minimization of the time difference, the maximization of efficiency, and the number of man-machine.

(2) Personnel conditions: the consideration of the technical level, position and rest time of the staff, and even the physical conditions and personal efficiency [31].

(3) Workstation problems: the concern of the time difference caused by workstation cycles and workstation efficiency.

(4) Product problems: the time imbalance caused by different times required for disassembly of different parts within a single product or multiple products, and the disassembly allocation among multiple products.

As one of the main problems of disassembly lines, RL has been applied to the above problems and a large number of successful cases have been obtained. In [36], the authors solved the multi-robot disassembly line balancing problem through RL, and compared and analyzed the results by testing three kinds of DRL. In [37], the authors used RL and Monte Carlo methods to test the uncertainty on the disassembly lines, highlighting the certainty and universality of RL in the disassembly line balancing problem.

2.3. Other Applications

RL has also been used in a large number of successful studies on other issues in the field of disassembly lines. Here are a few examples:

(1) The use of RL in dealing with the problem of disassembly skills in the remanufacturing industry. The skills required for contact disassembly are summarized with excellent results obtained in [38].

(2) The inventory problem in the disassembly lines of multi-product and multi-demand is studied. The Q-Learning algorithm and random parameters are used to solve a highly dynamic and uncertain environment [39].

(3) Pai [40] and his team introduced the industrial knowledge graph (IKG) and multi-agent reinforcement learning (MARL) based on self-x cognitive manufacturing networks to realize self-configuration solutions and task decomposition, where an example of multi-robot was given to verify accuracy and correctness.

(4) The authors of proposed a control method for a hybrid disassembly system in combination of the RL and comprehensive modeling method, and such a control method can be directly applied to real-world production systems. In addition, the potential of RL was proved in two different test cases in comparison with heuristic algorithms.

As mentioned above, RL approaches have many advantages over traditional methods. First of all, the requirements of RL on mathematical models are not as high as those of traditional methods. Secondly, RL can realize online optimization and real-time disassembly control and feedback. Moreover, RL can better face high-dimensional data and complex environments with faster learning speed. Last but not least, it has been proved by more and more researchers that the expansibility of RL is far higher than that of traditional methods such as heuristic algorithms. However, in the field of disassembly lines, RL also faces many challenges. For example, unlike in the field of image processing, there are not enough standards, unified test cases and comparison index groups to compare and test. The dynamic environment is mostly simulated by random variables, which is not professional enough. Besides, for mixed problems in many aspects, it is sometimes difficult to clearly define and divide the state and action of RL.

3. Reinforcement Learning

The development of RL is a long process from proposing to applying, and many classifications have been gradually derived [41], such as value-based RL, policy-based RL, model-based RL, and model-free RL [42,43] (Figure 4). In Table 1 and Table 2, we compared and distinguished their characteristics to a certain extent, and note that they all need the five basic elements of RL, namely, the subject, state, action, reward, and environment. This paper mainly introduces Q-Learning and Deep-Q-Learning (DQN), which are of important significance for RL and hence widely used in the field of the disassembly production lines. Q-Learning is the basic algorithm of RL and the cornerstone of most value-based RL [44]. DQN is a typical case from RL to the deep learning technology and has many variations and improvements [45].

Table 1 Model-free/model-based comparison

Characteristic	Model-Based	Model-Free
Agent knows all potential rewards for moving from the current state to the next state	Yes	No
Algorithms that sample only from experience	No	Yes
High sampling efficiency	Yes	No
High degree of general-purpose	No	Yes
Use transition or reward functions	Yes	No

Table 2 Value-based/policy-based comparison

Characteristic	Value-Based	Policy-Based
Whether it is effective under continuous action	No	Yes
Learning Polices can be changed	No	Yes
Tend to be globally optimal	Yes	No
Tend to be locally optimal	No	Yes
Directly optimize the policy	No	Yes

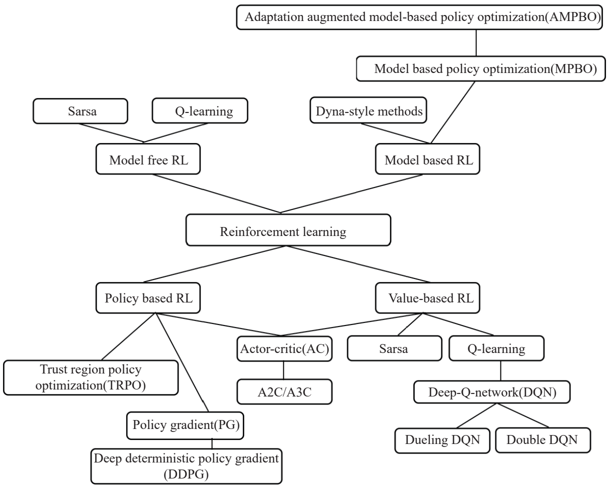

Figure 4. Classification of reinforcement learning.

3.1. Q-Learning

This section shows the construction process of RL algorithms and discusses the mathematical models of Q-Learning.

3.1.1. Fundamentals of Q-Learning

In RL, the agent wanders in an unknown environment and tries to maximize its long-term return by performing operations and receiving rewards. The timing differential control algorithm under the off-track strategy is an important breakthrough in the early stage of RL. This algorithm is called Q-Learning algorithm and is a typical value-based RL. Q is both $Q(s_t, a_t)$, which is equivalent to the s_t state ($s_t \in$ S) that gets the state value in the current state. The action a_t ($a_t \in$ A) taken by the agent is based on the known action income expectation, and the environment E is based on the agent's action feedback (e.g. the new state reached by the action and the corresponding reward r obtained by the action). The main idea of the algorithm is to obtain a corresponding Q-table through the state S and behavior A to store the Q value, and proceed according to the obtained Q value study and choice [46].

In Q-Learning, Q-table is a projection of the target environment. Because RL is also a kind of trial-and-error learning, it can explore and learn knowledge in an unfamiliar environment [47]. Q-Learning uses an agent to interact with the environment, obtain the best behavior for the current environment faced by the agent or strategy through multiple trial-and-errors and evaluations, and update the behavior according to the reward feedback until the Q-table is stable.

The agent, environment, reward, and action can be regarded as a Markov decision process and each process can be counted as a state s_t. $\pi(a_t|s_t)$ refers to the strategy of taking action a_t in the s_t state. $P(s_{t+1}|s_t, a_t)$ represents the probability of the selection action a_t in state s_t that is required to reach the next state s_{t+1}.

3.1.2. Mathematical Model of Q-Learning

Q-Learning is defined as:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma max_a Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \quad (1)$$

Therefore, the objective function of Q-Learning is actually to find the expectation of the largest cumulative reward:

$$\text{Goal}: max_\pi E[\sum_{t=0}^{T} \gamma^t r(s_t, a_t, s_{t+1})|\pi] \quad (2)$$

In RL, the five elements that make up the structure of the RL algorithm are: the state set S, action set A, instant reward r_t, attenuation factor γ. Given strategy π, the algorithm goal is to solve the state value function $v(\pi)$ (Figure 5).

One of the distinguished features of Q-Learning is that it uses the Temporal-Difference (TD) method for sampling and updating. Compared with the Monte-Carlo (MC) method, TD does not require a complete state sequence, and it does not need to run a complete state sequence. The state value is evaluated or updated. Because the code is simple and easy to understand and reform, Q-Learning has been applied in various environments in recent years, such as multi-product and multi-demand disassembly production line inventory management [48], where dynamic fuzzy Q-Learning is adopted to adjust the fuzzy inference system on-line superior [49].

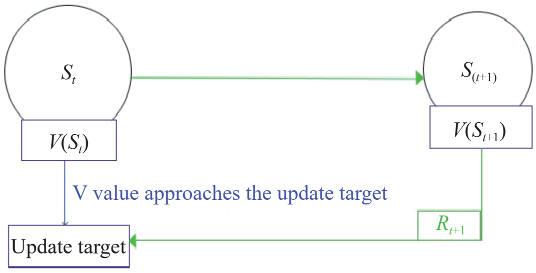

Figure 5. Basic model.

Q-Learning solves the optimal sequence in the Markov decision process through the Bellman equation, in which the state value function $V_\pi(s)$ is used to judge the value of the current state. Each value of the state is not only determined by the value of the current state itself, but also the later reachable state. Therefore, the cumulative reward expectation of the requested state can guide the state value $V(s)$ of the current state s. The Bellman equation of the state value function is given as follows:

$$\begin{aligned} V_\pi(s) &= E_\pi[r_{t+1}+\gamma[r_{t+2}+\gamma[r_{t+3}+\gamma[\cdots]]]|s_t = s] \\ &= E_\pi[r_{t+1}+\gamma V(s_{t+1})|s_t = s] \end{aligned} \tag{3}$$

Combined with the objective function of Q-Learning, we can get the state action-value function of the cumulative optimal value function $V^*\ (s)$ and $Q(s,a)$:

$$V^*\ (s) = max_\pi E[\sum_{t=0}^{H} \gamma^t R(S_t, A_t, S_{t+1})|\pi, s_0 = s] \tag{4}$$

$$q_\pi(s,a) = E_\pi[r_{t+1}+\gamma r_{t+1}+\gamma^2 r_{t+1}+\cdots|A_t = a, S_t = s] \tag{5}$$

where, γ in the formulas (7) and (9) is the attenuation coefficient. When the attenuation coefficient γ is closer to 1, it means that the corresponding agent can see the value of the future state more clearly, and more attention should be paid to the cumulative value of subsequent states. When γ is close to 0, it means that the agent pays more attention to the value of the current state, and is more conservative to the farther value. From 0 to 1, the agent looks farther and farther and pays more attention to future value [50].

$$V(s_t) \leftarrow V(s_t)+\alpha[r_{t+1}+\gamma V(s_{t+1})-V(s_t)] \tag{6}$$

Among them, $r_{t+1}+\gamma V(s_{t+1})$ is called the TD objective function, and $r_{t+1}+\gamma\ V(s_{t+1})-V(s_t)$ is called the TD deviation. α is the rewarding decay coefficient of the decay rate γ. According to the updated formula of TD(0), we adopted, the Q value can be derived to obtain the updated formula of Q-Learning, which is also the Q-Learning mentioned by the above definition:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t)+\alpha[r_{t+1}+\gamma max_{a_{t+1}} Q(s_{t+1},a_{t+1})-Q(s_t,a_t)] \tag{7}$$

The following is the pseudo-code of Q-Learning:

Algorithm 1 Q-learning Algorithm

```
Input: episode,$\gamma,\alpha,\varepsilon$
Output: Q-table
Initialize $Q(s_t,a_t)$, for all $s_t \in S, a_t \in A$, arbitrarily except that $Q(terminal) = 0$
Repeat (for each episode):
    Initialize $s_t$ :
    Choose $a_t$ from $s_t$ using policy derived from
    $Q(e.g,\varepsilon-greedy)$
    Take action $a_t$, observe $r_t, s_{t+1}$
    $Q(s_t,a_t) \leftarrow Q(s_t,a_t)+$
    $\alpha[r_t+\gamma max_{a_{t+1}} Q(s_{t+1},a_{t+1})-Q(s_t,a_t)]$
    $s_t \leftarrow s_{t+1}$
Until $s_t$ is terminal
```

3.2. Deep Q-Network

DQN is a typical case of DRL that combines Q-Learning and neural networks [45]. In the face of the "dimension disaster" formed by the excessive state space and action space, the neural network is used to replace the Q table to avoid the storage of unnecessary data. The basic principle is to: (1) use the memory of the neural network and the processing of large quantities of data, (2) adopt the deep convolutional neural network (DCNN) to approximate the value function, and (3) utilize the empirical playback mechanism to learn from the old or processed data. As a result, the memory of DQN helps the Q value approach the target value.

For the exploration of trial-and-error behaviors we mentioned above, the solution is ε-greedy exploration. Each time the agent randomly selects an action with a certain probability ε. Otherwise, the agent selects an action with the largest Q value among the currently selectable actions.

The pseudo-code of DQN is listed in Algorithm 2. From the algorithm perspective, DQN's modifications to Q-Learning are mainly reflected in the following three aspects.

(1) DQN uses a neural network to approximate the value function;

(2) DQN uses the learning process of experience replay training and RL;

(3) DQN independently sets the target network to deal with the TD deviation in the time difference algorithm separately.

The following is the convergence formula for weight w in the DQN algorithm;

$$\hat{v}(s,w) \approx v_{\pi}(s) \ or \ \hat{q}(s,a,w) \approx q_{\pi}(s,a) \tag{8}$$

where w is the weight, and the neural network or regression algorithm extracts the characteristic value of the input state, uses the TD to calculate the output, and then uses the function to train and converge to w. DQN uses the target network $\hat{Q}$ to calculate y, and updates $\hat{Q}$ with the parameters of Q at every certain step, so that y is not affected by the latest parameters, in exchange for higher stability.

Algorithm 2 Deep Q-Learning with experience replay

```
Initialize replay memory $D$ to capacity $N$
Initialize action-value function $Q$ with random weights $\theta$
Initialize target action-value function $\hat{Q}$ with weight $\theta^{-} = \theta$
for episode $1..M$ do
  Initialize sequence $S_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$
  for $t = 1..T$ do
    With probability $\varepsilon$ select a random action $a_t$;
    otherwise select $a_t = argmax_a Q(\phi(s_t), a; \theta)$
    Execute action $a_t$ in the emulator and observe the
  reward $r_t$ and image $x_{t+1}$
    Set $s_{t+1} = s_t, a_t, x_{t+1}$, $\phi_{t+1} = \phi(s_{t+1})$
    Store experience $(\phi_t, a_t, r_t, \phi_{t+1})$ in $D$
    Sample random minibatch of experiences:
    $(\phi_t, a_t, r_t, \phi_{t+1})$ from $D$
    Set $y_j$ according to Eq. (3)
    Perform a gradient descent step on Eq. (4)
    for the weights $\theta$
    Every $C$ steps reset $\hat{Q} = Q$
  end for
end for
```

$$y_j = \begin{cases} r_j, & \text{if episode terminates at step} \\ r_j + \gamma max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^{-}), & \text{otherwise} \end{cases} \tag{9}$$

$$(y_i - Q(\phi_j, a; \theta))^2 \tag{10}$$

However, when training a neural network, there is an assumption that all training data are independent and identically distributed, but in general there is a certain correlation between the data collected by RL. Using these data for sequential training, the neural network will be unstable. Experience replay can break the correlation between these data and ensure neural network training remains stable. Algorithm 3 lists the pseudo code of experience replay.

Algorithm 3 Experience replay

```
Input: Disassembly environment and initial state Steps:
(1) Make a transition: $(s_t, a_t, r_t, s_{t+1})$
(2) Store recent n transitions in a replay buffer.
(3) Remove old transitions so that the buffer has at most n transitions.
(4) Set buffer capacity $n$, a tuning hyper-parameter and application-specific.
    ·$n$ is typically large. e.g. $10^5$ - $10^6$.
```

DQN is an off-policy algorithm (different policy algorithm). That is, the strategy value functions that update strategy of the generated sample $< s_t, a_t, r_t, s_{t+1} >$ are not the same. The strategy is the ε-greedy strategy, and the strategy for updating the value function is the original strategy. So you can learn from the past, current, and other agents' samples. The experience pool contains samples of the past, or past experiences and memories, which is, on one hand, consistent with the fact that human beings learn knowledge relying on memories. On the other hand, the random addition of experience mentioned in (4) will make neural networks more efficient [51,52]. Moreover, the number of data extracted each time is suitable for the forward and backward propagation of the neural network.

4. RL in Disassembly Lines

In this section, we discuss the application of RL to the field of disassembly lines and meanwhile review existing results. Table 3 provides a quick reference.

Table 3 RL applications on disassembly

Algorithm	Application	Basic element definition				Refs.
		Environmental	**State**	**Action**	**Reward**	
Q-Learning	DLPB	Precedence graph	Disassembly task	Available disassembly task	Idle time Number of workstations High demand parts	[37]
	DLPB	Matrix	Disassembly operation	Operation transfer	Time consumption	[57]
	SDSP	Matrix	Disassembly state	State transfer	Component value	[59]
	SDP	Matrix	Disassembly state	Disassembly operation	Component value	[30]
	ODP	AND/OR graph	Disassembly state	Disassembly operation	Component value	[59]
	Robot	Gazebo	Tool position	Tool position movement	Action function	[60]
Dual Q-Learning	AJSSP	Sheduling rules	Feature combination	Select time Select level	Processing time	[61]
DQN	SDSP	Matrix	Disassembly parts	Available disassembly parts	Disassembly time Disassembly profit	[53]
	HRC	Buffer set	Human state Robot state	Human operation Robot operation	Time consumption	[54]
	DLBP	Buffer set	Workstation state	Number of workstations	Time consumption	[36]
DDPG	HRC	State combination	Human state Robot state Execution state	Individual task selection Collaborative task selection	Global assembly progress	[28]
A2C	DLBP	Matrix	Feature combination	Task selection Workstation selection	Time consumption	[62]
PPO	DLBP	AND/OR graph	Subassembly Workstation	Subassembly selection Workstation selection	Comprehensive indicators	[63]
RL	DSM	Multiple graphs	Decision information	Operations Workstations	Resource cost Punishes time Failure punishment	[31]
DRL	HRC	RGB image	RGB image	Image change	Comprehensive indicators	[64]
Model-Based RL	CMfg	Decision model	Current decision condition	Decision	Schedule quality	[56]

4.1. Basic Elements of RL

4.1.1. Environment

RL is a trial and error learning in the environment, and the experience gained is also based on the environment. In the current application field, there are many ways to map disassembly lines to the environment. For example, precedence graphs in DLBP are used as the basis of the environment in [37]. The authors of [30] and [53] adopted Q-Learning and DQN, respectively, to learn the disassembly sequences, but both used a matrix as the basic environment to facilitate the definition of the precedence relationship. Liu et al. [54] set up a man-machine combination environment through OpenAI, adopted the dual agent mode, took the operation of the machine as the basis, and took the agent as the simulation of workers' activities, thus obtaining a virtual environment of man-machine combination.

4.1.2. State

For different application problems, the definition of states is different. According to the characteristics of the disassembly line field, most of the current research defines the disassembly tasks that can be divided separately in the environment as states, but their parameters are different according to the research problems. For example, in [37], each state contained attributes such as workstation and idle time, and one state is determined by the combination of multiple attributes. The work presented in [30] the disassembled parts as the state, which includes the component information, disassembly time, and other attributes, meanwhile providing the agent with effective information about the disassembled parts accurately and strengthening the agent's ability to disassemble EOL products.

4.1.3. Action

The ability of action selection and planning is also an important indicator of the evaluation algorithm in RL. The action is the embodiment of the transition between states, and the agent transfers from one state to the next through the action. For example, in [30], the disassembly operation was taken as an action which was subsequently adjusted after each new state reached. Later on, the action set faced by each state was minimized to improve the exploration efficiency and calculation speed. The authors of [55] regarded the disassembly operation as a state and the transition between operations as an action, so that the matrix of the disassembly environment became an $n \times n$ matrix that helped better utilize its properties. In [53], the action was chosen as the set of the disassembled parts that can be obtained when each disassembly state reached, and such disassembled parts were expressed by the relational matrix to effectively improve the universality and make the neural network more convenient to process relevant data.

4.1.4. Reward

In a learning process, the agent gets different rewards when choosing different removable parts. The reward is the basis of RL using to feedback on the action of the agent, and is also regarded as the benchmark to measure the probability of reaching any state.

The reward in RL is generally set according to the target value. In [56] , according to the characteristics of minimizing the time, a two-objective optimization model was established to minimize the total maximum completion time and logistics distance, and the reward was set as a two-dimensional vector. Tuncel [37] took the reciprocal after normalizing the multi-objective to ensure that Q-Learning obtained the maximum value of Q value. In [54], in the case of dynamic environment, the time of human-machine coordination was used as a reward. Zhao et al. [53] based on the characteristics of selective disassembly, the two indicators of disassembly time and disassembly profit were taken as rewards for the agent, and the hierarchical mechanism set by its algorithm was effectively used to further expand the problems faced by selective disassembly.

4.2. Relevant Application and Research

After years of research, many papers have been published on RL in the field of disassembly and remanufacturing, especially after 2020. These applications involve not only the basic problems mentioned above but also cross-cutting fields, including disassembly planning, cloud manufacturing, product scheduling, operation control, etc. Next, we will discuss the latest progress in related fields in recent years.

For example, in one of the latest publications , based on DRL, a Markov decision-making process was used to further deepen the problem of disassembly sequences. Combined with virtual reality technology, the sequence can be created in a dynamic environment according to the basic input in the virtual reality training system. In [65], considering the modern manufacturing environment, RL was used in combination with industrial insertion tasks with visual inputs to solve the inaccuracy problem of the controller caused by the difficulty of establishing the relevant physical effect model in the first-order modeling. In the research of [64] about sustainable manufacturing, DRL was taken as one of the key objects for in-depth research, and a specific system in human-robot cooperative disassembly (HRCD) was realized, which proved the feasibility and effectiveness of RL.

In the application of disassembly sequencing, RL has been more widely used in recent years. Because disassembly sequencing itself can be regarded as a decision-making process, RL has more application variants, which ranges from the RL model (for the evaluation of future actions based on the value of components mentioned earlier [66]) to the selective disassembly model (derived from the characteristics of disassembled products proposed in [30]) and the multi-objective high-dimensional model (solved by the DQN algorithm with neural networks in [53]). In terms of implementation, an assembly sequence planning system for workpieces (ASPW) was proposed in [67], where DRL was adopted in case of sparsity rewards and lack of training environments. The presented algorithm ASPW-DQN combined the physical simulation engine Gazebo to provide the required training environment, and established a training platform to facilitate automatic sequence planning of complex assembly products.

At the same time, some researchers have combined RL with traditional heuristic algorithms to handle more complex problems and improve the operation efficiency [51,52]. For example, RL and the genetic algorithm (GA) were used together in [68] for effective maintenance optimization with intermediate buffer inventory. When multi-agent RL is faced with a complex reward function, the excellent global optimization ability of GA was used to guide the decision-making of each agent, and a bilateral interaction was established between the multi-agent RL and GA, which improved the solution quality and speed.

The multi-constraint disassembly problem is also an issue caused by the increase of number of products to be disassembled. The traditional optimization algorithm based on a single product cannot handle the dynamic environment with mixed conditions. For example, in [59], the cooperation with industrial robots, training logic, and environment interaction was considered where a case study of human-robot cooperative assembly task was carried out. The integrated optimization and scheduling problem of multi-state single machine production in industrial applications was studied in [69] by considering the integrated optimization problem (as a Markov decision process) and various factors (such as processing and maintenance costs), where a new heuristic learning method was proposed to deal with the integrated model, thereby improving the learning efficiency of RL. In [70], combined with the Internet of things (IoT), a DRL-based intelligent transfer framework for partial detection was proposed, in which vehicles were observed (that equipped with IoT communication technologies and within the sensing range) and excellent results were obtained under extensive simulations of different topologies, traffic flows and detection intervals.

Some recently published results [57, 58] used Q-Learning as a basic algorithm for the reason that Q-Learning is much faster than improved DQN that is integrated with neural networks [58]. A value matrix was used to optimize SDSP by means of converting the original optimization problem into a directed graph path problem according to the mathematical model, and this expanded the model of the problem [57]. To deal with DLPB, stage goals were set so that Q-Learning can meet the requirements of multiple indicators under certain conditions.

Some reported research works, such as [53], treated the decision problem of SDSP as MDP and proposed DQN-SDSP for EOL products with uncertain structure. DQN-SDSP can adaptively obtain a selective disassembly sequence, and take into account the impact of tools on the disassembly steps for use when the structure of EOL products changes. The DQN algorithm was expanded to avoid the dimension disaster through the neural network, and the memory brought by neural networks was used to accelerate the speed of obtaining the optimal sequence under large-scale data. Furthermore, comparative experiments were conducted with other algorithms in the case.

To summarize, the advantages of applying RL to the field of disassembly are obvious: (1) RL provides an adaptable scheme capable of learning things that are not easy to model; (2) RL is an adaptive algorithm from low-dimensional data to high-dimensional data, especially considering the fact that DRL can handle large-scale high-dimensional data in complex systems [71]; (3) RL can be combined with other approaches to solve a variety of problems, such as image processing and decision-making problems; and (4) RL has a high expansion ability – from Q-Learning to DQN, RL can be expanded into a variety of algorithms [72]. However, RL cannot be used at will in the field of disassembly lines. At present, there are at least the following difficulties. First, the environment is different, and there is no unified basic environment, which leads to a large difference in the same problem. Secondly, the division of basic elements such as states and actions sometimes leads to certain information loss. In the disassembly line system, the variables and constraints are complex, and the selection of the algorithm is a serious problem. Moreover, at present, there is no unified comparison index like deep learning in RL, which makes it difficult to compare various studies.

To solve these problems, we need to pay attention to the following points. First, when determining the RL algorithm, we need to design the environment and actions in advance, and select different kinds of RL according to the characteristics of the problem. Then, when studying action space and state space, we need to design the data structure of reward in combination with the research objectives to avoid RL falling into local optimal solutions. Next, we should be careful at transforming special problems into general RL models such as a Markov decision process, and abstracting specific practical problems into mathematical problems [73]. Last but not the least, specific attention is required when introducing cross fields to optimize the algorithm and model based on the basic principles.

5. Challenges and Future Directions

5.1. Disassembly Line Problems

Most of the existing research results on RL for DLBP take the attribute information of the disassembled product as the state, the changes of the parts in the disassembly as the action, and the self-determined objective function as the reward. However, the following challenges are encountered at present. (1) Once the multi-objective problem is involved, it is necessary to improve the data dimension or preprocess the data such as reward. (2) The data of training environment is relatively small at present, and many cases need to be transformed before they can be used as RL. Therefore, we should first deal with the multi-objective problem, customize a special solution method for the multi-objective Pareto solution set, and use multi-agent or hierarchical design. Secondly, we should study how to investigate a general basic environment. Take the basic environment as the cornerstone, introduce the development direction of each research problem, and finally overcome a large number of inefficient reconstruction caused by the environment.

5.2. Practical Application

Just like the data and environmental challenges mentioned above, there are few professional reference data sets in the field of disassembly lines at present, and the training of RL agent requires a large amount of data to obtain satisfactory results. In fact, most of the current research simulates the specific conditions of the real environment in the virtual environment, and rarely involves the system with sensors or real disassembly operations, which fails to extend the application of RL in the disassembly line to a broad scope of commercial cases. On the other hand, the dismantlement line has high requirements for the overall efficiency, recovery cost, dismantlement profit and other indicators, resulting in the situation that RL can only be in the exploration and research stage for a long time in the absence of cases. Therefore, in order to promote the application of RL in the field of disassembly lines, some key challenges need to be overcome. First, we should continue to improve the algorithm theory of RL to improve the robustness and global convergence ability. Secondly, the mathematical model should be established for the disassembly problem, the application scenario should be expanded, and the interface should be set between multiple problems to ensure the overall coordination. In addition, both support and collaborative research on policy issues such as carbon emissions and remanufacturing are needed.

5.3. Future Development

Even with some challenges, the current RL is still an important part of the AI technology, which has been successful used in many scenarios and application areas, and is attracting more and more attention. Although there are still some obstacles in both theory and technique, there is no doubt about its future development. As for the future development directions of RL, at least the following aspects should be considered. First, for abstraction ability, RL should follow certain standards when defining basic elements, so as to generalize the model. Second, the neural network structure of DRL needs to be combined with the improvement research of DL and other approaches to boost learning ability, e.g. training speed and accuracy [74]. Finally, it is the practical application that strengthens the connection between the virtual environment and real world, and opens the interface between research and industrial technology [75]. Therefore, more effort should be spent on RL applications.

6. Conclusion

A comprehensive review of various studies in the application of reinforcement learning (RL) in disassembly lines is presented in this paper, with the intention to help related researchers to understand what has been done, what still needs to investigate and where the challenges are. The disassembly line balancing problem (DLBP) and its sub branches are discussed with their characteristics also involved. Secondly, the basic classification of RL is discussed, and the most popular and widely used RL algorithms are reviewed in detail. Then, the applications of RL in disassembly sequencing including SDP, DLBP and other issues are discussed and analyzed. Specifically, the corresponding environment definitions, reward settings and basic mechanisms are studied. Finally, we point out the difficulties and future research directions of RL in the field of disassembly lines, so as to promote the further development of the related research.

Author Contributions: Jiacun Wang: The proposer of the paper idea, instructor and the reviser of the paper; **Xiwang Guo:** instructor and the reviser of the paper; **Zhiliang Bi:** data collection and paper writing; **Shixin Liu:** data collation; **Liang Qi:** data collation. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by National Key RD Program of China (2021YFB3301200) and NSFC under Grant Nos. (62073069, 51405075 and 61903229). Liaoning Revitalization Talents Program under Grant No.XLYC1907166; Archival Science and Technology Project of Liaoning Province, No.2021-B-004a; The Natural Science Foundation of ShandongProvince under Grant ZR2022MF270.

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

References

1. Liu, Z.Y.; Li, C.; Fang, X.Y.; *et al*. Energy consumption in additive manufacturing of metal parts. *Procedia Manuf.*, **2018**, *26*:

834−845.

2. Azaria, A.; Richardson, A.; Kraus, S.; *et al*. Behavioral analysis of insider threat: A survey and bootstrapped prediction in imbalanced data. *IEEE Trans. Comput. Soc. Syst.*, **2014**, *1*: 135−155.
3. Mete, S.; Serin, F. A reinforcement learning approach for disassembly line balancing problem. In *Proceedings of 2021 International Conference on Information Technology (ICIT), Amman, Jordan, 14–15 July 2021*; IEEE: Amman, Jordan, 2021; pp. 424–427. doi: 10.1109/ICIT52682.2021.9491689
4. Liu, J.Y.; Zhou, Z.D.; Pham, D.T.; *et al*. Collaborative optimization of robotic disassembly sequence planning and robotic disassembly line balancing problem using improved discrete Bees algorithm in remanufacturing. *Robot. Comput. Integr. Manuf.*, **2020**, *61*: 101829.
5. Igarashi, K.; Yamada, T.; Gupta, S.M.; *et al*. Disassembly system modeling and design with parts selection for cost, recycling and CO_2 saving rates using multi criteria optimization. *J. Manuf. Syst.*, **2016**, *38*: 151−164.
6. Baazouzi, S.; Rist, F.P.; Weeber, M.; *et al*. Optimization of disassembly strategies for electric vehicle batteries. *Batteries*, **2021**, *7*: 74.
7. Battaïa, O.; Dolgui, A.; Heragu, S.S.; *et al*. Design for manufacturing and assembly/disassembly: Joint design of products and production systems. *Int. J. Prod. Res.*, **2018**, *56*: 7181−7189.
8. Tian, G.D.; Zhou, M.C.; Li, P.G. Disassembly sequence planning considering fuzzy component quality and varying operational cost. *IEEE Trans. Autom. Sci. Eng.*, **2018**, *15*: 748−760.
9. Guo, X.W.; Zhang, Z.W.; Qi, L.; *et al*. Stochastic hybrid discrete grey wolf optimizer for multi-objective disassembly sequencing and line balancing planning in disassembling multiple products. *IEEE Trans. Autom. Sci. Eng.*, **2022**, *19*: 1744−1756.
10. Guo, X.W.; Zhou, M.C.; Liu, S.X.; *et al*. Multiresource-constrained selective disassembly with maximal profit and minimal energy consumption. *IEEE Trans. Autom. Sci. Eng.*, **2021**, *18*: 804−816.
11. Harib, K.H.; Sivaloganathan, S.; Ali, H.Z.; et al. Teaching assembly planning using AND/OR graph in a design and manufacture lab course. In *Proceedings of 2020 ASEE Virtual Annual Conference, 22 Jun 2020-26 Jun 2020;* 2020.
12. Tian, G.D.; Ren, Y.P.; Feng, Y.X.; *et al*. Modeling and planning for dual-objective selective disassembly using and/or graph and discrete artificial bee colony. *IEEE Trans. Industr. Inform.*, **2019**, *15*: 2456−2468.
13. Barrett, T.; Clements, W.; Foerster, J.; *et al*. Exploratory combinatorial optimization with reinforcement learning. *Proc. AAAI Conf. Artif. Intelli.*, **2020**, *34*: 3243−3250.
14. Qu, S.H. Dynamic Scheduling in Large-Scale Manufacturing Processing Systems Using Multi-Agent Reinforcement Learning. Ph.D. Thesis, Stanford University, Palo Alto, CA, USA, 2019.
15. Chang, M.M.L.; Ong, S.K.; Nee, A.Y.C. Approaches and challenges in product disassembly planning for sustainability. *Procedia CIRP*, **2017**, *60*: 506−511.
16. Aguinaga, I.; Borro, D.; Matey, L. Parallel RRT-based path planning for selective disassembly planning. *Int. J. Adv. Manuf. Technol.*, **2008**, *36*: 1221−1233.
17. Tseng, H.E.; Chang, C.C.; Lee, S.C.; *et al*. A Block-based genetic algorithm for disassembly sequence planning. *Expert Syst. Appl.*, **2018**, *96*: 492−505.
18. Wu, H.; Zuo, H.F. Using genetic annealing simulated annealing algorithm to solve disassembly sequence planning. *J. Syst. Eng. Electron.*, **2009**, *20*: 906−912.
19. Guo, X.W.; Zhou, M.C.; Abusorrah, A.; *et al*. Disassembly sequence planning: A survey. *IEEE/CAA J. Autom. Sin.*, **2021**, *8*: 1308−1324.
20. Wang, H.; Xiang, D.; Duan, G.H. A genetic algorithm for product disassembly sequence planning. *Neurocomputing*, **2008**, *71*: 2720−2726.
21. Xing, Y.F.; Wang, C.E.; Liu, Q. Disassembly sequence planning based on Pareto ant colony algorithm. *J. Mech. Eng.*, **2012**, *48*: 186−192.
22. Chapman, D.; Kaelbling, L.P. Input generalization in delayed reinforcement learning: An algorithm and performance comparisons. In *Proceedings of the 12th International Joint Conference on Artificial intelligence, Sydney New South Wales Australia, 24–30 August 1991;* Morgan Kaufmann Publishers Inc.: Sydney New South Wales Australia, 1991; pp. 726–731.
23. Guo, X.W.; Zhou, M.C.; Liu, S.X.; *et al*. Lexicographic multiobjective scatter search for the optimization of sequence-dependent selective disassembly subject to multiresource constraints. *IEEE Trans. Cybern.*, **2020**, *50*: 3307−3317.
24. Guo, X.W.; Liu, S.X.; Zhou, M.C.; *et al*. Disassembly sequence optimization for large-scale products with multiresource constraints using scatter search and petri nets. *IEEE Trans. Cybern.*, **2016**, *46*: 2435−2446.
25. Ji, Y.J.; Liu, S.X.; Zhou, M.C.; *et al*. A machine learning and genetic algorithm-based method for predicting width deviation of hot-rolled strip in steel production systems. *Inf. Sci.*, **2022**, *589*: 360−375.
26. Zhao, Z.Y.; Liu, S.X.; Zhou, M.C.; *et al*. Decomposition method for new single-machine scheduling problems from steel production systems. *IEEE Trans. Autom. Sci. Eng.*, **2020**, *17*: 1376−1387.
27. Zhao, Z.Y.; Zhou, M.C.; Liu, S.X. Iterated greedy algorithms for flow-shop scheduling problems: A tutorial. *IEEE Trans. Autom. Sci. Eng.*, **2022**, *19*: 1941−1959.
28. Zhang, R.; Lv, Q.B.; Li, J.; *et al*. A reinforcement learning method for human-robot collaboration in assembly tasks. *Robot. Comput. Integr. Manuf.*, **2022**, *73*: 102227.
29. de Mello, L.S.H.; Sanderson, A.C. AND/OR graph representation of assembly plans. *IEEE Trans. Robot. Autom.*, **1990**, *6*: 188−199.
30. Xia, K.; Gao, L.; Li, WD.; et al. A Q-learning based selective disassembly planning service in the cloud based remanufacturing system for WEEE. In *Proceedings of the ASME 2014 International Manufacturing Science and Engineering Conference collocated with the JSME 2014 International Conference on Materials and Processing and the 42nd North American Manufacturing Research Conference, Detroit, Michigan, USA, 9–13 June 2014;* ASME: Detroit, USA, 2014; pp. V001T04A012. doi: 10.1115/MSEC2014-4008
31. Wurster, M.; Michel, M.; May, M.C.; *et al*. Modelling and condition-based control of a flexible and hybrid disassembly system with manual and autonomous workstations using reinforcement learning. *J. Intell. Manuf.*, **2022**, *33*: 575−591.
32. Mao, H.Y.; Liu, Z.Y.; Qiu, C. Adaptive disassembly sequence planning for VR maintenance training via deep reinforcement learning. *Int. J. Adv. Manuf. Technol.* **2021**, in press. doi: 10.1007/s00170-021-08290-x
33. McGovern, S.M.; Gupta, S.M. A balancing method and genetic algorithm for disassembly line balancing. *Eur. J. Oper. Res.*, **2007**, *179*: 692−708.
34. Guo, X.W.; Liu, S.X.; Zhou, M.C.; *et al*. Dual-objective program and scatter search for the optimization of disassembly sequences subject to multiresource constraints. *IEEE Trans. Autom. Sci. Eng.*, **2018**, *15*: 1091−1103.

35. Bentaha, M.L.; Battaïa, O.; Dolgui, A. An exact solution approach for disassembly line balancing problem under uncertainty of the task processing times. *Int. J. Prod. Res.*, **2015**, *53*: 1807−1818.
36. Mei, K.; Fang, Y.L. Multi-robotic disassembly line balancing using deep reinforcement learning. In *Proceedings of the ASME 2021 16th International Manufacturing Science and Engineering Conference, 21–25 June 2021;* ASME, 2021; pp. V002T07A005. doi: 10.1115/MSEC2021-63522
37. Tuncel, E.; Zeid, A.; Kamarthi, S. Solving large scale disassembly line balancing problem with uncertainty using reinforcement learning. *J. Intell. Manuf.*, **2014**, *25*: 647−659.
38. Serrano-Muñoz, A.; Arana-Arexolaleiba, N.; Chrysostomou, D.; et al. Learning and generalising object extraction skill for contact-rich disassembly tasks: An introductory study. *Int. J. Adv. Manuf. Technol.* **2021**, in press. doi: 10.1007/s00170-021-08086-z
39. Tuncel, E.; Zeid, A.; Kamarthi, S. Inventory management in multi-product, multi-demand disassembly line using reinforcement learning. In *Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management, Istanbul, Turkey, 3–6 July 2012;* Istanbul, Turkey, 2012; pp. 1866–1873.
40. Zheng, P.; Xia, L.Q.; Li, C.X.; *et al*. Towards self-X cognitive manufacturing network: An industrial knowledge graph-based multi-agent reinforcement learning approach. *J. Manuf. Syst.*, **2021**, *61*: 16−26.
41. Wiering, M.; van Otterlo, M. *Reinforcement Learning*; Springer: Berlin, Heidelberg, Germany, **2012**. doi:10.1007/978-3-642-27645-3.
42. Kaiser, L.; Babaeizadeh, M.; Milos, P.; et al. Model based reinforcement learning for Atari. In *Proceedings of the 8th International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020;* OpenReview.net: Addis Ababa, Ethiopia, 2020.
43. Rafati, J.; Noelle, D. C. Learning representations in model-free hierarchical reinforcement learning. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, Honolulu, 27 January–1 February* 2019; AAAI Press: Honolulu, 2019; p. 1303.
44. Watkins, C.J.C.H.; Dayan, P. *Q*-learning. *Mach. Learn.* **1992**, 8, 279–292. doi: 10.1007/BF00992698
45. Osband, I.; Blundell, C.; Pritzel, A.; et al. Deep exploration via bootstrapped DQN. In *Proceedings of the 30th International Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016;* Curran Associates Inc.: Barcelona, Spain, 2016; pp. 4033–4041.
46. van Hasselt, H. Double Q-learning. In *Proceedings of the 23rd International Conference on Neural Information Processing Systems, Vancouver British, Columbia, Canada, 6 –9 December 2010;* Curran Associates Inc.: Vancouver British, Canada, 2010; pp. 2613–2621.
47. Clifton, J.; Laber, E. Q-learning: Theory and applications. *Annu. Rev. Stat. Appl.*, **2020**, *7*: 279−301.
48. Montazeri, M.; Kebriaei, H.; Araabi, B.N. Learning Pareto optimal solution of a multi-attribute bilateral negotiation using deep reinforcement. *Electron. Commer. Res.*, **2020**, *43*: 100987.
49. Er, M.J.; Deng, C. Online tuning of fuzzy inference systems using dynamic fuzzy Q-learning. *IEEE Trans. Syst., Man, Cybern., Part B Cybern.*, **2004**, *34*: 1478−1489.
50. Melo, F.S. *Convergence of Q-Learning: A Simple Proof*; Institute of Systems and Robotics: Lisboa, 2001; pp. 1–4.
51. Zhang, S.T.; Sutton, R.S. A deeper look at experience replay. arXiv preprint arXiv: 1712.01275, 2017. doi:10.48550/arXiv.1712.01275.
52. Fedus, W.; Ramachandran, P.; Agarwal, R.; et al. Revisiting fundamentals of experience replay. In *Proceedings of the 37th International Conference on Machine Learning, 13–18 July 2020;* PMLR, 2020; pp. 3061–3071.
53. Zhao, X.K.; Li, C.B.; Tang, Y.; *et al*. Reinforcement learning-based selective disassembly sequence planning for the end-of-life products with structure uncertainty. *IEEE Robot. Autom. Lett.*, **2021**, *6*: 7807−7814.
54. Liu, Z.H.; Liu, Q.; Wang, L.H.; *et al*. Task-level decision-making for dynamic and stochastic human-robot collaboration based on dual agents deep reinforcement learning. *Int. J. Adv. Manuf. Technol.*, **2021**, *115*: 3533−3552.
55. Zhang, H.J.; Liu, P.S.; Guo, X.W.; et al. An improved Q-learning algorithm for solving disassembly line balancing problem considering carbon emission. In *Proceedings of 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, 9–12 October 2022;* IEEE: Prague, Czech Republic, 2022; 872–877. doi: 10.1109/SMC53654.2022.9945321
56. Chen, S.K.; Fang, S.L.; Tang, R.Z. A reinforcement learning based approach for multi-projects scheduling in cloud manufacturing. *Int. J. Prod. Res.*, **2019**, *57*: 3080−3098.
57. Liu, Y.Z.; Zhou, M.C.; Guo, X.W. An improved q-learning algorithm for human-robot collaboration two-sided disassembly line balancing problems. In *Proceedings of 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, 9–12 October 2022;* IEEE: Prague, Czech Republic, 2022; 568–573. doi: 10.1109/SMC53654.2022.9945263
58. Bi, Z.L.; Guo, X.W.; Wang, J.C.; et al. A Q-learning-based selective disassembly sequence planning method. In *Proceedings of 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, 9 –12 October 2022;* IEEE: Prague, Czech Republic, 2022; pp. 3216–3221. doi: 10.1109/SMC53654.2022.9945073
59. Reveliotis, S.A. Modelling and controlling uncertainty in optimal disassembly planning through reinforcement learning. In *Proceedings of the IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, 26 April 2004-1 May 2004;* IEEE: New Orleans, USA, 2004; pp. 2625–2632. doi: 10.1109/ROBOT.2004.1307457
60. Kristensen, C.B.; Sørensen, F.A.; Nielsen, H.B.; *et al*. Towards a robot simulation framework for E-waste disassembly using reinforcement learning. *Procedia Manuf.*, **2019**, *38*: 225−232.
61. Wang, H.X.; Sarker, B.R.; Li, J.; *et al*. Adaptive scheduling for assembly job shop with uncertain assembly times based on dual Q-learning. *Int. J. Prod. Res.*, **2021**, *59*: 5867−5883.
62. Cai, W.B.; Guo, X.W.; Wang, J.C.; et al. An improved advantage actor-critic algorithm for disassembly line balancing problems considering tools deterioration. In *Proceedings of 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, 9 –12 October 2022;* IEEE: Prague, Czech Republic, 2022; 3336 –3341. doi: 10.1109/SMC53654.2022.9945173
63. Zhong, Z.K.; Guo, X.W.; Zhou, M.C.; et al. Proximal policy optimization algorithm for multi-objective disassembly line balancing problems. In *Proceedings of 2022 Australian & New Zealand Control Conference, Gold Coast, Australia, 24–25 November 2022;* IEEE: Gold Coast, Australia, 2022; pp. 207–212. doi: 10.1109/ANZCC56036.2022.9966864
64. Liu, Q.; Liu, Z.H.; Xu, W.J.; *et al*. Human-robot collaboration in disassembly for sustainable manufacturing. *Int. J. Prod. Res.*, **2019**, *57*: 4027−4044.
65. Schoettler, G.; Nair, A.; Luo, J.L.; et al. Deep reinforcement learning for industrial insertion tasks with visual inputs and natural rewards. In *Proceedings of 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV,*

USA, 24 October 2020-24 January 2021; IEEE: Las Vegas, USA, 2020; pp. 5548–5555. doi: 10.1109/IROS45743.2020.9341714

66. Lowe, G.; Shirinzadeh, B. Dynamic assembly sequence selection using reinforcement learning. In *Proceedings of the IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, 26 April 2004-1 May 2004;* IEEE: New Orleans, USA, 2004; pp. 2633–2638. doi: 10.1109/ROBOT.2004.1307458
67. Zhao, M.H.; Guo, X.; Zhang, X.B.; *et al*. ASPW-DRL: Assembly sequence planning for workpieces via a deep reinforcement learning approach. *Assem. Autom.*, **2020**, *40*: 65−75.
68. Li, B.C.; Zhou, Y.F. Multi-component maintenance optimization: An approach combining genetic algorithm and multiagent reinforcement learning. In *Proceedings of 2020 Global Reliability and Prognostics and Health Management (PHM-Shanghai), Shanghai, China, 16–18 October 2020;* IEEE: Shanghai, China, 2020; pp. 1–7. doi: 10.1109/PHM-Shanghai49105.2020.9280997
69. Yang, H.B.; Li, W.C.; Wang, B. Joint optimization of preventive maintenance and production scheduling for multi-state production systems based on reinforcement learning. *Reliab. Engin. Syst. Saf.*, **2021**, *214*: 107713.
70. Chu, T.S.; Wang, J.; Codecà, L. *et al*. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Trans. Intell. Transp. Syst.*, **2020**, *21*: 1086−1095.
71. Heuillet, A.; Couthouis, F.; Díaz-Rodríguez, N. Explainability in deep reinforcement learning. *Knowl.-Based Syst.*, **2021**, *214*: 106685.
72. Saleh, I.K.; Beshaw, F.G.; Samad, N.M. Deep reinforcement learning WITH a path-planning communication approach for adaptive disassembly. *J. Optoelectronics Laser*, **2022**, *41*: 307−314.
73. Cobbe, K.; Klimov, O.; Hesse, C.; et al. Quantifying generalization in reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, USA, 9–15 June 2019;* PMLR: Long Beach, 2019; pp. 1282–1289.
74. Arulkumaran, K.; Deisenroth, M.P.; Brundage, M.; *et al*. Deep reinforcement learning: A brief survey. *IEEE Signal Process. Mag.*, **2017**, *34*: 26−38.
75. Sitcharangsie, S.; Ijomah, W.; Wong, T.C. Decision makings in key remanufacturing activities to optimise remanufacturing outcomes: A review. *J. Clean. Prod.*, **2019**, *232*: 1465−1481.

Citation: Guo, X.; Bi, Z.; Wang, J.; *et al*. Reinforcement Learning for Disassembly System Optimization Problems: A Survey. *International Journal of Network Dynamics and Intelligence*. 2023, 2(1): 1−14. doi: 10.53941/ijndi0201001