Deep Reinforcement Learning Approaches for Smart Manufacturing

Subjects: Computer Science, Artificial Intelligence

Contributor: Alejandro del Real Torres , Doru Stefan Andreiana , Álvaro Ojeda Roldán , Alfonso Hernández Bustos , Luis Enrique Acevedo Galicia

AI and, in particular, the Deep Reinforcement Learning (DRL) algorithms, which are a perfect response to the unpredictability and volatility of modern demand, are studied in detail. Through the introduction of RL concepts and the development of those with ANNs towards DRL, the potential and variety of these kinds of algorithms are highlighted. Moreover, because these algorithms are data based, their modification to meet the requirements of industry operations is also included. Digital twins are a technology that is increasingly important in I.40 and I.5.0, which seems to be crucial to the development of smart manufacturing.

deep reinforcement learning | smart manufacturing | energy management | Industry 4.0 | AI | Artificial Networks | Robotics | Review | Process control

1. Introduction

Roughly a decade ago, industry 4.0 (I4.0) emerged as the term to define the fourth industrial revolution. Its objective is the transition from the mass production automation of the third industrial revolution to more efficient and flexible production [1]. It can be defined as a technology-driven revolution focusing on further automation and the digitalisation of industrial processes. This results in smart factories, which make use of improved technologies, such as artificial intelligence (AI), Internet of Things (IoT), cloud computing and cyber-physical systems (CPS) [2]. However, it lacks a human-centric and sustainability-centred vision. Moreover, the COVID-19 crisis revealed some deficiencies in global industrial production, which lacks enough flexibility to deal with abrupt changes in production demand [3]. For this reason, the term industry 5.0 (I5.0) has been introduced [4]. This new concept strengthens and complements the objectives of I4.0 through a human-centric, sustainable and resilient industry, reinforcing the contribution of industry to worker welfare and green transition. To this end, it combines the advances in I4.0 technologies in terms of digital twins, CPS, Big Data and AI, among others, with innovative technologies that have surged in the last years [5]. all in all, with a human and sustainable-centred perspective [6].

In 2021, manufacturing recovered to pre-pandemic levels of activity, generating approximately 17% of the gross domestic product (GDP) on average around the world [7] and 14.9% in the European Union, making it the most important industrial activity at the economic level [8].

Independently of the sector, manufacturing comprises many processes, from planning and scheduling to executing physical operations in the production line until the product is ready for distribution [9]. Among these processes, there are tasks involving production scheduling, assembly, decision support systems and path planning. Currently, many of them are carried out by digital systems and robots thanks to the automation of factories, improving their efficiency [10]. However, applying artificial intelligence, in particular machine learning (ML), takes a step forward in this enhancement. Without being explicitly programmed, machine learning algorithms endow automatons with cognitive capabilities that allow them to learn a task [11]. However, the bulk of these algorithms require data in order to learn, and it is not always possible to obtain accurate data in some industrial settings. Reinforcement learning (RL) is a machine learning paradigm that is ideal since its algorithms immediately learn from interaction with the environment. Additionally, the use of deep neural networks (DNNs) with RL algorithms gave rise to deep reinforcement learning (DRL), whose algorithms are capable of learning more complex tasks [12]. Furthermore, those algorithms are relevant for both I4.0 and the upcoming I5.0 since they align with the objective of industry 5.0 easily adapting to a more human-centred approach [13][14].

As reflected in most of the reviews concerning smart manufacturing, I4.0 and I5.0, AI is identified as a key enabling technology. However, AI is a huge study field, and the majority of those reviews do not go deep into how and what to implement given a specific problem. Furthermore, the results in that direction are even scarcer when focusing on more specific AI fields such as DRL. In this sense, the research provides a review of the most commonly used DRL algorithms in manufacturing processes, including their main characteristics and performance, real applications and implementation. Therefore, the research is intended to serve as a guideline for the development and improvement of factories in line with industry 4.0 and 5.0, promoting the use of DRL techniques and algorithms.

Reinforcement Learning

In the early history of reinforcement learning, there were three threads; the first one focused on learning by trial and error; the second one centred on the problem of optimal control; and the third one surged later on, based on ideas from the first two, concerned temporal–difference methods. All of them came together in the late 1980s to give birth to the modern field of reinforcement learning [15]. Nowadays, reinforcement learning has been consolidated as one of the three main machine learning paradigms, together with supervised and unsupervised learning [16][17].

Reinforcement learning algorithms are based on an iterative learning process. The learning process is based on trial and error and the interaction between an agent and an environment [18]. This interaction is modelled as a Markov Decision Process (MDP), a concept first introduced by Bellman R. E. in 1957 [19]. Through this idea, the interaction is reduced to three signals: state s (the current situation of the environment); action (operation or decision taken by the agent based on the state and its experience); and reward r (numerical feedback that the environment returns to the agent to indicate how good or bad is the action taken by the agent) [20]. **Figure 1** illustrates this interaction.

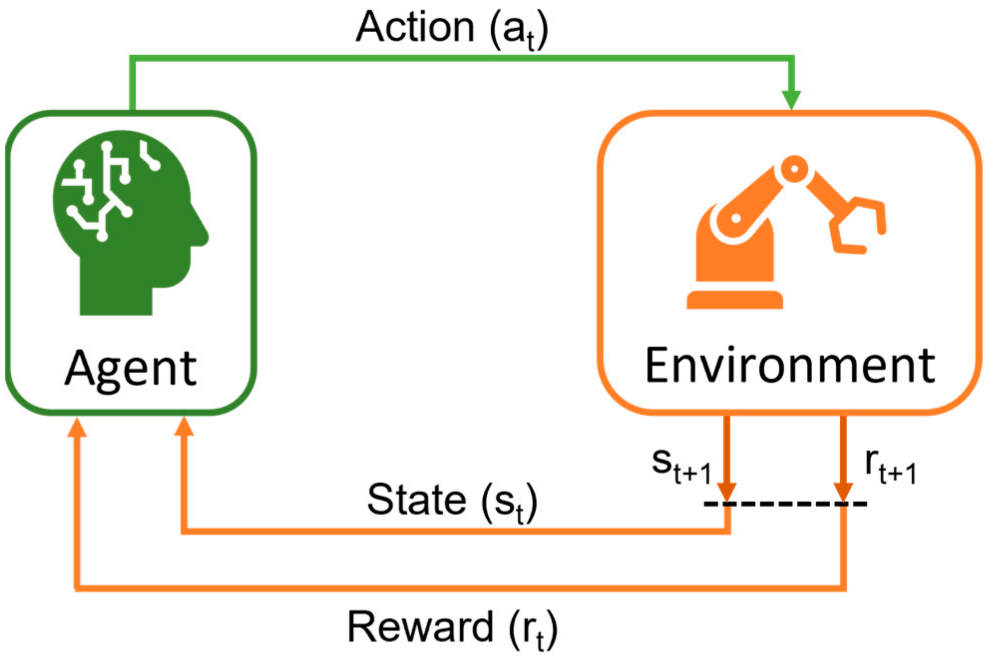

Figure 1. Structure of Markov Decision Process.

In the learning process, the value function and the policy are updated and improved regarding each other. Under this, the exploration–exploitation problem exists, and a trade-off must be found [21][22]. On the one hand, exploration of new actions is necessary to learn alternative paths to achieve the goal and learn the task, whereas exploitation leverages the acquired knowledge to maximise the accumulative reward [23]. On the other hand, exploitation consists of applying the knowledge learned and mostly taking the optimum action known [24].

Firstly, RL algorithms go through an exploratory phase to learn the dynamics of the environment. In this sense, there are several exploration techniques. The most common technique is a random exploration which usually gives great results, as reflected through impressive performances in self-driving cars [25], autonomous landing [26], Atari games [27], Mujoco simulator [28], controller tuning [29] and much more. There are also complex techniques, such as reward shaping [30][31], where the algorithm designer arbitrarily modifies the agent's rewards. However, this technique highly depends on the designer's experience and knowledge of the problem. Errors in reward shaping may lead to infinite repetition of action [32] or no actions at all [33]. An extensive analysis concerning the exploratory techniques and their benefits and drawbacks can be found in Pawel L. et al.'s (2022) survey [34].

Secondly, once the agent explores the environment and learns the consequence of its actions, it passes to exploit that knowledge. However, depending on the problem handled, the agent usually maintains part of its exploratory behaviour just to ensure that the policy of actions followed is still the best. The balance between exploration and exploitation is still an open issue under investigation since there is no unique and perfect solution, but every problem has its own solution [35][36][37].

2. Deep Reinforcement Learning

To address higher dimensional and more complex problems, deep neuronal networks (DNNs) were incorporated into RL, leading to deep RL (DRL) [38]. DNNs are used as function approximators to estimate the policy and value function. Moreover, leveraging their capacity to compact input data dimensionality, hence more complex observations, such as images and non-linear problems, can be processed [39][40][41]. This DRL field started with the Deep Q-Networks (DQN) algorithm [27] which has exponentially increased over the last few years. Researcher describes the catalogue of DRL algorithms, including their primary properties and classification schemes, illustrated in **Figure 2**.

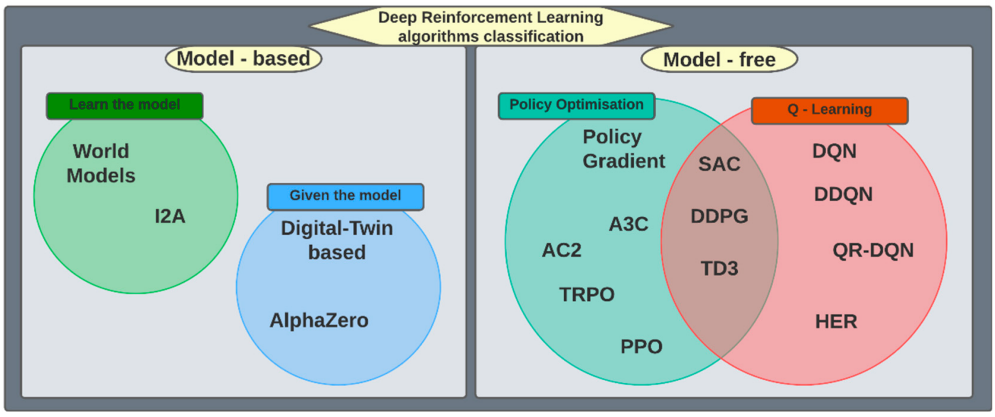

Figure 2. Classification of deep reinforcement learning algorithms.

Figure 2 depicts the most extended classification of DRL algorithms [42]. The main grouping is based on the available information about the dynamics of the environment, which determines the learning process of the agent.

On the one hand, model-based algorithms can be distinguished. These algorithms have access to information on the environment dynamics, including the reward function, which allows the agent to estimate how the environment will react to an action [43]. Typically, these algorithms are integrated with metaheuristics and optimisation techniques [44]. Moreover, they are particularly good at solving high-dimensional problems, as reflected in Aske P. et al. (2020) [45] survey. Furthermore, those methods reflect a higher sample efficiency, as reflected through empirical [46][47] and theoretical [48] studies. A complete overview concerning model-based DRL is presented by Luo, F. et al. (2022) [49] in their survey. Inside model-based DRL algorithms [43], there are two different situations depending on if the model is known or not.

Concerning the first group, if the model is known, this knowledge is used to improve the learning process, and the algorithm is integrated with metaheuristics, planning and optimisation techniques. However, since the environments usually have large action spaces, the application of these techniques is highly resource demanding. Thus, a complete optimisation of the learning process cannot be carried out. Moreover, although there are no algorithms defined as such, except for well-known algorithms such as Alpha Zero [50] and Single Agent [51], most of them are adapted to the application and the characteristics of the model environment. In recent years, DRL model-based algorithms that make use of digital twin models may be highlighted, such as the algorithms presented by Matulis

and Harvey (2021) [52] and Xia et al. (2021) [53]. Therefore, in the implementation of the model-based algorithms, the following aspects must be addressed:

- In which state the planning starts;
- How many computational resources are assigned to it;
- Which optimisation or planning algorithm is used;
- What is the relation between the planning and the DRL algorithm.

In the other group of model-based algorithms, the model of the environment is not fully known, and the algorithms train with a learned model [54]. Normally, a representation of the environment is extracted by using supervised/unsupervised algorithms, which is carried out in a previous step as a model learning process [55][56][57]. Algorithms such as World Models [58] and Imagination-Augmented Agents (I2A) [59] belong to this group. Nonetheless, the accuracy of the model depends on the observable information and the capacity to adapt to changes in the model dynamics. For this reason, these algorithms are more suitable for dealing with deterministic environments. Based on the experience acquired through the interaction with the environment, three model approaches can be obtained [60][61]:

- Forward model: based on the current state and the selected action by the agent, it estimates the next state;
- Backward model: a retrospective model that predicts which state and action led to the current state;
- Inverse model: it assesses which action makes moving from one state to another.

On the other hand, model-free algorithms cannot anticipate the evolution of the environment after an action because the dynamics of the environment is unknown [62]. Thus, the algorithm estimates the most suitable action at the current state based on the acquired experience through interaction. This latter is the most frequent scenario in practice; hence, more algorithms exist [63]. The model-free DRL algorithms focus on the management of acquired experience by algorithms and how they use this information to learn a policy. This distinguishes on-policy algorithms from off-policy algorithms [64]. In the former case, the agent applies its policy generating short-term experience, which frequently consists of a fixed number of transitions (trajectory) [65][66]. Based on this information, the policy is updated, and then the experience is discarded. On the other hand, off-policy algorithms have a memory that stores the transitions created by several past policies [67]. This memory is finite and has a memory management method, for instance, FIFO (first-in, first-out) [68]. In this case, the policy is updated with a sampled batch of the stored transitions, considering the experience generated with old policies [69].

Although this latter classification is not exclusive to model-free algorithms, there is a certain parallelism with the two families of model-free algorithms represented in **Figure 2**, policy optimisation (PO) and Q-learning families. The first family began with the Policy Gradient algorithm and was later expanded to include the Advantage Actor Critic

(A2C) [70], Asynchronous Advantage Actor Critic (A3C) [71], and proximal policy optimization (PPO) [72] algorithms. This class of algorithms is capable of handling continuous and discrete action spaces, and the action at each state is determined by a probability distribution. The second family was derived from Deep Q-Networks (DQN) [73], and algorithms such as Quantile Regression DQN (QR-DQN) [74] and hindsight experience replay (HER) [75] belong to it. In contrast to the other family, they can only deal with discrete action space environments, and the policy calculates the Q-value of each state-action pair to take a decision.

Lastly, it should be highlighted that these classifications are not exclusive, and there are algorithms that integrate features and techniques of different groups, such as the hybrid algorithms that are halfway between policy optimisation and Q-learning families (see **Figure 2**). Some algorithms of this group are Soft Actor-Critic (SAC) [76], Deep Deterministic Policy Gradient (DDPG) [77] and Twin Delayed Deep Deterministic Policy Gradient (TD3) [78]. These algorithms address some of the weaknesses of the other algorithms that allow the implementation of approaches to more complex problems. In addition, combinations of algorithms from different groups can be found in the literature, such as DDPG + HER [79] and model-free and model-based algorithms [45].

Integration in the Industry

Manufacturing involves a set of tasks that generally entail decision making by plant operators. These tasks are related to scheduling [80] (e.g., predicting the production based on future demand, guaranteeing the supply chain, planning processes to optimise production and energy consumption); process control [81] (i.e., automated processes such as assembly lines, pick-and-place and path planning); and monitoring [10] (e.g., decision support systems, calibration and quality control) [12]. As can be observed, most of these tasks are complex, and their efficient performance needs expert knowledge and time to be programmed. In the manufacturing sector, the former exists for many tasks, but the availability of time is limited even more if flexible production wants to be achieved under the framework of I4.0. Moreover, for smart factories of I5.0, other factors, such as benefits for the well-being of workers and the environment, must be considered. All in all, the automation of manufacturing tasks is a complex optimisation problem that requires novel technologies to be addressed, such as ML. Based on a few investigations [11][82][83], below the main requirements of an ML application in the industry are listed:

- Dealing with high-dimensional problems and datasets with moderate effort;
- Capability to simplify potentially difficult outputs and establish an intuitive interaction with operators;
- Adapting to changes in the environment in a cost-effective manner, ideally with some degree of automation;
- Expanding the previous knowledge with the acquired experience;
- Ability to deal with available manufacturing data without particular needs for the initial capture of very detailed information;

- Capability to discover relevant intra- and inter-process relationships and, preferably, correlation and/or causation.

Among ML paradigms, reinforcement learning is suitable for this type of task. The trial-and-error learning through the interaction with the environment and not requiring pre-collected data and prior expert knowledge allow RL algorithms to adapt to uncertain conditions [12]. Moreover, thanks to the capacity of ANNs to create simple representations of complex inputs and functions, DRL algorithms can address complex tasks, maintaining adaptability and robustness [84]. Indeed, some applications can be found in manufacturing, for instance, in scheduling tasks [85][86] and robot manipulation [87][88].

However, the application of DRL in industrial processes presents some challenges that must be considered during the implementation. A complete list of challenges is gathered in studies such as [11][89]; however, the most common ones perceived by the researchers in real-world implementations are described below.

- Stability. In industrial RL applications, the sample efficiency of off-policy algorithms is desirable. However, these show an unstable performance in high-dimensional problems, which worsens if the state and action spaces are continuous. To mitigate this deficiency, two approaches predominate: (i) reducing the brittleness to hyperparameter tuning and (ii) avoiding local optima and delayed rewards. The former can be solved by using tools that optimise the selection of hyperparameters values, such as Optuna [90], or employing algorithms that internally optimise some hyperparameters, such as SAC [91]. The other approach can be addressed by stochastic policies, for example, introducing entropy maximisation such as SAC and improved exploration strategies [92].

- Sample efficiency. Learning better policies with less experience is key for efficient RL applications in industrial processes. This is because, in many cases, the data availability is limited, and it is preferable to train an algorithm in the shortest possible time. As stated before, among model-free DRL algorithms, off-policy algorithms are more sample efficient than on-policy ones. In addition, model-based algorithms have better performance, but obtaining an accurate model of the environment is often challenging in the industry. Other alternatives to enhance sample efficiency are input remapping, which is often implemented with high-dimensional observations [93], and offline training, which consists of training the algorithm with a simulated environment [94].

- Training with real processes. Albeit training directly with the real systems is possible, it is very time consuming and entails the wear and tear of robots and automatons [89]. Moreover, human supervision is needed to guarantee safety conditions. Therefore, simulated environments are used in practice, allowing the generation of much experience at a lower cost and faster training. Nonetheless, a real gap exists between simulated and real-world environments, making applying the policy learned during the training difficult [95].

- Sparse reward. Manufacturing tasks usually involve a large set of steps until reaching their goal. Generally, this is modelled with a zero-reward most of the time and a high reward at the end if the goal is reached [96]. This can

discourage the agent in the exploration phase, thus attaining a poor performance. To this end, some solutions are aggregating demonstration data to the experience of the agent in a model-based RL algorithm to learn better models; including scripted policies to initialise the training, such as in QT-Opt [97] and reward shaping provides additional guidance to exploration, boosting the learning process.

- Reward function. The reward is the most important signal the agent receives because it guides the learning process [98]. For this reason, clearly specifying the goals and rewards is key to achieving a successful learning process. This becomes more complex as the task and the environment becomes more complicated, e.g., industrial environments and manufacturing tasks. To mitigate this problem, some alternatives are integrating intelligent sensors to provide more information, using heuristics techniques and replacing the reward function with a model that predicts that reward [99].

3. Deep Reinforcement Learning in the Production Industry

Nowadays, manufacturing industries face major challenges, such as mass customisation and shorter development cycles. Moreover, there is a need to meet the ever-rising bar for product quality and sustainability in the shortest amount of time through an ambiguous and fluctuant market demand [83]. However, those challenges also open up new opportunities for innovative technologies brought by the I4.0 and I5.0 [13][100]. Among those, AI plays a special role, and furthermore, DRL, after the outstanding results presented by OpenAI [101] and DeepMind [102], among others, is progressively shifted to the production industry [103]. In this sense, some of the main DRL features, such as the adaptability and ability to generalise and extract information from past experiences, have already been demonstrated in a few sectors, as reflected in other reviews. Among them are robotics [87][104], scheduling [105][106], cyber-physical systems [107] and energy systems [108].

3.1. Path Planning

In manufacturing, path planning is crucial for machines such as computer numerical control (CNC) machines [109] and robot manipulation [110] to perform tasks such as painting, moving in space and welding, and additive manufacturing [94][111]. Moreover, path planning is part of the mobile robot navigation system that has an increasing presence in factories [112]. The main objective of this task is to find the optimal trajectory to move the robot or part of it from one point in space to another while maybe performing an operation. In industrial environments, other factors must be considered due to the features of the task or the environment or the potentially severe consequences of a failure. These make path planning more complex, and some of the most popular ones are the avoidance of obstacles, dynamic environments and constraints of the movements of the robots and systems.

For this application, model-free DRL algorithms are predominant, probably due to the complexity of modelling a dynamic environment [113]. DQN, together with its variants, is the most used one [114][115][116]. Despite some issues, such as overestimating q-values and instability, DQN applications are widely used in path planning. An important task of this field is active object detection (AOD), whose purpose is to determine the optimal trajectory so that a robot has the viewpoints that allow it to gather the necessary visual information to recognise an object. DQN is still used for this purpose, outperforming other AOD methods. Fang et al. (2022) [117] recently presented a self-

supervised DQN-based algorithm that improves the success rate and reduces the average trajectory length. Moreover, the developed algorithm was successfully tested with a real robot arm. However, the applications of DQN variants need to become popular in order to overcome the aforementioned drawbacks.

Prioritised DQN (P-DQN) is used to upgrade the convergence speed of DQN, assigning more priority to those samples that contain more information in comparison with the experience [118]. These samples are more likely to be selected to update the parameters of the ANNs. Liu et al. (2022) [119] present a P-DQN-based path-planning algorithm to address path planning in very complex environments with many obstacles. This priority assignment can be detached, constituting a technique called priority experience replay (PER). This technique is combined with Double DQN (DDQN) in [120], increasing the stability of the learning process. Moreover, DDQN also offers satisfactory performance without PER. An example is the path planning application presented in [40], where the DDQN agent is pre-trained in a virtual environment with a 2D-LiDAR and then tested in a real environment using a monocular camera.

In line with I5.0, path planning has a challenge in robotic applications to achieve the estimation of time-efficient and free-collision paths. In this context, crowd navigation of mobile robots can be highlighted due to the need to predict the movement of other objects in the environment, such as humans. For this purpose, the DQN variant of Dueling DQN in combination with an online planner proposed in [121] results in equivalent or even better performance of the state-of-the-art methods (95% of success in complex environments) with less than half the computational cost. Furthermore, based on Social Spatial–Temporal Graph Convolution Network (SSTGCN), a model-based DRL algorithm is developed in [122], highlighting its robustness to changes in the environment.

Lastly, the use of hybrid DRL algorithms should be remarked on because they can work with continuous action space and are not like DQN, which is limited to discrete spaces. For example, Gao et al. (2020) [123] present a novel path planner for mobile robots that combines TD3 and the traditional path planning algorithm Probabilistic Roadmap (PRM). PRM + TD3 is trained in an incremental way, achieving an outstanding generalisation for planning long-distance paths. In addition, a variant of DDPG called mixed experience multi-agent DDPG (ME-MADDPG) is applied to coordinate the displacement of several mobile robots. This algorithm enhances the convergence properties of other DRL algorithms in this field [124].

3.2. Process Control

With the automation of factories, process control became a key element in manufacturing. This control is scalable from large SCADA panels that monitor the whole production chain of a factory to specific processes [125]. Moreover, this manufacturing task addresses simple control operations, such as opening valves, and complex control operations, such as coordinating several robot arms for assembling. For this purpose, control strategies have typically been applied; however, the application of artificial intelligence methods, such as neural networks, is growing thanks to the development of smart factories [10]. Given the plethora of process control tasks, this section focuses on the most recent DRL applications in this field. In addition, a subsection is dedicated to robotic control, especially robot manipulation, due to its significant role in manufacturing [126].

The literature search reflects that DRL algorithms are generally applied to control specific processes and that model-free algorithms predominate. Since control tasks usually involve continuous variables, the algorithms from the policy optimisation family and hybrid algorithms are the most used ones. Regarding the former, PPO is widely applied because it is the most cutting-edge and established algorithm within the PO family. Szarski et al. (2021) apply PPO to control the temperature in a composite curing process to reduce the cycle time [127]. The developed controller is tested with the simulation of a complex curing process in two realistic different aerospace parts, reducing up to 40% of the ramp time. Moreover, this test demonstrates the controller's applicability because it was only trained for one of the parts. Other PPO applications can be found in other manufacturing processes, such as controlling the power and velocity of a laser in charge of melting via powder bed fusion [48] and controlling the rolls of a strip rolling process to achieve the desired flatness [128]. It should be noted that this last application is also compared with DRL hybrid algorithms, outperforming them regarding results and stability.

Although PPO has been applied to some control tasks, its on-policy nature generally entails larger training. Off-policy DRL algorithms improve it thanks to being more sample efficient [129], and DDPG is the most popular off-policy hybrid algorithm for control applications. This algorithm is an extension of DQN for continuous action spaces, and it is the first off-policy algorithm for this type of space, showing positive performance in the control of complex systems. Fusayasu et al. (2022) [130] present a novel application of DDPG in the control of multi-degree-of-freedom spherical actuators, characterised by their difficult control due to their strong non-linearities of torque. DDPG achieves a highly accurate and robust control, outperforming PID and neural network controllers. In the chemical process control, Ma et al. (2019) [131] demonstrate how a DDPG controller can control a polymerisation system, which is a complex, multi-input, non-linear chemical reaction system with a large time delay and noise tolerance. In this case, the main adaptation of the original algorithm is the inclusion of historical experience to deal with time delay. Another application of DDPG in the optimisation of chemical reactions is [132], where the maximisation of hydrogen production through the partial oxidation reaction of methane is reached. Moreover, TD3, as an improved version of DDPG, is also applied in this type of process, for instance, the multivariable control of a continuous stirred tank reactor (CSTR) [133]. The importance of DDPG and TD3 in process control in the chemical industry is shown in [134], where hybrid and PO algorithms are compared for five use cases, and DDPG and TD3 outperform all of them in all use cases.

3.3. Robotics

Robot manipulation encompasses a wide range of tasks, from assembly operations, such as screwing and peg-in-hole, to robot grasping and pick-and-place operations [135][136]. The characteristics of DRL make it very suitable for robotic tasks, which has produced a close relationship between both fields for many years, leading to promising results in the future [104][137].

Firstly, this research starts with the peg-in-hole assembly, the robotic manipulation task with the most DRL applications according to the literature search, and its high precision characterises it. For this task, PPO is the most commonly applied algorithm with applications such as [87][138][139]. Among them, the PPO controller developed by Leyendecker et al. (2021) [87] should be noted, where the algorithm is trained through curriculum learning. This

technique consists of dividing the learning problem into several subtasks and learning them in ascending order of complexity, which allows the learning of the simpler tasks to be used to learn the more complex ones and improves generalisation skills [140].

Although PPO applications abound, other DRL algorithms can be found. For example, Deng et al. (2021) propose an actor-critic-based algorithm that improves the stability and sample efficiency of other state-of-the-art algorithms such as DDPG and TD3 [87][88]. In addition, training this algorithm with hierarchical reinforcement learning (HRL) notably increases the generalisation capability to other assembly tasks. HRL consists of decomposing tasks into simpler and simpler sub-tasks, establishing levels of hierarchy in which more complex parent tasks are formed by simpler child tasks. With this technique, the most basic tasks are learned, which allow for the development of more complex tasks [141]. Furthermore, among the applications of hybrid algorithms, the work of Beltran-Hernández et al. (2020) [88], which uses SAC to learn contact-rich manipulation tasks and tests the algorithm with a real robot arm, and the proposed uses of DDPG to control the force in contact-rich manipulation in [142] and to enhance the flexibility of assembly lines in [143] are noteworthy. The latter is particular in that it uses a digital twin model of the assembly line to train the DDPG algorithm, and once trained, this model is used to monitor the assembly lines and predict failures during the production stage.

Digital twins are a technology that is increasingly important in I.40 and I.5.0, which seems to be crucial to the development of smart manufacturing. Indeed, some DRL control applications, such as [143], leverages this technology to increase their data efficiency and robustness. Liu et al. (2022) train a DQN algorithm with the digital twin model of a robot arm that has to perform a grasping task [144]. In this line, Xia et al. (2021) do the same with DQN and DDQN + PER for a pick-and-place task [53]. Both cases highlighted the smoother transfer of knowledge from the simulation to the real environment thanks to digital twin models.

Finally, another robot manipulation task to which DRL is currently applied is pick-and-place, which in turn includes other tasks such as motion planning, grasping and reaching a point in space [137]. As in other robotic tasks, the use of DDPG is predominant [145]. Some recent examples are [146], whose objective is reaching a point and measuring the influence of different reward functions, and [147], where the application of DDPG results in robust grasping in pick-and-place operations. In addition, the joint use of DDPG and HER is common, highlighting the work of Marzari et al. (2021), that DDPG + HER is used together with HRL to learn complex pick-and-place tasks [148]. Nonetheless, other state-of-the-art algorithms are used in this field, such as TD3 + HER for the motion planning of robot manipulators [149] and PPO and SAC for a grasping task with an outstanding success rate [150]. In this latter work, it should be noted that SAC training requires fewer episodes, but they last longer.

3.4. Scheduling

The aim of scheduling is to optimise the use of time to reduce the consumption of resources in all senses, hence improving the overall efficiency of the industrial processes. In this, several sub-objectives must be considered. It plays an essential role within any kind of industry and has always been a significant research topic approached from different fields. However, due to its interdisciplinary nature, the size of the problem can easily scale up.

Consequently, the optimisation problem has multiple objectives and is usually complex given the uncertainties that must be faced and the high interconnectivity of the elements involved [151]. In this sense, DRL arises as an enabling technology, as reflected in literature reviews concerning smart scheduling in the industry 4.0 framework [152].

On the one hand, in order to solve the multi-objective optimisation problem, a common approach is the implementation of multi-agent DRL algorithms. Several successful studies can be found about this in different production sectors [12]. Lin et al. (2019) [85] implemented a multi-agent DQN algorithm for a semiconductor manufacturing industry in order to cover the human-based decisions and reduce the complexity of the problem, resulting in enhanced performance. Through a similar approach, Ruiz R. et al. (2022) [86] focus on the maintenance scheduling of several machines presenting up to ≈ 75% improvement in overall performance. Other studies combine those algorithms with IoT devices for smart resource allocation [153] or with other algorithms, such as Lamarckian local search for emergency scheduling activities [154]. For the latter, Baer et al. (2019) [155] propose an interesting approach by implementing a multi-stage learning strategy, training different agents individually but optimising them together towards the global goal, presenting great results. On the other hand, in order to face the increasing fluctuation in production demand and product customisation, actor-critic DRL approaches are usually implemented [156].

The actor-critic approach is characterised by its robustness [157] and acts as an upgrade of the traditional Q-learning, which could act as a decision-support system easing operators scheduling tasks [158][159]. Through the actor-critic approach, the policy is periodically checked and recalibrated to the situation, which highly increases the adaptability and eases the implementation in real-time scheduling [80][160]. In addition, several studies reflect that it can be implemented with cloud-fog computing services [161][162]. Furthermore, the performance can be increased by implementing a processing approach divided into batches, as reflected in Palombarini et al. (2018, 2019) studies [163][164]. There are also some novel approaches integrating different neuronal networks that aim to cope with complexity and expand the applications. For example, Park et al. (2020) implemented a proximal policy optimisation (PPO) neuronal network trained with relevant information from scheduled processes, such as the setup status [165].

For latter, despite the great results presented by the research, unfortunately, most of those approaches are not adopted in a practical context. Due to the scheduling policies already established in the production industries, it is quite complex to introduce novel approaches even if the research shows good results. Consequently, increasing research efforts are required in this direction.

3.5. Maintenance

The maintenance objective is to reduce breakdowns and promote overall reliability and efficiency [166]. The term mainly refers to tasks required to restore full operability, such as repairing or replacing damaged components. It significantly impacts the operational reliability and service life of the machinery in any industry. There are four types of maintenance: reactive, preventive, predictive and reliability-centred [167][168]. Historically, reactive maintenance

has predominated, which was performed after the failure of the machine, mainly due to limited knowledge about their operation and failures. Nowadays, this strategy is still in use for unpredictable failures and failures of cheap objects. Over time, the understanding of the process has increased, and preventive maintenance has come up. Further on, I4.0 technologies and advances in AI have enabled predictive and reliability-centred maintenance [169].

As part of AI advances for maintenance activities in the industry, RL algorithms play an important role due to their self-learning capability [170]. Moreover, the integration of neuronal networks, resulting in DRL, expands the applications and performances even further [171]. Their application can help anticipate failures by predicting key parameters and also prevent failures through in-line maintenance, enlarging the lifetime of components.

The anticipation of failures is usually combined with scheduling optimisation to maximise the results [172][173]. In order to speed up the learning phase, Ong, K.S.H, et al. (2022) boards the predictive maintenance problem with a model-free DRL conjoined with the transfer learning method to assist the learning by incorporating expert demonstrations, reducing the training phase time by 58% compared with baseline methods [174]. On the other hand, Acernese, A. et al. board fault detection for a steel plant through a double deep-Q network (DDQN) with prioritised experience replay to enhance and speed up the training [175].

There are also hybrid approaches, such as the one proposed by Chen Li et al. (2022) [176], where feedback control is implemented based on an advantage actor-critic (A2C) RL algorithm to predict the machine status and control the cycle time accordingly. In addition, Yousefi, N. et al. (2022), in their study, propose a dynamic maintenance model based on a Deep Q-learning algorithm to find the optimal maintenance policy at each degradation level of the machine's components [177].

3.6. Energy Management

Nowadays, and especially with the I5.0 and worldwide policies (e.g., Paris agreement [178]), energy consumption and environmental impact are in the spotlight. In this sense, AI algorithms such as DRL can boost energy efficiency and reduce the environmental impact of the manufacturing industry [179]. The algorithms are usually implemented into the energy market to reduce costs and energy flow control in storage and machines operation to increase their energy consumption effectiveness [180]. In resource- and energy-intensive industries such as printed circuit boards (PCB) fabrication, Leng et al. demonstrated that the DRL algorithm was able to improve lead time and cost while increasing revenues and reducing carbon use when compared to traditional methods (FIFO, random forest) [181]. Lu R. et al. (2020) faced a multi-agent DRL algorithm against a conventional mathematical modelling method simulating the manufacturing of a lithium–ion battery. The benchmark presents a 10% reduction in energy consumption [182].

References

1. Pereira, A.; Romero, F. A review of the meanings and the implications of the Industry 4.0 concept. Procedia Manuf. 2017, 13, 1206–1214.
2. Lasi, H.; Fettke, P.; Kemper, H.G.; Feld, T.; Hoffmann, M. Industry 4.0. Bus. Inf. Syst. Eng. 2014, 6, 239–242.
3. Meena, M.; Wangtueai, S.; Mohammed Sharafuddin, A.; Chaichana, T. The Precipitative Effects of Pandemic on Open Innovation of SMEs: A Scientometrics and Systematic Review of Industry 4.0 and Industry 5.0. J. Open Innov. Technol. Mark. Complex. 2022, 8, 152.
4. Industry 5.0—Publications Office of the EU. Available online: https://op.europa.eu/en/publication-detail/-/publication/468a892a-5097-11eb-b59f-01aa75ed71a1/ (accessed on 10 October 2022).
5. Xu, X.; Lu, Y.; Vogel-Heuser, B.; Wang, L. Industry 4.0 and Industry 5.0—Inception, conception and perception. J. Manuf. Syst. 2021, 61, 530–535.
6. Crnjac, Z.M.; Mladineo, M.; Gjeldum, N.; Celent, L. From Industry 4.0 towards Industry 5.0: A Review and Analysis of Paradigm Shift for the People, Organization and Technology. Energies 2022, 15, 5221.
7. The World Bank. Manufacturing, Value Added (% of GDP)—World|Data. 2021. Available online: https://data.worldbank.org/indicator/NV.IND.MANF.ZS (accessed on 11 October 2022).
8. The World Bank. Manufacturing, Value Added (% of GDP)—European Union|Data. 2021. Available online: https://data.worldbank.org/indicator/NV.IND.MANF.ZS?locations=EU&name_desc=false (accessed on 11 October 2022).
9. Yin, R. Concept and Theory of Dynamic Operation of the Manufacturing Process. In Theory and Methods of Metallurgical Process Integration; Academic Press: Cambridge, MA, USA, 2016; pp. 13–53.
10. Stavropoulos, P.; Chantzis, D.; Doukas, C.; Papacharalampopoulos, A.; Chryssolouris, G. Monitoring and Control of Manufacturing Processes: A Review. Procedia CIRP 2013, 8, 421–425.
11. Wuest, T.; Weimer, D.; Irgens, C.; Thoben, K.D. Machine learning in manufacturing: Advantages, challenges, and applications. Prod. Manuf. Res. 2016, 4, 23–45.
12. Panzer, M.; Bender, B. Deep reinforcement learning in production systems: A systematic literature review. Int. J. Prod. Res. 2021, 60, 4316–4341.
13. Maddikunta, P.K.R.; Pham, Q.-V.; B, P.; Deepa, N.; Dev, K.; Gadekallu, T.R.; Ruby, R.; Liyanage, M. Industry 5.0: A survey on enabling technologies and potential applications. J. Ind. Inf. Integr. 2022, 26, 100257.
14. Bigan, C. Trends in Teaching Artificial Intelligence for Industry 5.0. In Sustainability and Innovation in Manufacturing Enterprises; Springer: Singapore, 2022; pp. 257–274.

15. Sutton, R.S.; Barto, A.G. Finitie Markov Decision Processes. In Reinforcement Learning: An Introduction, 2nd ed.; The MIT Press: Cambridge, CA, USA, 2020; pp. 47–68. Available online: http://incompleteideas.net/book/RLbook2020.pdf (accessed on 17 September 2022).
16. Virvou, M.; Alepis, E.; Tsihrintzis, G.A.; Jain, L.C. Machine Learning Paradigms; Springer: Cham, Switzerland, 2020.
17. Coursera. 3 Types of Machine Learning You Should Know. 2022. Available online: https://www.coursera.org/articles/types-of-machine-learning (accessed on 5 November 2022).
18. Wiering, M.; Otterlo, M. Reinforcement learning. Adaptation, learning, and optimization. In Reinforcement Learning State-of-the-Art; Springer: Berlin/Heidelberg, Germany, 2012.
19. Bellman, R.E. A Markovian Decision Process. J. Math. Mech. 1957, 6, 679–684.
20. van Otterlo, M.; Wiering, M. Reinforcement learning and markov decision processes. In Reinforcement Learning; Springer: Berlin, Germany, 2012; Volume 12.
21. Yogeswaran, M.; Ponnambalam, S.G. Reinforcement learning: Exploration-exploitation dilemma in multi-agent foraging task. OPSEARCH 2012, 49, 223–236.
22. Coggan, M. Exploration and exploitation in reinforcement learning. In CRA-W DMP Project; Working Paper of the Research Supervised by Prof. Doina Precup; McGill University: Montreal, QC, Canada, 2004.
23. Mcfarlane, R. A Survey of Exploration Strategies in Reinforcement Learning. J. Mach. Learn. Res. 2018, 1, 10.
24. Furelos-Blanco, D.; Law, M.; Jonsson, A.; Broda, K.; Russo, A. Induction and exploitation of subgoal automata for reinforcement learning. J. Artif. Intell. Res. 2021, 70, 1031–1116.
25. Kiran, B.R.; Sobh, I.; Talpaert, V.; Mannion, P.; Al Sallab, A.A.; Yogamani, S.; Perez, P. Deep Reinforcement Learning for Autonomous Driving: A Survey. IEEE Trans. Intell. Transp. Syst. 2022, 23, 4909–4926.
26. Polvara, R.; Patacchiola, M.; Sharma, S.; Wan, J.; Manning, A.; Sutton, R.; Cangelosi, A. Autonomous quadrotor landing using deep reinforcement learning. ArXiv 2017, arXiv:1709.03339.
27. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. Nature 2015, 518, 529–533.
28. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. In Proceedings of the 4th International Conference on Learning Representations (ICLR 2016), San Juan, Puerto Rico, 2–4 May 2016.

29. Lee, S.; Bang, H. Automatic Gain Tuning Method of a Quad-Rotor Geometric Attitude Controller Using A3C. Int. J. Aeronaut. Space Sci. 2020, 21, 469–478.
30. Laud, A.D. Theory and Application of Reward Shaping in Reinforcement Learning. Ph.D. Dissertation, University of Illinois, Urbana-Champaign IL, USA, 2004.
31. Marom, O.; Rosman, B. Belief Reward Shaping in Reinforcement Learning. Proc. AAAI Conf. Artif. Intell. 2018, 32, 3762–3769.
32. Clark, J.; Amodei, D. Faulty Reward Functions in the Wild. 2016. Available online: https://openai.com/blog/faulty-reward-functions/ (accessed on 9 November 2022).
33. Irpan, A. Deep Reinforcement Learning Doesn't Work Yet. 2018. Available online: https://www.alexirpan.com/2018/02/14/rl-hard.html (accessed on 9 November 2022).
34. Ladosz, P.; Weng, L.; Kim, M.; Oh, H. Exploration in deep reinforcement learning: A survey. Inf. Fusion 2022, 85, 1–22.
35. Asiain, E.; Clempner, J.B.; Poznyak, A.S. Controller exploitation-exploration reinforcement learning architecture for computing near-optimal policies. Soft Comput. 2019, 23, 3591–3604.
36. Schäfer, L.; Christianos, F.; Hanna, J.; Albrecht, S.V. Decoupling exploration and exploitation in reinforcement learning. ArXiv 2021, arXiv:2107.08966.
37. Chen, W.H. Perspective view of autonomous control in unknown environment: Dual control for exploitation and exploration vs reinforcement learning. Neurocomputing 2022, 497, 50–63.
38. François-Lavet, V.; Henderson, P.; Islam, R.; Bellemare, M.G.; Pineau, J. An Introduction to Deep Reinforcement Learning. Found. Trends Mach. Learn. 2018, 11, 219–354.
39. Chen, L. Deep reinforcement learning. In Deep Learning and Practice with MindSpore; Springer: Singapore, 2021.
40. Arulkumaran, K.; Deisenroth, M.P.; Brundage, M.; Bharath, A.A. Deep reinforcement learning: A brief survey. IEEE Signal Process. Mag. 2017, 34, 26–38.
41. Sewak, M. Deep Reinforcement Learning—Frontiers of Artificial Intelligence, 1st ed.; Springer: Singapore, 2019.
42. Yang, Y.; Kayid, A.; Mehta, D. State-of-the-Art Reinforcement Learning Algorithms. IJERT J. Int. J. Eng. Res. Technol. 2020, 8, 6.
43. Moerland, T.M.; Broekens, J.; Jonker, C.M. Model-based Reinforcement Learning: A Survey. arXiv 2020, arXiv:2006.16712.
44. Kaiser, Ł.; Babaeizadeh, M.; Miłos, P.; Osinski, B.; Campbell, R.H.; Czechowski, K.; Erhan, D.; Finn, C.; Kozakowski, P.; Levine, S.; et al. Model-Based Reinforcement Learning for Atari. In

Proceedings of the International Conference on Learning Representations (ICLR 2020), Addis Ababa, Ethiopia, 30 April 2020.

45. Plaat, A.; Kosters, W.; Preuss, M. Deep model-based reinforcement learning for high-dimensional problems, a survey. arXiv 2020, arXiv:2008.05598.
46. Janner, M.; Fu, J.; Zhang, M.; Levine, S. When to trust your model: Model-based policy optimization. In Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019.
47. Wang, T.; Bao, X.; Clavera, I.; Hoang, J.; Wen, Y.; Langlois, E.; Zhang, S.; Zhang, G.; Abbeel, P.; Ba, J. Benchmarking model-based reinforcement learning. arXiv 2019, arXiv:1907.02057.
48. Sun, W.; Jiang, N.; Krishnamurthy, A.; Agarwal, A.; Langford, J. Model-based RL in contextual decision processes: PAC bounds and exponential improvements over model-free approaches. In Proceedings of the Thirty-Second Conference on Learning Theory, Phoenix, AZ, USA, 25–28 June 2019.
49. Luo, F.-M.; Xu, T.; Lai, H.; Chen, X.-H.; Zhang, W.; Yu, Y. A survey on model-based reinforcement learning. arXiv 2022, arXiv:2206.09328.
50. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. Mastering the game of Go without human knowledge. Nature 2017, 550, 354–359.
51. Feng, D.; Gomes, C.P.; Selman, B. Solving Hard AI Planning Instances Using Curriculum-Driven Deep Reinforcement Learning. In Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI 2020), Yokohama, Japan, 7–15 January 2021; pp. 2198–2205.
52. Matulis, M.; Harvey, C. A robot arm digital twin utilising reinforcement learning. Comput. Graph. 2021, 95, 106–114.
53. Xia, K.; Sacco, C.; Kirkpatrick, M.; Saidy, C.; Nguyen, L.; Kircaliali, A.; Harik, R. A digital twin to train deep reinforcement learning agent for smart manufacturing plants: Environment, interfaces and intelligence. J. Manuf. Syst. 2021, 58, 210–230.
54. Wiering, M.A.; Withagen, M.; Drugan, M.M. Model-based multi-objective reinforcement learning. In Proceedings of the 2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL), Orlando, FL, USA, 9–12 December 2014.
55. Kurutach, T.; Clavera, I.; Duan, Y.; Tamar, A.; Abbeel, P. METRPO: Model-ensemble trust-region policy optimization. In Proceedings of the 6th International Conference on Learning Representations (ICLR 2018), Vancouver, BC, Canada, 30 April–3 May 2018.
56. Rajeswaran, A.; Mordatch, I.; Kumar, V. A game theoretic framework for model based reinforcement learning. In Proceedings of the 37th International Conference on Machine Learning

(ICML 2020), Virtual, 13–18 July 2020.

57. Shen, J.; Zhao, H.; Zhang, W.; Yu, Y. Model-based policy optimization with unsupervised model adaptation. Adv. Neural Inf. Process. Syst. 2020, 33, 2823–2834.
58. Ha, D.; Schmidhuber, J. World Models. Forecast. Bus. Econ. 2018, 201–209.
59. Racanière, S.; Weber, T.; Reichert, D.; Buesing, L.; Guez, A.; Rezende, D.J.; Badia, A.P.; Vinyals, O.; Heess, N.; Li, Y.; et al. Imagination-Augmented Agents for Deep Reinforcement Learning. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; pp. 5691–5702.
60. Edwards, A.D.; Downs, L.; Davidson, J.C. Forward-backward reinforcement learning. arXiv 2018, arXiv:1803.10227.
61. van Hasselt, H.; Hessel, M.; Aslanides, J. When to use parametric models in reinforcement learning? In Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019.
62. Ramírez, J.; Yu, W.; Perrusquía, A. Model-free reinforcement learning from expert demonstrations: A survey. Artif. Intell. Rev. 2022, 55, 3213–3241.
63. Otto, F. Model-Free Deep Reinforcement Learning—Algorithms and Applications. In Reinforcement Learning Algorithms: Analysis and Applications; Springer: Cham, Switzerland, 2021; Volume 883.
64. Hausknecht, M.; Stone, P.; Mc, O. On-Policy vs. Off-Policy Updates for Deep Reinforcement Learning. In Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI 2016), New York City, NY, USA, 9–15 July 2016.
65. Tan, Z.; Karakose, M. On-Policy Deep Reinforcement Learning Approach to Multi Agent Problems. In Interdisciplinary Research in Technology and Management; CRC Press: Boca Raton, FL, USA, 2021.
66. Andrychowicz, M.; Raichuk, A.; Stańczyk, P.; Orsini, M.; Girgin, S.; Marinier, R.; Hussenot, L.; Geist, M.; Pietquin, O.; Michalski, M.; et al. What matters in on-policy reinforcement learning? a large-scale empirical study. arXiv 2020, arXiv:2006.05990.
67. Agarwal, R.; Schuurmans, D.; Norouzi, M. Striving for Simplicity in Off-Policy Deep Reinforcement Learning. 2019. Available online: https://openreview.net/forum?id=ryeUg0VFwr (accessed on 14 October 2022).
68. Zimmer, M.; Boniface, Y.; Dutech, A. Off-Policy Neural Fitted Actor-Critic. In Proceedings of the Deep Reinforcement Learning Workshop (NIPS 2016), Barcelona, Spain, 5–10 December 2016.
69. Fujimoto, S.; Meger, D.; Precup, D. Off-policy deep reinforcement learning without exploration. In Proceedings of the 36th International Conference on Machine Learning (ICML 2019), Long

Beach, CA, USA, 9–15 June 2019.

70. Clemente, A.V.; Castejón, H.N.; Chandra, A. Efficient Parallel Methods for Deep Reinforcement Learning. arXiv 2017, arXiv:1705.04862.
71. Mnih, V.; Badia, A.P.; Mirza, M.; Graves, A.; Lillicrap, T.P.; Harley, T.; Silver, D.; Kavukcuoglu, K. Asynchronous Methods for Deep Reinforcement Learning. arXiv 2016, arXiv:1602.01783.
72. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Openai, O.K. Proximal Policy Optimization Algorithms. arXiv 2017, arXiv:1707.06347.
73. Huang, Y. Deep Q-networks. In Deep Reinforcement Learning: Fundamentals, Research and Applications; Dong, H., Ding, Z., Zhang, S., Eds.; Springer Nature: Singapore, 2020.
74. Dabney, W.; Rowland, M.; Bellemare, M.G.; Munos, R. Distributional Reinforcement Learning with Quantile Regression. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, Hilton New Orleans Riverside, New Orleans, LA, USA, 2–7 February 2018; pp. 2892–2901.
75. Andrychowicz, M.; Wolski, F.; Ray, A.; Schneider, J.; Fong, R.; Welinder, P.; McGrew, B.; Tobin, J.; Abbeel, P.; Zaremba, W. Hindsight Experience Replay. Available online: https://goo.gl/SMrQnI (accessed on 4 October 2022).
76. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. January 2018. Available online: http://arxiv.org/abs/1801.01290 (accessed on 4 October 2022).
77. Casas, N. Deep deterministic policy gradient for urban traffic light control. arXiv 2017, arXiv:1703.09035.
78. Fujimoto, S.; van Hoof, H.; Meger, D. Addressing Function Approximation Error in Actor-Critic Methods. In Proceedings of the 35th International Conference on Machine Learning (ICML 2018), Stockholm, Sweden, 10–15 July 2018; pp. 2587–2601.
79. Saeed, M.; Nagdi, M.; Rosman, B.; Ali, H.H.S.M. Deep Reinforcement Learning for Robotic Hand Manipulation. In Proceedings of the 2020 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE 2020), Khartoum, Sudan, 26 February–1 March 2021.
80. Serrano-Ruiz, J.C.; Mula, J.; Poler, R. Smart manufacturing scheduling: A literature review. J. Manuf. Syst. 2021, 61, 265–287.
81. Kuo, R.J.; Cohen, P.H. Manufacturing process control through integration of neural networks and fuzzy model. Fuzzy Sets Syst. 1998, 98, 15–31.
82. Chien, C.F.; Dauzère-Pérès, S.; Huh, W.T.; Jang, Y.J.; Morrison, J.R. Artificial intelligence in manufacturing and logistics systems: Algorithms, applications, and case studies. Int. J. Prod. Res. 2020, 58, 2730–2731.

83. Morgan, J.; Halton, M.; Qiao, Y.; Breslin, J.G. Industry 4.0 smart reconfigurable manufacturing machines. J. Manuf. Syst. 2020, 59, 481–506.
84. Oliff, H.; Liu, Y.; Kumar, M.; Williams, M.; Ryan, M. Reinforcement learning for facilitating human-robot-interaction in manufacturing. J. Manuf. Syst. 2020, 56, 326–340.
85. Lin, C.C.; Deng, D.J.; Chih, Y.L.; Chiu, H.T. Smart Manufacturing Scheduling with Edge Computing Using Multiclass Deep Q Network. IEEE Trans. Ind. Inform. 2019, 15, 4276–4284.
86. Rodríguez, M.L.R.; Kubler, S.; de Giorgio, A.; Cordy, M.; Robert, J.; Le Traon, Y. Multi-agent deep reinforcement learning based Predictive Maintenance on parallel machines. Robot. Comput. Integr. Manuf. 2022, 78, 102406.
87. Leyendecker, L.; Schmitz, M.; Zhou, H.A.; Samsonov, V.; Rittstieg, M.; Lutticke, D. Deep Reinforcement Learning for Robotic Control in High-Dexterity Assembly Tasks-A Reward Curriculum Approach. In Proceedings of the 2021 Fifth IEEE International Conference on Robotic Computing (IRC), Taichung, Taiwan, 15–17 November 2021; pp. 35–42.
88. Beltran-Hernandez, C.C.; Petit, D.; Ramirez-Alpizar, I.G.; Harada, K. Variable Compliance Control for Robotic Peg-in-Hole Assembly: A Deep-Reinforcement-Learning Approach. Appl. Sci. 2020, 10, 6923.
89. Ibarz, J.; Tan, J.; Finn, C.; Kalakrishnan, M.; Pastor, P.; Levine, S. How to train your robot with deep reinforcement learning: Lessons we have learned. Int. J. Robot. Res. 2021, 40, 698–721.
90. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the Proceedings of the 25th International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 4–8 August 2019.
91. Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; et al. Soft Actor-Critic Algorithms and Applications. arXiv 2018, arXiv:1812.05905.
92. Yang, T.; Tang, H.; Bai, C.; Liu, J.; Hao, J.; Meng, Z.; Liu, P.; Wang, Z. Exploration in Deep Reinforcement Learning: A Comprehensive Survey. arXiv 2021, arXiv:2109.06668.
93. He, L.; Aouf, N.; Whidborne, J.F.; Song, B. Integrated moment-based LGMD and deep reinforcement learning for UAV obstacle avoidance. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 7491–7497.
94. Aumjaud, P.; McAuliffe, D.; Rodríguez-Lera, F.J.; Cardiff, P. Reinforcement Learning Experiments and Benchmark for Solving Robotic Reaching Tasks. Adv. Intell. Syst. Comput. 2021, 1285, 318–331.
95. Salvato, E.; Fenu, G.; Medvet, E.; Pellegrino, F.A. Crossing the reality gap: A survey on sim-to-real transferability of robot controllers in reinforcement learning. IEEE Access 2021, 9, 153171–

153187.

96. Sutton, R.; Barto, A. Frontiers. In Reinforcement Learning: An Introduction, 2nd ed.; The MIT Press: Cambridge, CA, USA, 2020; pp. 459–475. Available online: http://incompleteideas.net/book/RLbook2020.pdf (accessed on 1 October 2022).
97. Kalashnikov, D.; Irpan, A.; Pastor, P.; Ibarz, J.; Herzog, A.; Jang, E.; Quillen, D.; Holly, E.; Kalakrishnan, M.; Vanhoucke, V.; et al. QT-Opt: Scalable Deep Reinforcement Learning for Vision-Based Robotic Manipulation. arXiv 2018, arXiv:1806.10293.
98. Matignon, L.; Laurent, G.J.; le Fort-Piat, N. Reward function and initial values: Better choices for accelerated goal-directed reinforcement learning. In International Conference on Artificial Neural Networks; Springer: Berlin/Heidelberg, Germany, 2006; pp. 840–849.
99. Eschmann, J. Reward Function Design in Reinforcement Learning. Stud. Comput. Intell. 2021, 883, 25–33.
100. Lee, J.; Bagheri, B.; Kao, H.A. A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. Manuf. Lett. 2015, 3, 18–23.
101. OpenAI. Available online: https://openai.com (accessed on 1 November 2022).
102. DeepMind. Available online: https://www.deepmind.com (accessed on 1 November 2022).
103. Azeem, M.; Haleem, A.; Javaid, M. Symbiotic Relationship between Machine Learning and Industry 4.0: A Review. J. Ind. Integr. Manag. 2021, 7.
104. Nguyen, H.; La, H. Review of Deep Reinforcement Learning for Robot Manipulation. In Proceedings of the 2019 Third IEEE International Conference on Robotic Computing (IRC), Naples, Italy, 25–27 February 2019.
105. Liu, Y.; Ping, Y.; Zhang, L.; Wang, L.; Xu, X. Scheduling of decentralized robot services in cloud manufacturing with deep reinforcement learning. Robot. Comput.-Integr. Manuf. 2023, 80, 102454.
106. Xing, Q.; Chen, Z.; Zhang, T.; Li, X.; Sun, K.Y. Real-time optimal scheduling for active distribution networks: A graph reinforcement learning method. Int. J. Electr. Power Energy Syst. 2023, 145, 108637.
107. Rupprecht, T.; Wang, Y. A survey for deep reinforcement learning in markovian cyber–physical systems: Common problems and solutions. Neural Netw. 2022, 153, 13–26.
108. Cao, D.; Hu, W.; Zhao, J.; Zhang, G.; Zhang, B.; Liu, Z.; Chen, Z.; Blaabjerg, F. Reinforcement Learning and Its Applications in Modern Power and Energy Systems: A Review. J. Mod. Power Syst. Clean Energy 2020, 8, 1029–1042.

109. Sun, Y.; Jia, J.; Xu, J.; Chen, M.; Niu, J. Path, feedrate and trajectory planning for free-form surface machining: A state-of-the-art review. Chin. J. Aeronaut. 2022, 35, 12–29.
110. Sánchez-Ibáñez, J.R.; Pérez-Del-Pulgar, C.J.; García-Cerezo, A. Path planning for autonomous mobile robots: A review. Sensors 2021, 21, 7898.
111. Jiang, J.; Ma, Y. Path planning strategies to optimize accuracy, quality, build time and material use in additive manufacturing: A review. Micromachines 2020, 11, 633.
112. Patle, B.; L, G.B.; Pandey, A.; Parhi, D.; Jagadeesh, A. A review: On path planning strategies for navigation of mobile robot. Def. Technol. 2019, 15, 582–606.
113. Qiu, T.; Cheng, Y. Applications and Challenges of Deep Reinforcement Learning in Multi-robot Path Planning. J. Electron. Res. Appl. 2021, 5, 25–29.
114. Zhao, Y.; Zhang, Y.; Wang, S. A Review of Mobile Robot Path Planning Based on Deep Reinforcement Learning Algorithm. J. Phys. Conf. Ser. 2021, 2138, 012011.
115. Huo, Q. Multi-objective vehicle path planning based on DQN. In Proceedings of the International Conference on Cloud Computing, Performance Computing, and Deep Learning (CCPCDL 2022), Wuhan, China, 11–13 March 2022; p. 12287.
116. Wang, J.; Zhang, T.; Ma, N.; Li, Z.; Ma, H.; Meng, F.; Meng, M.Q. A survey of learning-based robot motion planning. IET Cyber-Syst. Robot. 2021, 3, 302–314.
117. Fang, F.; Liang, W.; Wu, Y.; Xu, Q.; Lim, J.-H. Self-Supervised Reinforcement Learning for Active Object Detection. IEEE Robot. Autom. Lett. 2022, 7, 10224–10231.
118. Lv, L.; Zhang, S.; Ding, D.; Wang, Y. Path Planning via an Improved DQN-Based Learning Policy. IEEE Access 2019, 7, 67319–67330.
119. Liu, Y.; Chen, Z.; Li, Y.; Lu, M.; Chen, C.; Zhang, X. Robot Search Path Planning Method Based on Prioritized Deep Reinforcement Learning. Int. J. Control. Autom. Syst. 2022, 20, 2669–2680.
120. Wang, Y.; Fang, Y.; Lou, P.; Yan, J.; Liu, N. Deep Reinforcement Learning based Path Planning for Mobile Robot in Unknown Environment. J. Phys. Conf. Ser. 2020, 1576, 012009.
121. Zhou, Z.; Zhu, P.; Zeng, Z.; Xiao, J.; Lu, H.; Zhou, Z. Robot Navigation in a Crowd by Integrating Deep Reinforcement Learning and Online Planning. Appl. Intell. 2022, 52, 15600–15616.
122. Lu, Y.; Ruan, X.; Huang, J. Deep Reinforcement Learning Based on Social Spatial–Temporal Graph Convolution Network for Crowd Navigation. Machines 2022, 10, 703.
123. Gao, J.; Ye, W.; Guo, J.; Li, Z. Deep Reinforcement Learning for Indoor Mobile Robot Path Planning. Sensors 2020, 20, 5493.

124. Wu, D.; Wan, K.; Gao, X.; Hu, Z. Multiagent Motion Planning Based on Deep Reinforcement Learning in Complex Environments. In Proceedings of the 2021 6th International Conference on Control and Robotics Engineering (ICCRE 2021), Beijing, China, 16–18 April 2021; pp. 123–128.

125. Nolan, D.P. Process Controls. In Handbook of Fire and Explosion Protection Engineering Principles, 2nd ed.; Elsevier: Amsterdam, The Netherlands, 2011; pp. 113–118.

126. Karigiannis, J.N.; Laurin, P.; Liu, S.; Holovashchenko, V.; Lizotte, A.; Roux, V.; Boulet, P. Reinforcement Learning Enabled Self-Homing of Industrial Robotic Manipulators in Manufacturing. Manuf. Lett. 2022, 33, 909–918.

127. Szarski, M.; Chauhan, S. Composite temperature profile and tooling optimization via Deep Reinforcement Learning. Compos. Part A Appl. Sci. Manuf. 2021, 142, 106235.

128. Deng, J.; Sierla, S.; Sun, J.; Vyatkin, V. Reinforcement learning for industrial process control: A case study in flatness control in steel industry. Comput. Ind. 2022, 143, 103748.

129. Li, Y. Deep Reinforcement Learning: An Overview. arXiv 2017, arXiv:1701.07274.

130. Fusayasu, H.; Heya, A.; Hirata, K. Robust control of three-degree-of-freedom spherical actuator based on deep reinforcement learning. IEEJ Trans. Electr. Electron. Eng. 2022, 17, 749–756.

131. Ma, Y.; Zhu, W.; Benton, M.G.; Romagnoli, J. Continuous control of a polymerization system with deep reinforcement learning. J. Process. Control. 2019, 75, 40–47.

132. Neumann, M.; Palkovits, D.S. Reinforcement Learning Approaches for the Optimization of the Partial Oxidation Reaction of Methane. Ind. Eng. Chem. Res. 2022, 61, 3910–3916. Available online: https://doi.org/10.1021/ACS.IECR.1C04622/ASSET/IMAGES/LARGE/IE1C04622_0010.JPEG (accessed on 31 October 2022).

133. Yifei, Y.; Lakshminarayanan, S. Multi-Agent Reinforcement Learning System for Multiloop Control of Chemical Processes. In Proceedings of the 2022 IEEE International Symposium on Advanced Control of Industrial Processes (AdCONIP), Vancouver, BC, Canada, 7–9 August 2022; pp. 48–53.

134. Dutta, D.; Upreti, S.R. Upreti. A survey and comparative evaluation of actor-critic methods in process control. Can. J. Chem. Eng. 2022, 100, 2028–2056.

135. Suomalainen, M.; Karayiannidis, Y.; Kyrki, V. A survey of robot manipulation in contact. Robot. Auton. Syst. 2022, 156, 104224.

136. Mohammed, M.Q.; Kwek, L.C.; Chua, S.C.; Al-Dhaqm, A.; Nahavandi, S.; Eisa, T.A.E.; Miskon, M.F.; Al-Mhiqani, M.N.; Ali, A.; Abaker, M.; et al. Review of Learning-Based Robotic Manipulation in Cluttered Environments. Sensors 2022, 22, 7938.

137. Gu, S.; Holly, E.; Lillicrap, T.; Levine, S. Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3389–3396.
138. Zhou, Z.; Ni, P.; Zhu, X.; Cao, Q. Compliant Robotic Assembly based on Deep Reinforcement Learning. In Proceedings of the 2021 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE), Chongqing, China, 9–11 July 2021.
139. Hebecker, M.; Lambrecht, J.; Schmitz, M. Towards real-world force-sensitive robotic assembly through deep reinforcement learning in simulations. In Proceedings of the 2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Delft, The Netherlands, 12–16 July 2021; pp. 1045–1051.
140. Narvekar, S.; Peng, B.; Leonetti, M.; Sinapov, J.; Taylor, M.E.; Stone, P. Curriculum learning for reinforcement learning domains: A framework and survey. arXiv 2020, arXiv:2003.04960.
141. Bosch, A.V.D.; Hengst, B.; Lloyd, J.; Miikkulainen, R.; Blockeel, H. Hierarchical Reinforcement Learning. In Encyclopedia of Machine Learning; Springer: Boston, MA, USA, 2011; pp. 495–502.
142. Wang, C.; Lin, C.; Liu, B.; Su, C.; Xu, P.; Xie, L. Deep Reinforcement Learning with Shaping Exploration Space for Robotic Assembly. In Proceedings of the 2021 3rd International Symposium on Robotics & Intelligent Manufacturing Technology (ISRIMT), Changzhou, China, 24–26 September 2021.
143. Li, J.; Pang, D.; Zheng, Y.; Guan, X.; Le, X. A flexible manufacturing assembly system with deep reinforcement learning. Control Eng. Pract. 2022, 118, 104957.
144. Liu, Y.; Xu, H.; Liu, D.; Wang, L. Wang. A digital twin-based sim-to-real transfer for deep reinforcement learning-enabled industrial robot grasping. Robot. Comput. Integr. Manuf. 2022, 78, 102365.
145. Lobbezoo, A.; Qian, Y.; Kwon, H.-J. Reinforcement Learning for Pick and Place Operations in Robotics: A Survey. Robotics 2021, 10, 105.
146. Zeng, R.; Liu, M.; Zhang, J.; Li, X.; Zhou, Q.; Jiang, Y. Manipulator Control Method Based on Deep Reinforcement Learning. In Proceedings of the 2020 Chinese Control and Decision Conference (CCDC), Hefei, China, 22–24 August 2020; pp. 415–420.
147. Dai, J.; Zhu, M.; Feng, Y. Stiffness Control for a Soft Robotic Finger based on Reinforcement Learning for Robust Grasping. In Proceedings of the 2021 27th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Shanghai, China, 26–28 November 2021; pp. 540–545.
148. Marzari, L.; Pore, A.; Dall'Alba, D.; Aragon-Camarasa, G.; Farinelli, A.; Fiorini, P. Towards Hierarchical Task Decomposition using Deep Reinforcement Learning for Pick and Place

Subtasks. In Proceedings of the 2021 20th International Conference on Advanced Robotics (ICAR 2021), Virtual Event, 6–10 December 2021; pp. 640–645.

149. Kim, M.; Han, D.-K.; Park, J.-H.; Kim, J.-S. Motion Planning of Robot Manipulators for a Smoother Path Using a Twin Delayed Deep Deterministic Policy Gradient with Hindsight Experience Replay. Appl. Sci. 2020, 10, 575.
150. Shahid, A.A.; Piga, D.; Braghin, F.; Roveda, L. Continuous control actions learning and adaptation for robotic manipulation through reinforcement learning. Auton. Robot. 2022, 46, 483–498.
151. Wang, L.; Pan, Z.; Wang, J. A Review of Reinforcement Learning Based Intelligent Optimization for Manufacturing Scheduling. Complex Syst. Model. Simul. 2022, 1, 257–270.
152. Prashar, A.; Tortorella, G.L.; Fogliatto, F.S. Production scheduling in Industry 4.0: Morphological analysis of the literature and future research agenda. J. Manuf. Syst. 2022, 65, 33–43.
153. Rosenberger, J.; Urlaub, M.; Rauterberg, F.; Lutz, T.; Selig, A.; Bühren, M.; Schramm, D. Deep Reinforcement Learning Multi-Agent System for Resource Allocation in Industrial Internet of Things. Sensors 2022, 22, 4099.
154. Hu, C.; Wang, Q.; Gong, W.; Yan, X. Multi-objective deep reinforcement learning for emergency scheduling in a water distribution network. Memetic Comput. 2022, 14, 211–223.
155. Baer, S.; Bakakeu, J.; Meyes, R.; Meisen, T. Multi-agent reinforcement learning for job shop scheduling in flexible manufacturing systems. In Proceedings of the 2019 Second International Conference on Artificial Intelligence for Industries (AI4I), Laguna Hills, CA, USA, 25–27 September 2019.
156. Esteso, A.; Peidro, D.; Mula, J.; Díaz-Madroñero, M. Reinforcement learning applied to production planning and control. Int. J. Prod. Res. 2022.
157. Liu, L.; Zhu, J.; Chen, J.; Ye, H. Cooperative optimal scheduling strategy of source and storage in microgrid based on soft actor-critic. Dianli Zidonghua Shebei/Electr. Power Autom. Equip. 2022, 42.
158. Andreiana, D.S.; Galicia, L.E.A.; Ollila, S.; Guerrero, C.L.; Roldán, Á.O.; Navas, F.D.; Torres, A.D.R. Steelmaking Process Optimised through a Decision Support System Aided by Self-Learning Machine Learning. Processes 2022, 10, 434.
159. Roldán, Á.O.; Gassner, G.; Schlautmann, M.; Galicia, L.E.A.; Andreiana, D.S.; Heiskanen, M.; Guerrero, C.L.; Navas, F.D.; Torres, A.D.R. Optimisation of Operator Support Systems through Artificial Intelligence for the Cast Steel Industry: A Case for Optimisation of the Oxygen Blowing Process Based on Machine Learning Algorithms. J. Manuf. Mater. Process. 2022, 6, 34.
160. Fu, F.; Kang, Y.; Zhang, Z.; Yu, F.R. Transcoding for live streaming-based on vehicular fog computing: An actor-critic DRL approach. In Proceedings of the IEEE INFOCOM 2020—IEEE

Conference on Computer Communications Workshops (INFOCOM WKSHPS), Toronto, ON, Canada, 6–9 July 2020.

161. Xu, Y.; Zhao, J. Actor-Critic with Transformer for Cloud Computing Resource Three Stage Job Scheduling. In Proceedings of the 2022 7th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), Chengdu, China, 22–24 April 2022; pp. 33–37.
162. Fu, F.; Kang, Y.; Zhang, Z.; Yu, F.R.; Wu, T. Soft Actor-Critic DRL for Live Transcoding and Streaming in Vehicular Fog-Computing-Enabled IoV. IEEE Internet Things J. 2020, 8, 1308–1321.
163. Palombarini, J.A.; Martinez, E.C. Automatic Generation of Rescheduling Knowledge in Socio-technical Manufacturing Systems using Deep Reinforcement Learning. In Proceedings of the 2018 IEEE Biennial Congress of Argentina (ARGENCON), San Miguel de Tucuman, Argentina, 6–8 June 2018.
164. Palombarini, J.A.; Martínez, E.C. Closed-loop rescheduling using deep reinforcement learning. IFAC-PapersOnLine 2019, 52, 231–236.
165. Park, I.-B.; Huh, J.; Kim, J.; Park, J. A Reinforcement Learning Approach to Robust Scheduling of Semiconductor Manufacturing Facilities. IEEE Trans. Autom. Sci. Eng. 2020, 17.
166. Upkeep. Industrial Maintenance. Available online: https://www.upkeep.com/learning/industrial-maintenance (accessed on 28 October 2022).
167. ATS. The Evolution of Industrial Maintenance. Available online: https://www.advancedtech.com/blog/evolution-of-industrial-maintenance/ (accessed on 28 October 2022).
168. Moubray, J. RCM II-Reliability-centered Maintenance; Butterworth-Heinemann: London, UK, 1997.
169. Menčík, J. Maintenance. In Concise Reliability for Engineers; IntechOpen: London, UK, 2016; Available online: https://www.intechopen.com/chapters/50096 (accessed on 29 October 2022).
170. Pelantová, V. The Maintenance Management. In Maintenance Management-Current Challenges, New Developments, and Future Directions; IntechOpen: London, UK, 2022; Available online: https://www.intechopen.com/online-first/82473 (accessed on 29 October 2022).
171. Nguyen, V.-T.; Do, P.; Vosin, A.; Iung, B. Artificial-intelligence-based maintenance decision-making and optimization for multi-state component systems. Reliab. Eng. Syst. Saf. 2022, 228, 108757.
172. Yan, Q.; Wu, W.; Wang, H. Deep Reinforcement Learning Approach for Maintenance Planning in a Flow-Shop Scheduling Problem. Machines 2022, 10, 210.
173. Mohammadi, R.; He, Q. A deep reinforcement learning approach for rail renewal and maintenance planning. Reliab. Eng. Syst. Saf. 2022, 225, 108615.

174. Ong, K.S.H.; Wang, W.; Hieu, N.Q.; Niyato, D.; Friedrichs, T. Predictive Maintenance Model for IIoT-Based Manufacturing: A Transferable Deep Reinforcement Learning Approach. IEEE Internet Things J. 2022, 9, 15725–15741.
175. Acernese, A.; Yerudkar, A.; Del Vecchio, C. A Novel Reinforcement Learning-based Unsupervised Fault Detection for Industrial Manufacturing Systems. In Proceedings of the 2022 American Control Conference (ACC), Atlanta, GA, USA, 8–10 June 2022; pp. 2650–2655.
176. Li, C.; Chang, Q. Hybrid feedback and reinforcement learning-based control of machine cycle time for a multi-stage production system. J. Manuf. Syst. 2022, 65, 351–361.
177. Yousefi, N.; Tsianikas, S.; Coit, D.W. Dynamic maintenance model for a repairable multi-component system using deep reinforcement learning. Qual. Eng. 2022, 34, 16–35.
178. United Nations for Climate Change (UNFCCC). The Paris Agreement. Available online: https://unfccc.int/process-and-meetings/the-paris-agreement/the-paris-agreement (accessed on 1 November 2022).
179. Cheng, L.; Yu, T. A new generation of AI: A review and perspective on machine learning technologies applied to smart energy and electric power systems. Int. J. Energy Res. 2019, 43, 1928–1973.
180. Perera, A.; Kamalaruban, P. Applications of reinforcement learning in energy systems. Renew. Sustain. Energy Rev. 2021, 137, 110618.
181. Leng, J.; Ruan, G.; Song, Y.; Liu, Q.; Fu, Y.; Ding, K.; Chen, X. A loosely-coupled deep reinforcement learning approach for order acceptance decision of mass-individualized printed circuit board manufacturing in industry 4.0. J. Clean. Prod. 2021, 280, 124405.
182. Lu, R.; Li, Y.-C.; Li, Y.; Jiang, J.; Ding, Y. Multi-agent deep reinforcement learning based demand response for discrete manufacturing systems energy management. Appl. Energy 2020, 276, 115473.